

31003-0

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
PÓS GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

GERAÇÃO AUTOMÁTICA DE LÓGICA ALEATÓRIA
UTILIZANDO A METODOLOGIA TRANCA

por

Marcelo Soares Lubaszewski

Dissertação submetida como requisito parcial
para obtenção do grau de Mestre em
Ciência da Computação



SABi



05225402

Prof. Ricardo Augusto da Luz Reis
Orientador

UFRGS
INSTITUTO DE INFORMÁTICA
BIBLIOTECA

Porto Alegre, maio de 1990.

ESTE LIVRO DEVE SER DEVOLVIDO NA
ÚLTIMA DATA EXIBIDA

CATALOGAÇÃO NA FONTE

Lubaszewski, Marcelo Soares

Geração Automática de Lógica Aleatória
utilizando a Metodologia TRANCA. Porto Alegre,
PGCC da UFRGS, 1990.

lv.

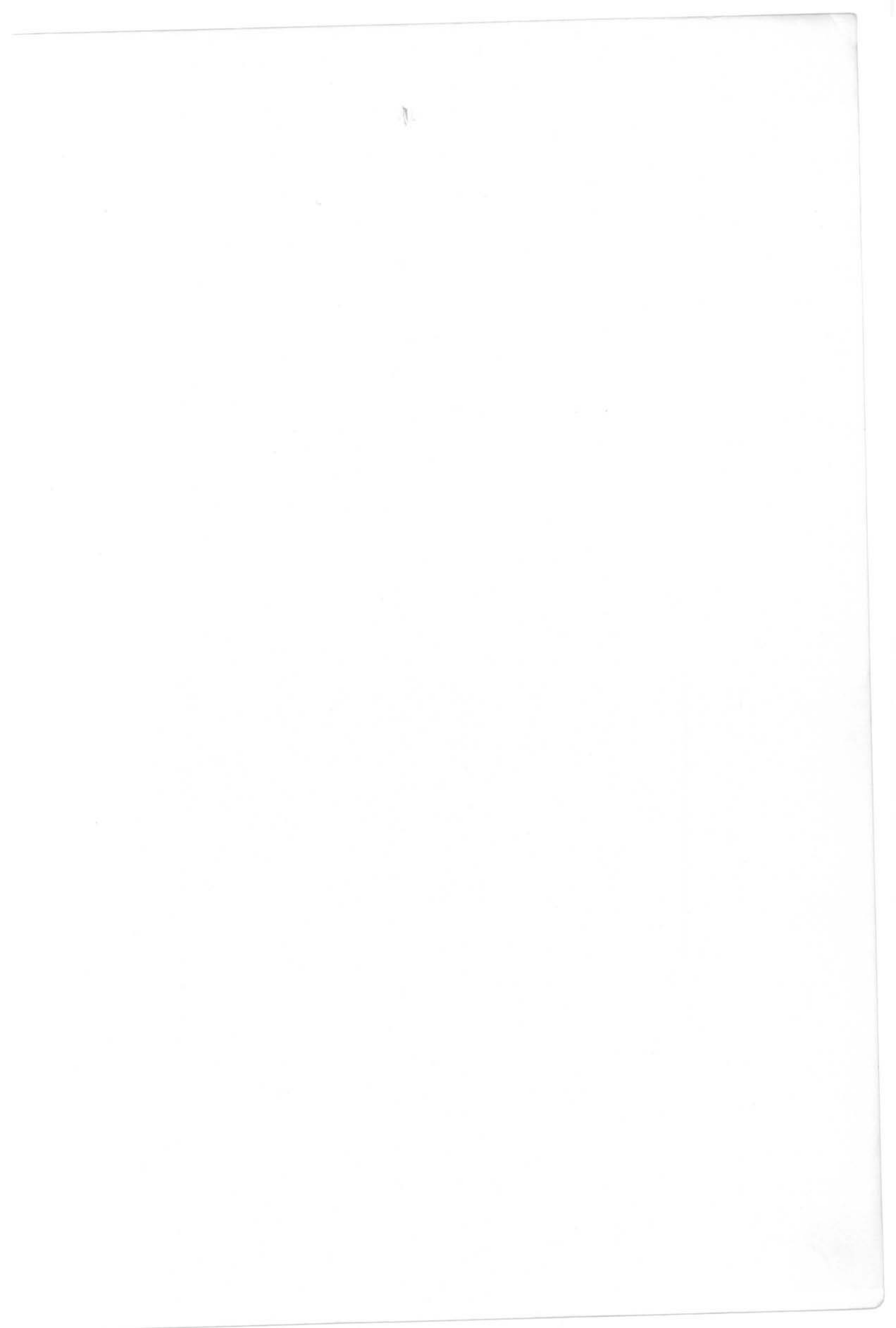
Diss. (mestr. ci, comp.) UFRGS-PGCC, Porto
Alegre, BR-RS, 1990.

Dissertação: Microeletrônica: PAC de Circuitos
Integrados: Geração de Módulos: Projeto VLSI:
Circuitos Complexos.

"É através da ação, do testar,
do usar suas capacidades que
o pensamento se desenvolve."

Madalena Freire

Aos meus pais e à Andréa,
companheiros incansáveis desta
e de tantas outras investidas.



AGRADECIMENTOS

Agradeço a Ricardo Reis que, além de ter dado mostras consecutivas de profissionalismo durante a orientação deste trabalho, tornou-se uma pessoa importante em minha vida pela sua amizade incondicional.

Aos companheiros de projeto, Rogério Gomes, Fernando Moraes, André Reis e Fábio Somenzi, pela aposta no sucesso deste trabalho e conseqüente contribuição a nível de idéias e cooperação braçal.

A Fernando Rosa, André Baggio e Avelino Zorzo que, em tempos diferentes, canalizaram seus esforços para que o sistema aqui apresentado viesse a assumir um perfil de ferramenta comercial.

A Sérgio Bampi, pela concessão de recursos de seu projeto de pesquisa em prol do desenvolvimento deste trabalho.

A Gilberto Marchioro, pelo interesse permanente em adaptar as ferramentas sob sua responsabilidade às necessidades deste projeto.

A todos os professores do PGCC que, de forma direta ou indireta, participaram do desenvolvimento desta dissertação, seja pela formação em sala de aula e pelo interesse na evolução do sistema, seja pelas proveitosas discussões e valiosas sugestões ao andamento e conclusão do trabalho.

Aos colegas Carlos Eduardo Pereira e Luigi Carro, pela constante troca de idéias relacionadas ao nosso universo profissional e à vida em geral.

Aos funcionários da secretaria do PGCC e do DI, aos funcionários do laboratório e da biblioteca, por se demonstrarem sempre extremamente receptivos ao fornecimento de informações, ao empréstimo de material e à prestação de

serviços.

A todos os meus amigos, colegas ou não de mestrado e profissão, distantes ou não do país, que fizeram parte de minha vida nestes últimos anos, influenciando diretamente na conquista do equilíbrio razão-emoção - indispensável ao sucesso de qualquer empreendimento.

Por fim, mas igualmente de coração, o meu agradecimento à minha família, à Andréa e à minha sogra, pelo carinho e a compreensão dispensados e, principalmente, por viverem tanto quanto eu este trabalho, mesmo sendo leigos no assunto.

SUMÁRIO

GLOSSÁRIO	13
LISTA DE ABREVIATURAS	15
LISTA DE FIGURAS	17
LISTA DE TABELAS	21
RESUMO	23
ABSTRACT	25
1 PROJETO AUTOMÁTICO DE CIRCUITOS INTEGRADOS	27
1.1 Compilação de Silício	27
1.2 Planejamento Topológico	30
1.3 Geração de Módulos	32 (30 e 31)
2 SÍNTESE DE LEIAUTE	37
2.1 Geração de Célula	39
2.2 Composição de Células	42
2.2.1 Posicionamento	43
2.2.1.1 O Problema	43
2.2.1.2 Ferramentas	45
2.2.1.3 Geração de Posicionamento	46
2.2.1.3.1 Geração Aleatória de Posicionamento .	47
2.2.1.3.2 Posicionamento por Crescimento de	
Agglomerados	47
2.2.1.3.3 Posicionamento por Partição	49
2.2.1.3.4 Posicionamento Global	52
2.2.1.4 Otimização de Posicionamento	53
2.2.1.4.1 Otimização por Intercâmbio de	
Células	54
2.2.1.4.2 Otimização baseada em Fenômenos	
Físicos	54
2.2.2 Orientação de Células	56
2.2.3 Assinalamento de Conectores	57
2.2.4 Roteamento	59
2.2.4.1 O Problema	59

2.2.4.2	Classes de Roteamento	61
2.2.4.3	Roteamento Detalhado	63
2.2.4.3.1	Roteadores de Propósito Geral	63
2.2.4.3.2	Roteadores de Canal	65
2.2.4.4	Refinamento do Roteamento	71
3	A METODOLOGIA TRANCA	73
3.1	Estrutura de Banda	75
3.2	Maleabilidade	76
3.3	Transparência	77
3.4	Gerenciamento de Trilhas	81
4	O GERADOR DE LÓGICA ALEATÓRIA TRANCA	83
4.1	Ambiente de Projeto	84
4.2	Descrição do Circuito	86
4.3	Geração de Célula	88
4.4	Composição de Células	92
4.4.1	POTRANCA	93
4.4.1.1	O Particionador	94
4.4.1.2	O Posicionador Intrabanda	105
4.4.2	RETRANCA	115
4.5	Considerações sobre a Implementação	136
5	AVALIAÇÃO DA FERRAMENTA E DA METODOLOGIA	139
5.1	Experiência com o Circuito CONVERSO	140
5.2	Experiência com o Circuito MODEM	144
5.3	Experiência com o Circuito TREVO	145
5.4	Comparação com o Sistema LES	157
5.5	Propostas para Futuras Otimizações	160
6	CONCLUSÃO	171
ANEXO 1	EXEMPLO DE DESCRIÇÃO NILOTRANCA	173
ANEXO 2	DOCUMENTAÇÃO DE UMA CÉLULA DA BIBLIOTECA .	177
ANEXO 3	ALGORITMO GERAL DO PARTICIONADOR	179
ANEXO 4	ALGORITMO GERAL DO POSICIONADOR INTRABANDA	183
ANEXO 5	ALGORITMO GERAL DO ROTEADOR	187

ANEXO 6	DESCRIÇÃO DE UM MULTIPLICADOR COMBINACIONAL DE 4 BITS	191
ANEXO 7	RELATÓRIO DE PARTICIONAMENTO	195
ANEXO 8	RELATÓRIO DE POSICIONAMENTO INTRABANDA ...	199
ANEXO 9	RELATÓRIO DE ROTEAMENTO	201
ANEXO 10	EXEMPLO DE UTILIZAÇÃO DA INTERFACE HOMEM-MÁQUINA	209
ANEXO 11	BIBLIOTECA DE "STANDARD CELLS" CONVENCIONAIS	215
BIBLIOGRAFIA		219

GLOSSÁRIO

Benchmark: exemplo de problema proposto para possibilitar a comparação entre as soluções geradas por algoritmos diferentes.

Dogleg: troca de trilha no roteamento de uma rede, realizada em uma coluna onde inexistem pinos da mesma rede.

Gate-Array: estilo de projeto de leiaute, cujas células são mapeadas em um arranjo de transistores bidimensional e previamente fabricado. Este arranjo é personalizado pela realização de conexões entre transistores utilizando uma ou mais camadas de metal. Grupos de transistores são agrupados em fileiras, as quais são separadas por canais horizontais de roteamento.

Gate-Matrix: estratégia de concepção de leiaute utilizada no projeto de circuitos baseado em lógica complementar. Os transistores são dispostos horizontalmente em fileiras de dispositivos de mesmo tipo - transistores p acima de transistores n. Os transistores que compartilham a mesma porta são dispostos sobre uma mesma coluna, onde a camada de polissilício implementa entradas, saídas e conexões internas. As conexões entre drenos de transistores posicionados sobre a mesma linha são realizadas utilizando difusão ou metal; as conexões entre dreno e porta são realizadas na direção horizontal utilizando metal; as conexões entre drenos de transistores posicionados em linhas diferentes e as conexões com as linhas de alimentação utilizam difusão ou uma segunda camada de metal.

Look-Ahead: qualificativo de procedimentos que, em um determinado passo, tomam decisões em função das possibilidades para os passos seguintes.

Sea-of-Gates: variação do estilo "gate-array" de projeto, onde inexitem canais de roteamento ou os canais são pequenos quando comparados à área ocupada pelos transistores. Neste caso, todo roteamento é realizado sobre os transistores.

Standard Cell: estilo de projeto de leiaute baseado em uma biblioteca de células com altura padrão. O projeto de tais células é semi-dedicado e dependente de tecnologia. As células são dispostas em fileiras e as conexões são realizadas em canais dedicados ao roteamento, localizados entre fileiras de células. Os canais possuem alturas variáveis, determinadas a partir das necessidades específicas do circuito que se projeta.

LISTA DE ABREVIATURAS

ALU	Arithmetic and Logic Unit
AMPLO	AMbiente integrado para o Projeto LOGico de sistemas digitais
ASIC	Application Specific Integrated Circuit
CMOS	Complementary Metal Oxide Silicon
CMP	Circuito Multi-Projeto
EXTRAMO	EXibidor gráfico do gerador TRAMO
GME	Grupo de MicroEletrônica
GND	GrouND
LES	Layout Expert System
MODEM	MOdulador-DEModulador
NILO	NIvel LOGico
NILOTRANCA	NIvel LOGico TRANCA
PAC	Projeto Auxiliado por Computador
PGCC	Pós-Graduação em Ciência da Computação
PLA	Programmable Logic Array
POTRANCA	POsicionador TRANCA
RAM	Random Access Memory
RETRANCA	RotEador TRANCA
ROM	Read Only Memory
TRAGO	TRAnca Gate-matrix generatOr
TRAMO	TRAnca MOdule generator
TRANCA	TRANSparent Cell Approach
UFRGS	Universidade Federal do Rio Grande do Sul
VCC	Voltage Continuos-Current
VLSI	Very Large Scale Integration
vs.	versus

UFRGS
INSTITUTO DE INFORMÁTICA
BIBLIOTECA

LISTA DE FIGURAS

Figura 1.1	Abordagens do modelo da divisão-e-conquista	28
Figura 1.2	Ambiente de compilação de Silício	29
Figura 1.3	Otimização local vs. otimização global .	31
Figura 1.4	Posicionamento de conectores	32
Figura 1.5	Roteamento entre blocos não adjacentes .	32
Figura 1.6	Anatomia de um gerador de módulos	34
Figura 2.1	Processo de concepção: formas de representação do circuito	38
Figura 2.2	Hierarquização do leiaute	39
Figura 2.3	Posicionamento: área vs. desempenho	44
Figura 2.4	Aspecto geométrico do posicionamento ...	45
Figura 2.5	Posicionamento por crescimento de aglomerados	48
Figura 2.6	Um exemplo de alocação relativa	49
Figura 2.7	A idéia da heurística Kernighan-Lin	50
Figura 2.8	Representação de redes múltiplas em grafos	51
Figura 2.9	Formas de particionamento	52
Figura 2.10	Possibilidades de orientação de uma célula	56
Figura 2.11	Redução do número de trilhas por espelhamento	56
Figura 2.12	Orientações permitidas e proibidas	57
Figura 2.13	Rotação de 90 graus	57
Figura 2.14	Classes de pinos	58
Figura 2.15	Efeitos do assinalamento de conectores .	58
Figura 2.16	Administração de camadas de roteamento .	60
Figura 2.17	Identificação de pares de pinos	61
Figura 2.18	Ordem de realização das conexões	61
Figura 2.19	Roteamento global em "Standard Cell" ...	63
Figura 2.20	Algoritmo de Lee: propagação de caminhos	64
Figura 2.21	Roteamento por projeção de linhas	65

Figura 2.22	Roteamento de canal	66
Figura 2.23	"Dogleg"	67
Figura 2.24	A idéia do algoritmo "left-edge"	68
Figura 2.25	Benefícios do uso de "doglegs"	68
Figura 2.26	Cumprimento de restrições verticais	69
Figura 2.27	Roteamento coluna a coluna	70
Figura 2.28	Minimização de contatos	72
Figura 3.1	Conexão entre células adjacentes	74
Figura 3.2	Estruturação em bandas	75
Figura 3.3	Passo de metal	75
Figura 3.4	Opções de leiaute para uma célula de memória	78
Figura 3.5	Conexões com bandas adjacentes	80
Figura 3.6	Redução de área e de comprimento de conexões	80
Figura 3.7	Alocação de trilhas	81
Figura 4.1	Ambiente de projeto TRANCA	84
Figura 4.2	Prioridades para alocação de trilhas ...	89
Figura 4.3	Leiaute de uma "nand" de 4 entradas	90
Figura 4.4	Exemplo de intercâmbio	95
Figura 4.5	Partição em fatias	97
Figura 4.6	Estratégia de redes imaginárias vs. pesos diferenciados	99
Figura 4.7	Etapas do particionamento	100
Figura 4.8	Fator de compensação do critério de equilíbrio	103
Figura 4.9	Redução do número de soluções mínimas ..	104
Figura 4.10	Exemplo de particionamento	106
Figura 4.11	Estratégia de posicionamento intrabanda	107
Figura 4.12	Posicionamento em uma banda	113
Figura 4.13	Exemplo completo de posicionamento	116
Figura 4.14	Classes de células de interconexão	117
Figura 4.15	Célula de interconexão intrabanda vs. conexão por justaposição	117

Figura 4.16	Redes divididas entre bandas adjacentes	118
Figura 4.17	Possibilidades de disposição de células de interconexão interbanda	119
Figura 4.18	Soluções com múltiplas células de interconexão alinhadas	120
Figura 4.19	Solução com pares de células de interconexão alinhadas	120
Figura 4.20	O assinalamento de conectores no RETRANCA	122
Figura 4.21	Roteamento global das bandas	124
Figura 4.22	Cálculo do deslocamento	126
Figura 4.23	Acúmulo de deslocamentos	127
Figura 4.24	Reaproveitamento de área	127
Figura 4.25	Refinamento do roteamento global	129
Figura 4.26	Roteamento intrabanda	131
Figura 4.27	Influência da ordenação no número de trilhas	132
Figura 4.28	Saída simbólica	133
Figura 4.29	Exemplo de leiaute completo	135
Figura 4.30	Cadeia de síntese de leiaute	136
Figura 4.31	Organização de arquivos em diretórios ..	137
Figura 5.1	Planta-baixa do conversor	142
Figura 5.2	Leiaute final do conversor	143
Figura 5.3	Leiaute manual do circuito MODEM	146
Figura 5.4	Leiaute automático do circuito MODEM ...	147
Figura 5.5	Ganho em área pelo uso de canais com transparência	150
Figura 5.6	Topologias dos módulos comparados	150
Figura 5.7	Relações altura vs. número de bandas ...	152
Figura 5.8	Relações largura vs. número de bandas ..	152
Figura 5.9	Relações largura/altura vs. número de bandas	153
Figura 5.10	Relações área vs. número de bandas	154
Figura 5.11	Relações área vs. número de bandas com o uso de transparência vertical	156

Figura 5.12	Diagrama lógico do flip-flop D	159
Figura 5.13	Leiaute do flip-flop D	159
Figura 5.14	Eliminação de célula de interconexão ...	162
Figura 5.15	Fusão de células de interconexão	163
Figura 5.16	Roteamento com o uso de metal 2	166
Figura 5.17	Implementação de barramentos	167

LISTA DE TABELAS

Tabela 4.1	Abordagens de geração de célula	88
Tabela 4.2	Células da biblioteca TRANCA	91
Tabela 4.3	Custos e ganhos individuais	95
Tabela 4.4	Fatores de correção	95
Tabela 4.5	Ganhos de intercâmbio: GP do par (C1,C2)	96
Tabela 4.6	Escores globais e apontador de posicionamento	113
Tabela 5.1	Comparação entre abordagens	151
Tabela 5.2	Novos valores para o uso de metal 2	154

RESUMO

Este trabalho trata da geração de módulos em lógica aleatória em um ambiente de compilação de silício.

Apresenta-se uma ferramenta automática de projeto, cuja base é um subconjunto de características da metodologia de concepção de leiaute TRANCA. Esta metodologia é fundamentada, principalmente, na realização de conexões sobre as áreas ativas do circuito.

Descreve-se as estratégias de geração e composição de células adotadas pela ferramenta. Estas estratégias resultaram de adaptações de técnicas reconhecidas de síntese automática de leiaute e de outras, desenvolvidas para suportarem as novas vantagens e restrições impostas pela metodologia em uso.

Compara-se leiautes gerados automaticamente, com versões "manuais", "Standard Cell" e de outras ferramentas de síntese, constatando-se a compactação de área inerente à adoção da metodologia TRANCA. Projeções realizadas mostram que as vantagens de economia em silício acentuam-se com o uso de todo o potencial da metodologia.

Realiza-se uma primeira avaliação dos resultados obtidos pela ferramenta, propondo-se, em função das deficiências observadas, otimizações nos procedimentos utilizados. Apresentam-se, também, sugestões para a síntese automática de leiaute suportando a metodologia como um todo.

ABSTRACT

This work deals with the generation of random logic modules within a silicon compilation environment.

A design automatic tool is presented. Its design-basis is a subset of features of the TRANCA layout methodology, which is mainly based on routing over the circuit gates.

The cell-building and cell-composition strategies adopted by the tool are described. They resulted from the adaption of well-established techniques and from other techniques developed to meet the advantages and constraints imposed by the methodology in use.

Some automatic layouts are compared to handcrafted versions, Standard Cell layouts and versions by other physical design tools. The results show the area compaction inherent to the adoption of the TRANCA methodology. Accomplished projections show that the advantages of silicon saving increase with the use of the whole potential of the methodology.

A first evaluation of the results obtained by the tool takes place. As a function of the observed deficiencies, optimizations for the used procedures are proposed. Suggestions for the layout automatic synthesis using the whole methodology are also presented.

1 PROJETO AUTOMÁTICO DE CIRCUITOS INTEGRADOS

No estágio atual de desenvolvimento tecnológico da indústria de semicondutores, não tem sentido falarmos em projeto de circuitos VLSI sem considerarmos a automação do processo de concepção. Considere-se que uma relação custo-benefício favorável só é possível com a redução do ciclo de projeto e o conseqüente afastamento do fantasma da obsolescência por um prazo que justifique o investimento.

Em função disso, o setor de microeletrônica volta decisivamente sua atenção ao desenvolvimento de ambientes integrados de projeto de circuitos. A efetividade de tais ambientes se dá à medida que confiabilidade e flexibilidade de projeto sejam oferecidas, e a partir do momento que as ferramentas que compõem o ambiente sejam capazes de suportar a crescente complexidade dos sistemas digitais passíveis de integração.

1.1 Compilação de Silício

A divisão-e-conquista tem sido o modelo mais freqüentemente empregado na abordagem da complexidade de projeto, observando-se uma tendência generalizada de seu uso com um enfoque descendente e baseado em refinamentos sucessivos.

A idéia do modelo, considerando-se o enfoque descendente, é dividir o problema maior em subproblemas e atacá-los individualmente de forma seqüencial, complementar e/ou concorrente (figura 1.1).

A abordagem seqüencial pressupõe que um determinado subproblema só possa ser resolvido a partir da solução para o subproblema que o antecede na hierarquia. O projeto da máquina de estados de um sistema digital, por exemplo, resulta da determinação prévia da parte operativa a ser controlada. Isto não significa, absolutamente, que não possa haver realimentação no processo.

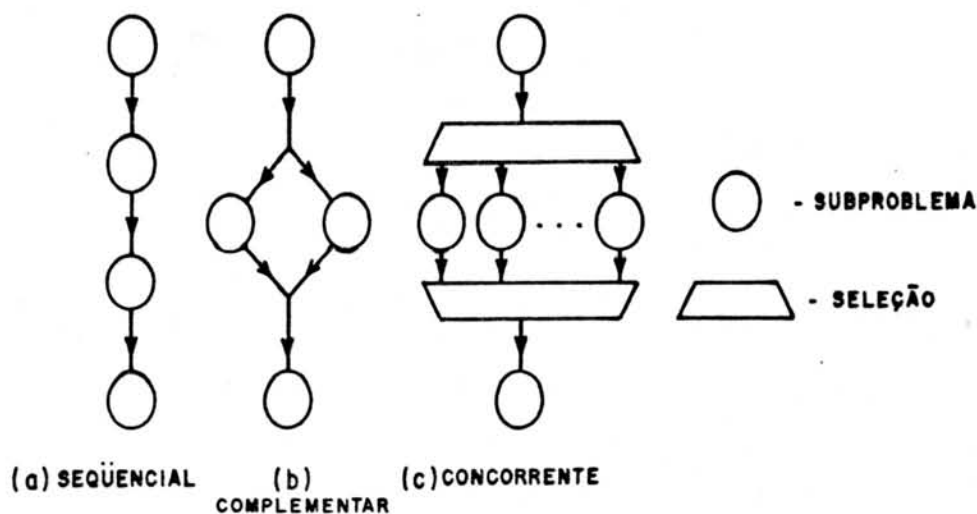


Figura 1.1 - Abordagens do modelo da divisão-e-conquista.

A abordagem complementar, por sua vez, pressupõe que um determinado problema possa ser dividido em subproblemas passíveis de tratamento disjunto e que a conjunção de soluções locais resulte na solução global. O leiaute final de um sistema digital, por exemplo, resulta da conjunção dos leiautes da parte operativa e da parte de controle devidamente conectados.

Por fim, a abordagem concorrente, resultado da diversidade de procedimentos para a solução de um mesmo subproblema, pressupõe momentos de decisão que antecedem e/ou sucedem o ataque do subproblema, conforme a adequacidade de cada procedimento às condições de contorno impostas pela aplicação e/ou as vantagens e desvantagens das soluções propostas por cada procedimento. A parte de controle de um sistema digital normalmente pode ser implementada por microprogramação, por lógica estruturada a dois níveis ou em lógica aleatória. Área, desempenho e testabilidade são alguns exemplos de condições de contorno geralmente impostas, bem como medidas para avaliação das soluções propostas [OBR 82].

As três abordagens, quando presentes em um mesmo sistema - o que normalmente ocorre, relacionam-se entre si de uma forma tão intrincada quanto harmônica.

Do exposto, imediatamente conclui-se que

hierarquia e modularidade são características não só necessárias a um ambiente de projeto, como também inerentes ao modelo da divisão-e-conquista.

Aliando-se hierarquia e modularidade às características de estruturação e abstração - também imprescindíveis à concepção automática de circuitos integrados, a aplicação do modelo ao problema resulta na sua divisão em projeto de sistema e projeto em silício [DEM 87], conforme mostra a figura 1.2.

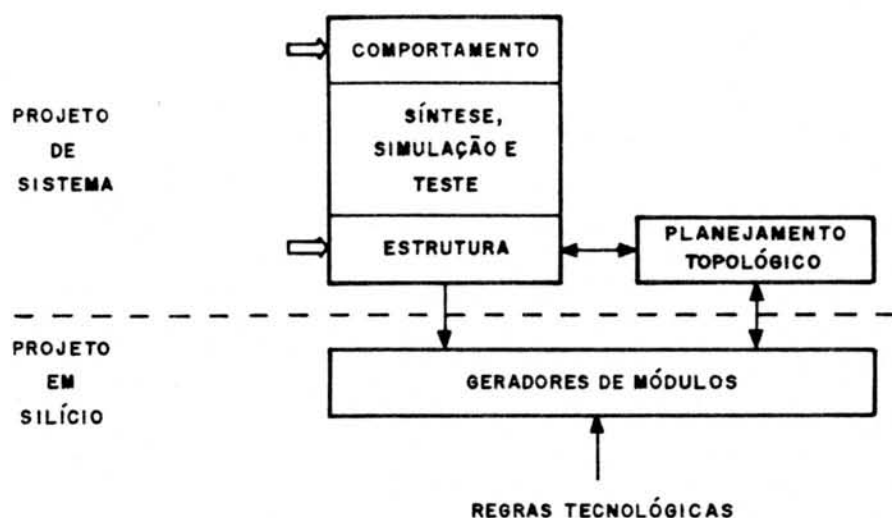


Figura 1.2 - Ambiente de compilação de silício.

No nível de projeto de sistema, realiza-se transformações sobre uma descrição comportamental do circuito, na busca de uma representação estrutural equivalente e de um nível de abstração a partir do qual o mapeamento em silício seja possível.

No nível de projeto em silício, cada gerador de módulos é responsável pela transformação de uma parte dessa estrutura em uma representação geométrica equivalente que esteja de acordo com as regras tecnológicas em uso.

O módulo de planejamento topológico (figura 1.2), por sua vez, gerencia a iteração entre os dois níveis de projeto, o que sugere que, no nível de projeto de sistema, as possibilidades em silício sejam efetivamente

consideradas.

A composição dos níveis de projeto de sistema e projeto em silício resulta na compilação de silício, que pode ter um caráter comportamental ou estrutural conforme o ponto de entrada no processo de concepção automática (figura 1.2).

[DEM 87] e [CAR 89] abordam com mais detalhes o tema compilação de silício, tanto do ponto de vista conceitual, como do ponto de vista evolutivo. Já [DEM 89] apresenta e descreve os principais sistemas de compilação de silício disponíveis comercialmente e em desenvolvimento nas universidades americanas e européias mais conceituadas em termos de microeletrônica.

1.2 Planejamento Topológico

A prática da realização de um planejamento da planta baixa do circuito, antes mesmo da realização dos blocos funcionais, data da era dos circuitos integrados concebidos "manualmente". Já naquela época, os engenheiros de silício optavam por uma metodologia de projeto que fosse tanto quanto possível descendente.

A distribuição prévia dos blocos funcionais na planta baixa visa, em última análise, a redução de área do chip como um todo, na busca de um bom rendimento em termos de custos e efetividade de produção e de um melhor desempenho elétrico do circuito que se concebe [REI 89a]. Para tanto, a filosofia de otimização global deve imperar, visto que uma solução local ótima pode não significar a melhor opção para o conjunto (figura 1.3).

O planejamento topológico é tanto mais flexível quanto maior é a receptividade dos geradores de módulos ao atendimento de restrições de forma de cada bloco funcional. Evidentemente, quem impõe tais restrições deve conhecer de antemão as limitações topológicas dos tipos de módulos

disponíveis, de forma a não ultrapassar seus limites de maleabilidade [REI 83].

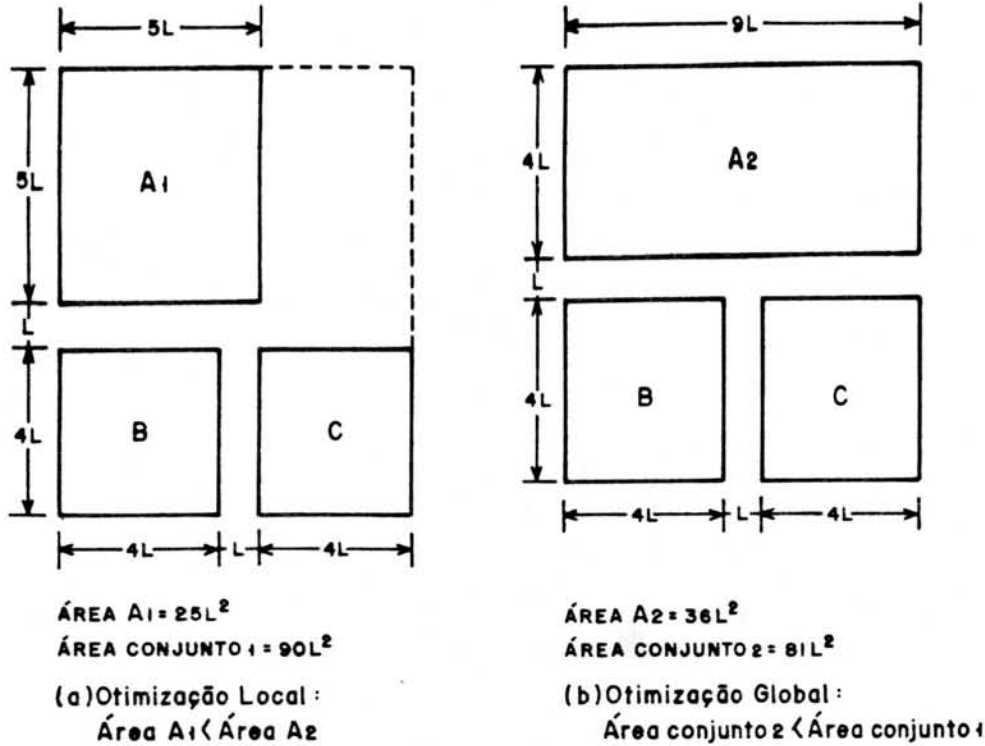


Figura 1.3 - Otimização local vs. otimização global.

A responsabilidade de bem administrar os recursos para a realização de interconexões fica dividida entre o planejador topológico e os geradores de módulos. Ao planejador topológico cabe propor um posicionamento tal de conectores, que as zonas dedicadas a interconexões entre os blocos funcionais se tornem localizadas e ocupem a menor área possível (figura 1.4). Aos geradores de módulos cabe a realização das conexões internas aos módulos de forma ótima (de preferência sobre as zonas ativas) e a reserva de camadas ou previsão de regiões de roteamento que possibilitem a conexão direta entre blocos não adjacentes (figura 1.5) [ANC 82].

Em função da diversidade de características a serem consideradas, o planejamento topológico vem sendo realizado de forma semi-automática, ou seja, mediante intervenção humana. O uso de técnicas de inteligência artificial parece ser extremamente promissor no sentido da automação do processo de planejamento topológico.

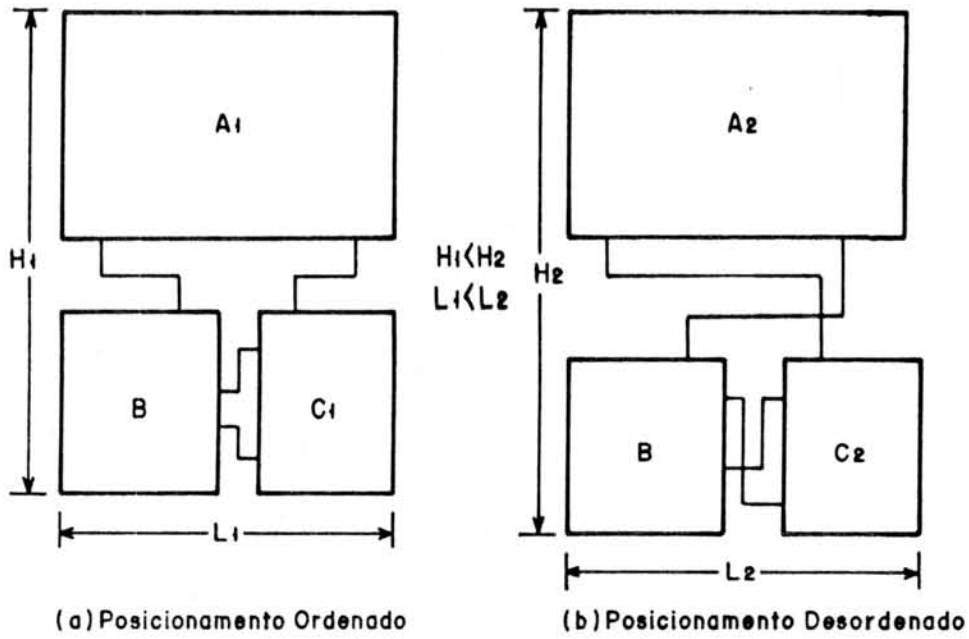


Figura 1.4 - Posicionamento de conectores.

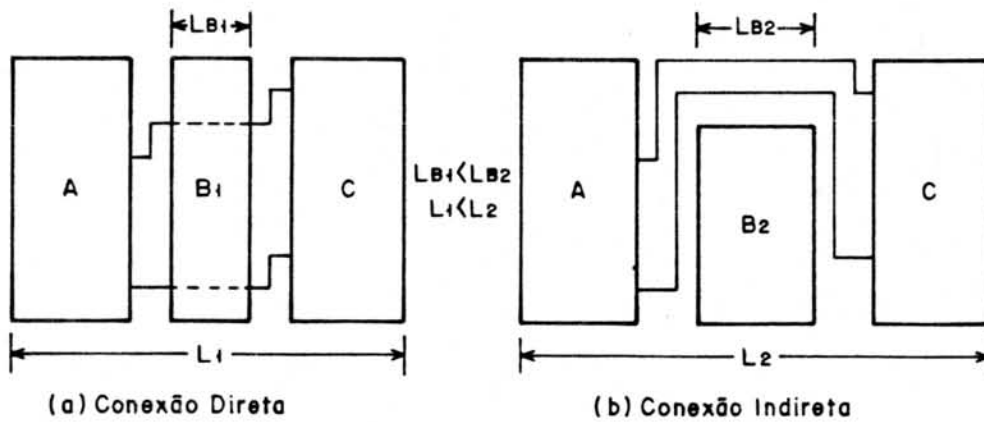


Figura 1.5 - Roteamento entre blocos não adjacentes.

1.3 Geração de Módulos

Geradores de módulos podem ser definidos como ferramentas capazes de produzirem leiautes, em uma topologia previamente estabelecida, que implementem fisicamente uma determinada estrutura. A personalização de cada módulo, que atribui ao gerador especializado um caráter geral dentro da classe topológica que ele representa, é fornecida via parâmetros e dados oriundos do projeto de sistema.

Estas ferramentas, em função do tipo de geometria que produzem, podem ser classificadas em geradores de módulos regulares ou geradores de módulos em lógica aleatória.

Módulos regulares são aqueles definidos por um "gabarito" e um conjunto limitado de células primitivas que populam este "gabarito". A personalização de tais módulos é feita por alongamento e justaposição das instâncias das células primitivas e/ou pela definição da posição de transistores em pontos de cruzamento entre dois padrões de roteamento ortogonais entre si. Alguns exemplos de módulos regulares são: memórias em geral (ROMs e RAMs) e arranjos lógicos programáveis (PLAs).

Módulos em lógica aleatória são aqueles que, pela própria irregularidade de sua geometria, não possuem um "gabarito" onde a disposição das instâncias das células primitivas seja realizada. Neste caso, a variedade de tipos e de complexidade povoa o conjunto de células primitivas. Observe-se que a personalização do módulo passa a ser função do elevado grau de liberdade para a disposição das instâncias das células primitivas no leiaute e para a realização das conexões entre elas. Exemplos clássicos de módulos em lógica aleatória são blocos funcionais concebidos segundo a abordagem "Standard Cell" e módulos projetados utilizando-se a técnica de "Gate-Matrix" [MOR 89b].

Partes operativas, tais como ALUs, contadores e multiplicadores, podem ser implementadas tanto como módulos regulares, como em lógica aleatória.

Abstraindo-se os tipos de geradores de módulos existentes, podemos enumerar as características desejáveis destas ferramentas sob a filosofia de compilação de silício apresentada anteriormente. O fornecimento de um modelo topológico para cada módulo é imprescindível do ponto de

vista do planejamento da planta baixa; a avaliação de desempenho do módulo só é possível de ser realizada a partir de um modelo temporal que o identifique; a adaptabilidade do gerador de módulos a novas tecnologias é decisiva na determinação da vida útil da ferramenta; por fim, o desenho do leiaute é o que justifica a existência do gerador de módulos.

A partir destas considerações, chega-se à proposta para a anatomia de um gerador de módulos apresentada na figura 1.6. Os procedimentos de geração de célula e de composição de células (figura 1.6) encarregam-se, respectivamente, da construção das células primitivas que serão instanciadas no leiaute, e da montagem do conjunto de instâncias e estabelecimento das relações entre elas.

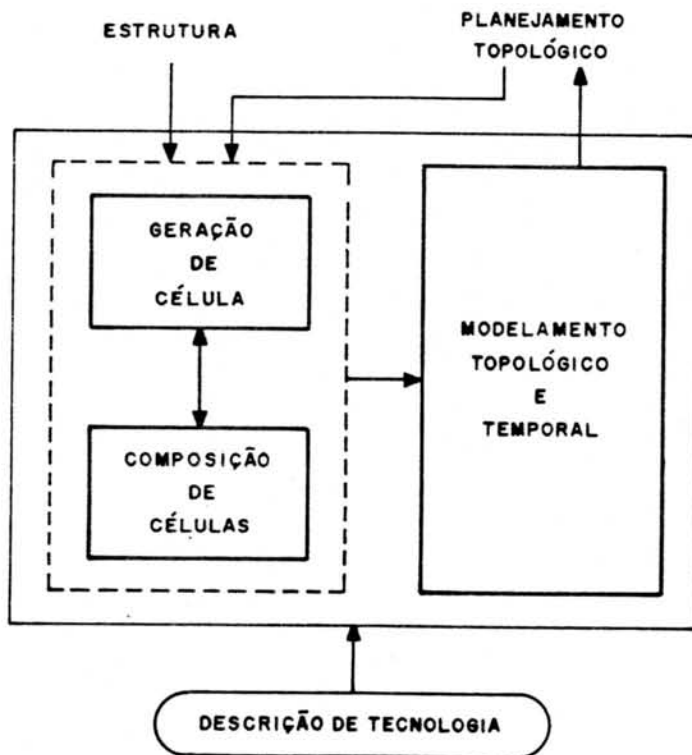


Figura 1.6 - Anatomia de um gerador de módulos.

[DEM 87] e [PRE 88] discutem mais amplamente o tema geração de módulos e classificam os tipos de módulos

de forma diferente daquela apresentada nesta seção.

Como o próprio desenvolvimento deste capítulo introdutório sugere, a linha deste trabalho consiste na geração de módulos, conforme os requisitos básicos para o planejamento topológico citados anteriormente. Em última análise, o que aqui se apresenta pretende ser uma contribuição para a construção de um compilador de silício do GME/UFRGS.

O enfoque deste trabalho reside na geração de lógica aleatória segundo uma metodologia de concepção de leiaute (TRANCA), cuja adoção comprovadamente resulta em compactação de área. Portanto, os aspectos topológico-geométricos da geração de módulos são enfatizados nos capítulos que seguem.

A organização do texto procura percorrer o tema proposto do geral em direção ao específico. Para tanto, parte-se do problema da síntese de leiaute (capítulo 2), apresenta-se a metodologia TRANCA (capítulo 3), descreve-se o gerador de módulos desenvolvido (capítulo 4) e avalia-se esta ferramenta e a metodologia adotada (capítulo 5).

2 SÍNTESE DE LEIAUTE

Uma metodologia descendente de projeto pressupõe a existência de fases que trabalham em diferentes níveis de abstração e que realizam transformações sobre uma representação de entrada, gerando outra representação para o circuito. Além de acrescentar à representação de entrada informações que a aproximam cada vez mais da implementação física, a representação de saída de cada fase deve ser equivalente à sua antecessora, deve atender às restrições de projeto que lhe são impostas e propagá-las às fases posteriores.

A partir da especificação de um determinado circuito, então, se avança no processo de concepção, obtendo-se de cada fase, sucessivamente, uma representação comportamental, uma representação estrutural, uma representação geométrica e, por fim, chega-se a uma implementação física possível (figura 2.1) [GAJ 83]. Cabe salientar que os tipos de representação citados são classificados conforme suas características predominantes, uma vez que construções de diferentes níveis de abstração freqüentemente aparecem combinadas durante o percurso das diversas fases de projeto.

A fase responsável pela geração de uma representação geométrica a partir de uma representação estrutural para o circuito é conhecida como projeto físico. Como todas as outras fases, esta é composta pelas etapas de síntese, análise e verificação [PRE 88]. A etapa de síntese do projeto físico deriva uma representação geométrica para o circuito; a etapa de análise, por sua vez, avalia a representação geométrica segundo sua correção, completeza e cumprimento das restrições topológicas e elétricas impostas; por fim, a etapa de verificação cumpre demonstrar a equivalência sob todos aspectos entre a representação sintetizada e a representação estrutural de origem.

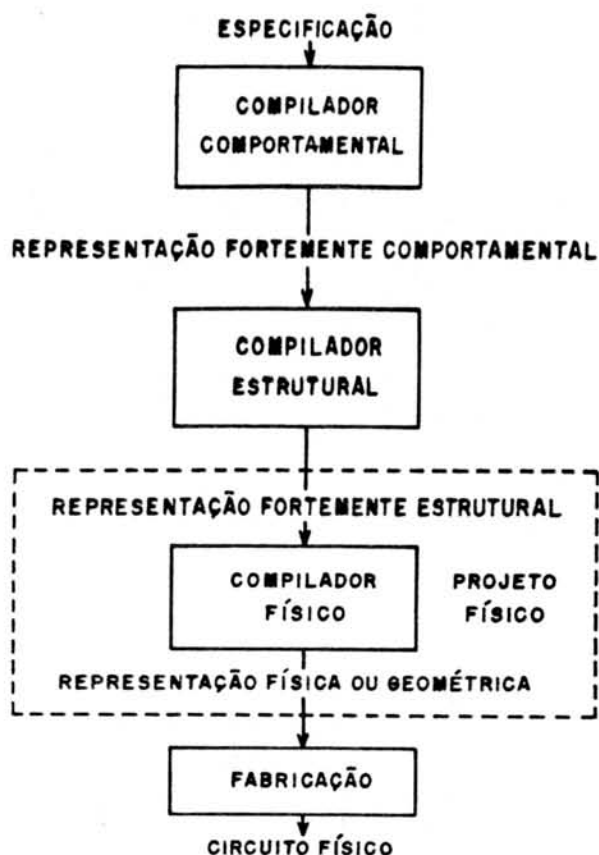


Figura 2.1 - Processo de concepção: formas de representação do circuito.

De particular interesse para este trabalho é a etapa de síntese da fase de projeto físico, aqui chamada de síntese de leiaute.

O leiaute final de um chip pode ser decomposto segundo a hierarquia apresentada na figura 2.2, a qual sugere uma classificação das células em módulos, células compostas e células primitivas. Cada módulo contém tipos de células normalmente não compartilhados com outros módulos, o que caracteriza sua especialização topológico-geométrica. As células compostas são definidas por instâncias de outras células - compostas ou primitivas, e pelo relacionamento entre estas instâncias em termos de posições e de conexões. Já as células primitivas, construídas a partir de uma rede de transistores, são consideradas indivisíveis do ponto de vista do projeto lógico-estrutural.

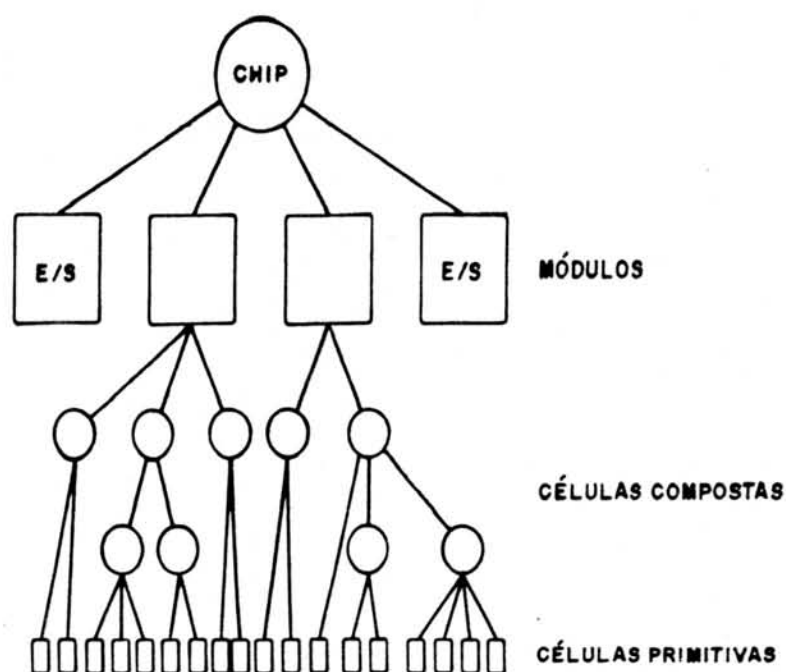


Figura 2.2 - Hierarquização do leiaute.

Da hierarquia proposta, prontamente identifica-se os procedimentos inerentes à síntese de leiaute, quais sejam: o estabelecimento da planta baixa do circuito e a concepção dos diversos módulos - realizada a partir do projeto das células primitivas e da composição iterativa de instâncias destas células.

Dando ênfase, então, à síntese automática de leiaute de blocos funcionais, as seções seguintes apresentam e discutem a geração de células primitivas e os diversos aspectos da composição de células. Dedicase especial atenção àqueles procedimentos que se adaptam à síntese de módulos ou de circuitos inteiros em lógica aleatória.

2.1 Geração de Célula

O problema de combinar transistores, fios e contatos, para a construção de primitivas lógicas, pode ser abordado segundo os seguintes níveis de abstração: geométrico, procedural, simbólico e estrutural a nível de transistores.

Gerar células no nível geométrico implica em trabalhar diretamente com a descrição das máscaras para

integração, mediante o uso de editores gráficos especializados. Este procedimento é caracterizado pelo alto grau de liberdade geométrica, pela impossibilidade de automação completa do processo de geração, pelo elevado tempo de projeto e pela grande susceptibilidade humana a erros. Pelo fato da atividade criativa ser realizada pelo projetista, a qualidade dos leiautes obtidos com este método é extremamente elevada. No entanto, o uso de verificadores de regras de projeto e de extratores elétricos para verificação de conectividade se torna obrigatório. Uma grande desvantagem da edição direta de máscaras é a vinculação inevitável da biblioteca de células primitivas a uma determinada tecnologia.

A opção de gerar células de forma procedural pressupõe a expressão dos retângulos que implementam as máscaras e a expressão das distâncias entre estes retângulos, não em tamanho físico real, mas em uma dimensão parametrizável. Muito embora com esta aproximação obtenha-se independência de tecnologia e geração automática completa, inúmeros são os problemas advindos da multiplicidade de regras de separação entre camadas, peculiar de leiautes irregulares. Principalmente por esta limitação, a abordagem procedural é típica da geração de células de módulos do tipo memórias [CAR 88], PLAs [GAL 89] e partes operativas.

A abordagem do problema de forma simbólica abstrai as regras de projeto e passa a trabalhar no domínio topológico da geração de célula. A disposição relativa de símbolos, representativos de transistores, fios e contatos, é realizada sobre uma grade e posteriormente convertida para um leiaute que preserva a topologia sugerida por tal posicionamento. A disposição de símbolos é feita via edição interativa; a conversão, completamente automática, garante a adaptação da célula a novas tecnologias e assegura a correção do leiaute gerado para as regras de projeto em

uso. A flexibilidade de geração se faz sentir à medida em que meios para parametrização dos elementos e para atendimento de restrições de posicionamento de conectores estão presentes. A desvantagem de tal abordagem é que os leiautes obtidos geralmente são menos densos que aqueles concebidos via edição direta de máscaras ou por geração procedural. Métodos de compactação aliviam esta situação, muito embora estes sejam extremamente difíceis de sistematizar e de serem controlados pelo projetista. Um exemplo de sistema simbólico de geração de células é apresentado em [MAR 89]. [MOR 89a] descreve um sistema de síntese de leiaute de memórias RAM utilizando leiaute simbólico.

A abordagem estrutural de geração de célula é a que trabalha no nível mais abstrato do problema. A partir de um conjunto de transistores e de relações bem definidas entre eles, a topologia detalhada e a geometria da célula são derivadas automaticamente, mantendo-se a independência de tecnologia e a parametrização dos elementos ativos. Normalmente a área de célula obtida pela abordagem estrutural é maior do que com os outros métodos. No entanto, a automação completa da geração, aliada à relativa flexibilidade topológica, sugerem a dinâmica de iteração com a composição de células, em tempo de execução desta. Os resultados obtidos são tão melhores à medida que se evolui de técnicas mais restritivas, como "Gate-Matrix" [PIG 88][MOR 89b], na direção de sistemas especialistas orientados para topologias mais flexíveis, como aquele apresentado em [KOL 85].

Independente da abordagem que se escolha para a geração, a caracterização elétrica da biblioteca de células primitivas tem de ser feita em algum momento. Só assim, será possível a avaliação do desempenho dos módulos gerados pela composição de células, em termos de frequência máxima de operação e consumo para uma determinada tecnologia.

Na abordagem geométrica, a única possibilidade de caracterização se dá via uso de extrator e simulador elétrico. Na abordagem procedural, em função da extrema regularidade de células e módulos, a avaliação elétrica pode ser realizada através de modelos específicos, tratados de forma implícita na composição. Se o método for o simbólico ou o estrutural, o modelamento é função do conhecimento prévio do papel elétrico de cada símbolo ou elemento, da parametrização de suas instâncias e da arquitetura do módulo como um todo. Nas duas últimas abordagens, caso a arquitetura do módulo não seja regular, a caracterização de cada célula individualmente é imprescindível e a avaliação elétrica do módulo só é possível após a composição de células. O único método que pressupõe a geração de leiaute completo, para posterior análise de desempenho, é o geométrico.

Embora as abordagens de geração de célula trabalhem em níveis diferentes de abstração, o objetivo último é sempre o da geração do leiaute do módulo, ou seja, a obtenção da geometria que implementa as máscaras de integração. No entanto, qualquer dos métodos pode permanecer em seu próprio nível de abstração, até que a composição de células seja totalmente realizada. A partir daí, então, a conversão de representação para a geometria, se for o caso, deve ser efetuada.

2.2 Composição de Células

A composição de células sobre uma superfície, para a obtenção do leiaute de um bloco funcional, pode ser dividida nos seguintes subproblemas:

- a) posicionamento de instâncias das células primitivas;
- b) determinação da orientação mais apropriada para cada instância;
- c) assinalamento dos conectores da interface de

cada instância;

d) alongamento das instâncias para realização de conexões por justaposição; e,

e) roteamento das redes que relacionam as instâncias entre si.

Afora o item d), bastante específico do projeto procedural e simbólico [MAR 89], os outros serão discutidos a seguir.

2.2.1 Posicionamento

2.2.1.1 O Problema

De uma maneira geral, posicionar as células que constituem um bloco significa estabelecer a localização de cada célula dentro de uma determinada área, segundo alguns objetivos pré-definidos.

O problema de alocação pode ser dividido em duas grandes classes [LIE 85]:

a) posicionamento relativo, quando a um conjunto de células, cujas relações de adjacência (interconexões) são conhecidas, se atribui parte de uma área sem a definição da localização de cada célula individualmente; e,

b) posicionamento absoluto, quando a cada célula se atribui uma localização fixa sobre a superfície que deve receber o módulo.

Os objetivos universais de um processo de posicionamento de células são:

a) obtenção de uma pequena área de implementação pela distribuição uniforme e efetiva de células e interconexões; e,

b) redução da complexidade de roteamento, tornando possível a sua realização integral e de forma compacta.

Alguns objetivos particulares a determinadas

implementações podem ser: minimização de área de leiaute, alto grau de localidade de interconexões, equalização da dissipação de calor sobre a superfície ou maximização de desempenho do circuito. Combinações de tais objetivos são freqüentemente adotadas, embora muitas vezes ocorram conflitos somente solucionados a partir de uma priorização prévia. Observe a figura 2.3: se considerarmos que as conexões é que determinam a velocidade de operação do circuito, vemos que o posicionamento que oferece a menor área de implementação não corresponde à solução de maior desempenho.

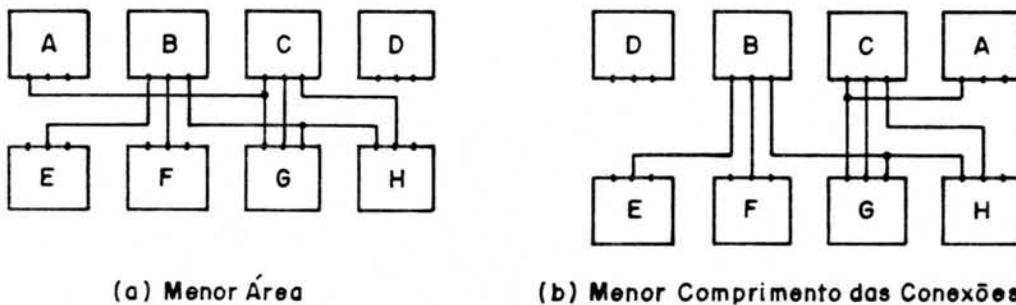


Figura 2.3 - Posicionamento: área vs. desempenho.

Uma vez que estes objetivos são difíceis de serem colocados sob a forma de funções que possibilitem uma avaliação computacional imediata, funções-objetivo mais restritas são adotadas na esperança que o seu uso resulte, indiretamente, no cumprimento dos objetivos reais. Assim, por exemplo, na busca de um posicionamento que resulte no menor comprimento total das interconexões ou na minimização do maior comprimento de conexão, procura-se obter a solução mais apropriada com relação à freqüência máxima de operação do circuito.

Os próprios parâmetros destas funções-objetivo restritas são utilizados como medidas para a comparação entre posicionamentos alternativos. Os parâmetros mais freqüentemente encontrados relacionam-se à completeza do roteamento, ao comprimento das conexões, ao número de conexões que cortam linhas imaginárias, ao número de

trilhas necessárias ao roteamento de um canal (densidade) ou à área do leiaute resultante.

Independente dos objetivos de um procedimento de posicionamento, uma solução exata para um número grande de células é via de regra impraticável. A única possibilidade garantida de termos a solução ótima do problema para qualquer aplicação é analisarmos todas as combinações possíveis de posicionamento das células. Mesmo em casos médios, a avaliação de todas as combinações é computacionalmente infactível, o que leva à procura de heurísticas que ofereçam resultados razoáveis, tanto em termos de posicionamento, como em termos de tempo de máquina.

2.2.1.2 Ferramentas

Sob o aspecto geométrico, as ferramentas de posicionamento podem ser divididas em ferramentas de posicionamento regular ("polycell") e irregular ("general cell") [HAS 82]. No primeiro caso, as células, de altura comparável, são posicionadas em fileiras; no segundo, células retangulares de dimensões arbitrárias devem ser convenientemente distribuídas sobre uma superfície (figura 2.4).

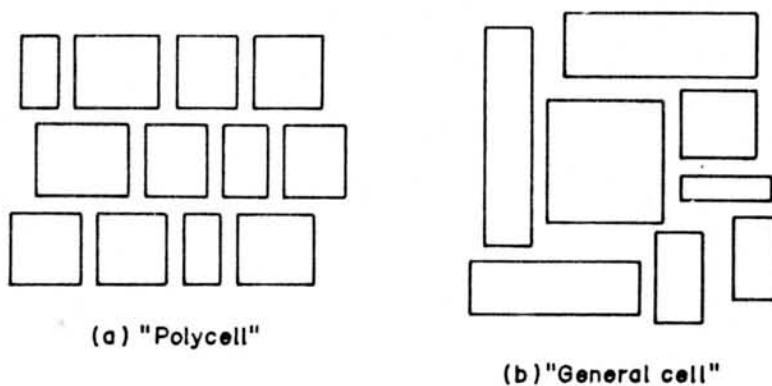


Figura 2.4 - Aspecto geométrico do posicionamento.

Como na geração de módulos o número de células é normalmente elevado, o estilo regular é preferido devido à

menor complexidade de posicionamento introduzida pela restrição topológica de altura fixa. A abordagem irregular, por sua vez, é típica do problema de planejamento topológico, onde o número de blocos funcionais é bastante mais reduzido.

Observa-se na literatura que a grande maioria das ferramentas que tratam com posicionamento irregular são baseadas em generalizações das estratégias utilizadas na abordagem regular.

Do ponto de vista funcional, duas outras classes podem ser estabelecidas: a classe das ferramentas geradoras e a das ferramentas otimizadoras de posicionamento [SOU 81].

As ferramentas geradoras de posicionamento partem de uma descrição das células e interconexões do circuito e procuram chegar a um posicionamento que venha de encontro aos objetivos pré-definidos. As ferramentas otimizadoras realizam algumas alterações sobre um posicionamento prévio, visando obter valores mais adequados para a função-objetivo.

Visto que a maioria dos geradores de posicionamento existentes produzem resultados satisfatórios em algumas situações e falham drasticamente em outras, não se concebe a ausência de procedimentos de otimização de posicionamento em um sistema de síntese de leiaute que se diga completo.

2.2.1.3 Geração de Posicionamento

As principais estratégias utilizadas por ferramentas geradoras de posicionamento fundamentam-se em geração aleatória, crescimento de aglomerados, partição e técnicas globais.

2.2.1.3.1 Geração Aleatória de Posicionamento

A geração aleatória baseia-se na seleção arbitrária de uma das células do grupo das ainda não submetidas ao processo, e na escolha, também arbitrária, de uma posição livre para ela. Ao fim de cada iteração, quando tem-se um posicionamento completo, este é avaliado, segundo algum critério [GOM 90], e posteriormente comparado à solução anterior, mantendo-se a melhor delas. O melhor posicionamento encontrado após um determinado tempo ou um número de tentativas pré-especificado é adotado como solução.

Esta técnica, apesar de rápida e fácil de programar, fornece resultados raramente satisfatórios, devido ao fato de não considerar informações específicas do circuito. A sua adoção só se justifica no posicionamento de um número pequeno de células ou quando pressupõe-se o uso posterior de uma boa ferramenta de otimização.

2.2.1.3.2 Posicionamento por Crescimento de Aglomerados

A técnica de crescimento de aglomerados é tipicamente uma abordagem ascendente para o problema de posicionamento. Neste caso, parte-se de uma ou várias células sementes e, a cada iteração do procedimento, seleciona-se outra célula do conjunto das não posicionadas e aloca-se uma posição absoluta ou relativa para ela.

A determinação de sementes, quando não realizada pelo projetista, é normalmente baseada na identificação de aglomerados (subcircuitos, pouco conectados a outros subcircuitos, cujas células relacionam-se fortemente entre si) e na escolha das células de maior conectividade dentro de cada aglomerado.

As seleções posteriores envolvem o cálculo de escores para cada célula não posicionada, como medida da conectividade destas com o grupo das células já

posicionadas. O maior escore é aceito e, em caso de empate, critérios que avaliam comprimento de interconexões, possibilidades de justaposição e outros, são adotados.

A alocação de uma posição para a célula escolhida é, naturalmente, baseada na busca de localidade das conexões, podendo ser absoluta - se o posicionamento for realizado sobre uma superfície previamente estabelecida e dividida em posições válidas, ou relativa - se o processo estabelecer simplesmente a vizinhança entre células. Na alocação relativa, caracterizada por informações do tipo a célula X deve ficar à direita de Y, não se pode prever em que direção crescerá o bloco ou próxima de que extremidade ficará uma determinada célula (figura 2.5).

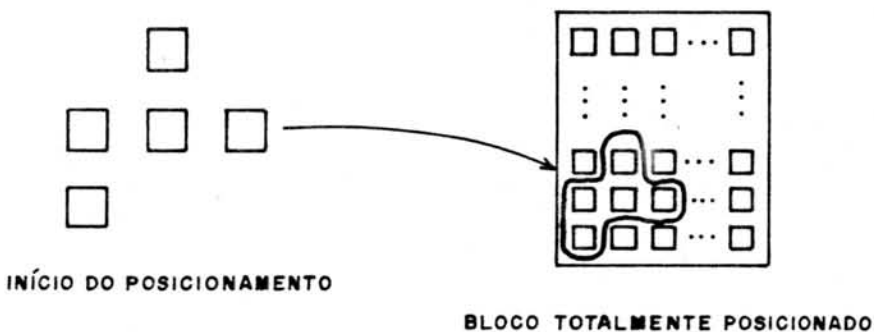
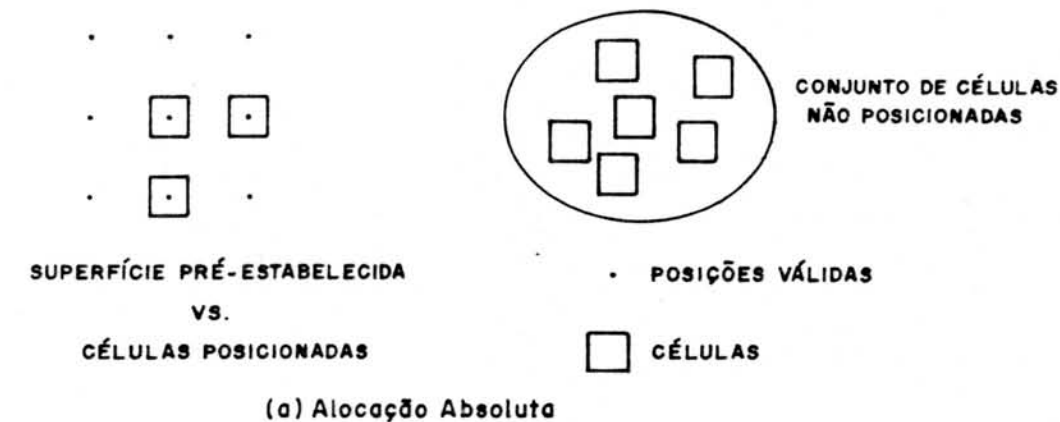


Figura 2.5 - Posicionamento por crescimento de aglomerados.

Um exemplo típico de alocação relativa é o posicionamento das células em linha, seguido de quebras

sucessivas em zigue-zague e conseqüente divisão da linha em fileiras (figura 2.6).

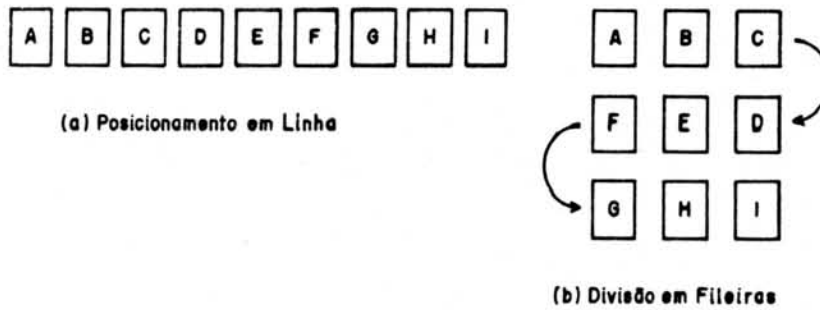


Figura 2.6 - Um exemplo de alocação relativa.

Como no crescimento de aglomerados as decisões são tomadas a partir de informações incompletas de posicionamento (os escores, por exemplo, são calculados sem considerar as seleções que estão por vir), os resultados obtidos são freqüentemente insatisfatórios. As referências [AKE 81], [KAM 82], [KOZ 83], [ROD 86] e [LUB 88a] apresentam exemplos de utilização desta técnica.

2.2.1.3.3 Posicionamento por Partição

Entre tantos subproblemas do processo de concepção de circuitos que se valem da teoria de grafos para sua modelagem [CHE 88], a partição é o exemplo mais significativo. Procedimentos de partição, quando aplicados ao posicionamento de células, configuram uma abordagem descendente de tratamento do problema.

Algoritmos baseados em partição dividem o circuito em conjuntos de células (normalmente dois) e tentam minimizar o número de conexões entre estes conjuntos pelo intercâmbio de células. Esta estratégia é aplicada recursivamente aos subcircuitos gerados, até que se obtenha partições equilibradas com um número relativamente pequeno de células.

Os fundamentos práticos da partição de grafos tiveram origem em 1970, quando Kernighan e Lin propuseram uma heurística para o ataque à complexidade combinatória do

problema [KER 70]. A idéia da heurística é que, partindo-se de dois subgrafos quaisquer de um grafo, determina-se um conjunto de nodos de cada subgrafo, tal que, transferindo-se cada um destes conjuntos para o outro subgrafo, se obtenha uma redução na inter-relação entre as partições do grafo (figura 2.7). O procedimento é repetido até que não seja mais possível a minimização do número de arcos comuns às duas partições.

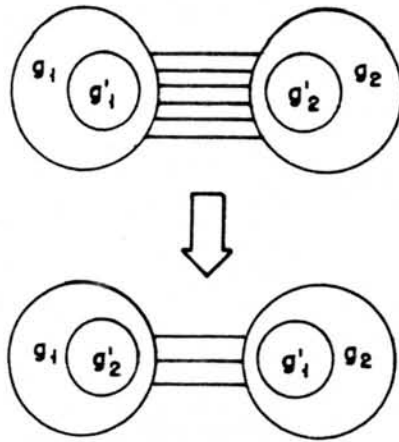


Figura 2.7 - A idéia da heurística Kernighan-Lin.

No tratamento de circuitos, no entanto, a heurística Kernighan-Lin falha drasticamente na modelagem de redes múltiplas por grafos. Isto decorre do fato de uma rede múltipla (rede compartilhada por mais de duas células), que é implementada com um único fio, ser representada por um subgrafo completo (subgrafo onde cada nodo se liga a todos os outros nodos) onde todos os arcos possuem pesos comparáveis àquele que corresponde à ligação física de somente duas células (figura 2.8). Schweikert e Kernighan propõem em [SCH 72a] uma nova estratégia para o cálculo de ganho no intercâmbio de células, de forma a tratar as redes múltiplas de maneira realística.

Fiduccia e Matheyses aprimoram a aplicação de partição de grafos à partição de circuitos pela adoção de um critério de balanceamento mais apropriado (baseado em tamanho das partições, e não mais em cardinalidade), e pelo uso de técnicas de programação - como redundância e

ordenação das estruturas de dados, que resultam em um tempo linear de processamento [FID 82].

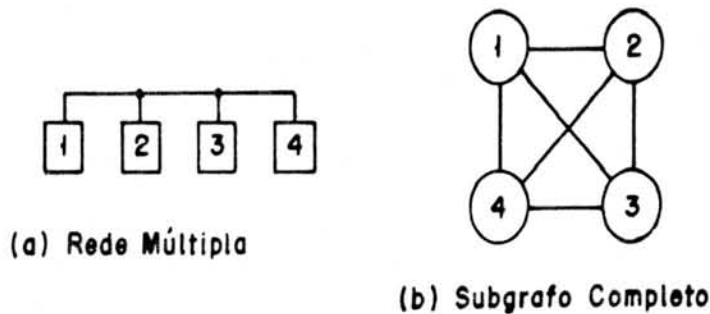


Figura 2.8 - Representação de redes múltiplas em grafos.

As heurísticas de particionamento citadas acima são apresentadas e discutidas em mais detalhes em [LUB 88b] e [LUB 88d].

A partir de 1982, então, proliferaram diferentes formas de aplicação da heurística Fiduccia-Matheyses e propostas pouco drásticas de modificações do procedimento, almejando soluções cada vez mais adequadas às diferentes subclasses do problema.

A utilização de tais heurísticas no particionamento de circuitos pode se dar segundo diferentes formas de implementação, como por exemplo, em quadratura, bissecção ou fatias. Independente da abordagem, observa-se a definição sucessiva de linhas imaginárias que cortam uma estrutura física e o direcionamento de esforços no sentido de minimizar o número de conexões que cruzam tais linhas. A figura 2.9 apresenta as três formas de implementação citadas acima, estabelecendo a ordem em que as linhas imaginárias são definidas em cada caso (a primeira linha corresponde sempre a P1).

A grande vantagem advinda da aplicação da técnica de partição ao posicionamento é que, à medida que o circuito como um todo é considerado desde o início do processo, os resultados obtidos são expressivamente melhores do que em abordagens ascendentes como o crescimento de aglomerados. O preço que se paga pela

eficiência é um maior tempo de processamento [MUR 80].

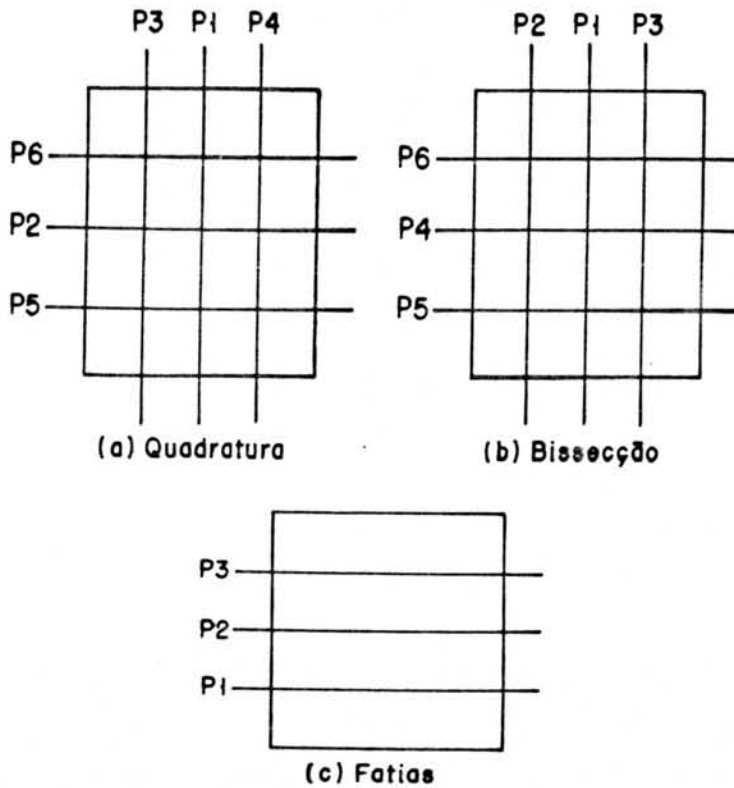


Figura 2.9 - Formas de particionamento.

Os leiautes obtidos via partição caracterizam-se, principalmente, pela distribuição efetiva da densidade de conexões. No entanto, estratégias baseadas nesta abordagem tendem a produzir partições grosseiramente desparelhas (em termos de área), quando a variação nos tamanhos das células é muito grande.

2.2.1.3.4 Posicionamento Global

Dentre as principais estratégias utilizadas para a geração de posicionamento nos resta apresentar, ainda, aquelas baseadas em técnicas globais.

Vimos, anteriormente, que o movimento das células em procedimentos baseados em crescimento de aglomerados é individual e seqüencial, enquanto que em procedimentos baseados em partição, primeiro realiza-se uma divisão das células em conjuntos menores e depois o movimento se desenrola por intercâmbio de células aos pares. Pois o que caracteriza o uso de técnicas globais é, justamente, a

capacidade de movimentação simultânea de todas as células. Até pelo caráter global da abordagem, acredita-se que a sua adoção resulte em posicionamentos extremamente superiores aos produzidos pelas outras técnicas. No entanto, principalmente por sua complexidade, o tema tem sido abordado por poucas publicações, o que sugere que muita pesquisa tenha que ser realizada até que se possa avaliar a efetividade de uso e o campo de aplicação de tais técnicas. Mais informações sobre esta técnica podem ser encontradas em [PRE 88].

2.2.1.4 Otimização de Posicionamento

De uma forma bastante semelhante aos procedimentos de geração, as técnicas de otimização de posicionamento dividem-se em três fases principais: seleção e movimento de células e avaliação de posicionamento. A diferença básica entre as duas abordagens reside no fato de que, em procedimentos de otimização, células já posicionadas é que estão sujeitas às fases de seleção e de movimento.

Na busca de otimização, então, o primeiro passo consiste na seleção de algumas células candidatas ao reposicionamento. Esta seleção pode ser inteligente, se fundamentada na adequacidade do local atribuído a cada célula, ou aleatória, se baseada em sorteio ou em uma seqüência pré-definida.

A partir da seleção, algumas das células, ou mesmo todas, são movidas para posições de outras células ou para locais deixados livres pelo procedimento de geração de posicionamento. A fase de movimento pode pressupor o intercâmbio de posições de células aos pares, ou um reposicionamento em cadeia até que todas as células sejam acomodadas - método este conhecido na literatura como relaxação. Naturalmente, o reposicionamento só é factível se cada célula puder se encaixar perfeitamente na sua

posição de destino.

A seguir, o novo posicionamento é comparado àquele que lhe deu origem e aceito ou não, conforme sejam constatadas melhorias ou perspectivas de otimização em iterações futuras. A comparação deve ser baseada na avaliação tão ou mais precisa dos mesmos parâmetros utilizados na medida de qualidade da solução fornecida pelo gerador de posicionamento.

Vistas as fases características dos procedimentos de otimização, passemos à análise das técnicas mais difundidas, que são: intercâmbio de células e métodos fundamentados em fenômenos físicos.

2.2.1.4.1 Otimização por Intercâmbio de Células

Na técnica de intercâmbio, a célula selecionada forma um par com cada uma das células restantes. Na base da tentativa, realiza-se a troca de posições entre as células que compõem o par e avalia-se o resultado, aceitando-o ou não.

Em função do número de células envolvidas, as combinações de trocas podem explodir exponencialmente. No sentido de reduzir o número de tentativas, freqüentemente adota-se a estratégia de considerar somente a vizinhança das células selecionadas.

[SHI 80] mostra o exemplo de um algoritmo que tenta eliminar o congestionamento do canal de roteamento pelo intercâmbio de células internas à mesma coluna, mas situadas em lados opostos de uma linha que cruza o ponto de densidade máxima de conexões.

2.2.1.4.2 Otimização baseada em Fenômenos Físicos

Os métodos baseados em modelos de fenômenos físicos, por sua vez, fundamentam-se em sistemas de forças

ou recozimento simulado ("simulated annealing").

Os métodos fundamentados em sistemas de forças utilizam um modelo baseado em vetores de forças, tanto para seleção, como para o movimento de células. [SOU 81] apresenta um exemplo que define um modelo onde às células são atribuídas massas e supõe-se que estas estejam flutuando em um óleo pesado. Forças de atração e repulsão configuram então a inter-relação entre células. É possível derivar equações diferenciais para o modelo que podem ser usadas para avaliar estados estáveis (posicionamentos) para o sistema.

Os métodos fundamentados em "simulated annealing", pressupõem que a otimização de posicionamento de células é análoga ao processo de recozimento, no qual um material é derretido e resfriado lentamente até que cristalize em um estado de baixa energia. Os métodos baseados em recozimento partem de um posicionamento inicial aleatório. Quando um posicionamento alternativo é gerado, seu score é avaliado e, se este for menor que o do posicionamento anterior, a nova solução é aceita, ou, se for maior, o movimento é aceito com uma probabilidade que decresce com a aproximação do limite máximo de iterações especificado [PRE 88].

Como se pode observar, estes métodos não suspendem a procura de um melhor posicionamento quando as melhorias temporariamente cessam. Justamente por este fato é que a técnica baseada em recozimento consegue superar mínimos locais e atingir o ótimo global. Muito embora esta forma de otimização gere resultados significativamente superiores às outras técnicas, isto se dá às custas de um elevadíssimo tempo de máquina.

[SEC 85] apresenta um exemplo de otimização de posicionamento baseado na simulação do processo de recozimento.

2.2.2 Orientação de Células

Normalmente a orientação de células é considerada como uma parte do problema de posicionamento. Aqui, prefere-se classificá-la como uma outra classe de problema, cuja aplicação na síntese de leiaute só tem sentido se for realizada imediatamente após o posicionamento de células.

A orientação, então, consiste na determinação de onde, dentro do espaço destinado pelo posicionamento, deve ficar o ponto de origem (canto inferior esquerdo) da célula (figura 2.10). A orientação de células pode pressupor tanto a operação de espelhamento, como a operação de rotação.

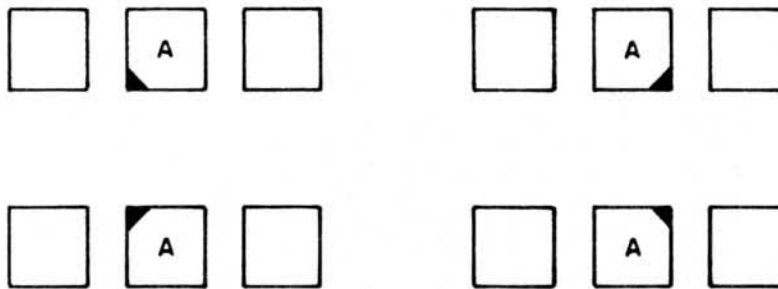
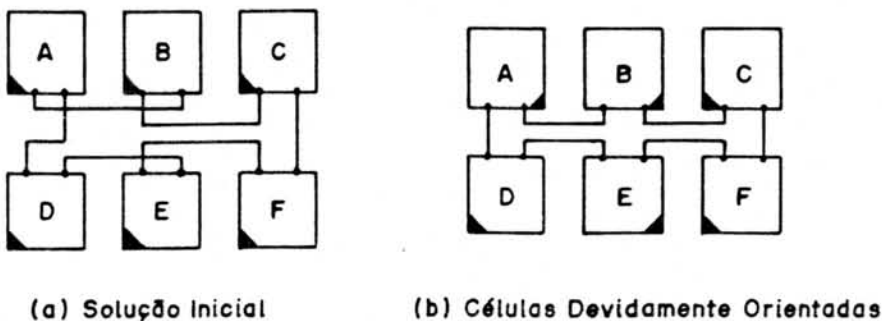


Figura 2.10 - Possibilidades de orientação de uma célula.

A escolha de orientação é, via de regra, baseada na procura da configuração que resulta em um menor comprimento de interconexões. Quanto maior o tamanho das células primitivas, maior é o impacto de uma boa orientação na redução dos comprimentos. Uma orientação adequada pode resultar, também, na redução do número de trilhas necessárias ao roteamento (figura 2.11).



(a) Solução Inicial

(b) Células Devidamente Orientadas

Figura 2.11 - Redução do número de trilhas por espelhamento.

A re-orientação de células pode ser realizada na horizontal, como na figura 2.11, ou na vertical. A consideração da localização das linhas de alimentação dentro da célula pode favorecer uma determinada orientação, ou impedi-la, mesmo que o comprimento de interconexões permaneça grande (figura 2.12). Rotações de 90 graus normalmente são proibidas, à medida que a adoção de tal orientação pode resultar na violação do espaço destinado à célula no bloco (figura 2.13).

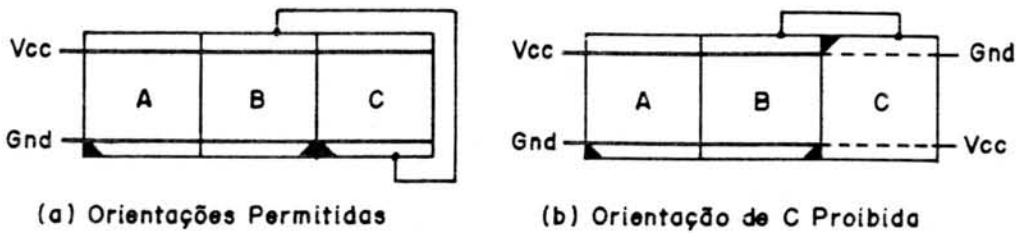


Figura 2.12 - Orientações permitidas e proibidas.

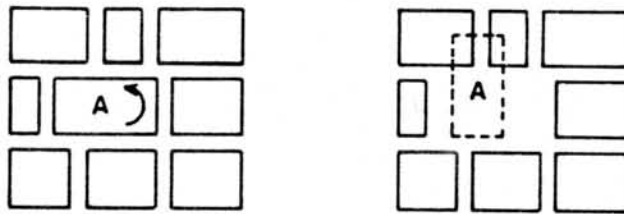


Figura 2.13 - Rotação de 90 graus.

[YAM 88] descreve um método analítico para a determinação ótima da orientação de células, em termos da estimativa do comprimento total das conexões do circuito.

2.2.3 Assinalamento de Conectores

O enfoque deste problema só tem sentido quando, dentre as células primitivas, existem aquelas que possuem alguns pinos (conectores da interface da célula) que sejam funcionalmente equivalentes entre si ou equipotenciais. Entradas de uma célula cujos efeitos sejam simétricos no sinal de saída são ditas pinos funcionalmente equivalentes; conectores que estejam fisicamente ligados no interior da célula são chamados de pinos equipotenciais. Um exemplo é

mostrado na figura 2.14.

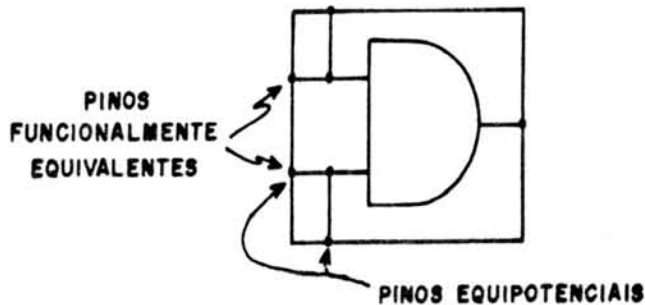


Figura 2.14 - Classes de pinos.

O assinalamento de conectores estabelece que redes devem ser conectadas aos pinos funcionalmente equivalentes e atribui a cada pino equipotencial uma porção da conexão que implementa a rede a ele destinada. A realização do assinalamento não tem efeito sobre a função do circuito, mas altera a sua representação estrutural. Para efeitos de análise e verificação, então, faz-se necessária a atualização da representação original.

A distribuição conveniente de redes e porções de redes entre os conectores da interface da célula não tem, em geral, um grande impacto no comprimento total das conexões. No entanto, seus efeitos podem ser sentidos quando se considera os problemas de cruzamento de redes, de congestionamento e de completude do roteamento. Na figura 2.15, onde os pinos 1 e 3 de A são funcionalmente equivalentes, pode-se avaliar os efeitos do assinalamento.

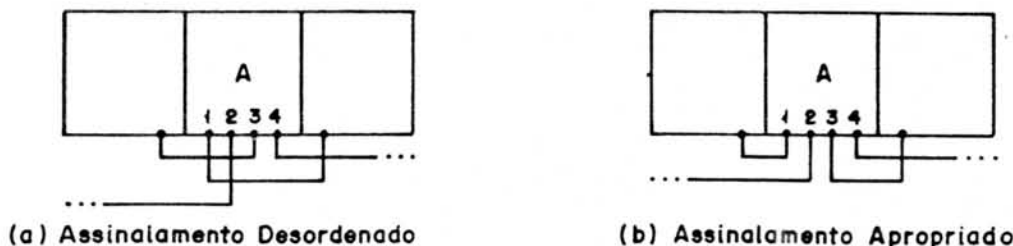


Figura 2.15 - Efeitos do assinalamento de conectores.

As técnicas de mapeamento em círculos concêntricos, assinalamento topológico e o método das nove zonas, discutidos em [PRE 88], são os exemplos mais característicos de ataque ao problema de assinalamento de

conectores.

2.2.4 Roteamento

2.2.4.1 O Problema

Seguindo o posicionamento, a orientação de células e o assinalamento de conectores, o roteamento desponta como a fase da síntese de leiaute responsável pela realização física das conexões entre os pinos das células. A partir do roteamento, então, a personalização do módulo está completa e a sua função perfeitamente estabelecida.

Na geração de módulos regulares, a síntese de leiaute pressupõe, geralmente, a realização do posicionamento e das conexões entre as células de forma concomitante. Isto é possível porque, nestes casos, as conexões, ou são feitas por justaposição, ou através de barramentos cuja localização é previamente estabelecida.

Na geração de módulos em lógica aleatória, no entanto, não se tem conhecimento de nenhuma informação específica sobre a forma como as conexões serão implementadas antes que o posicionamento de células seja realizado. Assim, posicionamento e roteamento são tratados de forma seqüencial, e a iteração no processo de síntese pressupõe a retomada do problema desde as considerações iniciais.

A implementação de conexões em módulos em lógica aleatória consiste na identificação de pares de pinos que serão ligados eletricamente, na determinação de que camadas serão utilizadas na sua realização, no estabelecimento de contatos que liguem fisicamente trechos correspondentes ao mesmo sinal, mas implementados em camadas diferentes, e na escolha do caminho a ser percorrido pela conexão. Todos estes passos são realizados para cada rede que compõe o bloco funcional.

Normalmente, as camadas disponíveis para a realização de conexões em uma tecnologia são polissilício e metal. Para que o roteamento seja feito de uma forma eficiente, a administração destas camadas, em um primeiro instante, deve ser tal que o seu uso seja permitido somente em uma direção, e que as direções correspondentes a cada uma das camadas sejam ortogonais entre si. Assim, segmentos horizontais serão sempre transparentes para segmentos verticais, e vice-versa, e as ligações necessárias entre eles serão identificadas por uma camada de contato (figura 2.16). Quando um segundo nível de metal estiver presente, ou adota-se somente as duas camadas de metal para o roteamento, ou trabalha-se como anteriormente e utiliza-se a nova camada em qualquer uma das direções, valendo-se de sua transparência com relação às outras duas. A tendência é que as camadas de metal sejam ortogonais.

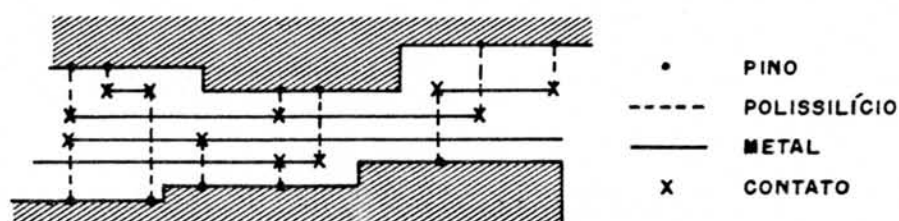


Figura 2.16 - Administração de camadas de roteamento.

Na identificação de pares de pinos de uma mesma rede para a implementação de trechos da conexão pertinente, o critério mais razoável para associação de pinos é a proximidade entre eles. Na figura 2.17a os pares A_1-A_2 e A_2-A_3 definem dois trechos consecutivos para a conexão relativa à rede A que, após implementados, garantem a realização física da rede A na íntegra. A solução proposta na figura 2.17b (pares A_1-A_2 e A_1-A_3) só é admissível se houver obstáculos na região de roteamento que impeçam a ligação de outra forma.

A ordem em que as conexões são realizadas pode ser determinante para a completeza do roteamento, pois a alocação de um determinado espaço para a ligação de dois

pinos pode significar o bloqueio da passagem para outras conexões. Considere o exemplo da figura 2.18, onde existem duas trilhas (ou linhas) para roteamento e a prioridade maior para alocação corresponde à trilha 1. A seqüência ADBC de realização garante uma solução para o problema; já a seqüência DACB impede a passagem das conexões A e B.

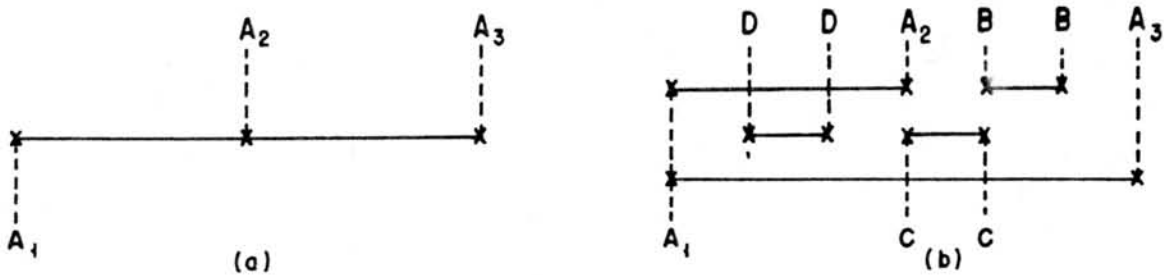


Figura 2.17 - Identificação de pares de pinos.

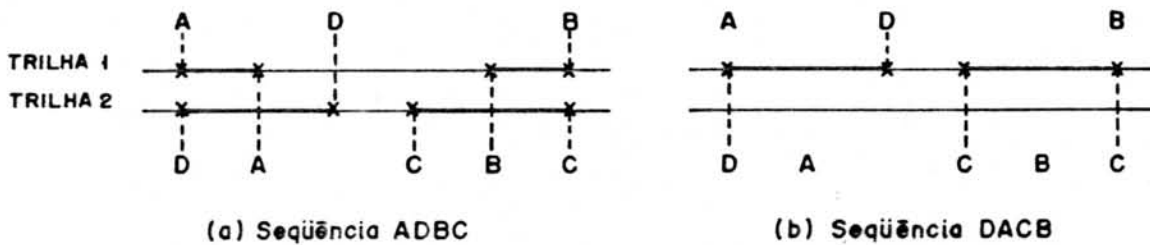


Figura 2.18 - Ordem de realização das conexões.

A partir da administração das camadas de roteamento, da identificação dos pares de pinos a serem ligados e da consideração de uma determinada ordem para a realização das conexões, a escolha do caminho entre pinos de uma rede para sua implementação física pode ser feita de várias formas diferentes. Pois é justamente este procedimento que diferencia os inúmeros algoritmos para roteamento constantes na literatura. Tal tema será abordado na seção 2.2.4.3.

2.2.4.2 Classes de Roteamento

As ferramentas utilizadas para roteamento em circuitos integrados podem ser classificadas em três grupos distintos: roteadores globais, roteadores detalhados

e roteadores especializados.

Os roteadores globais são responsáveis pela divisão do problema maior de roteamento de todo o circuito em subproblemas de menor complexidade, que serão tratados pelas duas outras classes de roteadores.

Ao roteador global de um sistema de síntese de leiaute cabe estabelecer em que regiões destinadas ao roteamento entre blocos funcionais serão realizadas as conexões que configuram cada rede do circuito. Esta planificação geral, que antecede a implementação física das conexões, tem como objetivos reduzir a área do chip, controlar o comprimento das ligações, garantir a realização completa do roteamento e balancear o congestionamento das conexões sobre as regiões de roteamento.

Pelo caráter global desta classe de roteamento, o problema por ela atacado é típico do planejamento topológico. No entanto, mesmo na geração de módulos, o roteamento global assume um papel importante. Quando utiliza-se as abordagens de projeto "Gate-Array" ou "Standard Cell" [WES 85] para a síntese de blocos funcionais, por exemplo, a definição de qual região de roteamento conterá que trechos de cada conexão é fator determinante da área de tais regiões ou da completeza do roteamento para regiões de tamanho fixo. Nestas abordagens, as células são dispostas em fileiras e regiões entre fileiras são dedicadas a conexões. Como, geralmente, pinos equipotenciais estão disponíveis nas extremidades superior e inferior de cada célula, são várias as possibilidades de implementação de conexões (figura 2.19).

O problema de roteamento global, em toda sua amplitude, é apresentado e discutido em [PER 90]. Um exemplo de estratégia para roteamento global de blocos em "Standard Cell" aparece em [CON 88].

À classe de roteadores detalhados pertencem

aquelas ferramentas capazes de estabelecerem, dentro de uma região limitada, os caminhos a serem seguidos por cada conexão pertinente. Assim, a distribuição de fios e contatos, para a implementação física das ligações, é responsabilidade do roteador detalhado. Esta classe, de particular interesse para este trabalho, é abordada em mais detalhes na seção seguinte.

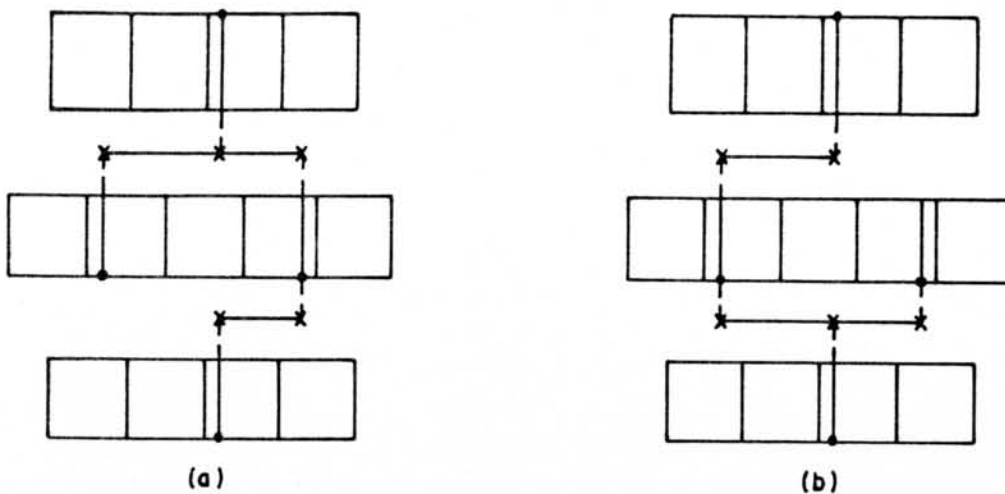


Figura 2.19 - Roteamento global em "Standard Cell".

Por fim, a resolução de problemas bastante específicos fica a cargo dos roteadores especializados. Um exemplo característico de problema de interesse desta classe é a distribuição das linhas de alimentação no circuito. [PRE 88] tece algumas considerações sobre este problema.

2.2.4.3 Roteamento Detalhado

Roteadores detalhados podem ser basicamente de dois tipos: roteadores de propósito geral ou roteadores de canal.

2.2.4.3.1 Roteadores de Propósito Geral

Os roteadores de propósito geral implementam conexões, uma a uma, sobre regiões, não necessariamente retangulares, onde obstáculos à passagem de fios podem estar presentes.

Neste caso, as regiões de roteamento são modeladas por um reticulado, cujos pontos são tais que as regras de largura e espaçamento entre camadas são obrigatoriamente respeitadas. Obstáculos ao roteamento são representados por pontos bloqueados no reticulado. A ligação entre dois pinos que compartilham a mesma rede é identificada por uma seqüência de pontos adjacentes entre os pontos de origem e destino da conexão.

O algoritmo de Lee [LEE 61] é a técnica mais conhecida dentre aquelas utilizadas em roteadores de propósito geral. Esta técnica pressupõe a expansão do roteamento a partir do ponto de origem, através da propagação de caminhos em todas as direções permitidas. A cada nova fronteira estabelecida a propagação é novamente realizada, até que se encontre o ponto de destino (figura 2.20). O algoritmo garante que, se pelo menos um caminho existir, este será encontrado e será o menor possível. No entanto, nada assegura que cada conexão seja realizada de forma a não impedir a passagem das subseqüentes.

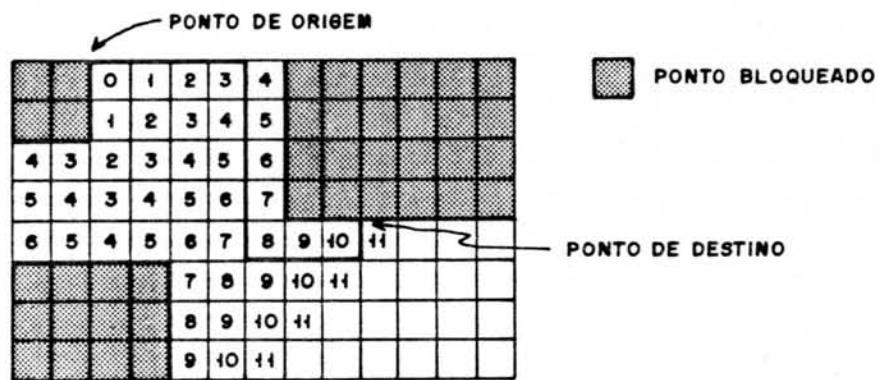


Figura 2.20 - Algoritmo de Lee: propagação de caminhos.

As maiores desvantagens do emprego da técnica proposta por Lee residem na necessidade de muita memória para armazenamento de informações e no elevado tempo de execução do procedimento. Duas formas de vencer tais limitações podem ser: a utilização de aceleradores em "hardware", como proposto em [AUD 87], ou a utilização de extensões do algoritmo que trabalham com seqüências de

linhas, ao invés de pontos.

Os algoritmos que lidam com seqüências de linhas trabalham paralelamente a partir dos dois pinos que devem ser conectados. Para cada pino são projetados segmentos, tanto na direção horizontal, quanto na vertical, e a cada obstáculo atingido por estes segmentos são definidos pontos de escape. A partir destes pontos se procede da mesma forma, recursivamente, até que dois segmentos de origens diferentes se cruzem. Quando isto ocorre, existe uma seqüência de segmentos que caracteriza um caminho entre os dois pinos (figura 2.21).

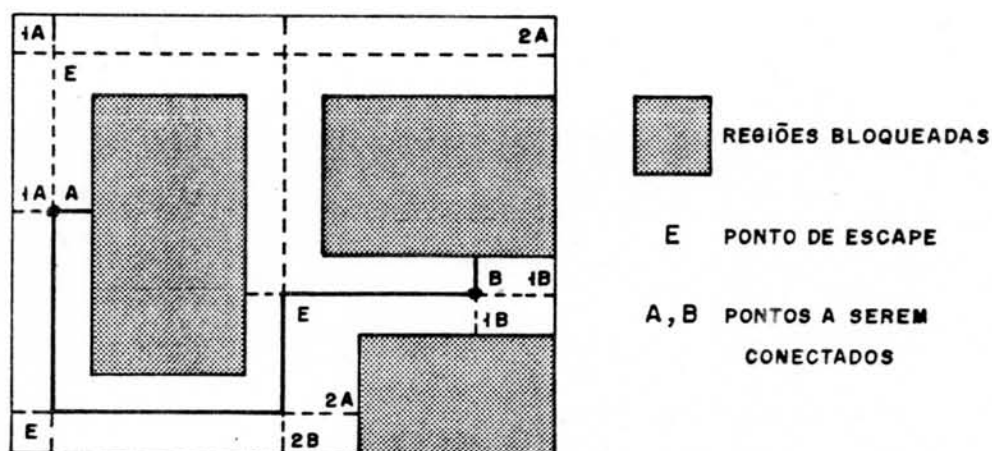


Figura 2.21 - Roteamento por projeção de linhas.

Dois exemplos de uso desta técnica são apresentados em [MIK 68] e [HIG 69]. O que diferencia uma abordagem da outra é o processo de escolha dos pontos de escape.

2.2.4.3.2 Roteadores de Canal

Antes de passarmos à análise das principais estratégias utilizadas pelos roteadores de canal - o segundo grupo de ferramentas pertencentes à classe de roteamento detalhado, se faz necessária a introdução de algumas definições e conceitos básicos.

Um canal de roteamento é definido como uma região

retangular caracterizada por listas ordenadas de pinos associadas às extremidades superior (topo), inferior (base), esquerda e direita. Considera-se que entre extremidades opostas o espaçamento entre pinos seja padrão e denota-se com 0 a inexistência de pino em uma determinada linha ou coluna (figura 2.22a).

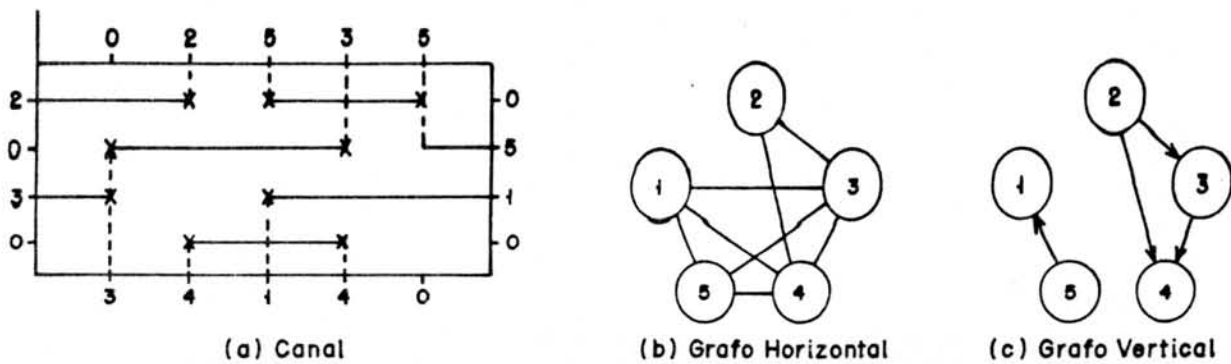


Figura 2.22 - Roteamento de canal.

Da análise dos trechos horizontais das redes contidas no canal da figura 2.22a, podemos observar que trechos de redes diferentes que possuam em comum pelo menos uma coluna não podem ser implementados na mesma linha. Este tipo de restrição é modelado por um grafo horizontal, cujos nodos representam as redes e arcos ligando dois nodos caracterizam o fato das redes associadas possuírem trechos com colunas em comum (figura 2.22b).

Se observarmos, então, os trechos verticais das redes contidas no canal da figura 2.22a, concluímos que trechos verticais de redes diferentes que compartilham a mesma coluna não se sobrepõem somente quando o trecho horizontal atribuído ao pino superior da coluna é colocado em uma linha acima do trecho horizontal atribuído ao pino inferior da coluna. Esta restrição é modelada por um grafo vertical dirigido, cujos nodos representam as redes e arcos dirigidos de um nodo a outro significam que a rede de origem do arco deve ser colocada acima da rede de destino do mesmo arco (figura 2.22c).

A densidade de um canal é dada pelo menor número

de linhas (ou trilhas) necessárias para o roteamento deste, valor este obtido a partir do número de clique do grafo horizontal. O número de clique corresponde ao número de nodos do maior subgrafo completo de um grafo. No exemplo da figura 2.22, a densidade do canal é 4, como podemos concluir a partir do grafo horizontal.

Convém mencionar, neste ponto, que conflitos expressos por um ciclo no grafo vertical só podem ser resolvidos por "doglegs", ou seja, pela quebra de um dos sinais do ciclo em mais de uma trilha (figura 2.23).

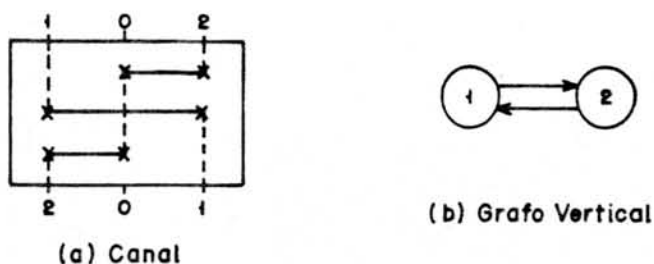


Figura 2.23 - "Dogleg".

Utilizando estes conceitos básicos, então, descreve-se, a seguir, os principais algoritmos aplicados ao roteamento de canal, quais sejam: "left-edge", guloso (do inglês "greedy") e hierárquico. Os dois primeiros tipos de algoritmos compartilham a característica de realização das conexões entre pinos simultaneamente, ao passo que o último realiza as conexões uma a uma, como nas técnicas de propósito geral.

Algoritmo "Left-Edge". O algoritmo "left-edge" básico [HAS 71] realiza o roteamento do canal por trilha, partindo do pressuposto de que sinais multipinos podem ser decompostos numa série de subsinais de dois pinos. Como considera-se que restrições verticais estejam ausentes, esta técnica gera soluções com um número de trilhas igual à densidade do canal.

Primeiramente, os sinais e subsinais são ordenados segundo seus pinos posicionados mais à esquerda.

Percorrendo esta lista, então, o algoritmo completa uma trilha do canal a cada iteração, a partir da seleção e agrupamento dos sinais que não se sobrepõem (figura 2.24). Estabelecidos os segmentos horizontais, os segmentos verticais são facilmente acrescentados ao canal.

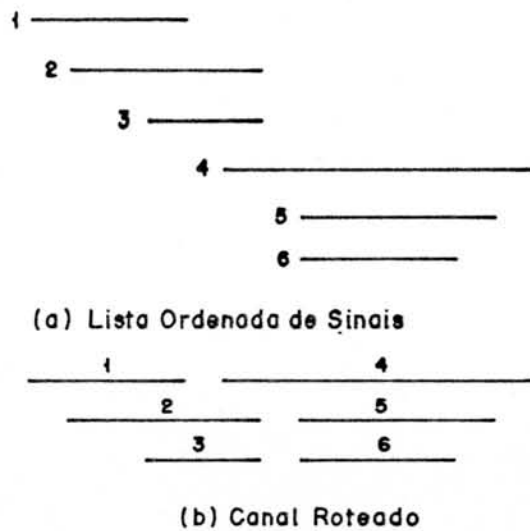


Figura 2.24 - A idéia do algoritmo "left-edge".

Como na maioria dos casos práticos existem restrições verticais, estas são consideradas na extensão do algoritmo "left-edge" proposta em [KER 73]. No entanto, passam a ser obtidos resultados com um número de trilhas excessivo na nova abordagem.

[DEU 76] vai mais longe e propõe o uso de "doglegs" no roteamento, no sentido de reaproximar o número de trilhas necessário da densidade real do canal (figura 2.25).

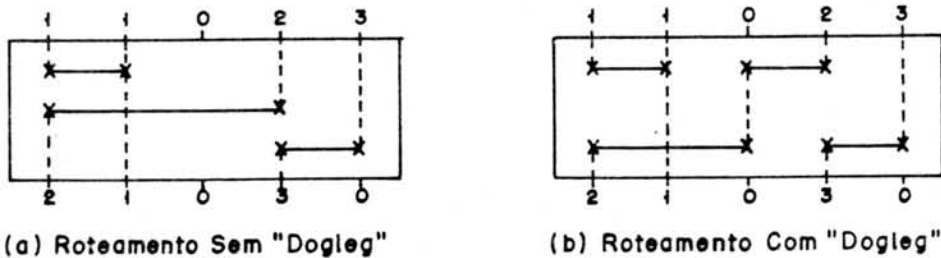


Figura 2.25 - Benefícios do uso de "doglegs".

Uma estratégia diferente para lidar com restrições verticais é apresentada em [REE 85]. Neste

caso, após o estabelecimento dos segmentos horizontais, a ocorrência de violação de restrição imposta pelo grafo vertical é tratada através do uso de padrões alternativos de roteamento. A figura 2.26 mostra um exemplo de padrão de roteamento utilizado.

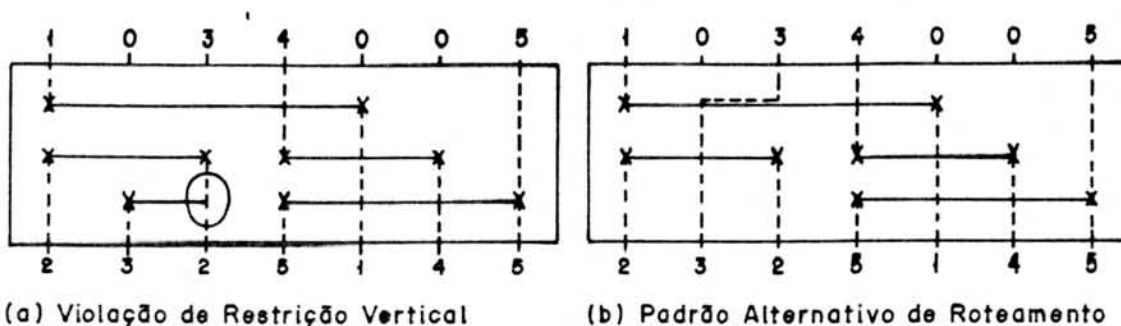


Figura 2.26 - Cumprimento de restrições verticais.

Algoritmo Guloso. Enquanto as técnicas baseadas no algoritmo "left-edge" realizam o roteamento de canal trilha a trilha, os algoritmos do tipo guloso [RIV 82] trabalham coluna a coluna. Estes algoritmos percorrem o canal da esquerda para a direita e atribuem a cada coluna um conjunto de segmentos verticais a partir de heurísticas simples. Para a realização das conexões, permite-se tanto a troca de trilhas, como a inclusão de novas colunas nas extremidades do canal.

O primeiro passo, em um algoritmo guloso, é introduzir no canal de roteamento os pinos pertencentes à coluna que se trata. Cada pino é conectado a uma trilha livre ou a uma trilha que já contenha a rede à qual ele pertence. Caso isto não seja possível, uma nova linha é adicionada ao canal.

O passo seguinte procura estabelecer segmentos verticais que conectem dois trechos de uma mesma rede que corram em trilhas diferentes. Caso isto não seja possível, tenta-se a redução da distância entre os trechos pela troca de trilha.

Finalmente, antes de passar à análise da coluna

seguinte, tenta-se mover cada rede para uma trilha mais próxima da extremidade do canal à qual pertence o próximo pino da rede a ser conectado.

A figura 2.27 mostra o exemplo de um canal roteado segundo a técnica descrita acima.

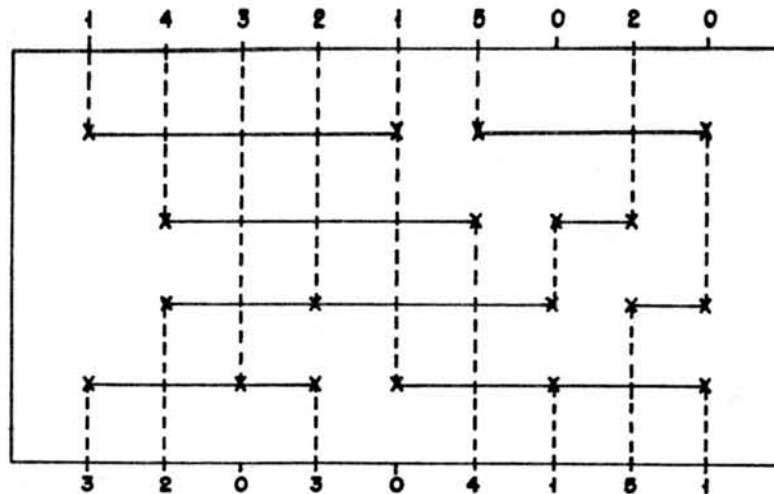


Figura 2.27 - Roteamento coluna a coluna.

Embora a utilização de algoritmos gulosos geralmente resulte em canais com menor número de trilhas do que gerados por algoritmos do tipo "left-edge", o número de contatos e de curvas produzidos é significativamente maior.

Algoritmo Hierárquico. O último tipo de roteadores de canal de interesse utiliza o algoritmo proposto por Burstein e Pelavin em [BUR 83a].

Este algoritmo trabalha de forma hierárquica sobre o canal, partindo de uma divisão deste em duas regiões horizontais. Cada conexão é então atribuída a uma destas regiões e as redes são roteadas uma após a outra. Quando todas as ligações estão concluídas, uma etapa de roteamento aleatório é realizada, no sentido de combater o congestionamento e de seguir às fases seguintes sem adicionar novas trilhas ao canal.

Em função da necessidade de um número maior de trilhas para o roteamento, cada região é recursivamente

dividida em duas subregiões, até que todas as conexões possam ser realizadas.

Segundo os resultados apresentados em [BUR 83b], o algoritmo hierárquico tende a produzir roteamentos muito próximos da densidade do canal.

Algoritmo "River-Routing". Um caso especial de roteamento de canal é aquele conhecido na literatura como "river routing". O problema de "river routing" é caracterizado por uma ordenação tal dos pinos no canal, que o roteamento pode ser realizado em uma única camada [PRE 88]. Uma vez que a ordenação de pinos pressuposta para o tratamento de tal problema é dificilmente obtida a partir do posicionamento de células de módulos em lógica aleatória, as estratégias referentes a esta subclasse de roteamento não serão abordadas neste trabalho.

2.2.4.4 Refinamento do Roteamento

Concluído o roteamento de uma determinada região, verifica-se freqüentemente que o número de contatos entre os segmentos horizontais e verticais é bastante elevado. Este fato leva a uma degradação das características elétricas do roteamento.

A minimização de contatos pode ser realizada sempre que um determinado segmento horizontal, que conecta dois outros segmentos verticais, não é interceptado por nenhum segmento vertical (figura 2.28a). Da mesma forma, a minimização é possível quando um segmento vertical, que conecta dois segmentos horizontais, não é cruzado por segmentos horizontais. Com estas condições satisfeitas, o segmento não interceptado será realizado na camada designada para segmentos ortogonais a ele. Conseqüentemente, dois contatos serão eliminados da conexão (figura 2.28b).

Uma vez que a minimização de contatos normalmente ocorre para segmentos pequenos, não há a necessidade de avaliação da resistência e da capacitância do trecho na camada substituta em relação ao mesmo trecho implementado na camada substituída.

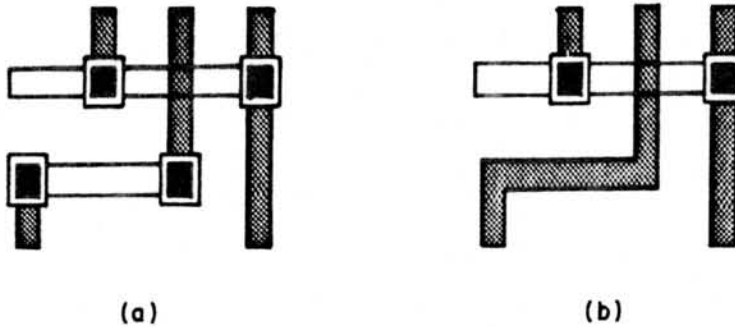


Figura 2.28 - Minimização de contatos.

Em [PER 89b] descreve-se um roteador de canal que, na fase de pós-processamento, realiza a minimização de contatos. Alguns resultados são apresentados.

Uma técnica de refinamento de caráter mais global é a estratégia de compactação do canal de roteamento. A idéia é reduzir a altura do canal através do encurvamento de trilhas, da realização de pequenos deslocamentos de contatos e da sobreposição de camadas não conflitantes. Todos estes passos pressupõem que a minimização de contatos tenha sido efetuada em uma fase anterior. Segundo [HSU 87], que inclui em seu sistema de síntese de leiaute uma fase de compactação de roteamento, os resultados podem atingir uma redução de até 20% em área.

3 A METODOLOGIA TRANCA

Considerando-se a síntese de leiaute de circuitos integrados, verifica-se que só é possível abstrair a geometria interna das células, se for adotado um padrão topológico de geração, tal que a composição de células seja feita a partir de um mínimo de informação que caracterize cada primitiva. Da mesma forma, a composição deve ser realizada segundo uma topologia que seja consequência daquela adotada na geração de célula e que garanta a ausência de conflitos entre os dois padrões.

Se, por um lado, a padronização reduz a complexidade da concepção automática, por outro, ela tende a desconsiderar algumas das soluções inteligentes outrora adotadas por projetistas. No momento, então, que se pensa em geração de módulos, torna-se imprescindível a análise de leiautes concebidos "manualmente", para levantamento de características topológicas que orientem adequadamente esta padronização. O objetivo é reunir a experiência de especialistas, em um único padrão, que resulte na sistematização eficiente da síntese de leiaute.

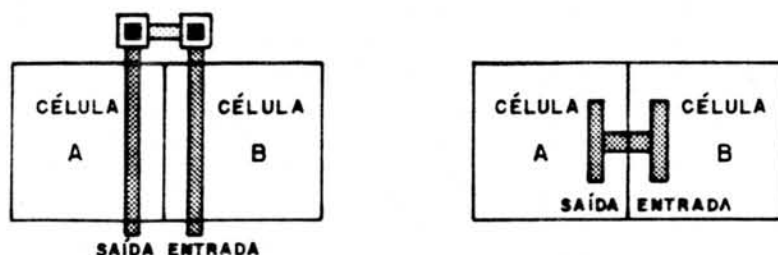
Seguindo esta linha, a metodologia de concepção de leiaute TRANCA ("TRANSPARENT Cell Approach") originou-se a partir de estudos estatísticos realizados sobre blocos em lógica aleatória de vários microprocessadores comerciais projetados "à mão" [REI 83]. Da análise de tais blocos funcionais, levantou-se algumas características consensuais, que comprovadamente são responsáveis pela compactação de seus leiautes, e apresentou-se uma proposta de administração destas características, de forma a obter-se resultados semelhantes para módulos gerados automaticamente [REI 87].

Como no caso de circuitos VLSI a área ocupada pelo roteamento é freqüentemente equivalente à área dos blocos funcionais (com uma tendência de superá-la), a

realização de conexões sobre as áreas ativas (transistores) resulta numa redução drástica da superfície necessária à implementação de cada módulo e do leiaute como um todo [ANC 82]. Esta filosofia garante a eficiência da metodologia TRANCA, quando comparada à abordagem "Standard Cell" convencional [WES 85], onde as conexões são implementadas em zonas dedicadas ao roteamento, situadas entre fileiras de células.

Assim como a abordagem "Standard Cell", a metodologia TRANCA é do tipo "polycell", ou seja, as células são posicionadas lado a lado, formando fileiras com altura fixa e com larguras aproximadamente iguais.

Na metodologia TRANCA, conexões entre células adjacentes podem ser realizadas por justaposição, ao passo que em "Standard Cell", os conectores são duplicados no topo e na base das células e as ligações só são implementadas em canais de roteamento. Conexões por justaposição são menos longas e resultam em aproveitamento efetivo de área e em redução de tempos de atraso (figura 3.1).



(a) Conexão em Canal de Roteamento (b) Conexão por Justaposição

Figura 3.1 - Conexão entre células adjacentes.

Passando à análise mais detalhada dos princípios da metodologia TRANCA, as seções seguintes apresentam seus quatro pontos fundamentais, quais sejam:

- estrutura de banda;
- maleabilidade;
- transparência horizontal e vertical; e,
- gerenciamento de trilhas.

3.1 Estrutura de Banda

As linhas de alimentação são implementadas como dois pentes com dentes intercalados (figura 3.2a). Cada faixa delimitada entre uma linha de VCC e outra de GND define uma banda. Internamente a cada banda, trilhas de metal paralelas às linhas de alimentação são utilizadas para a realização de conexões locais e externas (figura 3.2b). Duas trilhas adjacentes distanciam-se de um passo mínimo de metal, isto é, somente contatos alternados são permitidos (figura 3.3). Os transistores são projetados sob estas trilhas de metal, utilizando somente as camadas de difusão e polissilício.

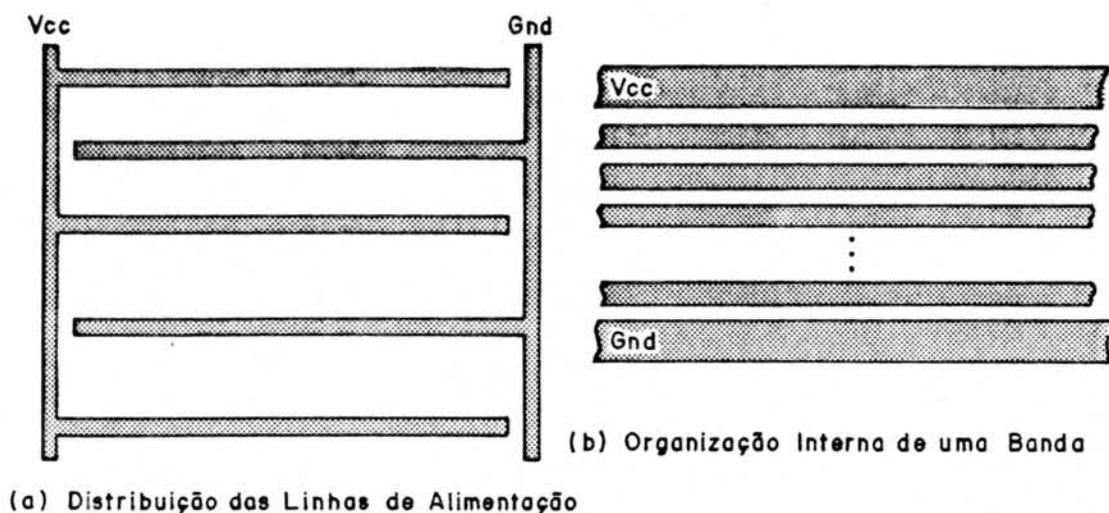


Figura 3.2 - Estruturação em bandas.

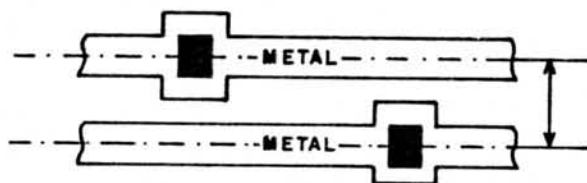


Figura 3.3 - Passo de metal.

A estrutura de banda é típica de blocos em lógica aleatória de circuitos comerciais projetados "à mão". Estudos estatísticos apresentados em [REI 83] demonstram que o número de trilhas por banda pode variar num mesmo circuito. No entanto, mais de 50% das bandas apresentam um

ou dois números diferentes de trilhas. Isto sugere a adoção de um valor médio para o projeto de uma biblioteca de células que integre uma ferramenta de síntese automática de leiaute.

No caso de circuitos NMOS, o número médio de trilhas por banda encontrado foi 7. Para circuitos CMOS, como o número de conexões locais cresce devido à necessidade de pontes entre os drenos de transistores do tipo p e do tipo n, o número de trilhas sugerido é 10 [REI 87].

Em função da complexidade do bloco funcional em desenvolvimento, as células da biblioteca podem ser projetadas sob uma estrutura de banda simples e utilizando um número de trilhas superior aos valores mencionados, ou sob uma estrutura de banda dupla (limitada por duas linhas de GND e com uma linha de VCC ao centro, por exemplo) conforme seu tamanho e restrições topológicas sugiram. [CAR 87] apresenta o desenvolvimento de uma biblioteca de células que utiliza uma estrutura de banda dupla.

3.2 Maleabilidade

A maleabilidade de um módulo ou de uma célula é definida como a sua capacidade de se adaptar a novas restrições de forma, pela modificação de sua organização interna.

Segundo sua maleabilidade, os blocos funcionais podem ser classificados de duros a moles. Enquanto um bloco do tipo ROM pode assumir somente algumas organizações internas permitidas devido à rigidez de sua topologia, blocos em lógica aleatória são suficientemente flexíveis para se adaptarem a várias formas diferentes de implementação [REI 85].

Em se tratando de uma célula, seus limites de maleabilidade são ditados por situações em que sua dimensão

vertical não pode mais ser reduzida a partir do aumento de sua dimensão horizontal, e vice-versa [REI 87].

Na metodologia TRANCA, a forma dos módulos, em uma direção, é função do número de bandas, e, na direção ortogonal, é função da largura das células que compõem as bandas e dos recursos utilizados para realização de conexões entre bandas.

A organização interna das células, por sua vez, é função do número de trilhas por banda e pressupõe o aproveitamento máximo da dimensão vertical. A escolha do número de trilhas por banda, então, deve considerar também a maleabilidade de cada célula da biblioteca e o potencial de roteamento necessário à realização de conexões internas à banda.

3.3 Transparência

A transparência de uma célula ou de um bloco funcional, em uma determinada direção, é definida como a relação entre o número de conexões externas que podem atravessar a célula ou o bloco nesta direção, e o número total de linhas disponíveis para roteamento local e externo na direção considerada.

A transparência horizontal, na direção do passo de metal 1, refere-se, então, ao número de trilhas da estrutura de banda que estão disponíveis para passagem de conexões sobre a célula ou sobre o bloco.

A transparência vertical, por sua vez, relaciona-se ao número de linhas verticais à estrutura de banda que estão livres para conexões entre bandas não adjacentes. As camadas utilizadas para roteamento vertical podem ser polissilício, difusão ou um segundo nível de metal.

Sem dúvida alguma, o fato de se efetuar as interconexões sobre as células, valendo-se do conceito de

transparência horizontal, resulta em uma redução considerável na área total de implementação. Observe a figura 3.4, onde 3 leiautes diferentes para um flip-flop RS são apresentados. Na figura 3.4a, a célula foi projetada sob uma estrutura de banda com 10 trilhas (fora as linhas de alimentação), das quais 6 restam para passagem de conexões. Na figura 3.4b, a mesma célula foi desenhada com compactação máxima na altura. Como a transparência horizontal desta versão é nula, a previsão de 6 trilhas para roteamento externo à célula resulta em uma área total (transistores+conexões) superior à da figura 3.4a. Já quando se concebe a mesma célula com transparência horizontal nula e de forma a se obter uma área equivalente à da figura 3.4a, verifica-se que o potencial de roteamento fica reduzido para 4 trilhas (figura 3.4c).

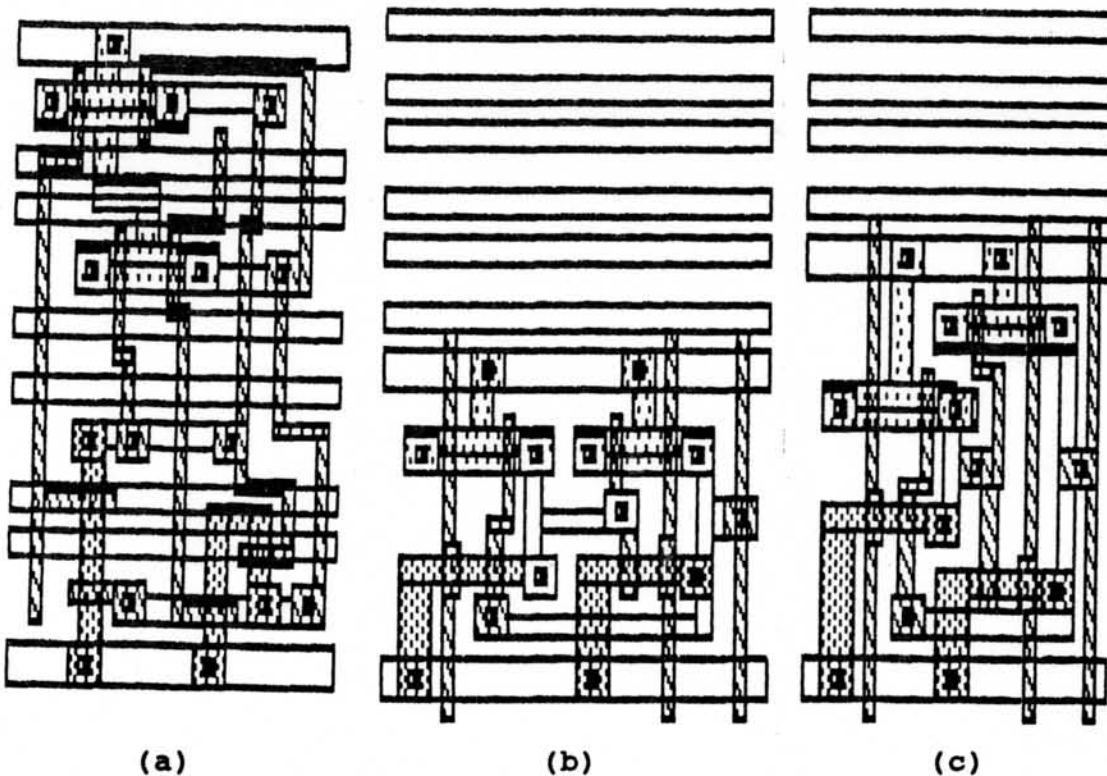


Figura 3.4 - Opções de leiaute para uma célula de memória.

Dois aspectos são relevantes para a administração eficiente dos recursos de transparência horizontal no módulo: a transparência individual de cada célula e a

transparência de um grupo de células adjacentes ou a transparência da banda como um todo.

Estudos sobre a transparência de células de circuitos reais [REI 83] resultaram em um intervalo de valores que oscilam entre 63 e 77%, independente do número de trilhas da estrutura de banda. Conclui-se, desta forma, que uma transparência média de cerca de 70% trata-se de uma boa escolha para a geração de célula.

O segundo aspecto da transparência horizontal é abordado em [REI 83] do ponto de vista da probabilidade de ocupação das trilhas, como uma função do comprimento da banda (medido em passos de polissilício). Em outras palavras, o interesse agora reside em conhecer o número de células que podem ser posicionadas lado a lado de forma a obtermos uma ocupação total das trilhas da estrutura de banda (transparência nula). Estudos estatísticos resultaram numa distribuição exponencial dada pela equação:

$$T = N * e^{-\frac{L}{20}} \quad [I]$$

onde,

T = transparência horizontal medida em número de trilhas livres;

N = número de trilhas da estrutura de banda (fora as linhas de alimentação); e,

L = comprimento da banda medido em número de passos de polissilício.

Com relação à transparência vertical, se somente as camadas de polissilício e difusão forem utilizadas para roteamento entre bandas, a probabilidade de que uma ligação possa atravessar uma determinada banda é extremamente baixa. Neste caso, a melhor solução é gerar células funcionais compactas na direção horizontal e usar células de interconexão para passagem de sinais (figura 3.5).

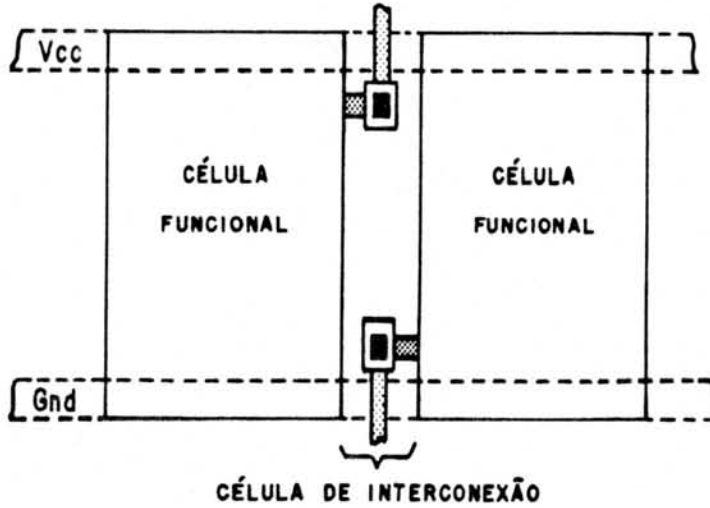


Figura 3.5 - Conexões com bandas adjacentes.

Se uma segunda camada de metal estiver disponível na tecnologia em uso, e se as células não a utilizarem em conexões internas a elas, a transparência vertical das células e bandas será de 100%. Evidentemente, considera-se que as restrições tecnológicas de relevo desta camada sejam pouco restritivas. Seguramente, nestas condições, a redução total de área, pela adoção da metodologia TRANCA, em relação à abordagem "Standard Cell" convencional, é garantida pela eliminação dos canais de roteamento e pela redução da largura das células funcionais [REI 88]. Pela análise da figura 3.6, verifica-se a tendência de obtenção de conexões menos longas no circuito, como consequência direta da compactação de área.



Figura 3.6 - Redução de área e de comprimento de conexões.

A redução da dimensão vertical continua existindo quando se utiliza células de interconexão. No entanto, o crescimento da dimensão horizontal é inevitável neste caso. Mesmo assim, assegura-se uma redução total de área para blocos com formatos onde a altura não predomine

demasiadamente sobre a largura, ou seja, para relações de aspecto inferiores a 1. Estas considerações são retomadas com o devido detalhamento no capítulo 5, onde se avalia a metodologia através de comparações com a abordagem "Standard Cell".

Uma opção para evitar a utilização de muitas células de interconexão é o uso de conexões verticais em metal 2.

3.4 Gerenciamento de Trilhas

De nada adianta a previsão de trilhas livres para roteamento externo às células, se a alocação destas trilhas para conexões locais for desordenada. Na figura 3.7, a alocação diferenciada de trilhas em duas células vizinhas simplesmente impede a passagem direta de uma conexão sobre elas. Neste caso, a realização de tal conexão só é possível através de uma ponte em polissilício.

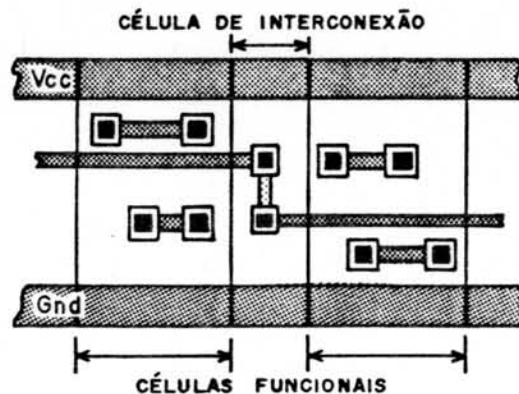


Figura 3.7 - Alocação de trilhas.

O gerenciamento de trilhas surge justamente para evitar esta troca de trilhas e o desperdício de área inerente a esta prática.

A idéia é dispor de um esquema de prioridades para alocação de trilhas que reserve as trilhas de mais alta prioridade a conexões locais e as de menor prioridade a conexões externas à célula. A escolha deste esquema de prioridades deve ser baseada nas possibilidades das regras

tecnológicas, no padrão topológico adotado para a biblioteca como um todo e na melhor topologia para implementação de cada célula individualmente.

Em tecnologias CMOS, por exemplo, as trilhas de maior prioridade são aquelas reservadas para ligar os drenos dos transistores p e n a uma ponte em polissilício que completa a conexão. Já às trilhas localizadas no interior da área reservada para separar os substratos tipo p e tipo n são atribuídas as menores prioridades, de forma que sinais globais, como "clock" e "reset", terminam percorrendo a banda por estas trilhas [REI 88].

Do exposto neste capítulo, concluímos que a adoção da metodologia TRANCA resulta na obtenção de leiautes compactos e de bom desempenho para blocos funcionais ou circuitos inteiros. Isto se justifica principalmente pela estruturação e eficiência na realização de interconexões.

A síntese de leiaute segundo esta metodologia é passível de automação completa e os princípios aqui apresentados são também válidos para as abordagens simbólica e estrutural de geração de célula.

4 O GERADOR DE LÓGICA ALEATÓRIA TRANCA

Perante uma nova metodologia de concepção de leiaute, a adequacidade de automação do processo de síntese é uma questão de desenvolvimento de ferramentas de projeto que, tanto quanto possível, tirem proveito das vantagens e contornem eficientemente as limitações advindas da adoção da metodologia considerada. Para tanto, os procedimentos de síntese de leiaute devem resultar da conjunção de adaptações de estratégias existentes e de novas técnicas desenvolvidas para tratar convenientemente o problema.

É importante salientar que, não necessariamente, o desempenho de uma ferramenta está relacionado à eficiência da metodologia de concepção de leiaute que ela adota. A análise de leiautes projetados "à mão" prova esta assertiva: independente da metodologia que se escolha, leiautes gerados sob a intervenção humana intensiva é que, via de regra, resultam na exploração efetiva do espaço de soluções provido pela metodologia. De qualquer forma, somente uma boa ferramenta é capaz de convencer os projetistas de circuitos integrados a adotarem uma nova metodologia, até porque sua validação só é possível a partir de inúmeras experiências com circuitos de diferentes complexidades.

No tocante a este trabalho, considere-se que o objetivo principal foi desenvolver uma ferramenta automática de síntese de leiaute que implementasse um subconjunto de características da metodologia TRANCA, como primeiro passo para avaliação desta.

A partir de estudos sobre a metodologia TRANCA (capítulo 3) e sobre as técnicas mais populares de síntese de leiaute (capítulo 2), chegou-se, então, à implementação do gerador de módulos em lógica aleatória TRANCA, batizado TRAMO ("TRANca MOdule generator"). Os aspectos referentes à compilação de silício, ao planejamento topológico e à

geração de módulos, abordados no capítulo 1, foram efetivamente considerados, conforme será visto no transcorrer deste capítulo.

4.1 Ambiente de Projeto

O projeto físico de módulos TRANCA é realizado em um ambiente de concepção, onde ferramentas responsáveis pelas fases de síntese, análise e verificação estão presentes, conforme mostra a figura 4.1. A proposta de divisão deste ambiente em projeto do sistema e projeto em silício não pretende sugerir, absolutamente, que a ferramenta TRAMO esteja sendo considerada um compilador de silício. Simplesmente, procura-se classificar as ferramentas de acordo com seus propósitos, de forma a colocar ferramentas compartilhadas por outros geradores de módulos, e portanto de aplicação geral, do lado do projeto do sistema.

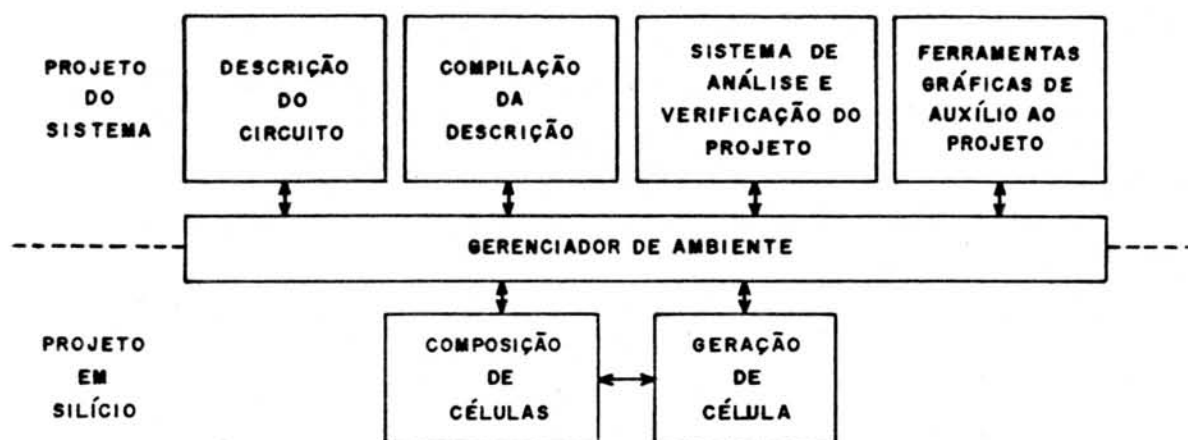


Figura 4.1 - Ambiente de projeto TRANCA.

O ambiente é gerenciado por uma interface orientada a menus [BAG 89], de forma que o compêndio de ferramentas, espalhado por diversos diretórios, e os meios de comunicação entre estas ferramentas sejam transparentes ao projetista. Por questões de simplicidade de implementação, as ferramentas "conversam" entre si por intermédio de arquivos. Versões futuras devem trabalhar a

nível de gerência de um banco de dados.

O ponto de partida para o projeto físico de um determinado circuito é a descrição do mesmo de forma textual ou gráfica. Portanto, compiladores da descrição, tanto para simulação lógica, quanto para síntese de leiaute, são indispensáveis.

Para análise e verificação do projeto físico, programas de verificação de regras de projeto [GOM 88], de extração [STE 89a] e simulação elétrica, e de extração [MOR 89c] e simulação lógica [CAL 88] são utilizados. As ferramentas do nível lógico se aplicam à validação do módulo como um todo; as do nível elétrico, à validação das células primitivas; já o verificador de regras de projeto atinge os leiautes oriundos da geração e da composição de células.

As ferramentas gráficas de auxílio ao projeto incluem editores de máscaras [STE 89b] e simbólico [MAR 89] e um exibidor gráfico dedicado (EXTRAMO). Ambos os editores podem ser utilizados tanto para geração de célula, como simplesmente para a visualização de um módulo inteiro; a ferramenta EXTRAMO tem um caráter exclusivo de exibição de etapas intermediárias da síntese de leiaute. Ao exibidor dedicado e ao editor de máscaras cumpre o auxílio gráfico na tarefa de planejamento topológico realizada pelo usuário do sistema.

Do lado do projeto em silício, uma biblioteca de células primitivas, um conversor e um montador simbólicos [MAR 89] e um gerador estrutural de células [MOR 89b] caracterizam o procedimento de geração de célula.

Por último, a composição de células consta de um subsistema de posicionamento (POTRANCA) e outro de roteamento (RETRANCA).

Muito embora todas as ferramentas aqui citadas

estejam disponíveis e em uso no presente momento, algumas delas não se encontram integradas ao ambiente de projeto. No entanto, o gerenciador já foi desenvolvido pressupondo o ambiente como descrito.

Aquelas ferramentas imprescindíveis à fase de síntese de leiaute, às quais as seções seguintes se dedicam, já integram efetivamente o ambiente apresentado.

4.2 Descrição do Circuito

A linguagem de descrição de "hardware" utilizada como entrada do gerador TRAMO, chamada de NILOTRANCA, resultou de algumas adaptações na linguagem estrutural NILO [WAG 87b]. NILO é a linguagem para descrição de circuitos nos níveis de portas lógicas e de chaves do sistema AMPLO - Ambiente integrado para o Projeto Lógico de sistemas digitais [WAG 88].

Esta opção foi feita devido ao fato do sistema AMPLO fornecer as facilidades de descrição textual e gráfica do circuito a nível lógico, por dispor de um simulador lógico [CAL 88] e por possibilitar o projeto de forma modular, pela composição puramente estrutural de subcircuitos descritos em NILO (ou em outros níveis de abstração) através da linguagem REDES [WAG 87a].

Uma descrição em NILO envolve, basicamente, a declaração dos conectores da interface do circuito, a enumeração e caracterização dos elementos que compõem o circuito e a declaração das interconexões que relacionam os elementos entre si e estes com a interface.

Na declaração da interface do circuito, nos interessa a determinação de direção dos sinais que a compõem (entrada, saída ou bidirecional); dos elementos disponíveis em NILO, somente as portas lógicas e as chaves bidirecionais estão definidas em NILOTRANCA; a caracterização dos elementos segundo seus tempos de atraso

e intensidade de sinais não é relevante para a síntese de leiaute e, portanto, é reconhecida e ignorada na compilação NILOTRANCA; a declaração de interconexões, como em NILO, é subdividida em uma seção que identifica as conexões entre os elementos do circuito, e em outra seção que define a equivalência entre estas conexões e os sinais da interface.

Como as linguagens de AMPLO foram desenvolvidas com uma orientação predominante para a simulação de sistemas digitais discretos [WAG 89], fez-se necessária a inclusão em NILOTRANCA de algumas construções que visassem a síntese de leiaute. Tais construções se aplicam ao posicionamento de conectores da interface, à identificação de redes críticas do circuito, à fixação de posições e orientações no leiaute para as células que implementam os elementos da linguagem e à imposição de caminhos para a realização de determinadas conexões. Constata-se, então, que o objetivo último de prover essas construções é considerar restrições topológicas para o módulo, que tornem o planejamento topológico e a interação com o usuário flexíveis no sentido da otimização do leiaute.

Outra consideração a fazer é que NILO não permite a descrição de circuitos que utilizem elementos com mais de uma saída ou superportas lógicas. NILOTRANCA, por sua vez, suporta primitivas simples do nível de transferência de registradores (como células de memória, por exemplo) e composições de portas lógicas que sejam mapeadas diretamente em uma única célula da biblioteca.

Depois de tantos adendos à linguagem original, evidentemente a integração completa entre os ambientes TRANCA e AMPLO ficou prejudicada. As ferramentas de AMPLO passam a ser utilizadas sob restrição no ambiente TRANCA, até que se convirja para um padrão único de descrição de circuitos. Por ora, a descrição limita-se à sua forma textual e a compilação é feita separadamente para a simulação e a síntese de leiaute.

Um exemplo de descrição em NILOTRANCA da parte de controle de um conversor analógico-digital de 4 bits aparece no anexo 1.

4.3 Geração de Célula

Conforme visto na seção 2.1, as abordagens de geração de célula aplicáveis à síntese de módulos em lógica aleatória são a geométrica, a simbólica e a estrutural.

O ambiente TRANCA prevê a utilização das três abordagens na síntese de leiaute, procurando dispor das vantagens de cada uma delas. A geração geométrica é a que fornece os leiautes mais compactos; a geração simbólica é independente de tecnologia; a estrutural, por sua vez, permite que a geração de célula seja feita e refeita para diferentes restrições topológicas e em tempo de execução do procedimento de composição de células. Um sumário das principais características de cada abordagem é mostrado na tabela 4.1.

Tabela 4.1 - Abordagens de geração de célula.

ABORDAGEM	RELAÇÃO COM A TECNOLOGIA	INTERAÇÃO COM A COMPOSIÇÃO DE CÉLULAS	ÁREA DE CÉLULA
GEOMÉTRICA	DEPENDENTE	ESTÁTICA	PEQUENA
SIMBÓLICA	INDEPENDENTE	ESTÁTICA	MÉDIA
ESTRUTURAL	INDEPENDENTE	DINÂMICA	GRANDE V

Na versão atual da ferramenta TRAMO, porém, a geração de célula é meramente geométrica e, portanto, baseada em uma biblioteca de células padrão.

As células foram concebidas segundo os

fundamentos da metodologia TRANCA, sob a estrutura de banda (seção 3.1) e o esquema de prioridades para alocação de trilhas (seção 3.2) apresentados na figura 4.2. O passo de metal utilizado prevê contatos entre duas camadas de metal e as prioridades das trilhas para conexões internas a uma célula decrescem de P9 a P0.

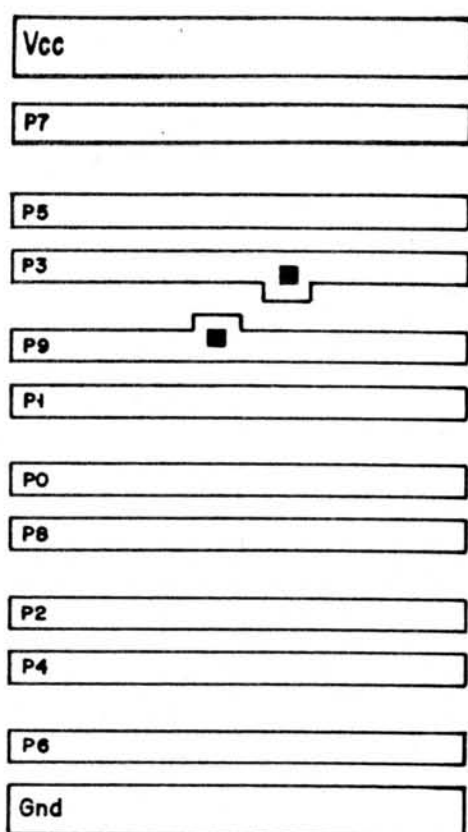


Figura 4.2 - Prioridades para alocação de trilhas.

Em linhas gerais, a configuração topológica das células é tal que: as entradas são implementadas por linhas verticais em polissilício que cruzam todas as trilhas em metal não utilizadas para conexões locais; e as saídas estão disponíveis em metal na extremidade direita da célula, procurando-se implementar pelo menos uma delas como uma conexão incompleta sobre as trilhas P9 e P8. A disposição das entradas, como descrito, permite a programação pura e simples de contatos no cruzamento de uma linha vertical com uma das trilhas da estrutura de banda; já a disposição das saídas adotada permite que conexões por justaposição economizem a área de implementação da ponte em

polissilício entre transistores p e n ou, na impossibilidade, exige a alocação de uma célula de interconexão que complete esta ligação e/ou oportunize a troca de trilha para roteamento.

Na figura 4.3 aparece o leiaute de uma das células da biblioteca TRANCA. Como se pode observar, a topologia adotada é perfeitamente válida para geração simbólica ou estrutural de célula.

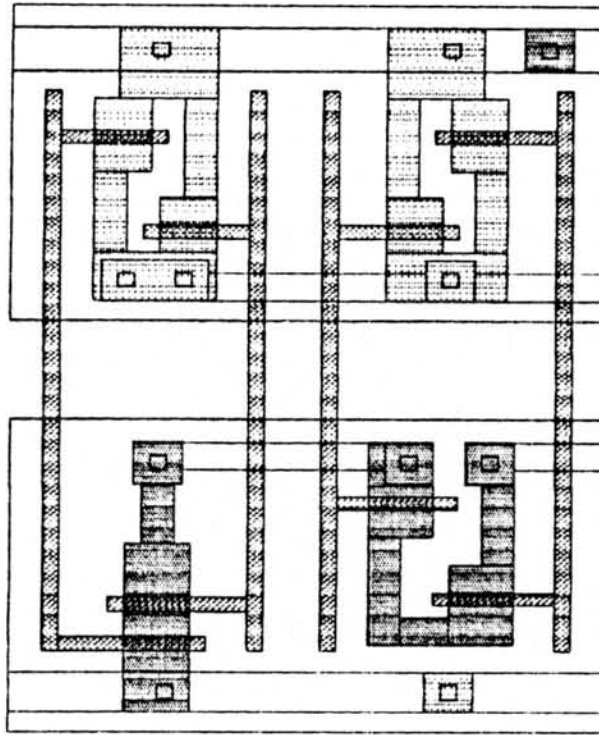


Figura 4.3 - Leiaute de uma "nand" de 4 entradas.

A referência [LUB 88c] apresenta em detalhes a configuração topológica brevemente descrita aqui.

A escolha de um conjunto inicial de células para compor a biblioteca foi baseada, principalmente, em catálogos de fabricantes, na relação número de posicionamentos vs. capacidade funcional [SUN 87] de células anteriormente utilizadas em circuitos concebidos pelo GME e nas primitivas disponíveis na linguagem de descrição de "hardware" apresentada na seção 4.2. A tabela 4.2 enumera as células que compõem atualmente a biblioteca

TRANCA, informando ainda a transparência e a largura de cada uma delas. A transparência média, ponderada pela largura das células, resultante dos dados fornecidos pela tabela 4.2 é de 63%.

Tabela 4.2 - Células da biblioteca TRANCA.

CÉLULA	TRANSPARÊNCIA (%)	LARGURA (λ)
INVERSOR	80	21
BUFFER INVERSOR 1 ESTÁGIO	80	31
BUFFER NÃO INVERSOR	80	42
BUFFER INVERSOR 3 ESTÁGIOS	80	72
NOR DE 2 ENTRADAS	80	33
NOR DE 3 ENTRADAS	80	52
NOR DE 4 ENTRADAS	80	77
NAND DE 2 ENTRADAS	80	35
NAND DE 3 ENTRADAS	80	56
NAND DE 4 ENTRADAS	80	73
NAND DE 6 ENTRADAS	80	103
OR DE 2 ENTRADAS	80	54
AND DE 2 ENTRADAS	80	53
AND-OR 2:2	70	89
XOR	80	71
CHAVE CMOS	60	45
INVERSOR/NÃO INVERSOR	70	43
MULTIPLEXADOR 2:1 (I)	60	99
MULTIPLEXADOR 2:1 (II)	60	82
FLIP-FLOP RS	50	138
FLIP-FLOP D ESTÁTICO	50	97
FLIP-FLOP JK	60	174
FLIP-FLOP JKSR	60	209
FLIP-FLOP D ESTÁTICO MS	50	177
FLIP-FLOP D DINÂMICO (I)	40	154
FLIP-FLOP D DINÂMICO (II)	60	131
FLIP-FLOP D DINÂMICO MS	50	247

A caracterização elétrica e a documentação das células foi realizada segundo os procedimentos e modelos sugeridos em [CAR 87]. Um exemplo extraído da documentação completa da biblioteca [MOR 88][SOM 89] é apresentado no anexo 2.

À medida que se dispõe das regras tecnológicas, da configuração topológica e de ferramentas de edição de máscaras e de análise e verificação de projeto, a biblioteca é aberta à inclusão de novas células que venham a ser necessárias na concepção de circuitos específicos.

4.4 Composição de Células

Durante o desenvolvimento do gerador TRAMO, em especial na fase de especificação dos procedimentos de composição de células, alguns pré-requisitos considerados fundamentais orientaram a escolha das estratégias de posicionamento e roteamento automáticos.

Do ponto de vista do planejamento topológico, restrições de forma do módulo e de orientação e ordenação dos sinais da sua interface deviam ser consideradas na busca do ótimo global para o circuito como um todo.

No tocante à metodologia TRANCA, as características de compactação de área e de bom desempenho elétrico de leiautes "manuais" tinham que ser reproduzidas pela ferramenta.

Com relação às limitações de utilização de uma estrutura de banda fixa e de uma abordagem sem transparência vertical de célula, considerou-se os problemas de evitar a saturação de trilhas, antes que o roteamento estivesse completo, e de restringir o uso de células de interconexão (consumidoras de área). Evidentemente, o compromisso entre este subconjunto de características da metodologia TRANCA e a complexidade de automação do leiaute limitou a liberdade de escolha de estratégias para a primeira versão da ferramenta.

Do ponto de vista da otimização individual do leiaute do módulo, uma interação com o projetista que possibilitasse intervenções diretas no posicionamento de células e no roteamento de sinais tinha que ser prevista.

Da consideração destes aspectos e do estudo das principais estratégias de composição de células, resultou o desenvolvimento de três ferramentas para a síntese automática de leiaute: um particionador, um posicionador intrabanda e um roteador.

As duas primeiras ferramentas compõem o subsistema de posicionamento de células - chamado POTRANCA, que a princípio pode ser utilizado para qualquer metodologia de projeto baseada em uma biblioteca de células padrão.

Por sua vez, o subsistema de roteamento - batizado de RETRANCA, é fundamentado em estratégias também aplicáveis ao projeto de circuitos em "sea-of-gates" [SHR 88].

Estas ferramentas são apresentadas nas próximas seções.

4.4.1 POTRANCA

Avaliando-se as estratégias de posicionamento existentes, verifica-se que as técnicas de partição aliam à procura de corte mínimo de linhas imaginárias por interconexões, a flexibilidade de atendimento de sugestões ou imposições de restrições posicionais de células (seção 2.2.1.3.3).

Estas características por si só justificam a escolha desta técnica como estratégia básica do POTRANCA. O fato do uso da técnica de partição resultar em melhores posicionamentos do que por crescimento de aglomerados reforça a escolha.

Aplicando-se imediatamente a idéia de partição, então, o problema de posicionamento bidirecional fica dividido em duas etapas no POTRANCA:

- a) particionamento do circuito em bandas; e,

b) posicionamento intrabanda das células.

4.4.1.1 O Particionador

A estratégia de particionamento do circuito em bandas parte da idéia da heurística FIDUCCIA-MATHEYSES brevemente apresentada na seção 2.2.1.3.3.

Como visto naquela seção, o particionamento de circuitos é baseado no intercâmbio de células entre as partições e no controle simultâneo dos seus pontos de equilíbrio.

A troca de células entre partições é guiada pelo cálculo do ganho de transferência de uma célula de uma partição à outra. O ganho de cada célula é função do seu custo interno e externo.

O custo interno (CI) de uma célula é dado pelo somatório dos pesos atribuídos às redes que a ligam exclusivamente a células que pertencem à mesma partição.

O custo externo (CE), por sua vez, é dado pelo somatório dos pesos das redes que ligam a referida célula somente a células pertencentes à outra partição.

Finalmente, o ganho para transferência individual (GI) é calculado por:

$$GI = CE - CI . \quad [II]$$

Se a transferência de células for realizada sob a forma de intercâmbio, o ganho (GP) de uma par (1,2) será:

$$GP = GI1 + GI2 - [FC(1,2) + FC(2,1)] \quad [III]$$

onde,

fator de correção $FC(i,j)$ = peso da rede que liga i a j ,
 se i é a única célula da rede na partição;
 = 0, caso contrário.

Os pesos das redes valem a princípio 1. Se uma

determinada rede é considerada crítica, e deseja-se estimular a aglomeração das células que a compartilham em uma mesma partição, a solução é atribuir a esta rede um peso superior. A linguagem NILOTRANCA dispõe da construção "weigh" para este propósito (veja anexo 1).

Um exemplo de avaliação de possíveis transferências entre partições de um circuito aparece na figura 4.4a e nas tabelas 4.3, 4.4 e 4.5.

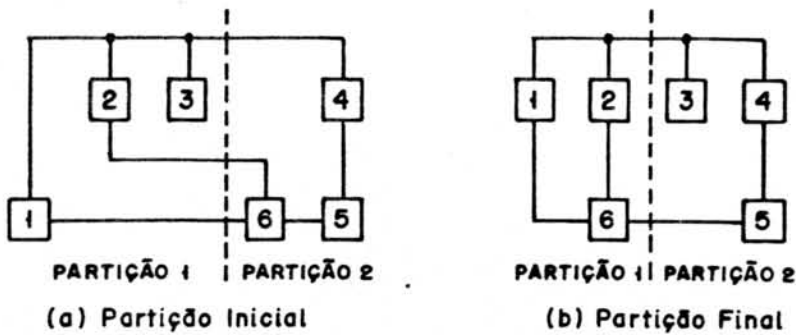


Figura 4.4 - Exemplo de intercâmbio.

Tabela 4.3 - Custos e ganhos individuais.

CÉLULA	CI	CE	GI
1	0	1	1
2	0	1	1
3	0	0	0
4	1	1	0
5	2	0	-2
6	1	2	1

Tabela 4.4 - Fatores de correção.

C2 \ C1	4	5	6
1	0	0	1
2	0	0	1
3	0	0	0

(a) FC(C1,C2)

C1 \ C2	1	2	3
4	1	1	1
5	0	0	0
6	1	1	0

(b) FC(C2,C1)

Tabela 4.5 - Ganhos de intercâmbio: GP do par (C1,C2).

C1 \ C2	4	5	6
1	0	-1	0
2	0	-1	0
3	-1	-2	①

No exemplo apresentado, o intercâmbio das células 3 e 6 é aquele que resulta em um ganho positivo e na manutenção de ambas as partições com a mesma cardinalidade (figura 4.4b).

O equilíbrio de uma partição de um circuito deve ser tal que uma razão (Q) pré-determinada entre a sua área (A) e a área total do circuito (W) seja respeitada. Logo,

$$Q = A / W . \quad [IV]$$

Como dificilmente a razão entre áreas obtida é exatamente igual à Q, um intervalo de tolerância é admitido no julgamento de aceitação ou não de um determinado intercâmbio ou transferência individual de célula. Assim, o controle do equilíbrio se dá segundo

$$Q \cdot W - S \cdot T \leq A \leq Q \cdot W + S \cdot T \quad [V]$$

onde,

S = percentual que varia de 10 a 100% (0,1 a 1); e

T = área da maior célula que compõe o circuito que se particiona.

A partir do momento que se lida com células de mesma altura, as áreas utilizadas em [IV] e [V] são automaticamente substituídas por larguras da partição do circuito e de célula.

Aplicando os critérios de cálculo de ganho e de controle de equilíbrio ao particionamento de circuitos

TRANCA e agregando as características da metodologia e as expectativas do projetista com relação à ferramenta, a estratégia de partição recursiva do módulo em fatias (seção 2.2.1.3.3) é adotada para o particionador.

Da bipartição inicial do circuito resulta um bloco principal e um bloco residual. O bloco residual é biparticionado da mesma maneira, até que o último bloco principal seja obtido por indivisibilidade do próprio bloco residual gerado. A cada iteração, a razão de equilíbrio Q para o bloco principal aumenta em função da redução do número de partições a serem ainda obtidas, conforme mostra a figura 4.5.

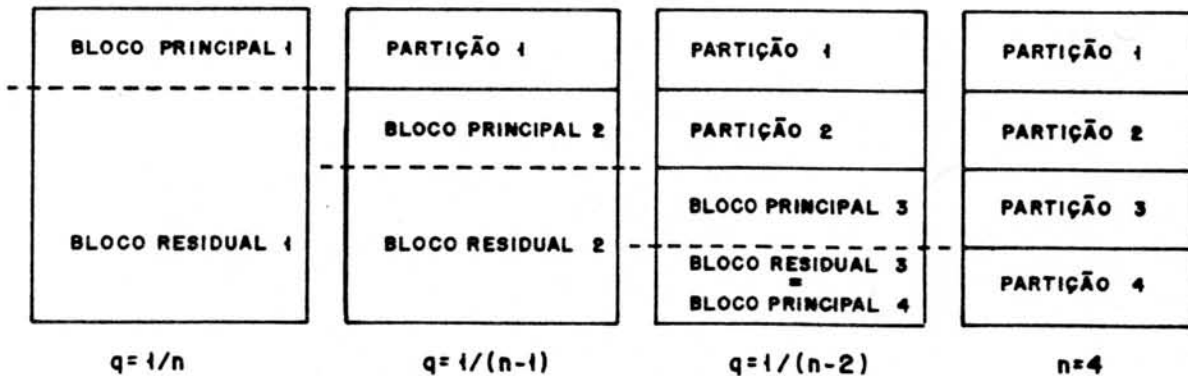


Figura 4.5 - Partição em fatias.

Como se observa na mesma figura, cada nova partição gerada é adjacente à partição anterior. Assim, o posicionamento tem um caráter absoluto com relação às partições, uma vez que elas ocupam posições bem definidas, e um caráter relativo com relação às células, que estão alocadas a determinadas partições mas não possuem uma posição definitiva dentro delas.

Devido ao caráter absoluto do posicionamento das partições, a adjacência de redes entre elas precisa ser estimulada, visando a maior localidade possível de conexões e a conseqüente redução da complexidade de alocação de células de interconexão no roteamento. A técnica adotada para o propósito de estímulo de adjacência baseia-se no

conceito de redes imaginárias, que também se aplica ao atendimento de orientações e ordenações de sinais da interface.

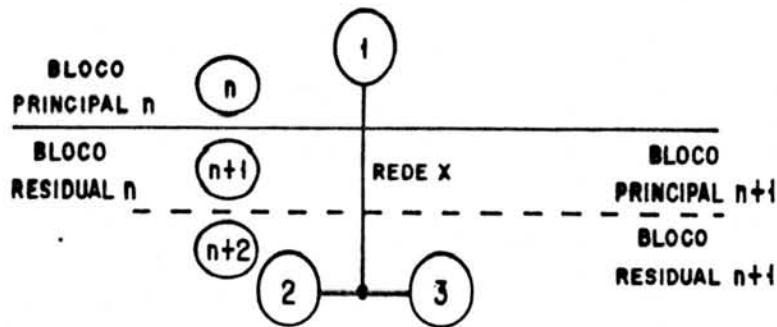
A idéia é associar, a cada célula ligada a uma rede sob restrição, uma rede imaginária com peso direcionado, ou seja, diferenciado para transferências ao bloco principal e ao residual, conforme a natureza da restrição. A rede original é mantida, permitindo seguir-se considerando a vizinhança entre células por ela estabelecida.

Neste ponto, cabe salientar que a adoção pura e simples da estratégia de atribuição de pesos diferenciados às redes reais é ineficiente sob o aspecto de otimização física. Considere o exemplo da figura 4.6, onde quer-se estimular a adjacência entre partições vizinhas. Suponha que a partição inicial do bloco residual n disponha as células 2 e 3 no resíduo $n+1$. Por maior que seja o peso da rede x para transferência ao bloco principal $n+1$, tanto a célula 2 como a 3 terão um ganho nulo relacionado a esta transferência, já que o movimento individual não reduz o número de conexões que cruzam a linha limítrofe entre partições do resíduo n (figura 4.6a). Se, ao contrário, redes imaginárias com peso superior ao da rede real forem atraídas pelo bloco principal, os custos das células 2 e 3 com relação à rede x serão necessariamente positivos para movimento ao bloco principal $n+1$ (figura 4.6b). Neste caso, a possibilidade de que as células 2 e 3 sejam transferidas à partição vizinha da partição n é real.

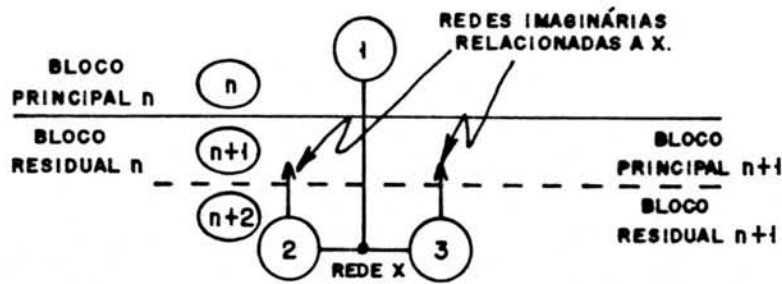
Aplicando-se o conceito de redes imaginárias ao atendimento de restrições dos sinais de interface, o direcionamento dos pesos passa a ser função da extremidade do módulo onde o sinal deve estar disponível.

É evidente que, pelo fato de ter-se que respeitar um determinado equilíbrio entre partições, eventualmente

algumas restrições não serão atendidas a contento. Para atenuar o efeito de conexões muito longas, a reavaliação e o incremento dos pesos das redes imaginárias precisa ser realizado.



(a) Estratégias de Pesos Diferenciados



(b) Estratégias de Redes Imaginárias com Peso Diferenciado

Figura 4.6 - Estratégias de redes imaginárias vs. pesos diferenciados.

A estratégia de redes imaginárias torna, sem dúvida alguma, o tempo computacional ascendente, porém em níveis ainda aceitáveis. Os resultados são todavia compensadores.

O particionador TRANCA realiza o posicionamento relativo de células, primeiro dividindo o módulo em regiões (partição horizontal) e depois cada região em bandas (partição vertical), conforme mostra a figura 4.7.

O número de banda (NB) do circuito, se não fornecido diretamente pelo projetista, é calculado a partir da relação de aspecto (RA) desejada para o módulo, ou seja

$$RA = \text{ALTURA} / \text{LARGURA} = (\text{NB} \cdot K) / (W / \text{NB}) \quad [\text{VI}]$$

onde,

K = constante dependente de tecnologia, função da altura padrão das células e da distância mínima entre bandas;

e,

W = somatório das larguras de todas as células que compõem o circuito.

De [VI] vem que:

$$NB = (W \cdot RA / K)^{1/2} \quad [VII]$$

truncado para valores fracionários.

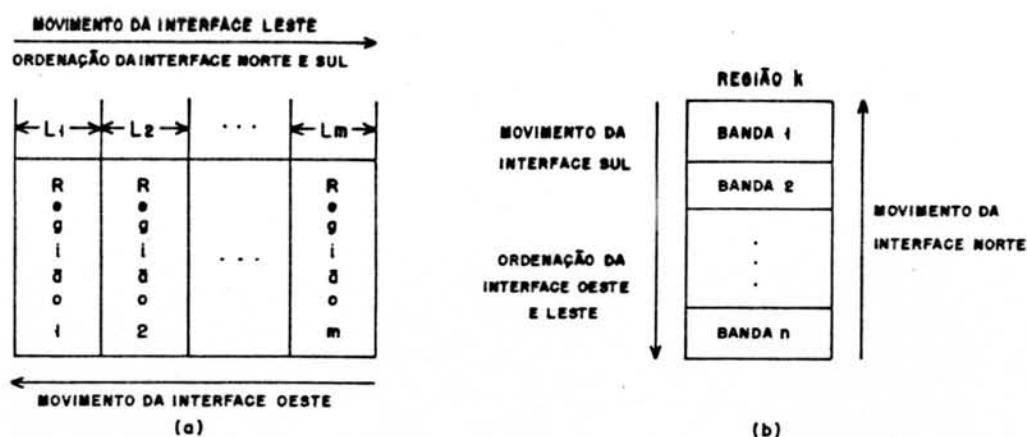


Figura 4.7 - Etapas do particionamento.

A partição horizontal é realizada com o objetivo principal de preservar a estrutura de banda contra a saturação de trilhas. Na seção 3.3 vimos que existe um comprimento máximo de banda (L_m), para o qual garante-se a existência de trilhas livres para o roteamento. Sendo assim, o módulo precisa ser particionado em regiões com um número reduzido de conexões em comum, de forma que as bandas que compõem cada região não possuam comprimento superior a L_m . Destas considerações, resulta que o número de regiões (NR) do circuito é dado por:

$$NR = (W / NB) / L_m \quad [VIII]$$

arredondado para o valor inteiro imediatamente superior, se NR resultar em valor fracionário. L_m é calculado a partir de [I] (página 79) resultando em:

$$L_m = -20 \cdot \log_e \frac{1}{N} \quad [IX]$$

A partição horizontal, além de assegurar condições de roteabilidade e primar pela adjacência de redes entre regiões vizinhas, procura atender às restrições

de roteamento relacionadas à orientação dos sinais da interface oeste/leste e relacionadas à ordenação dos conectores da interface norte/sul (figura 4.7a). No caso de orientação oeste, redes imaginárias com peso positivo para transferência à região principal e negativo para transferência à região residual são definidas; se a orientação a considerar é a leste, o procedimento é o oposto. No caso de ordenação, primeiro os conectores são destinados a determinadas regiões; a seguir, os pesos das redes imaginárias definidas são tais que, enquanto não se atinge a região alvo, o peso é positivo para transferência à região residual e negativo para transferência à região principal, mas ao atingi-la, os pesos têm suas polaridades invertidas. Nem para a orientação, nem tampouco para a ordenação de sinais de interface, as restrições são impositivas; elas são, outrossim, favorecidas com relação ao tratamento de redes internas ao circuito.

Enquanto a partição horizontal estabelece como direção preferencial, para realização de conexões internas a cada região, a direção vertical, favorecendo o uso de células de interconexão, a partição vertical procura redirecionar as conexões para exploração da estrutura de banda, e portanto, realização de ligações em metal na direção horizontal. Cada região é então particionada em bandas, estimulando-se a localidade de conexões e a adjacência de redes entre bandas vizinhas. A cada nova região submetida à partição vertical, a localização das redes divididas com a região vizinha anteriormente particionada é considerada, ou seja, a propagação de restrições internas ao módulo é realizada.

A partição vertical tem por objetivo, também, o atendimento das restrições de roteamento associadas à orientação de sinais da interface norte/sul e à ordenação dos conectores da interface oeste/leste (figura 4.7b). Ambas são tratadas de forma semelhante àquela empregada na

partição horizontal, considerando-se agora bandas principais e residuais.

Como usualmente as células apresentam larguras das mais variadas, nem as áreas das regiões serão idênticas, nem tampouco os comprimentos das bandas. O critério de equilíbrio (inequação [V] - página 96), neste caso, deve ser cuidadosamente avaliado, uma vez que a irregularidade das regiões se reflete nas bandas que, por sua vez, já possuem um fator de irregularidade associado (em função do intervalo de tolerância admitido no critério).

No particionador TRANCA, adota-se um fator de compensação (FC) no critério de equilíbrio da partição vertical, de forma a reduzir ou aumentar o comprimento do trecho de banda que é gerado em uma determinada região, em função do comprimento esperado e do somatório dos comprimentos dos trechos da mesma banda nas regiões já totalmente particionadas.

Alterando-se, então, a inequação [V] temos que:

$$Q \cdot W_r - S \cdot T - FC \leq L_r \leq Q \cdot W + S \cdot T - FC \quad [X]$$

e

$$FC = \sum_{i=1}^{R-1} L_i - [W / (NR \cdot NB)] \cdot (R - 1) \quad [XI]$$

onde,

W_r = somatório das larguras das células que compõem a região R;

L_r = comprimento do trecho da banda que se gera na região R;

L_i = comprimento do trecho da banda considerada na região i;

Q, S, T são definidos na página 96;

NB é definido na página 99; e,

NR e W são definidos na página 100.

Observando-se a figura 4.8, vemos que a adoção

desta estratégia conduz a resultados mais regulares.

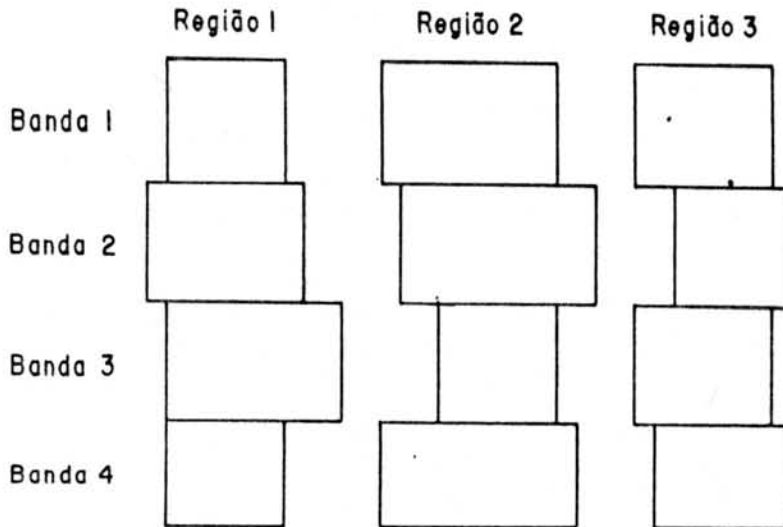


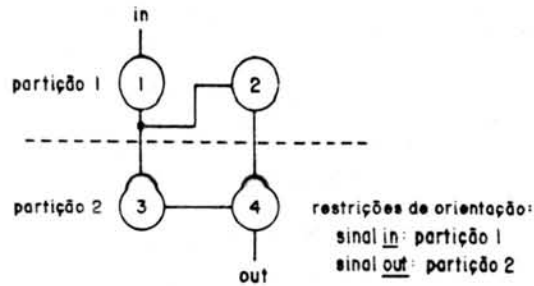
Figura 4.8 - Fator de compensação do critério de equilíbrio.

Sabidamente, a solução fornecida por um processo de partição é extremamente dependente da partição inicial sobre a qual se trabalha [SEC 88] e, com freqüência, as heurísticas adotadas esbarram em mínimos locais.

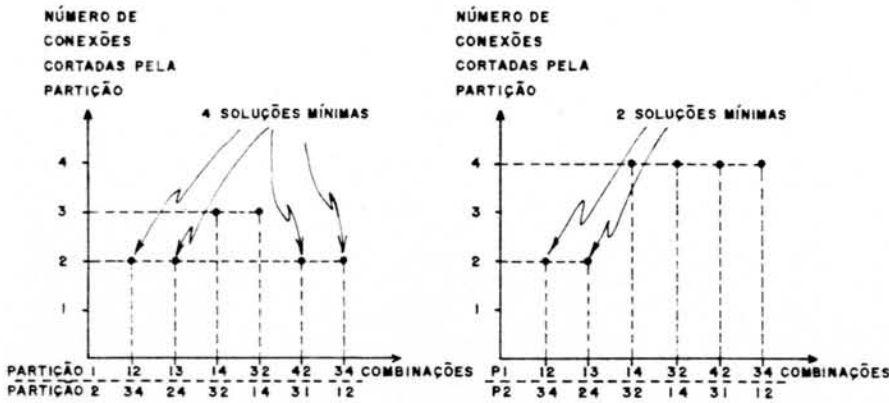
O fato do particionador TRANCA considerar restrições de interface atenua esta situação. Quanto maior o número de restrições de interface, menor o número de soluções aceitáveis (mínimos locais) para a partição. E, quanto menor o número de mínimos locais, maior a probabilidade de se atingir o mínimo global a partir de qualquer solução inicial proposta. Estes fatos são evidenciados se observarmos as possibilidades de partição do circuito da figura 4.9.

De qualquer forma, isto não significa a imunidade completa do particionador TRANCA, até porque a síntese de leiaute de módulos livres de restrições topológicas é também realizada. Assim, optou-se pelo particionamento de circuitos para vários intervalos diferentes de tolerância do equilíbrio. Variando-se, então, o valor de S na inequação [X], as partições iniciais geradas a partir da seqüência da descrição provida pelo projetista diferem de algumas células entre si. A partição é realizada do início

ao fim para cada valor de S adotado, permanecendo aquela que resulta na melhor solução em termos de área total do módulo, ou de atendimento a restrições de interface.



(a) Circuito Exemplo



(b) Soluções sem Restrições de Interface

(c) Soluções Considerando Restrições de Interface

ML = nº de mínimos locais.
 MG = nº de mínimos globais.
 P = probabilidade de se atingir um mínimo global.

$$P = \frac{MG}{ML+MG} = \frac{1}{\frac{ML}{MG} + 1}$$

$\lim_{ML \rightarrow 0} P = 1$ $\lim_{ML \rightarrow \infty} P = 0$

Figura 4.9 - Redução do número de soluções mínimas.

Visando a obtenção da relação de aspecto desejada para o módulo, nem sempre o primeiro valor obtido para o número de bandas (equação [VII] - página 100) corresponde à melhor solução de partição. Obviamente, isto decorre do fato de células de interconexão serem posteriormente acrescentadas ao leiaute. Para contornar a geração de partições que resulte em valores muito diferentes do desejado, uma estimativa de área e de relação de aspecto é realizada ao final do particionamento para um determinado número de bandas. Conforme os valores estimados, a partição é refeita para um número maior ou menor de bandas, e assim sucessivamente até que se atinja a relação de aspecto especificada. A convergência deste processo só pode ser

assegurada para relações de aspecto de até 0.66.

O anexo 3 apresenta os principais passos percorridos pelo particionador TRANCA no posicionamento relativo das células de um circuito. Maiores detalhes sobre cada procedimento envolvido aparecem em [LUB 88b].

As construções da linguagem NILOTRANCA que possibilitam a intervenção direta do projetista na determinação de partições para o circuito são "region" e "strip" (veja anexo 1), que impõem, respectivamente, a região e a banda onde uma determinada célula deve ser posicionada.

A figura 4.10 mostra a saída gráfica do programa EXTRAMO para o particionamento em 6 bandas do circuito descrito no anexo 6. O anexo 7 apresenta o relatório de saída do particionador para este mesmo particionamento.

4.4.1.2 O Posicionador Intrabanda

O posicionamento absoluto das células dentro de cada região de uma determinada banda é realizado pelo posicionador intrabanda.

A estratégia básica utilizada é um misto das técnicas de crescimento de aglomerados e das técnicas de partição. O movimento das células caracteriza um crescimento linear das bandas, ao passo que o critério de seleção de célula é fundamentalmente o de corte mínimo.

A heurística de posicionamento utilizada, primeiro, centraliza todas as bandas em relação à maior delas (mais longa).

A partir da centralização, apontadores de posicionamento para cada banda são estabelecidos. Estes apontadores correspondem à coordenada x onde a próxima célula selecionada da banda deve ser posicionada. A cada novo posicionamento, o apontador relacionado é incrementado

da largura da célula movida para aquela posição.

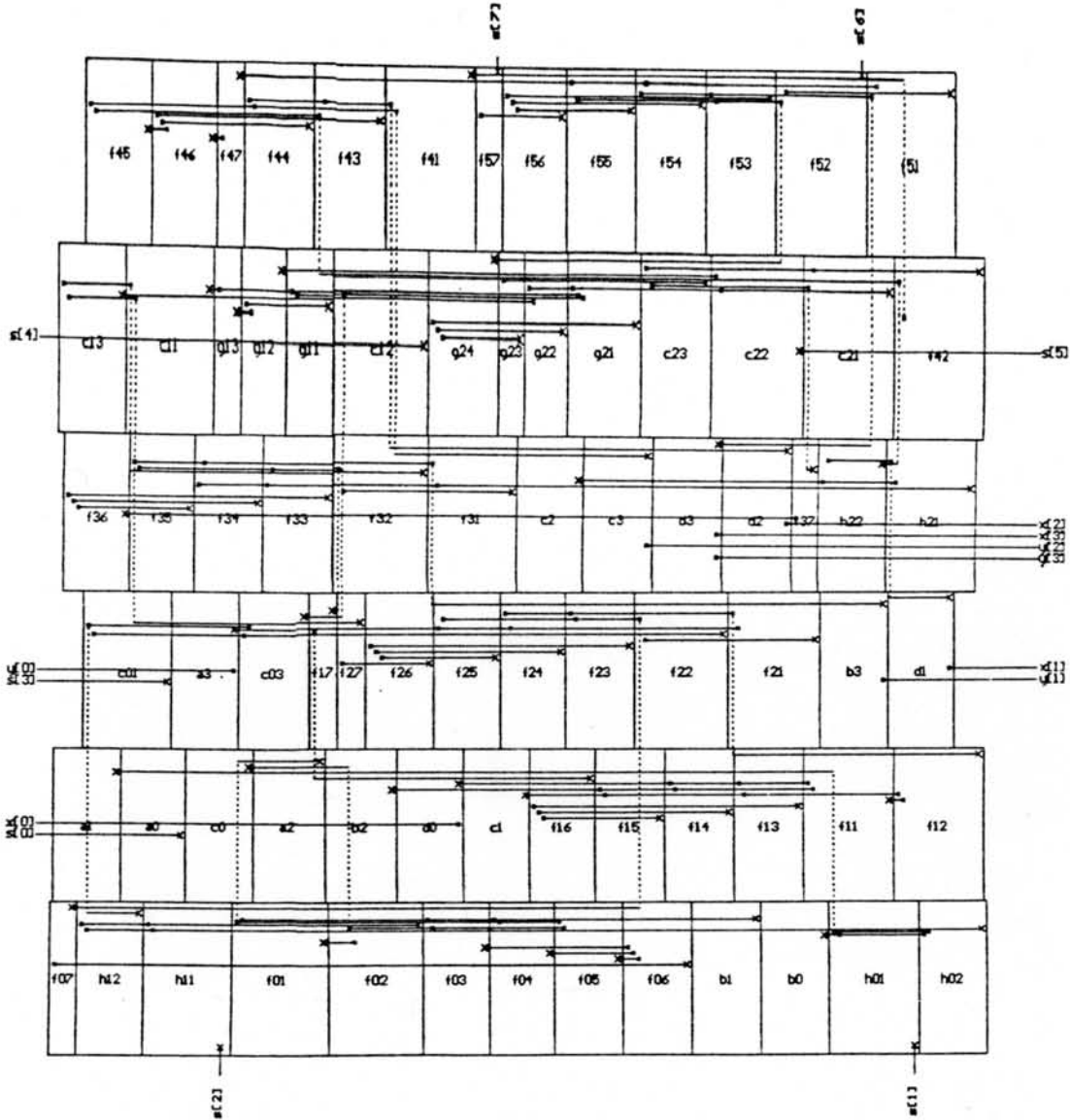


Figura 4.10 - Exemplo de particionamento.

O posicionamento é realizado de forma concorrente entre as bandas, ou seja, a escolha da banda que terá uma de suas células posicionada antecede a seleção e o movimento de uma célula.

A seleção de banda é ditada pelo menor apontador de posicionamento na iteração, considerando-se exclusivamente as bandas que ainda possuem células não posicionadas da região considerada.

A escolha de célula, por sua vez, é baseada na avaliação de escores das células da banda selecionada, que

não foram anteriormente posicionadas.

Para cada região que se segue, procede-se da mesma forma, partindo-se dos apontadores de posicionamento resultantes do tratamento da região imediatamente anterior.

A dinâmica do procedimento (figura 4.11), tal qual descrita, é a aplicação direta da estratégia sugerida em [CHO 88], salvo a divisão do módulo em regiões, não utilizada pelo autor.

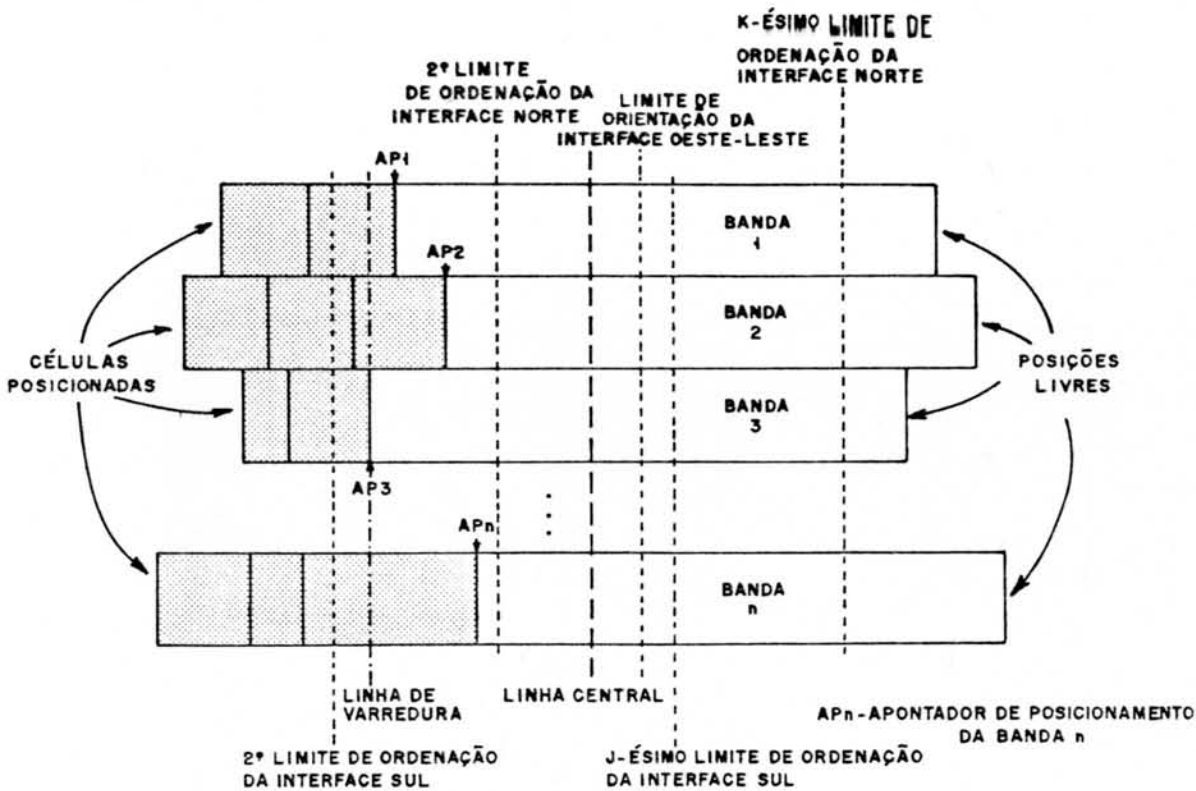


Figura 4.11 - Estratégia de posicionamento intrabanda.

Mesmo sem considerarmos os critérios de seleção de célula adotados, podemos verificar que a dinâmica do procedimento tem um caráter global, pois considera o circuito como um todo e ataca o problema de posicionamento de forma descendente. Este fato justifica a sua escolha para o posicionamento TRANCA.

Como, no entanto, o procedimento não provê meios para consideração de restrições de interface, alguns conceitos novos tiveram que ser incluídos.

A orientação de sinais da interface, a nível de posicionamento intrabanda, não precisa considerar nada além da direção oeste-leste, pois as restrições relativas à direção norte-sul já foram tratadas a nível de particionamento. Tomando-se como referência, então, a linha vertical limite para orientação, resta dirigir as células ligadas a redes da interface oeste para posições à esquerda de tal linha e células ligadas a redes da interface leste para posições à direita da mesma linha. Evidentemente, a proximidade das extremidades é desejável.

Com relação à ordenação de sinais da interface, a única direção de interesse é a norte-sul, já que a ordenação oeste/leste foi estabelecida pela divisão do circuito em bandas. Os problemas de ordenação norte e sul, embora utilizem a mesma estratégia, precisam ser atacados separadamente, uma vez que o número de sinais a serem ordenados e o número e as larguras das células atingidas são normalmente muito diferentes.

No tratamento da ordenação, assim como na orientação, a idéia é estabelecer linhas imaginárias verticais que representem limites de posicionamento. Desta forma, na porção do circuito delimitada pela extremidade esquerda do módulo, e pela segunda linha limite para ordenação norte, por exemplo, dá-se preferência ao posicionamento de células que compartilham a rede norte com ordenação 1; entre a segunda e a terceira linhas, o posicionamento preferencial é o de células que compartilham a rede com ordenação 2; e assim sucessivamente, até que se considere a n -ésima linha, onde n é número de ordenações, e a extremidade direita do módulo. As mesmas considerações podem ser feitas para ordenação de redes com orientação sul.

A determinação da localização das linhas limites de posicionamento é feita segundo a divisão da largura total do módulo, de forma proporcional aos comprimentos

associados a cada grupo de células que compartilham a mesma rede ordenada. Este comprimento é dado pelo maior somatório das larguras das células que compartilham a referida rede e que estão dispostas na mesma banda.

Na figura 4.11, a presença de linhas limites para orientação e para ordenação de sinais da interface pode ser observada.

Estabelecidos os apontadores de posicionamento, o critério para seleção de banda e as linhas limites para orientação e ordenação de interface, resta-nos simplesmente definir os escores de células a serem computados, já que a seleção de célula é guiada pelo maior escore global obtido.

Os escores utilizados no posicionador intrabanda são os seguintes: escore de orientação oeste/leste, escore de orientação de interface livre de restrição, escore de ordenação, escore de vizinhança e escore global.

O escore de orientação oeste/leste (EOL), computado para cada célula uma única vez durante todo o procedimento, é dado por:

$$EOL = \sum_{i=1}^n p_i \cdot (\text{oeste} - \text{leste}) \quad [\text{XII}]$$

onde,

n = número total de redes da célula;

p_i = peso da rede i ;

oeste = 1, se rede i é oeste, e
= 0, caso contrário;

leste = 1, se rede i é leste, e
= 0, caso contrário.

De [XII] concluímos que células ligadas a redes livres de orientação têm $EOL=0$, que orientações oeste contribuem para que EOL seja positivo, e que orientações leste contribuem para que EOL seja negativo. Esta convenção foi adotada porque o posicionamento é realizado da esquerda

(oeste) para a direita (leste).

O escore de orientação de interface livre de restrição (ELR) só se aplica quando o número de regiões resultantes do particionamento é superior a 1. Neste caso, antes do posicionamento das bandas pertencentes à primeira e à última região, o escore é calculado como segue:

$$ELR = \sum_{i=1}^n p_i \cdot (itf_lr) \cdot (primreg - ultreg) \quad [XIII]$$

onde,

$itf_lr = 1$, se rede i é interface sem orientação, e
 $= 0$, caso contrário;

$primreg = 1$, se posicionamento da primeira região, e
 $= 0$, caso contrário;

$ultreg = 1$, se posicionamento da última região, e
 $= 0$, caso contrário.

Da observação do comportamento de [XIII] vemos que redes da interface livres de restrições e situadas na região mais à esquerda do módulo são consideradas como se tivessem uma orientação oeste; já redes da interface livres de restrições e situadas na região mais à direita são consideradas como redes com origem ou destino no leste do módulo.

O escore de ordenação (ENS), por sua vez, é reavaliado a cada novo posicionamento de célula. Em função das linhas que limitam a porção do circuito destinada a uma determinada ordenação de sinal da interface, o escore é calculado. Assim:

$$ENS = \sum_{i=1}^n p_i \cdot (nsord) \cdot (mai_sup - men_inf) \quad [XIV]$$

onde,

$nsord = 1$, se rede i é norte ou sul ordenada, e
 $= 0$, caso contrário;

$mai_sup = 1$, se apontador de posicionamento maior que linha limite à direita do local preferencial, e

$\text{mai_sup} = 0$, caso contrário;
 $\text{men_inf} = 1$, se apontador de posicionamento menor que
 linha limite à esquerda, e
 $= 0$, caso contrário.

Verificamos, então, que ENS é negativo enquanto não se atinge o local preferencial; nulo, enquanto se estiver entre as linhas limites para posicionamento pertinentes; e positivo, a partir do momento que se ultrapassa o limite mais à direita.

O escore de vizinhança (EVZ) é aquele que considera as conexões internas ao módulo. Como a relação entre o conjunto das células já posicionadas e o conjunto das não posicionadas se altera a cada movimento de célula deste em direção àquele, o escore de vizinhança é reavaliado antes do próximo posicionamento. O escore é dado por:

$$\text{EVZ} = \sum_{i=1}^n p_i \cdot (\text{posic} - \text{nposic}) \quad [\text{XV}]$$

onde,

$\text{posic} = 1$, se a célula é a única representante da rede i no conjunto das não posicionadas, e
 $= 0$, caso contrário;
 $\text{nposic} = 1$, se nenhuma célula da rede i pertence ao conjunto das posicionadas, e
 $= 0$, caso contrário.

Analisando [XV], vemos que EVZ prima pelo corte mínimo, pois células fortemente ligadas ao conjunto das posicionadas e fracamente ligadas ao conjunto das células não posicionadas tendem a possuir escores tão mais positivos quanto maior é a intensidade de atração.

Por fim, o escore global (EGL), que determina a seleção ou não da célula para movimento, é dado por:

$$\text{EGL} = \text{EOL} + \text{ELR} + \text{ENS} + \text{EVZ} \quad [\text{XVI}]$$

Durante a fase de seleção de célula, a avaliação de escores pode resultar em empate entre células candidatas. Neste caso, estimativas dos comprimentos das redes que ligam as células empatadas ao bloco posicionado são consideradas.

Primeiro, tenta-se o desempate pela avaliação do comprimento das redes terminais, ou seja, das redes cuja última representante no bloco ainda não posicionado é a célula que se considera. A célula que possuir o maior somatório de comprimentos é selecionada.

Se o empate persistir, passa-se à análise das redes não terminais não pertencentes à interface do módulo, a partir das mesmas considerações feitas no passo anterior.

Por fim, caso o impasse permaneça, a célula candidata de menor largura é selecionada. Em último caso, a ordem de descrição das células é considerada.

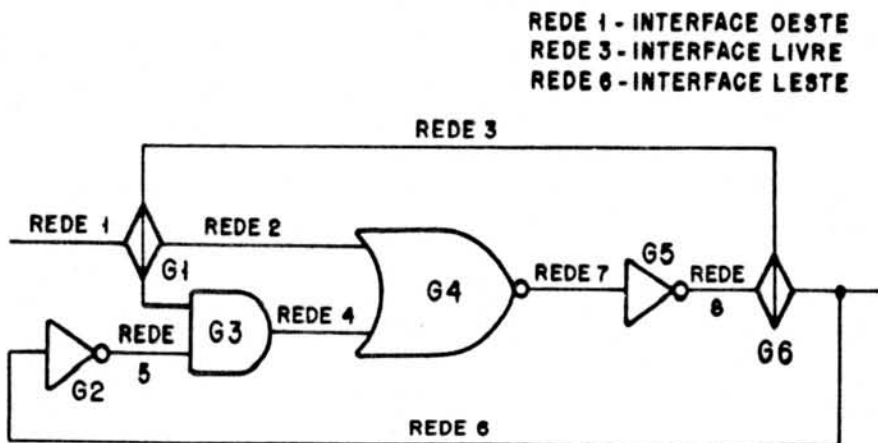
O citado comprimento de redes é obtido de maneira simplificada pelo cálculo da distância entre o atual apontador de posicionamento da banda e a coordenada de posicionamento da célula mais próxima que compartilha a mesma rede, considerando-se ainda que as conexões só são implementadas em trechos horizontais e verticais.

Tanto a opção pelo maior somatório de comprimentos, quanto a opção pela célula menos larga (se for o caso), primam pela minimização do comprimento total das interconexões do circuito.

Os critérios de desempate adotados resultaram de experiências realizadas com as estratégias de posicionamento apresentadas em [KAN 83] e [SCH 72b].

A figura 4.12 e a tabela 4.6 apresentam um exemplo de posicionamento de células em uma banda. A referida tabela mostra a evolução dos valores dos escores globais (EGL) a cada novo posicionamento e a atualização do

apontador de posicionamento da banda em função da largura da célula movida (veja tabela 4.2).



Ordem de Posicionamento: G1 G4 G5 G3 G6 G2

Figura 4.12 - Posicionamento em uma banda.

Tabela 4.6 - Escores globais e apontador de posicionamento.

CÉLULA	egl 1	egl 2	egl 3	egl 4	egl 5	egl 6
G1	(-1)	-	-	-	-	-
G2	-3	-3	-3	-3	-1	(1)
G3	-3	-2	0	(0)	-	-
G4	-3	(-1)	-	-	-	-
G5	-2	-2	(0)	-	-	-
G6	-4	-3	-3	-1	(0)	-
APONTADOR DE POSICIONAMENTO RESULTANTE	45	78	99	152	197	218

Obs: Empate entre G3 e G5 para egl3 resolvido por largura de célula (Veja tabela 4.2)

Neste ponto, cabe salientar que, assim como no particionamento, o resultado do posicionamento intrabanda é extremamente dependente do ponto de partida do processo. Se restrições de orientação oeste forem impostas, provavelmente elas determinarão as condições iniciais para o posicionamento. Já se o módulo não contar com restrições deste tipo, as primeiras células a serem posicionadas serão com certeza aquelas de menor complexidade funcional, que possuem um menor número de conectores e são menos largas

que outras células.

Completando-se o posicionamento absoluto de todas as células que compõem o módulo, a determinação de orientação destas é realizada.

Como visto na seção 2.2.2, o objetivo deste procedimento é minimizar o comprimento de interconexões pela determinação da localização definitiva do ponto inferior esquerdo de cada célula. No caso do posicionador TRANCA, somente espelhamentos sobre o eixo y são permitidos.

A idéia é obter o somatório dos comprimentos dos trechos de conexões sobre a célula - que partem dela ou a ela chegam, não interessando conexões que a cruzem (mesmo que haja ligação física). A avaliação é feita para a célula espelhada e não espelhada. Adota-se a configuração que resultar em menor comprimento de interconexões.

Já que neste ponto o assinalamento de conectores (seção 2.2.3) ainda não foi realizado, as posições consideradas para os pinos da interface da célula correspondem à média das coordenadas dos pinos funcionalmente equivalentes.

Como, em alguns casos, o somatório de comprimentos resulta em valores idênticos para as configurações espelhada e não espelhada, o desempate é realizado pela avaliação das possibilidades de implementação de redes da interface não orientadas e de conexões por justaposição com as células vizinhas na banda.

O anexo 4 apresenta os principais passos percorridos pelo procedimento de posicionamento intrabanda do subsistema POTRANCA.

As construções da linguagem NILOTRANCA que possibilitam a imposição de ordem de posicionamento na banda e de orientação de células são, respectivamente,

"ordering" e "mirrored/non_mirrored". Seu uso pode ser observado no exemplo de descrição fornecido no anexo 1.

Cabe ressaltar que, assim como no particionamento, o uso de construções impositivas para o posicionamento resulta em condições de contorno diferentes para o problema. Portanto, desde o início de seu tratamento, custos e escores de células ficam condicionados a estas imposições e, de um posicionamento livre de qualquer restrição a outro com apenas algumas imposições, as soluções obtidas podem diferir grandemente.

A figura 4.13 apresenta a saída gráfica do programa EXTRAMO para o posicionamento intrabanda correspondente à saída do particionador mostrada na figura 4.10. Observe a distribuição equilibrada da densidade de conexões horizontais e verticais e a redução do número de trilhas utilizadas para o roteamento intrabanda. Como a saída gráfica do subsistema de posicionamento não se preocupa com o uso efetivo das trilhas disponíveis na banda, tanto na figura 4.10, como na figura 4.13, as ligações apresentadas são apenas indicações das conexões a serem efetuadas, isto é, não correspondem ao roteamento real, o qual será executado pela ferramenta RETRANCA.

O anexo 8 apresenta o relatório de saída do posicionador intrabanda para este mesmo posicionamento.

4.4.2 RETRANCA

Uma vez que o conceito de transparência vertical não faz parte do subconjunto de características da metodologia TRANCA implementado pelo gerador TRAMO, células de interconexão precisam ser utilizadas para realização do roteamento.

Como visto na seção 3.3, as células de interconexão podem ser implementadas utilizando-se a camada de polissilício na direção ortogonal à estrutura de banda e

são posicionadas entre as células funcionais que compõem uma banda.

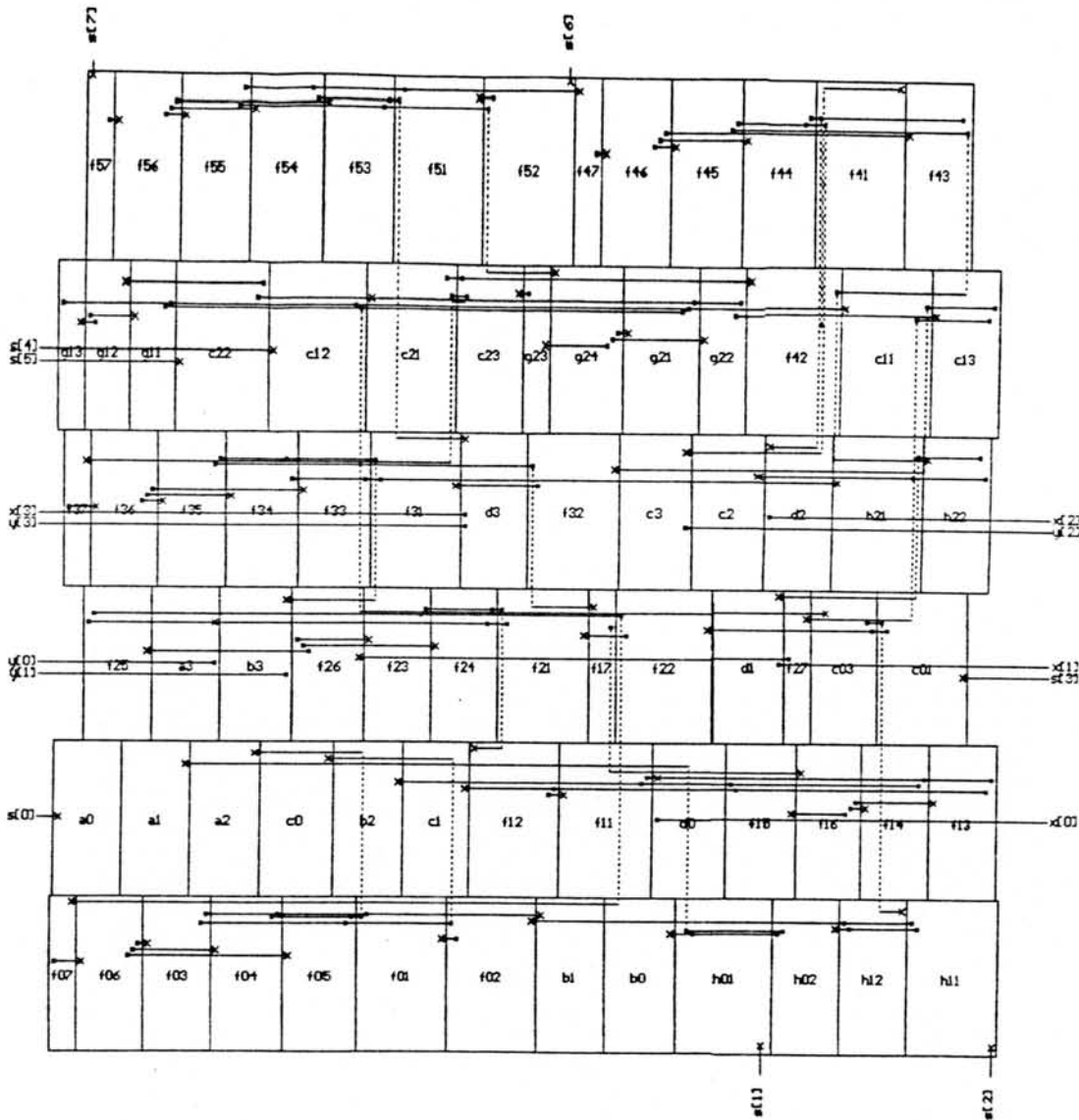


Figura 4.13 - Exemplo completo de posicionamento.

Em função da abrangência da ligação que uma célula de interconexão implementa, ela pode ser classificada como intrabanda, interbanda ou mista (figura 4.14). Células de interconexão intrabanda são utilizadas para a troca de percurso de trechos de redes internas à banda; células de interconexão interbanda passam ligações de uma banda à outra; por fim, células de interconexão mistas são células interbanda que, simultaneamente, são utilizadas para troca de trilha na banda das conexões que

elas implementam.

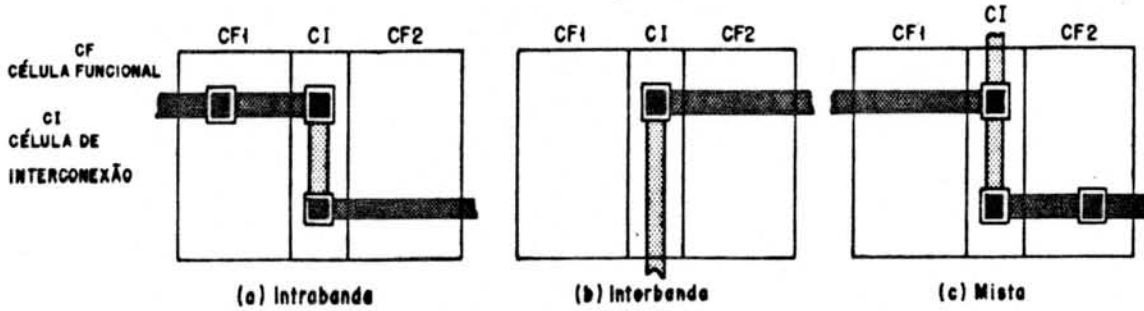


Figura 4.14 - Classes de células de interconexão.

Fazendo-se uma analogia com o problema de roteamento de canal (seção 2.2.4.3.2), podemos dizer que células de interconexão intrabanda correspondem a "doglegs". Já as células de interconexão interbanda correspondem às células de passagem utilizadas na abordagem "Standard Cell" como meio de comunicação entre canais de roteamento separados por uma fileira de células. Células de interconexão mistas, por sua vez, não possuem similares em outras abordagens.

No subsistema RETRANCA, cada célula de interconexão intrabanda é usada para completar a ligação entre os drenos dos transistores p e n, que configura a saída de uma célula (seção 4.3), e possibilitar que a rede considerada corra sobre outra trilha da estrutura (figura 4.15a). O uso de tais células de interconexão se justifica para os casos em que não é possível a realização de conexões por justaposição (figura 4.15b).

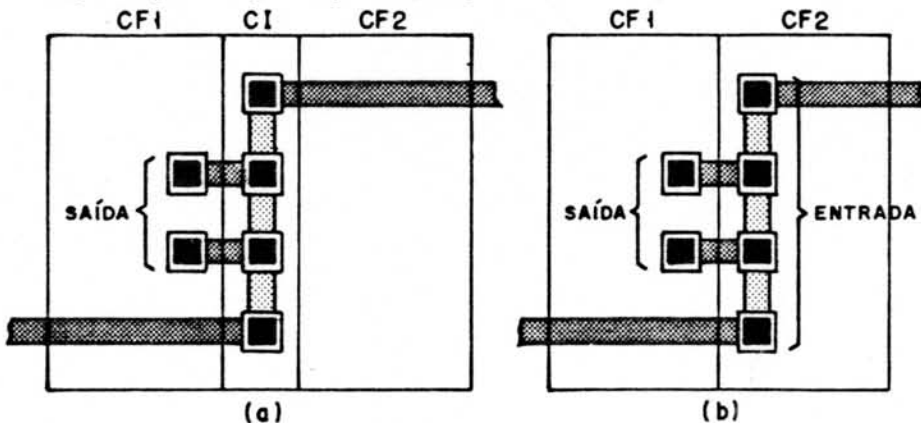


Figura 4.15 - Célula de interconexão intrabanda vs. conexão por justaposição.

O roteamento de redes divididas entre somente duas bandas adjacentes pressupõe a utilização de duas células de interconexão interbanda (ou mistas), cada qual localizada em uma das bandas e posicionadas exatamente uma sobre a outra (figura 4.16).

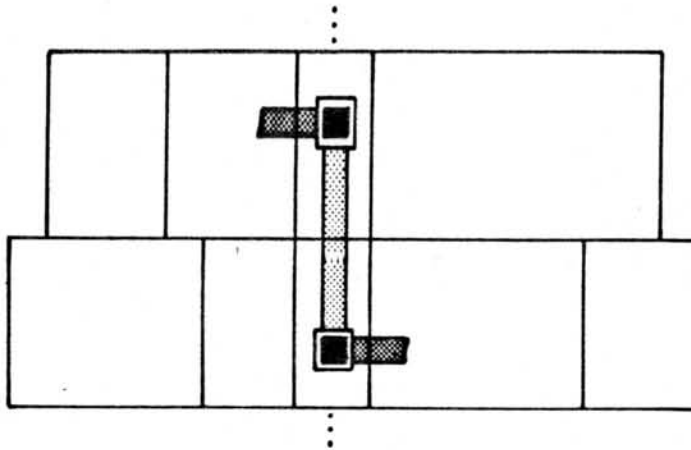


Figura 4.16 - Redes divididas entre bandas adjacentes.

A realização de conexões divididas entre mais de duas bandas, ou entre duas bandas não adjacentes, pode envolver números diferentes de células de interconexão. A solução mais imediata é a de utilizar uma célula de interconexão em cada banda e dispô-las umas sobre as outras (figura 4.17a). Uma solução de caráter local é considerar as bandas duas a duas e proceder recursivamente como na figura 4.16 (figura 4.17b). A solução mais inteligente, sem dúvida alguma, é a de avaliar as disponibilidades de espaços para alocação de células de interconexão em cada banda e, em função disto, posicioná-las livremente de forma a obter a menor perda de área possível (figura 4.17c).

A opção (a) da figura 4.17, apesar de utilizar um número menor de células de interconexão que a opção (b), é extremamente mais rígida que esta e pode resultar em layouts com áreas de implementação bem superiores ou com interconexões muito longas.

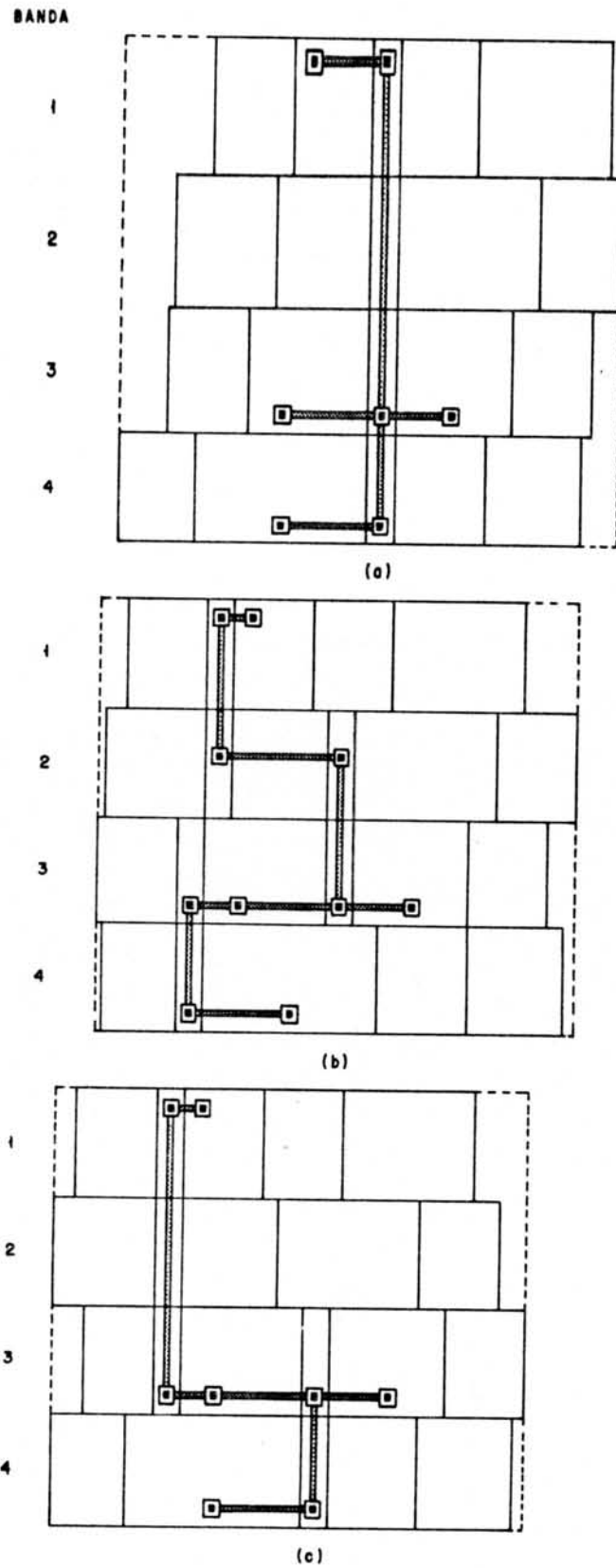


Figura 4.17 - Possibilidades de disposição de células de interconexão interbanda.

Para exemplificar esta afirmação, basta imaginarmos que, além da rede roteada na figura 4.17, tenhamos que realizar uma conexão entre as últimas células das bandas 2, 3 e 4. A figura 4.18 mostra as duas possibilidades para a opção (a) e a figura 4.19, a melhor solução para a opção (b).

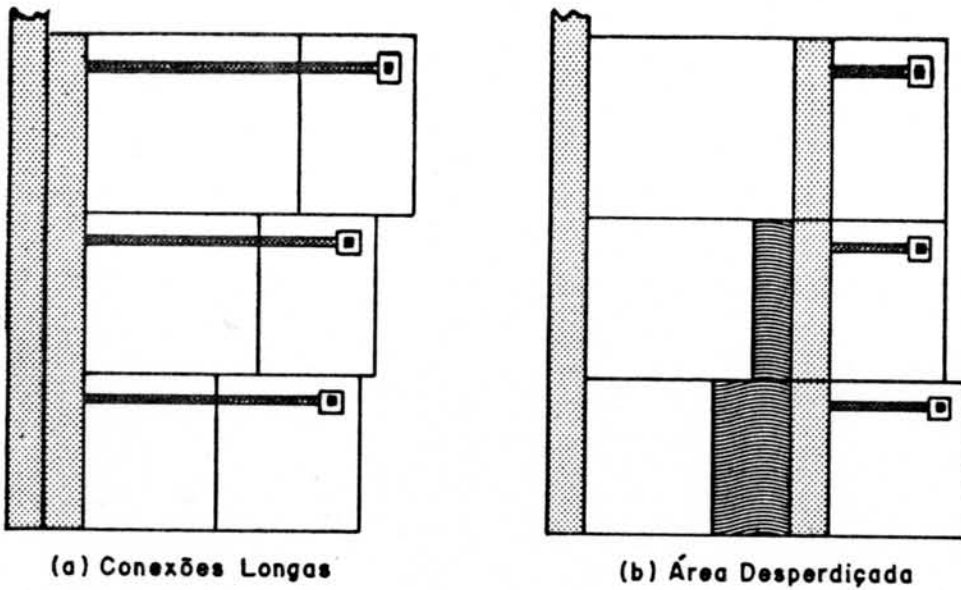


Figura 4.18 - Soluções com múltiplas células de interconexão alinhadas.

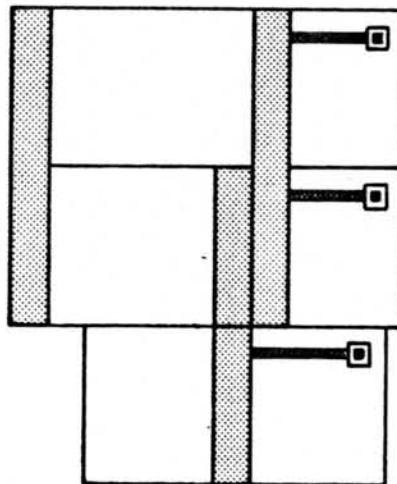


Figura 4.19 - Solução com pares de células de interconexão alinhadas.

Avaliando-se estas soluções, verificamos que,

quanto maior o número de redes divididas entre mais de duas bandas, tão mais oneroso se torna o alinhamento de um grande número de fronteiras de células.

É importante ressaltar que o desperdício de área (figura 4.18b) não é exclusividade da estratégia de múltiplas células de interconexão. Ele também ocorre quando se utiliza pares de células de interconexão, só que em proporções bastante menores.

Parece evidente que a melhor opção depende do caso e que, por isso, o ideal é utilizar uma estratégia mista para o ataque do problema. No caso de circuitos mais regulares, por exemplo, a utilização de múltiplas células de interconexão alinhadas é mais apropriada. Já para circuitos completamente irregulares, o uso de pares alinhados de células de interconexão tende a fornecer melhores resultados.

O fato dos circuitos alvos do RETRANCA serem blocos de lógica aleatória conduz à opção pela implementação de conexões por pares alinhados de células de interconexão. O estímulo de adjacência de redes entre bandas vizinhas, provido pelo particionador TRANCA, reforça a escolha. Por questões de redução da complexidade de programação, a primeira versão do roteador TRANCA não explora o espaço de soluções possíveis que utilizam, também, a estratégia de alinhamento múltiplo.

Estabelecidas as formas possíveis de implementação de conexões em módulos TRANCA, passamos à análise das estratégias básicas do procedimento de roteamento.

O primeiro passo é realizar o assinalamento de conectores (seção 2.2.3). Um assinalamento apropriado entre os pinos funcionalmente equivalentes de cada célula é imprescindível para evitar o congestionamento da estrutura de banda e a alocação desnecessária de células de

interconexão, como mostra a figura 4.20.

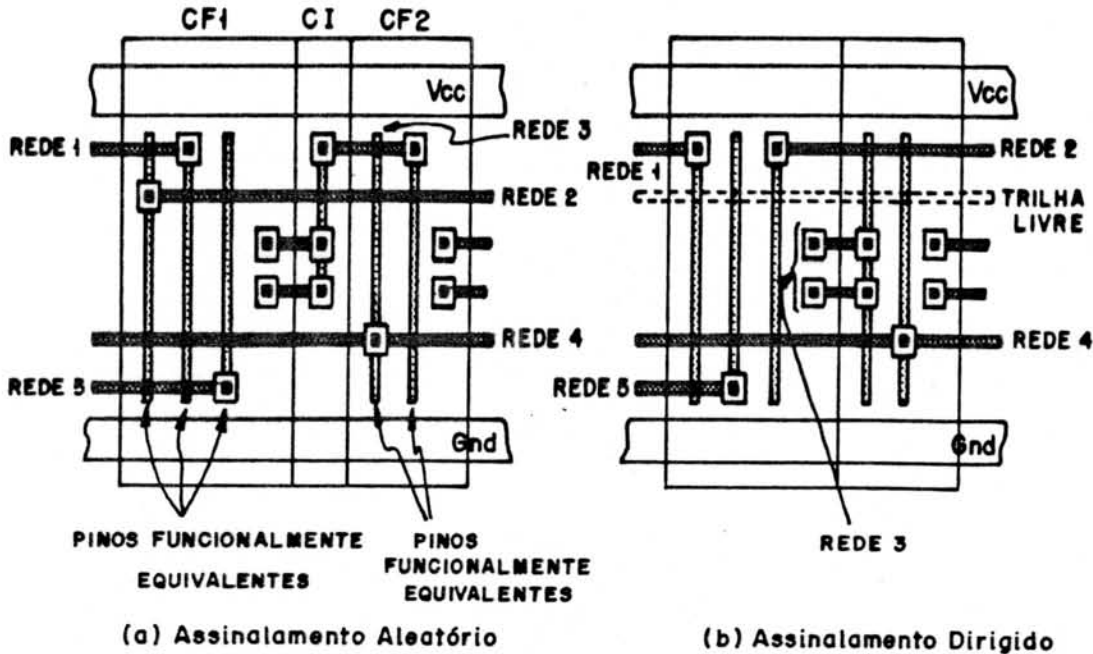


Figura 4.20 - O assinalamento de conectores no RETRANCA.

O assinalamento das saídas das células dá início ao processo. A seguir, se houver trilhas impostas para roteamento, o assinalamento é feito de forma a favorecer o comprimento das redes que devem correr sobre elas. As possibilidades de realização de conexões por justaposição regem o assinalamento da entrada situada na extremidade mais próxima da fronteira de cada célula. Por fim, em função da proveniência/destino das redes que ligam cada célula ao restante do circuito, estabelece-se pinos preferenciais para estas conexões. O desempate entre redes que "preferem" um mesmo pino é realizado a partir da priorização das conexões mais longas.

De posse da informação de que saídas não se justapõem à entrada da célula vizinha, ainda tenta-se encontrar caminhos diretos (sem troca de trilha) entre aquelas saídas que estejam disponíveis em uma única trilha e o conectores mais próximos que compartilham a mesma rede.

Esgotados todos os meios para evitar o uso de células de interconexão intrabanda, nada resta além de posicionar aquelas estritamente necessárias, e assinalá-las

com as redes que irão implementar. O posicionamento é feito sempre ao lado da extremidade da célula funcional onde se encontra disponível a saída que irá utilizar a célula de interconexão.

O passo seguinte consta da determinação de orientação para os sinais da interface livres de restrições. As redes da interface nestas condições, e que possuam células na primeira ou na última banda, passam a ter atributo norte e sul respectivamente. Às redes presentes em bandas intermediárias atribui-se orientação oeste ou leste. Se a distância à extremidade esquerda do módulo, da célula da rede de menor coordenada de posicionamento, for menor que a distância à extremidade direita do módulo, da célula da rede de maior coordenada de posicionamento, a orientação assumida para a rede é a oeste; caso contrário, a orientação é leste.

A partir do momento que algumas redes têm orientação norte e outras sul, a única forma de dispô-las nas extremidades referentes do módulo é através da realização de conexões ortogonais à estrutura de banda. A primeira providência é verificar a possibilidade de prolongar os pinos das células, relacionados a estas redes, sobre as linhas de alimentação da banda. Caso isto não seja possível, a solução é incluir novas células de interconexão intrabanda, que possibilitem a passagem dos sinais em polissilício sobre a linha de alimentação implementada em metal.

Quando as redes da interface norte/sul estiverem sujeitas a uma ordenação, a escolha do ponto de entrada/saída da rede do módulo tem que ser avaliada segundo as possibilidades oferecidas pelas células que as representam na banda. Todos os pinos na banda que compartilham redes sob tal restrição são, então, ordenados segundo sua posição e adota-se a seqüência de pinos que melhor atende a ordem requisitada na descrição.

O roteamento global das bandas sucede o tratamento de parte do problema de roteamento da interface. O objetivo é alocar células de interconexão interbanda, conforme a estratégia de pares alinhados descrita anteriormente. As bandas são tratadas duas a duas, recursivamente, até que todas as relações entre elas fiquem definidas no leiaute (figura 4.21).

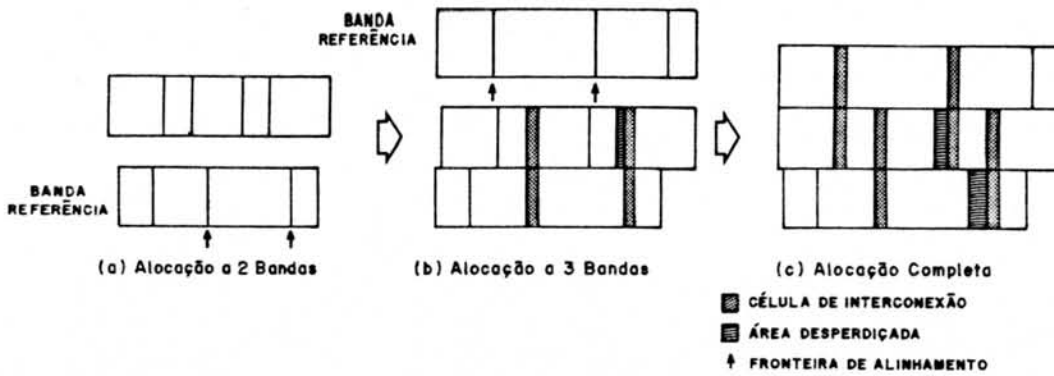


Figura 4.21 - Roteamento global das bandas.

À medida que células de interconexão intrabanda foram posicionadas em passos anteriores, as bandas são novamente centralizadas com relação à mais longa delas.

Para seleção do par de bandas que será considerado na alocação de células de interconexão, fatores de simplicidade relativa de roteamento (FSR) são calculados para cada banda. Observe-se que a alocação é tão mais simples em uma determinada banda, quanto menor o número de redes divididas da banda e quanto maior o número de possibilidades de posicionamento de células de interconexão. Desta forma,

$$FSR = (MLB / LB) \cdot (MLM / LM) \cdot (MNR / ND) \quad [XVII]$$

onde,

MLB = maior largura de banda;

LB = largura da banda considerada;

MLM = maior largura média de célula em uma banda;

LM = largura média de célula na banda;

MNR = maior número de redes divididas de uma banda;

ND = número de redes divididas da banda.

O par de bandas selecionado pressupõe sempre a adjacência entre as duas bandas e a maior soma de fatores de simplicidade possível. A banda que serve como referência para alocação é aquela que possui o menor fator de simplicidade do par.

A justificativa de optar-se pela alocação a partir das bandas mais simples de serem roteadas, é que as primeiras bandas submetidas ao processo sofrerão mais intensamente a perda de área pela busca de alinhamento entre fronteiras de células das bandas posteriores (figura 4.21). Como as bandas com maior FSR são normalmente menos largas e estão posicionadas mais próximas das extremidades superior e inferior do módulo, o menor retângulo envolvente do circuito tende a não ser afetado por esta perda associada.

A determinação do local onde cada par de células de interconexão deve ser posicionado é regida pelo deslocamento das células para o encaixe entre bandas e pelo comprimento da conexão a ser realizada. Estes fatores são avaliados para todas as fronteiras válidas entre células da banda de referência, optando-se pela posição que oferecer o melhor compromisso. O escore (EP) computado para cada posição é dado por:

$$EP = (N - 1) \cdot \text{deslocamento} + \text{comprimento} \quad [\text{XVIII}]$$

onde,

N = número de trilhas para roteamento da estrutura de banda.

O deslocamento equivale à distância exemplificada na figura 4.22a, se a alocação for feita da esquerda para a direita (avanço) na banda de referência, ou à distância mostrada na figura 4.22b, se o sentido de alocação for da direita para a esquerda (retrocesso). O produto "(N - 1) . deslocamento" da equação [XVIII] representa o acréscimo em comprimento que terão as outras (N - 1) redes da banda que venham a cruzar a posição que se avalia para alocação de um

par de células de interconexão.

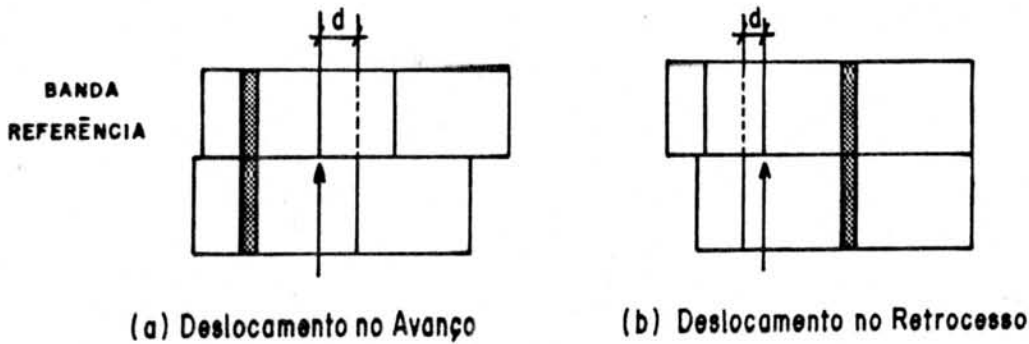


Figura 4.22 - Cálculo do deslocamento.

O comprimento utilizado na equação [XVIII] corresponde à distância entre a fronteira que se avalia e o pino mais próximo que compartilha a mesma rede, se este local não for cruzado pela rede considerada. Caso a rede não tenha células representantes na banda, o comprimento é calculado em função da distribuição dos conectores da rede no restante do circuito.

Antes de realizar a alocação definitiva das células de interconexão, avalia-se a melhor posição para cada rede dividida da banda de referência, como se as outras não existissem, e computa-se os valores de deslocamento para o procedimento em avanço e retrocesso. A escolha de um destes sentidos é baseada no menor valor resultante do acúmulo dos deslocamentos em cada sentido, como mostra o exemplo da figura 4.23. Observe que o acúmulo corresponde à perda de área nas bandas pelo alinhamento das fronteiras das células.

Se eventualmente já existirem áreas vazias internas a qualquer das bandas do par, estas são consideradas no acúmulo dos deslocamentos de células. Na figura 4.24 vemos que esta providência resulta no reaproveitamento de parte da área anteriormente sem utilidade.

Escolhido o sentido de movimento das células do par de bandas considerado, ordena-se os locais preferenciais para alocação de células de interconexão

segundo coordenadas crescentes, se o sentido for avanço, ou decrescentes, se for retrocesso. A partir daí, o primeiro local a ser efetivamente utilizado é aquele indicado pelo primeiro da lista ordenada.

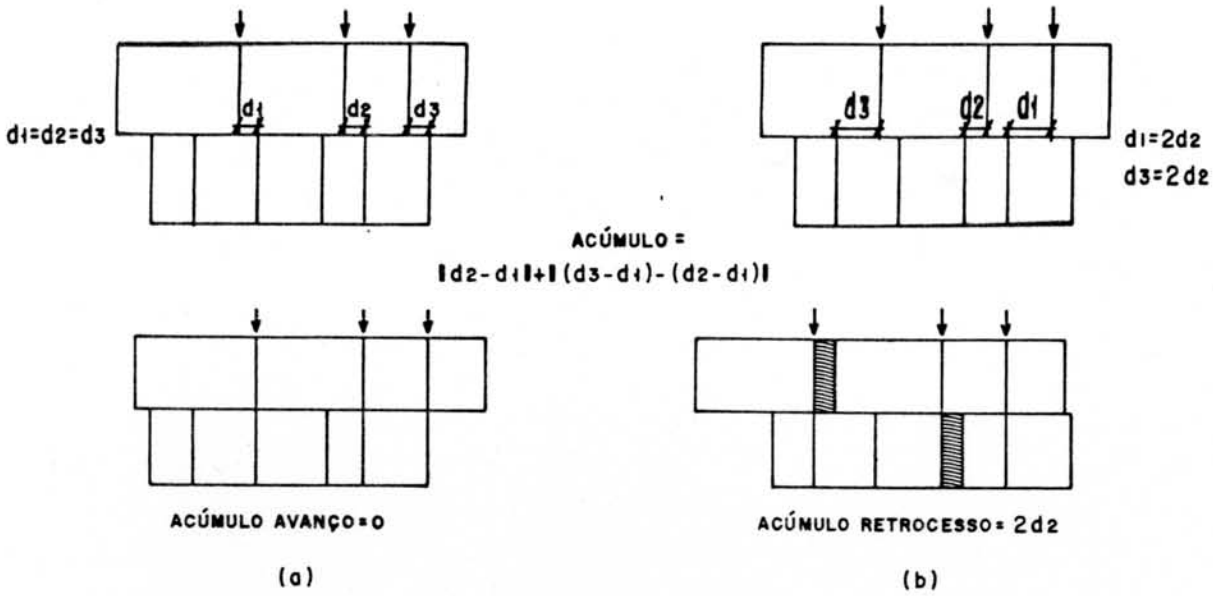
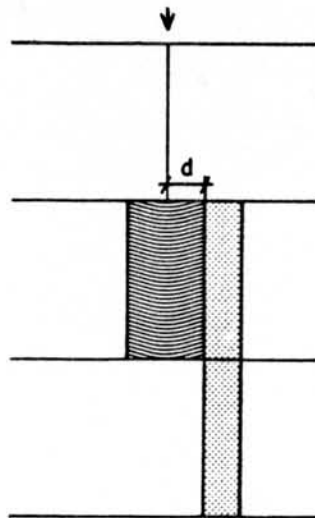


Figura 4.23 - Acúmulo de deslocamentos.



DESLOCAMENTO $\neq d$
DESLOCAMENTO = 0

Figura 4.24 - Reaproveitamento de área.

Alocando-se o primeiro par de células de interconexão nas bandas, as coordenadas de células e de pinos das redes divididas são atualizadas. A próxima iteração pesquisa locais preferenciais para as redes

restantes, a partir do local anteriormente utilizado, com base no escore dado pela equação [XVIII]. Os locais são reordenados e uma nova alocação é feita. O processo segue até que todas as células de interconexão tenham suas posições estabelecidas.

Encerrada a alocação para o par de bandas selecionado, o fator de simplicidade do conjunto é atualizado. Note-se que a partir do tratamento destas bandas, a relação topológica entre as duas passa a ser extremamente forte, ou seja, qualquer outro deslocamento de célula que uma delas venha a sofrer deve se refletir na outra banda, de forma a não causar o desalinhamento dos pares de células de interconexão estabelecidos entre elas. Justamente por este fato, o par de bandas passa às iterações seguintes do roteamento como um conjunto único - chamado de superbanda, e com um fator de simplicidade reduzido. O novo fator, função dos fatores das duas bandas que compõem o par (FSR1 e FSR2), é calculado por:

$$FSR = (FSR1 \cdot FSR2) / (FSR1 + FSR2). \quad [XIX]$$

Uma vez que a inclusão de células de interconexão interbanda causa a descentralização das bandas não consideradas na iteração, bandas e superbandas são recentralizadas ao final de cada alocação para um par de bandas.

As seleções seguintes de pares de bandas transcorrem da mesma forma como já descrito, sendo que superbandas passam a ser também candidatas no processo de escolha. À medida que se passa de uma iteração à outra, ou o número de superbandas cresce, ou o tamanho de uma delas (em número de bandas) cresce. Ao final do roteamento global entre bandas, teremos uma única superbanda que corresponde ao módulo como um todo.

Cada vez que uma superbanda é formada, ou simplesmente devido ao agregamento de uma nova banda, o

refinamento do roteamento é realizado. O objetivo único é tentar recuperar a área perdida pelo alinhamento de pares de células de interconexão. O refinamento é baseado na compactação da superbanda quando se verifica que a área total ou parte da área perdida em uma de suas bandas pode ser removida às custas de remoções encadeadas.

Um espaço livre situado entre dois pares de células de interconexão que relacionam uma banda com sua vizinha imediata só pode ser removido, se houver um outro espaço livre entre estes mesmos limites na banda vizinha, e assim sucessivamente para todas as integrantes subseqüentes da superbanda. Na figura 4.25a, por exemplo, observa-se que na banda 1 existe um espaço livre entre os pares 1 e 2 de células de interconexão; a remoção deste é possível porque na banda 2 há um espaço livre entre os pares 1 e 2 que pode ser suprimido. O referido espaço livre da banda 2 pode ser removido porque, na banda 3, existe um espaço entre os pares 3 e 4 que pode ser suprimido. Por último, o espaço da banda 3 pode ser removido porque existe um espaço livre entre os pares 5 e 6 na banda 4. Assim se completa o encadeamento, obtendo-se a compactação para a superbanda apresentada na figura 4.25b.

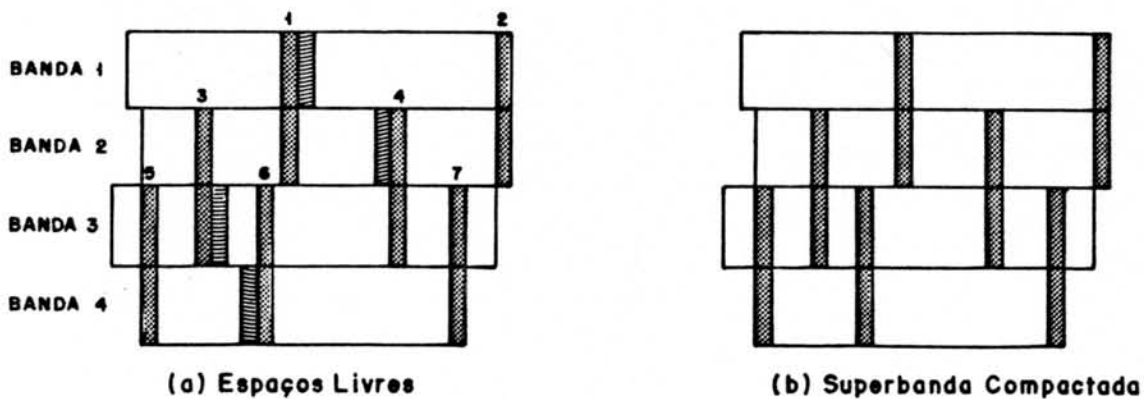


Figura 4.25 - Refinamento do roteamento global.

É importante salientar que este refinamento surge como alternativa para suprir as deficiências advindas da adoção de uma estratégia local para roteamento entre

bandas. A compactação a cada iteração aproxima a solução final obtida do resultado esperado para uma estratégia que trate todas as bandas simultaneamente.

Após o roteamento global, todos os pinos necessários para implementação física das conexões estão disponíveis em cada banda, de forma que o passo seguinte deve trabalhar a nível do roteamento intrabanda.

Por analogia com o problema de roteamento de canal (seção 2.2.4.3.2), observamos que canais de roteamento correspondem a estruturas de banda e que restrições verticais de roteamento inexistem, à medida que conectores de sinais diferentes jamais compartilham a mesma coluna nestas estruturas. Por outro lado, o problema de roteamento intrabanda em módulos TRANCA tem uma grande afinidade com as estratégias utilizadas por roteadores de propósito geral (seção 2.2.4.3.1): a procura de caminhos em um reticulado, onde alguns pontos - utilizados para roteamento interno à célula, configuram obstáculos à passagem de fios.

Desta forma, o procedimento para roteamento intrabanda resultou da conjunção das filosofias de roteamento de propósito geral e de roteamento de canal trilha a trilha.

O roteamento de cada banda parte da composição ordenada de matrizes das células que a constituem, sendo cada matriz a representante fiel da organização das conexões internas da célula sobre a estrutura de banda considerada (veja exemplo no anexo 2). As colunas destas matrizes correspondem aos nodos da interface das células; as linhas correspondem às trilhas da estrutura de banda.

A alocação de trilhas para a implementação de conexões é realizada da esquerda para a direita na banda, considerando-se para cada pino que se trata a localização do próximo conector da rede à direita e as trilhas livres

da estrutura no caminho entre eles. O esquema de prioridades para alocação, apresentado na seção 4.3, rege a seleção de trilha para roteamento.

A figura 4.26 apresenta o roteamento resultante para o exemplo da figura 4.12 (página 113), após o estabelecimento das orientações das células, o assinalamento de conectores e a alocação de células de interconexão intrabanda (CI).

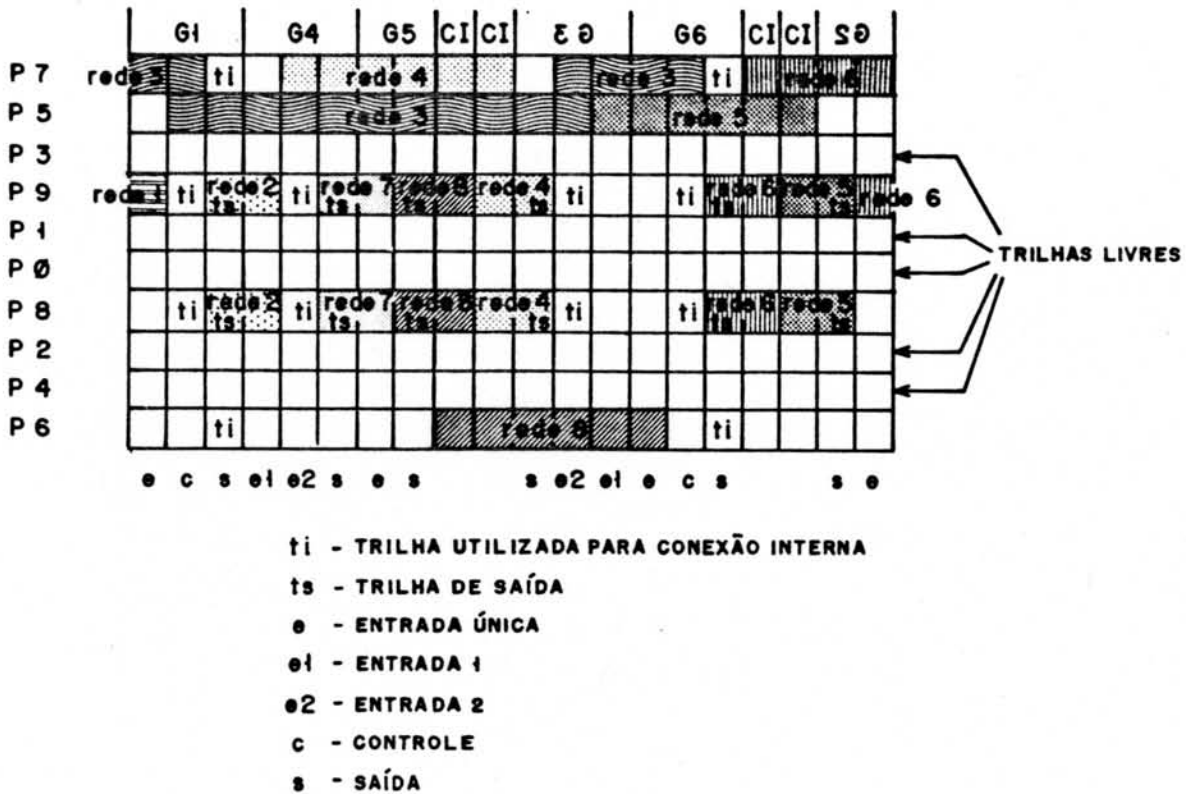


Figura 4.26 - Roteamento intrabanda.

Analisando-se o exemplo da figura 4.26, observa-se a semelhança do procedimento adotado com o algoritmo "left-edge" apresentado na página 67. No caso do RETRANCA, no entanto, não se assegura o roteamento com um número de trilhas igual à densidade do canal, pois, como se pode observar na figura 4.27, na presença de obstáculos à passagem de conexões, nem sempre a ordenação dos sinais segundo seus pinos mais à esquerda leva ao roteamento mais eficiente.

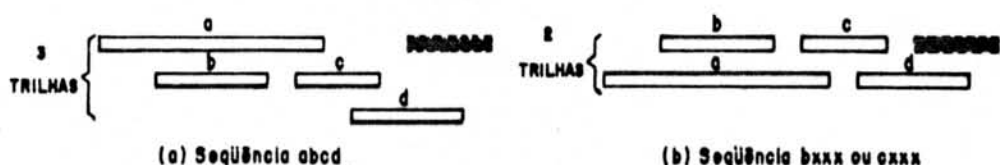


Figura 4.27 - Influência da ordenação no número de trilhas.

Segundo a estratégia apresentada, as bandas são roteadas uma a uma, sendo irrelevante a ordem de execução entre elas. O roteamento realizado, como se pode concluir imediatamente, é do tipo simbólico, uma vez que as regras tecnológicas em uso não são consideradas nesta fase.

Tão logo o roteamento intrabanda esteja concluído, dá-se início ao roteamento dos sinais da interface oeste/leste. O tratamento de eventuais ordenações, neste caso, é feito de maneira idêntica à ordenação da interface norte/sul. Com relação à orientação é evidente que somente os conectores mais à esquerda e mais à direita das redes da interface serão considerados, conforme se lide com a interface oeste ou leste respectivamente.

Antes da conversão de representação do roteamento para geometria, a composição das bandas é realizada de forma que as linhas de alimentação venham a formar dois pentes com dentes intercalados. Isto implica no espelhamento sobre o eixo x das células e de suas matrizes de roteamento, para as bandas pares ou ímpares, conforme o potencial de alimentação que deva estar disponível no topo do módulo.

O roteamento das linhas de alimentação fica limitado, então, ao prolongamento, em cada banda do circuito, das linhas de VCC, em um sentido, e de GND, em outro sentido, e na geração de retângulos em metal, situados nas laterais do módulo, que implementem a ligação física entre linhas de mesmo potencial.

A geração de uma representação simbólica para o módulo e do leiaute definitivo encerram o procedimento de roteamento.

Na figura 4.28, pode-se observar parte da saída simbólica gerada para o roteamento relacionado ao posicionamento da figura 4.13 (página 116). Como se vê, abstrai-se os detalhes de implementação de cada célula, representando-as por seus retângulos envolventes e pelos **nodos** da sua interface. A saída simbólica apresenta o posicionamento das células funcionais e das células de interconexão, e o roteamento intrabanda completo.

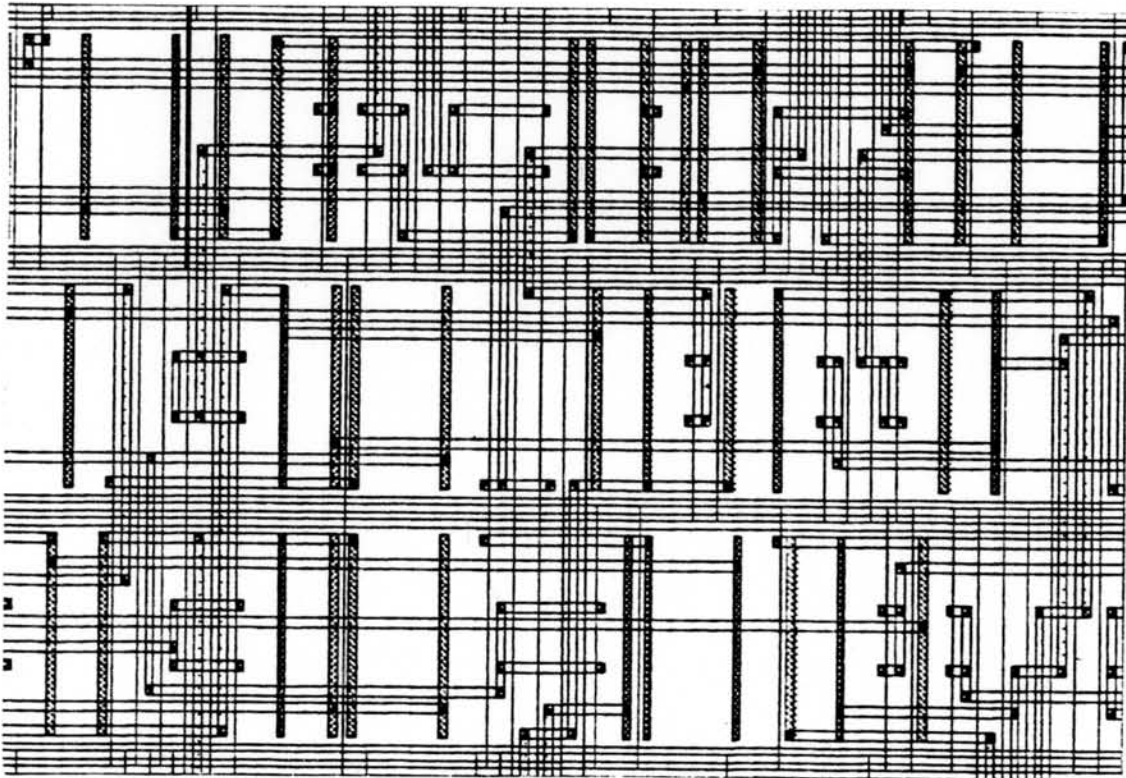


Figura 4.28 - Saída simbólica.

Uma vez que a representação simbólica se concentra exclusivamente em informações pertinentes ao roteamento, é extremamente mais fácil para o projetista avaliar o roteamento sobre ela, do que sobre o leiaute completo. Caso sejam verificadas possibilidades de otimização, imposições de trilhas para roteamento podem ser

feitas utilizando-se a construção "track" da linguagem NILOTRANCA (veja anexo 1).

Na figura 4.29 aparece o leiaute completo para o módulo descrito no anexo 6 e utilizado como exemplo na exposição dos subsistemas POTRANCA e RETRANCA. O anexo 9 apresenta o relatório de roteamento resultante para o leiaute da figura 4.29.

Uma visão global do procedimento realizado pelo RETRANCA é mostrada no anexo 5 sob forma algorítmica.

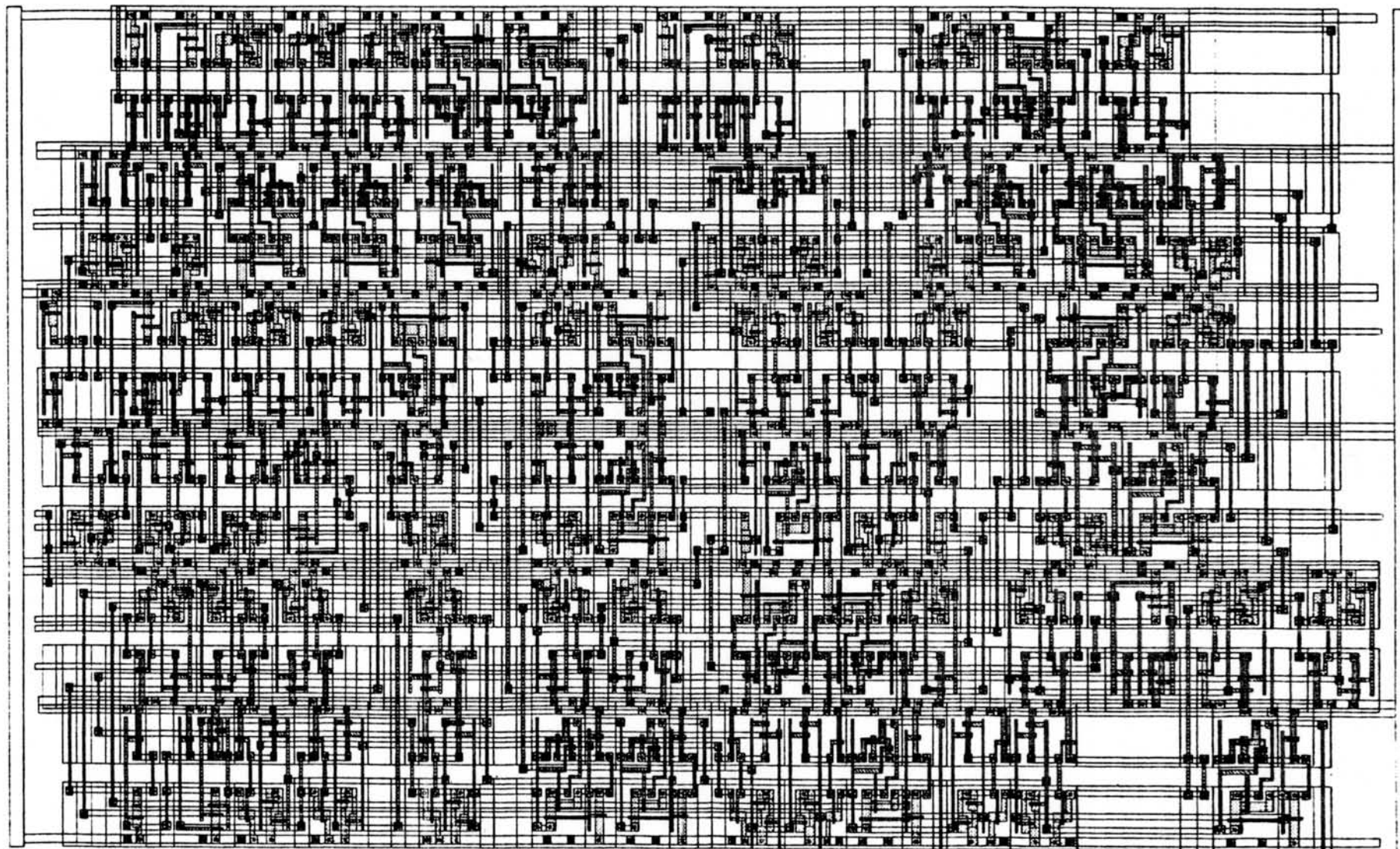


Figura 4.29 - Exemplo de leiaute completo.

4.5 Considerações sobre a Implementação

A própria ordem em que os subsistemas do gerador TRAMO são apresentados neste capítulo sugere a cadeia de procedimentos percorrida durante a síntese do leiaute de um módulo TRANCA. A figura 4.30 apresenta a seqüência de procedimentos para projetos que utilizam somente células já constantes na biblioteca.



Figura 4.30 - Cadeia de síntese de leiaute.

A interface de gerenciamento desenvolvida procura retratar fielmente a citada cadeia de procedimentos, incluindo recursos textuais e gráficos que facilitam o entendimento entre o projetista e o sistema [BAG 89]. Entre estes recursos constam: menus horizontais e verticais, acionamento por teclas e "mouse", "help", janelas de mensagens de execução e de erros, teclas especiais para seleção direta de funções, e entrada de parâmetros. O anexo 10 apresenta a seqüência de telas que caracteriza a síntese de leiaute realizada passo à passo.

A distribuição em diretórios dos arquivos que compõem o sistema, e daqueles gerados pelos procedimentos da figura 4.30, reflete a estruturação do ambiente de projeto, conforme se vê na figura 4.31.

Com relação aos arquivos de comunicação entre ferramentas, a figura 4.31 se limita a citar a origem destes, já que o seu destino se conclui a partir da análise

da figura 4.30. Acrescente-se que, tanto a saída do particionador, como a do posicionador, são também entradas para o programa de exibição gráfica EXTRAMO; no caso das duas saídas do roteador, os arquivos gerados são entradas para um editor de máscaras.



Figura 4.31 - Organização de arquivos em diretórios.

Os arquivos do tipo relatório são fornecidos como complemento à informação de resultados apresentada de forma gráfica.

No gerador TRAMO a síntese de leiaute pode ser realizada passo à passo, sob a supervisão do projetista, ou pode ser completamente automática. Em ambos os casos se trabalha com uma única versão de projeto.

Quando, na síntese supervisionada, o projetista realimenta o sistema com novas informações, a consistência de todo o processo precisa ser verificada e providências para mantê-la devem ser tomadas. Assim, se após a realização do posicionamento, o usuário modifica as restrições de interface na descrição e dispara a execução do roteamento, o gerenciador de ambiente orienta a síntese no sentido de retomar a compilação, o particionamento e o posicionamento, antes de partir para o roteamento; já se o projetista resolve alterar simplesmente a relação-a4680Hde aspecto do módulo e dispara novamente o posicionamento, a compilação não é realizada mas, antes de reposicionar as

células, o particionamento é executado para a nova restrição de forma.

A maioria das consistências da descrição feita pelo usuário é realizada a nível de compilação. No entanto, algumas falhas só podem ser detectadas mais adiante. Este é o caso das restrições posicionais e de trilhas para roteamento, que não podem ser consistidas antes que se saiba o número de bandas e de regiões do circuito, o número de células em cada partição ou a disponibilidade de trilhas na estrutura de banda.

A nível de técnicas de programação, é importante citar que as estruturas de dados utilizadas nas ferramentas apresentadas valem-se das características de redundância de informação e de encadeamento duplo. A redundância de informação reduz significativamente a profundidade de laços (laços internos a laços); o encadeamento duplo, por sua vez, agiliza ordenações e pesquisas mais elaboradas. Se por um lado ganha-se em velocidade de processamento, por outro lado perde-se em volume de dados armazenados.

As estruturas de dados presentes em todas as ferramentas de síntese do gerador TRAMO são um vetor de células, um vetor de redes e um vetor de bandas. Outras estruturas particulares a procedimentos internos aos diversos subsistemas são alocadas e desalocadas dinamicamente.

Com vistas à portabilidade das ferramentas para outras máquinas, os programas foram escritos em linguagem C, padrão Kernighan-Ritchie [KER 78]. A versão atual do gerador roda em máquinas do tipo IBM-PC compatível, prevendo-se a instalação em estações de trabalho.

5 AVALIAÇÃO DA FERRAMENTA E DA METODOLOGIA

Só o fato de se dispor de uma ferramenta de síntese que possibilite a concepção automática de leiaute já justifica, sem dúvida alguma, o seu desenvolvimento. A atividade entediante de desenho de máscaras, que outrora consumia dias de trabalho do projetista, passa a ser realizada em alguns minutos, ou em poucas horas (em função do nível de exigência do usuário), resultando em uma maior dedicação à atividade mais nobre de projeto do sistema.

Evidentemente, o uso disseminado da ferramenta só ocorre a partir do momento que ela conquista a confiança dos projetistas, primeiro garantindo a correção de leiaute, e depois assegurando a geração de resultados que se aproximam dos resultados produzidos pelo ser humano. Tanto uma característica, como a outra, só pode ser atingida através da prática, isto é, o próprio uso da ferramenta é que oportuniza a detecção de falhas nos procedimentos, a avaliação de resultados e a realimentação do seu projeto pela sugestão de melhorias nas estratégias adotadas.

No caso do gerador TRAMO, procurou-se realizar algumas experiências iniciais na tentativa de: 1) obter uma primeira avaliação da ferramenta e de seu estilo de projeto; 2) obter projeções relacionadas ao uso da metodologia TRANCA como um todo; e, 3) formular propostas para futuras otimizações ou novas implementações.

Desta forma, as primeiras experiências, realizadas durante o desenvolvimento do protótipo, visaram única e exclusivamente a depuração dos programas e a avaliação absoluta das estratégias, ou seja, a verificação ou não de que os procedimentos implementados atingiam os objetivos especificados. Nesta fase, o circuito alvo adotado tinha que ser um representante típico de blocos de lógica aleatória e devia possuir um número pequeno de elementos, de forma a possibilitar uma visão global do

módulo pelo projetista da ferramenta. Optou-se, então, pelo decodificador de instrução do circuito Micro-Controlador Industrial [SIL 87]. Como originalmente o decodificador de instrução é implementado por um PLA, fez-se necessária a conversão prévia da lógica a dois níveis que o caracteriza para uma lógica multinível equivalente. Alguns resultados obtidos para o particionamento e o posicionamento intrabanda de tal circuito aparecem em [LUB 89b]; com relação ao roteamento, um leiaute em 4 bandas é apresentado em [LUB 89a].

As experiências seguintes deviam preocupar-se com a validação das idéias de planejamento topológico - embutidas na ferramenta, com a comparação dos resultados fornecidos pelo gerador TRAMO com leiautes concebidos "manualmente" e leiautes utilizando a abordagem "Standard Cell", e com a comparação com outros sistemas de síntese automática. O objetivo, nesta nova fase, passa a ser a demonstração de competitividade da ferramenta, no sentido de impulsionar o seu uso e de estimular a sua otimização.

Como será visto nas seções seguintes, tais experiências foram orientadas procurando-se explorar alguns pontos do problema isoladamente, para facilitar a análise dos resultados. Os circuitos utilizados são de pequena complexidade, principalmente pela absoluta falta de amostras maiores, cuja documentação permitisse a síntese de leiaute sem a análise prévia do projeto lógico. Além do mais, mesmo que exemplos mais complexos estivessem disponíveis, em função do seu porte, talvez a máquina hospedeira da ferramenta não comportasse a sua síntese.

5.1 Experiência com o Circuito CONVERSO

Este circuito mapeia sinais analógicos em uma escala de 16 valores digitais, ou seja, trata-se de um conversor analógico-digital de 4 bits. O protótipo desenvolvido tem por objetivo o estudo de uma técnica de

conversão mais adequada à implementação em tecnologias digitais [DOS 90].

A parte operativa do conversor consta basicamente de um circuito de referência a resistor, de uma série de comparadores por balanço de cargas e de uma matriz de chaves CMOS. A codificação do sinal analógico é feita de forma serial-paralela, a partir da seleção de tensões de referência adequadas.

A parte de controle é responsável pelo seqüenciamento das fases de comparação, pelo chaveamento das tensões de referência e pela codificação digital das saídas dos comparadores. O esquemático e a descrição NILOTRANCA desta parte do circuito aparecem no anexo 1.

Como a implementação física de resistores e capacitores em circuitos integrados consome grande área em silício, é natural que a parte operativa ocupe a maior parte do chip. Nada mais lógico, então, que projetar a parte operativa, procurando-se a melhor topologia para ela, e submeter a parte de controle às restrições de planejamento topológico impostas pela porção analógica do circuito.

Segundo esta filosofia, a parte operativa foi projetada "manualmente" e a parte de controle através do gerador TRAMO. A interface para os sinais de controle da parte operativa ficaram disponíveis na extremidade inferior desta, conforme mostra a planta-baixa da figura 5.1.

Visando a obtenção do ótimo global para o circuito, abriu-se mão da opção de menor área para a parte de controle. A geração do leiaute foi realizada em 2 bandas, envolvendo a orientação e ordenação de conectores de sua interface para reduzir a área de roteamento entre os dois blocos. Todas as restrições constantes na descrição do anexo 1, que não se relacionam à interface, nela constam simplesmente a título de exemplo. Durante a síntese de

leiaute tais restrições não foram consideradas.

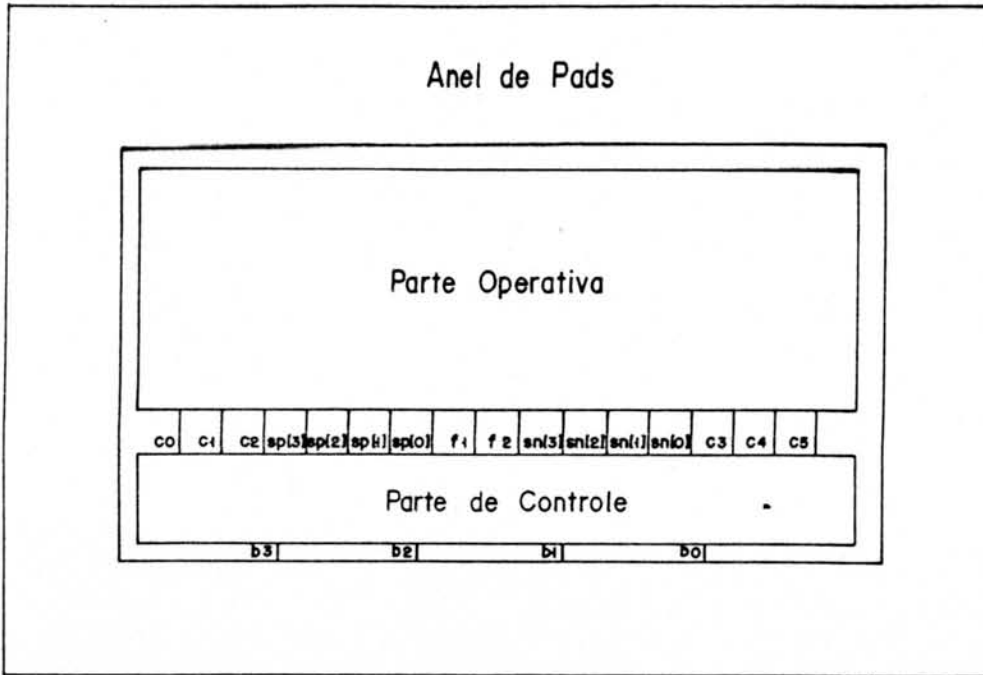


Figura 5.1 - Planta-baixa do conversor.

O leiaute final do circuito, cuja parte de controle consta de 254 transistores, aparece na figura 5.2. Observa-se, pelo leiaute obtido, a importância da geração de módulos em função do contexto. Desta forma, confirmam-se as vantagens da síntese de leiaute a partir de restrições de interface resultantes de um planejamento topológico prévio.

Um aspecto interessante a considerar é que, apesar das bandas geradas possuírem um comprimento muito superior ao valor de L_m (equação [IX] - página 100) para a tecnologia, a opção de particionamento em uma única região não resultou em saturação das estruturas de banda. Lembrese, ainda, que a transparência média das células da biblioteca é de 63% (seção 4.3), e portanto inferior ao recomendado pela metodologia TRANCA (70%).

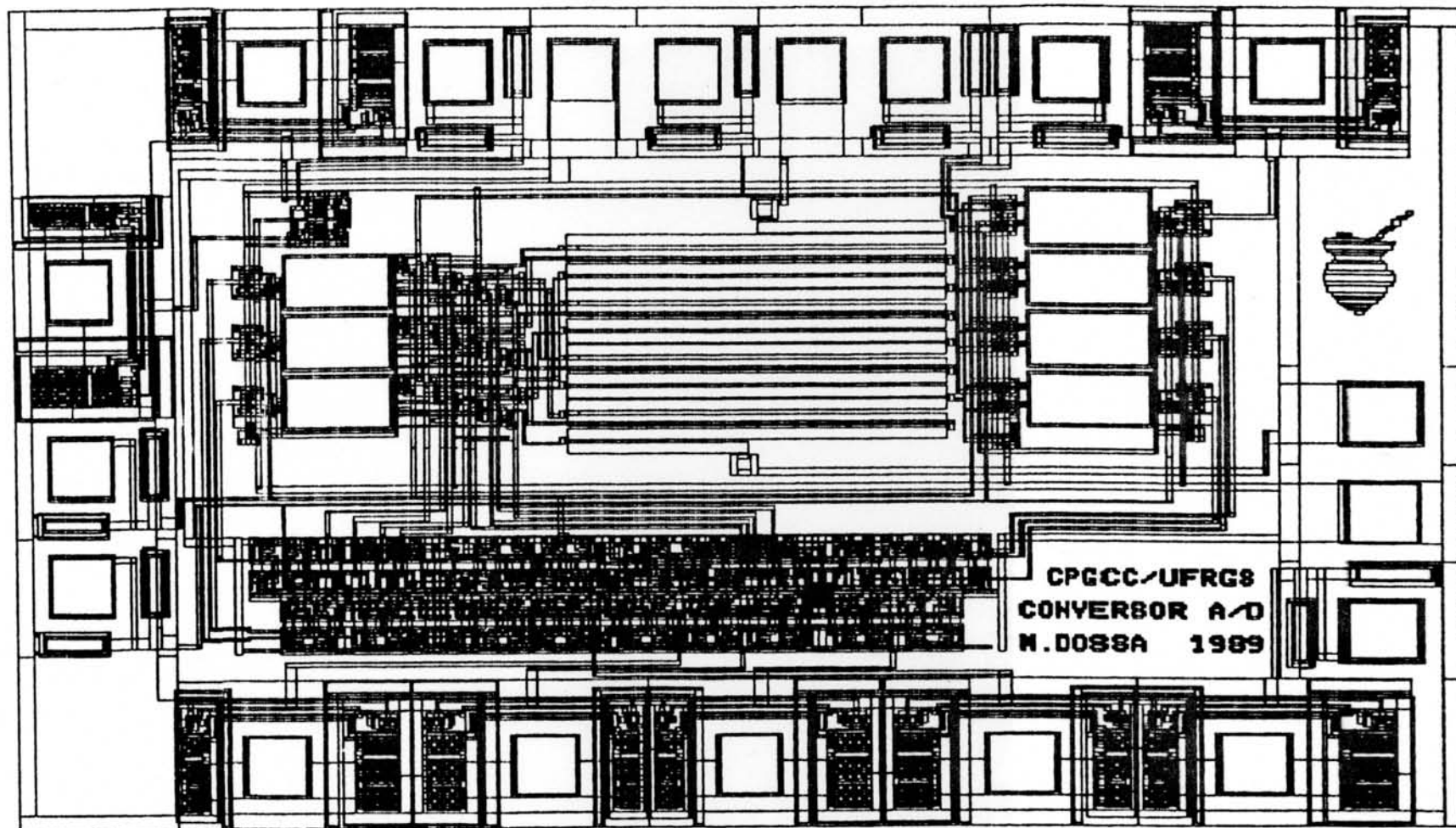


Figura 5.2 - Leiaute final do conversor.

A saturação das estruturas de banda não ocorre, neste caso, devido a dois fatores básicos: 1) a maioria das redes do circuito ligam somente duas células; e, 2) as trilhas da estrutura não são utilizadas para conexões entre células de interconexão interbanda que implementam redes sem células funcionais na banda. À medida que o número de bandas cresce, este último problema se agrava.

O circuito brevemente apresentado nesta seção faz parte do CMP Francês de setembro de 1989.

5.2 Experiência com o Circuito MODEM

O circuito MODEM trata-se de um ASIC para codificação e decodificação de códigos em modems de banda base. Sua função básica é converter dados em forma digital para forma bipolar ou ternária, e vice-versa [REI 89b].

O circuito consta de dois blocos funcionais completamente disjuntos: um codificador e um decodificador. O seu projeto lógico resultou na utilização de 41 elementos, entre portas lógicas e flip-flops, ligados entre si por intermédio de 60 redes.

O leiaute original do circuito MODEM, com 522 transistores distribuídos em 5 bandas, utiliza as células da biblioteca TRANCA, as quais já estavam disponíveis na época do projeto. A composição de instâncias destas células foi realizada "manualmente", valendo-se das idéias de concepção de leiaute da metodologia TRANCA.

A existência prévia de tal leiaute vem de encontro à necessidade de comparação entre os procedimentos de posicionamento e roteamento realizados "manual" e automaticamente.

Uma vez que o módulo gerado era o único bloco funcional do circuito, a síntese automática de leiaute foi realizada sem considerar restrições de orientação e

ordenação de redes da interface.

Da comparação entre os leiautes em questão, resulta que o módulo gerado automaticamente possui uma área cerca de 11% superior à do módulo concebido "manualmente". Analisando-se as figuras 5.3 e 5.4, observa-se, também, que a transparência final do módulo automático (3 trilhas livres) é idêntica à do módulo "manual".

Se considerarmos que a versão "manual" utiliza, em alguns trechos de redes, o conceito de transparência vertical em metal 2 para roteamento interbanda, concluímos que o leiaute gerado automaticamente é bastante satisfatório.

5.3 Experiência com o Circuito TREVO

O circuito TREVO consiste de um multiplicador combinacional de 4 X 4 bits, cujo projeto lógico utiliza técnicas de aceleração de carry. A síntese de leiaute do circuito original foi realizada em duas versões: uma "manual" e outra semi-automática, ambas sob o estilo "Gate-Array" de projeto [PER 89a].

A versão semi-automática, que utilizou ferramentas de PAC não integradas entre si, teve por objetivos básicos a comparação com a versão "manual" e a avaliação das dificuldades de trabalho em um ambiente não integrado de projeto. O subsistema de particionamento do gerador TRAMO foi utilizado neste estudo de caso. Segundo os dados fornecidos em [PER 89a], a solução obtida pelo uso desta ferramenta resultou em um aproveitamento de área - tanto a nível de posicionamento relativo, como a nível de roteamento, muito próximo do obtido na versão "manual".

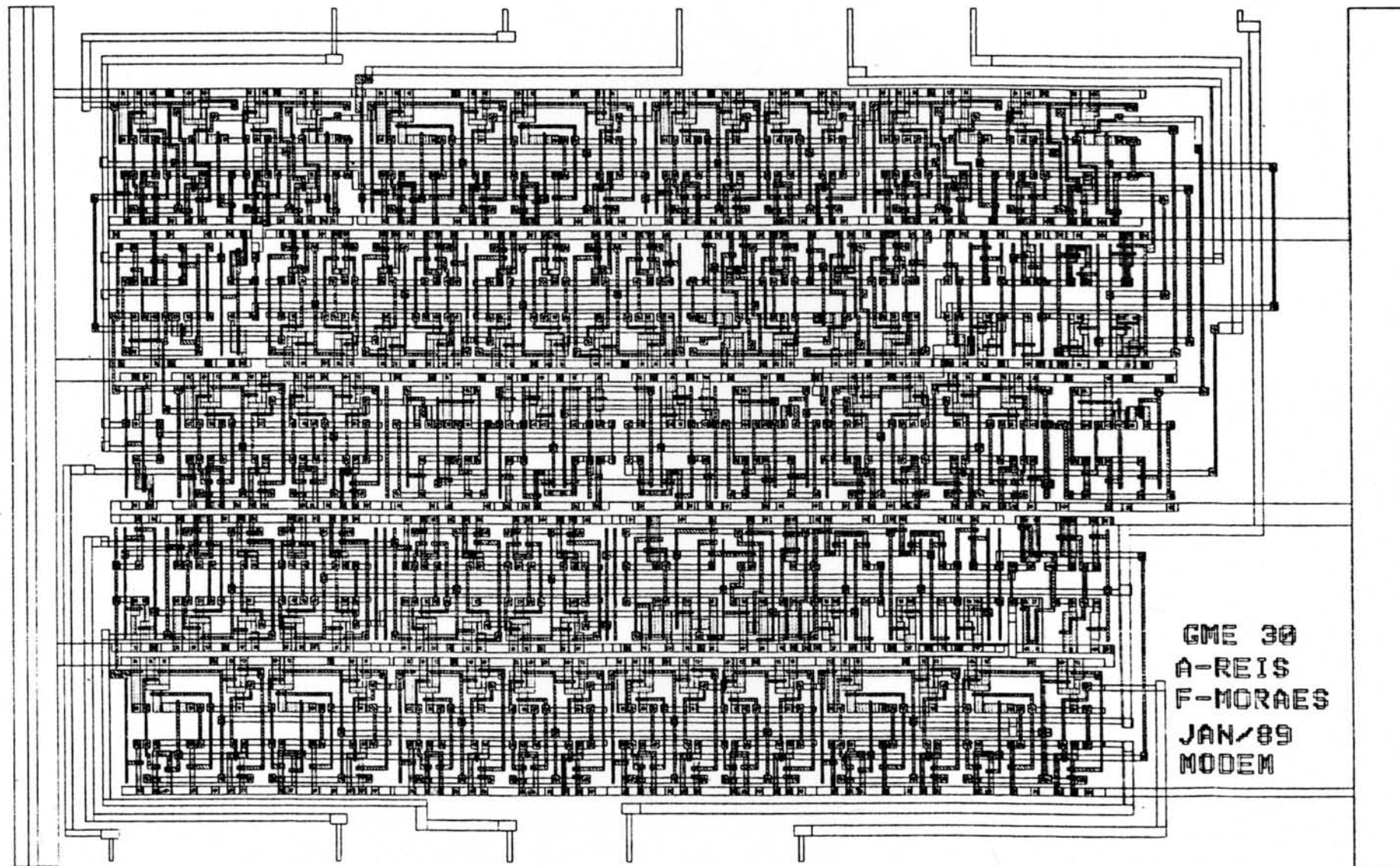


Figura 5.3 - Leiaute manual do circuito MODEM.

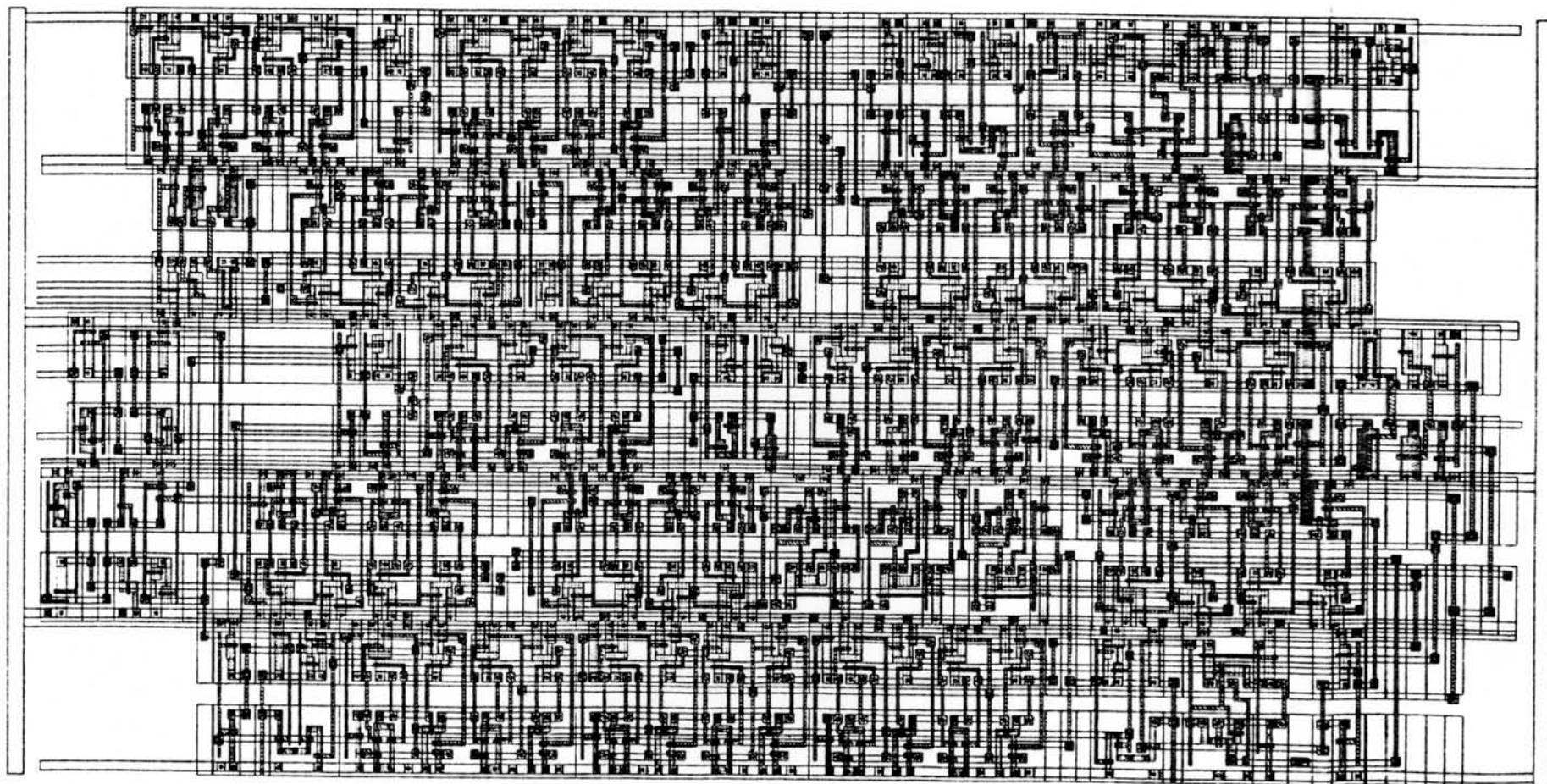


Figura 5.4 - Leiaute automático do circuito MODEM.

Posteriormente ao projeto original do circuito e ao estudo de caso mencionado, procurou-se realizar uma outra experiência que envolvesse todos os procedimentos de síntese da ferramenta TRAMO, e que resultasse na obtenção de dados que possibilitassem uma comparação mais significativa entre a metodologia TRANCA e o estilo de projeto "Standard Cell".

Partindo-se, então, do mapeamento do projeto lógico original na biblioteca de células TRANCA, obteve-se uma descrição do módulo com 79 portas lógicas e 87 ~~padões~~ (veja anexo 6), o que totaliza 436 transistores a nível de leiaute.

A experiência de reprojeção constou, basicamente, de: 1) geração do leiaute TRANCA para o módulo em 3, 4, 5, 6, 7 e 8 bandas; 2) estimativa de área e relação de aspecto para um leiaute equivalente na abordagem "Standard Cell" ; 3) comparação dos resultados obtidos; e, 4) projeção de resultados esperados pelo uso de transparência vertical na síntese de leiaute.

Uma vez que no GME/UFRGS não se dispunha de uma biblioteca de "standard cells", nem tampouco de ferramentas de PAC integradas que possibilitassem a síntese automática de leiaute segundo esta abordagem, optou-se pela realização de estimativas criteriosas para as diferentes formas do módulo a serem comparadas.

Primeiramente, projetou-se uma biblioteca de "standard cells" convencionais, constituída pelas células necessárias ao projeto do circuito TREVO e por uma célula funcionalmente equivalente à célula de maior complexidade topológica da biblioteca TRANCA. Esta última célula, apesar de não ser utilizada no projeto, é quem determina a altura da biblioteca "Standard Cell" equivalente à biblioteca TRANCA.

Todas as "standard cells" foram projetadas via

editor simbólico, segundo os seguintes critérios: 1) o desenho do leiaute da célula de maior complexidade topológica procurou a maior compactação possível na direção vertical, determinando assim uma altura mínima para a biblioteca; 2) a partir da obtenção desta altura, o projeto das outras células aproveitou ao máximo esta dimensão, na busca de larguras mínimas para elas. O uso de tais critérios assegurou a geração de células com leiautes tão compactos quanto leiautes gerados via editor de máscaras.

A biblioteca obtida (ANEXO 11) resultou em células 20% mais baixas e 4% em média mais largas que as células TRANCA - portanto, com áreas de cerca de 83% das áreas das células TRANCA.

A partir das dimensões das células da biblioteca "Standard Cell", obteve-se a área ocupada pelas fileiras nos leiautes dos módulos pertinentes, utilizando-se as mesmas estratégias de posicionamento do gerador TRAMO. Somente as células de interconexão necessárias à comunicação de canais consecutivos foram consideradas neste cálculo.

Com relação ao roteamento, a estimativa das dimensões dos canais foi efetuada utilizando-se os mesmos princípios para a realização de conexões intrabanda do gerador TRAMO. Manteve-se, desta forma, o roteamento linha a linha, a alternância de contatos e as características de transparência do módulo (como se o roteador de canal reservasse trilhas livres entre fileiras de células). Note que, com esta iniciativa, a abordagem convencional de síntese de leiaute saiu favorecida, uma vez que a área resultante para o roteamento entre supostos blocos não adjacentes ficou reduzida (figura 5.5).

Para cada fileira de células, então, obteve-se a estimativa de altura de um canal, onde seriam implementadas as conexões entre células da fileira e destas com as

células da fileira seguinte. Sendo assim, pode-se esperar as topologias apresentadas na figura 5.6 para os módulos a serem comparados. Na figura, observa-se a diferença de relações de aspecto para um mesmo número de bandas, fato resultante das características individuais de cada metodologia.

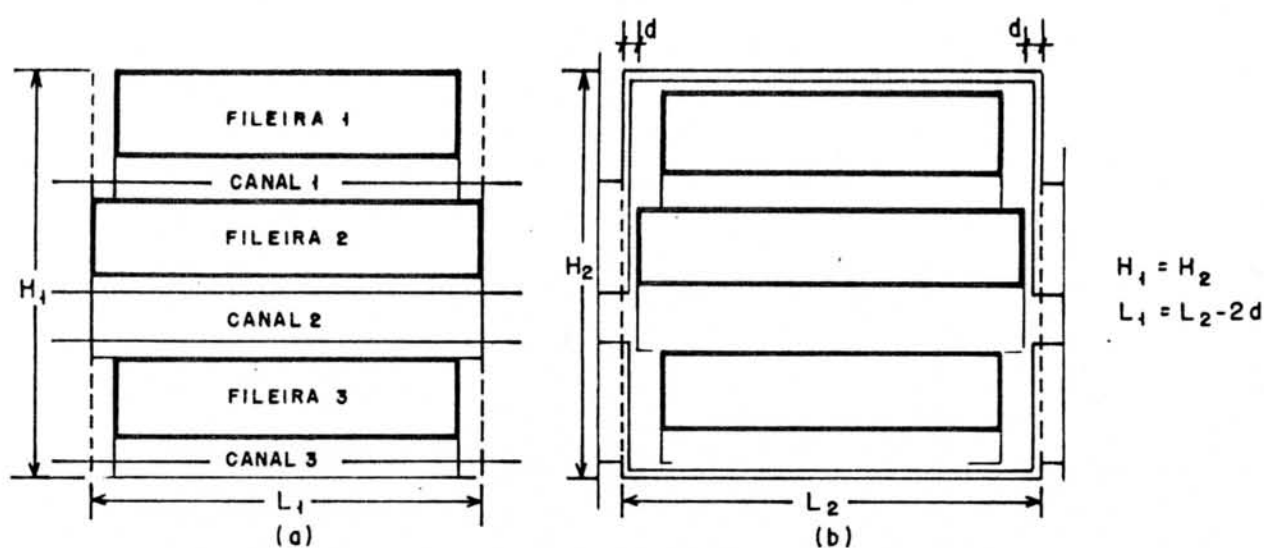


Figura 5.5 - Ganho em área pelo uso de canais com transparência.

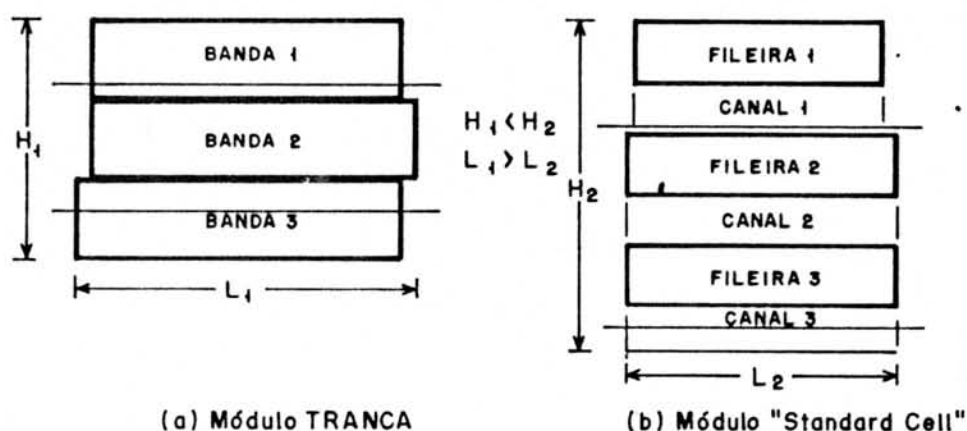


Figura 5.6 - Topologias dos módulos comparados.

No tocante aos módulos TRANCA, a única consideração feita foi o desconto da parcela na altura das células referente à utilização de contatos entre dois níveis de metal (lembre-se que a transparência vertical não é utilizada na síntese de leiaute). Assegura-se que a

redução de 5% na altura das células utilizadas no módulo não reflete na largura das mesmas. A nível de bloco como um todo, o desconto de tal parcela resulta na redução de cerca de 3% na área do módulo originalmente gerado.

A tabela 5.1 apresenta os valores obtidos pela experiência para: altura, largura, relação de aspecto e área de cada módulo "Standard Cell" e TRANCA, transparência do módulo TRANCA e relação entre áreas obtida para um mesmo número de bandas.

Tabela 5.1 - Comparação entre abordagens.

NÚMERO DE BANDAS	MÓDULO "STANDARD CELL" (SC)				MÓDULO TRANCA (TR)					ÁREA SC
	ALTURA	LARGURA	REL.ASPECTO	ÁREA	ALTURA	LARGURA	REL.ASPECTO	ÁREA	TRANSP	ÁREA TR
3	439	1524	0,29	667512	297	1945	0,15	577665	2	1,16
4	592	1161	0,51	687312	395	1565	0,25	618175	0	1,11
5	728	1002	0,73	729456	493	1409	0,35	694637	3	1,05
6	858	834	1,03	715572	591	1147	0,52	677877	4	1,06
7	1024	788	1,30	806912	689	1190	0,58	619910	9	0,98
8	1172	756	1,55	886032	787	1194	0,66	939678	11	0,94

OBS: VALORES EM λ (LAMBDA) E λ^2

Avaliando-se, primeiramente, as alturas e larguras fornecidas pela tabela 5.1, vemos nas figuras 5.7 e 5.8, que cada metodologia é caracterizada por derivadas diferentes. À medida que se aumenta o número de bandas, evidentemente a altura do módulo aumenta e a sua largura diminui. No entanto, a altura do módulo "Standard Cell" cresce mais rapidamente que a do módulo TRANCA (figura 5.7), devido ao aumento do número de canais de roteamento; já a largura do módulo TRANCA diminui mais lentamente que a do módulo "Standard Cell" (figura 5.8), devido ao crescimento do número de células de interconexão interbanda e ao aumento de área desperdiçada no seu ajuste.

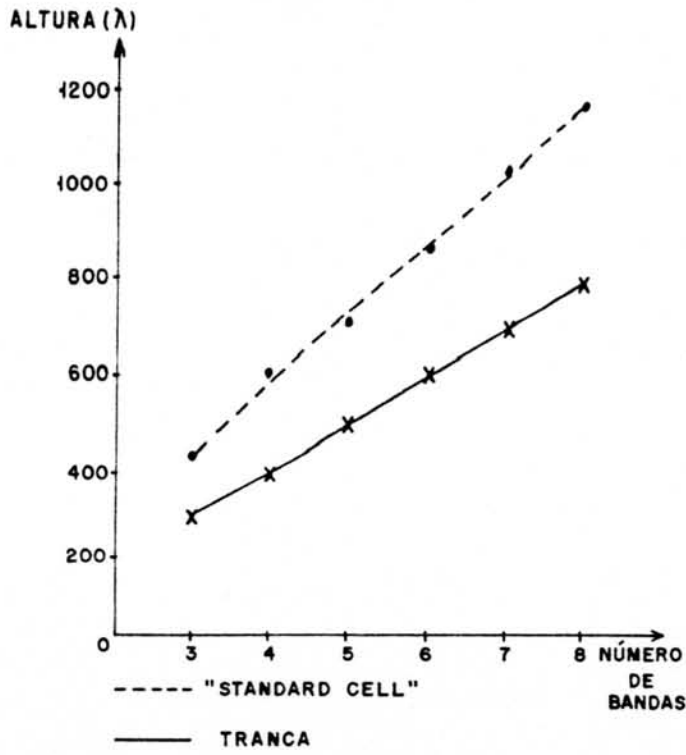


Figura 5.7 - Relações altura vs. número de bandas.

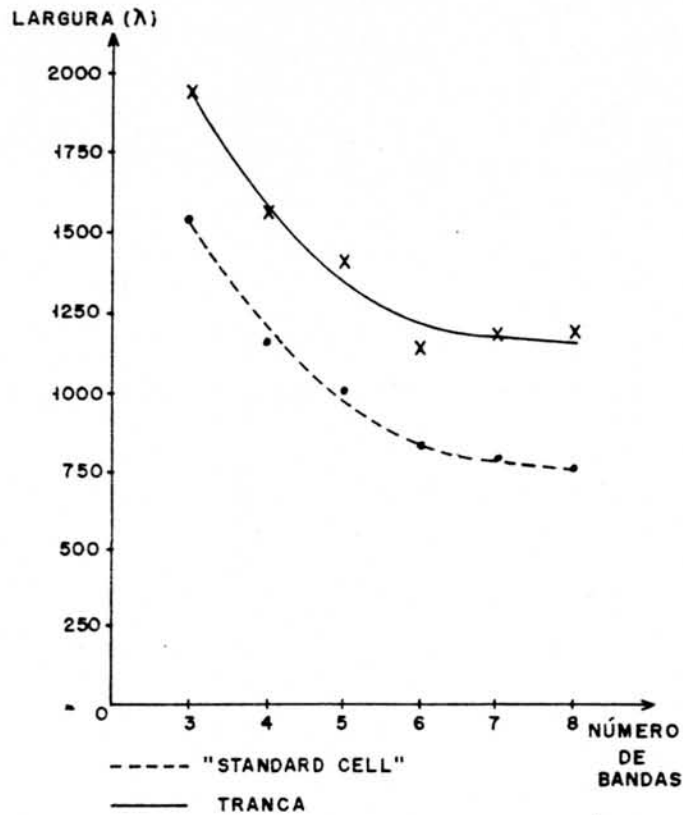


Figura 5.8 - Relações largura vs. número de bandas.

Se avaliarmos, a seguir, o inverso da relação de

aspecto (largura/altura), vemos que, embora as magnitudes sejam diferentes, como as derivadas das curvas mostradas na figura 5.9 muito se assemelham, a maleabilidade dos módulos concebidos segundo as duas abordagens é da mesma ordem.

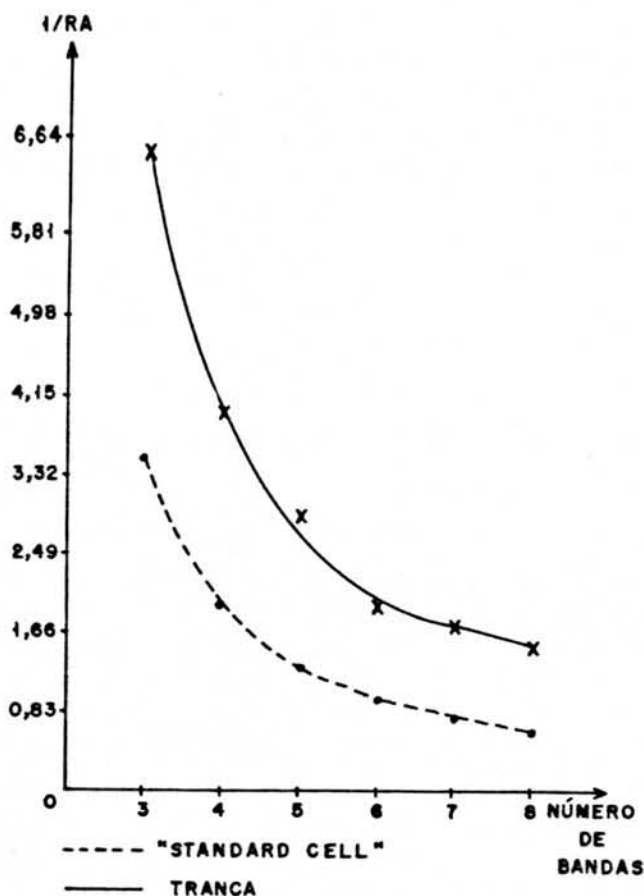


Figura 5.9 - Relações largura/altura vs. número de bandas.

Finalmente, a avaliação das relações área vs. número de bandas (figura 5.10), demonstra que, à medida que aumenta o número de bandas do circuito, a área do módulo TRANCA cresce mais rapidamente que a do módulo "Standard Cell". Este fato é esperado, uma vez que a área ocupada por células de interconexão interbanda e por espaços vazios aumenta com o número de bandas (veja tabela 5.2), e a taxa de ocupação das trilhas da estrutura diminui, ou seja, cresce o número de trilhas livres do módulo (tabela 5.1). Muito embora isto ocorra, observamos que, para o circuito TREVO, as áreas dos módulos "Standard Cell" são de 16 a 5% maiores que as áreas dos módulos TRANCA, para relações de aspecto destes que variam de 0.15 a 0.52 (tabela 5.1).

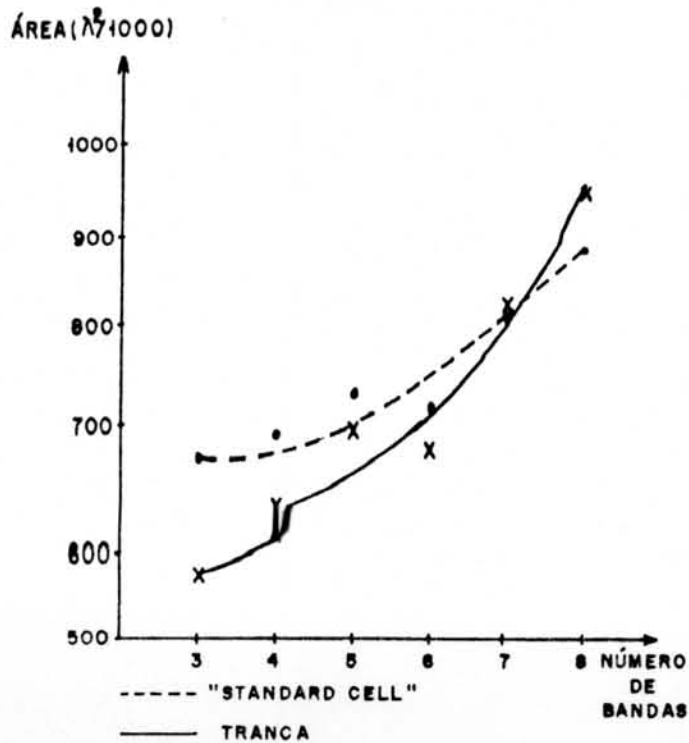


Figura 5.10 - Relações área vs. número de bandas.

Tabela 5.2 - Novos valores para o uso de metal 2.

NÚMERO DE BANDAS	ÁREA MÓDULO SC	MÓDULO TRANCA				ÁREA SC ÁREA TR.m ²
		ÁREA GERADA	PARCELA DESCONTO(*)	COMPENSAÇÃO ALTURA	ÁREA COM TRANSPARÊNCIA EM METAL 2(m ²)	
3	667512	577665	70389	14940	522216	1,28
4	697312	618175	98355	13948	533768	1,29
5	729456	694637	167127	13286	540796	1,35
6	715572	677877	150705	12660	539832	1,33
7	806912	819910	277667	12563	554806	1,45
8	886032	939678	356511	13167	596334	1,49

OBS: VALORES EM λ^2

(*) SOMATÓRIO DAS ÁREAS DE CÉLULAS DE INTERCONEXÃO INTERBANDA E DE ESPAÇOS VAZIOS.

Acredita-se que, para módulos de maior complexidade, e portanto com um maior número de conexões passíveis de serem realizadas sobre as áreas ativas do circuito, o ganho de área atinja até 30% e seja garantido para relações de aspecto de até 0.66. Fica evidente, pois,

que enquanto a comunicação entre bandas não utilizar o conceito de transparência vertical, a largura dos módulos gerados deve predominar sobre a sua altura para que se assegure a compactação de área do leiaute.

A partir dos leiautes dos módulos TRANCA gerados, pode-se fazer uma projeção dos resultados que seriam obtidos pelo uso de uma segunda camada de metal, supondo a absoluta transparência desta em relação às outras camadas. Para tanto, basta descontarmos das áreas obtidas, a área referente às células de interconexão interbanda e a área desperdiçada para encaixe destas células, e considerarmos a altura original das células da biblioteca. A parcela a ser descontada é dada pelo produto da altura do módulo pela diferença entre sua largura e o maior somatório das larguras de células funcionais e de células de interconexão intrabanda em uma banda. A retomada da altura original da biblioteca pressupõe a correção de área pelo acréscimo da parcela referente à diferença de alturas e à nova largura do módulo.

Observa-se na tabela 5.2, que os novos módulos TRANCA resultam em relações de área na faixa de 1.28 a 1.49. Já pela análise da figura 5.11, vemos que a área do módulo "Standard Cell" passa a crescer mais rapidamente que a área do módulo TRANCA, o que justifica a compactação crescente com o aumento do número de bandas. Deve ser lembrado que as células de passagem existentes na abordagem "Standard Cell" permanecem exercendo o papel de consumir área, à medida que se divide o módulo em um número maior de fileiras; o crescimento suave da área do módulo TRANCA reflete o uso crescente de células de interconexão intrabanda, uma vez que, com o aumento do número de bandas, diminuem as possibilidades de realização de conexões por justaposição.

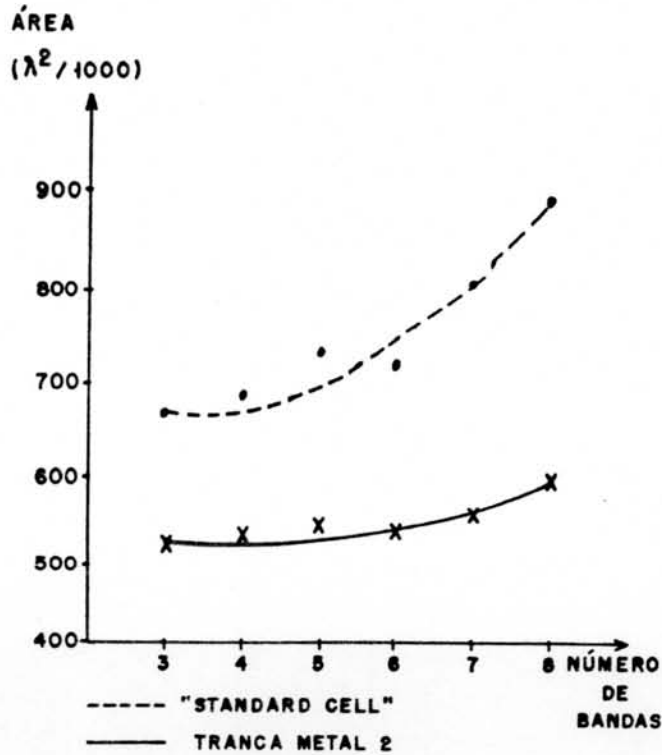


Figura 5.11 - Relações área vs. número de bandas com o uso de transparência vertical.

Muito embora esta projeção não leve em conta que haveria uma redução na área das "standard cells" convencionais, caso a camada de metal 2 fosse utilizada na sua geração, as estimativas não ficam absolutamente invalidadas. Considere que esta redução seria, com certeza, compensada pelo aumento da altura dos canais obtidos, já que as dimensões dos contatos entre metal 1 e metal 2 são maiores. No caso do circuito TREVO, muito possivelmente as áreas dos módulos "Standard Cell" cresceriam, pois as células utilizadas no seu projeto - com poucas conexões internas, teriam uma redução de área extremamente pequena comparada com o acréscimo de altura dos canais.

Para a experiência realizada, garantiu-se que as "standard cells" projetadas possuíssem as mesmas características elétricas que as células da biblioteca TRANCA. Desta forma, buscou-se obter módulos com desempenhos semelhantes, a menos das diferenças inevitáveis de comprimentos de conexões entre as duas abordagens.

Justamente para atenuar os efeitos de atraso causados por conexões muito longas, as bibliotecas de "standard cells" comerciais constam de células "bufferizadas" que, fatalmente, resultam em áreas de leiaute superiores às mencionadas nesta experiência. É evidente que, em função das especificações de projeto, eventualmente se faz necessária a "bufferização" de sinais que participam do caminho crítico do circuito. Para tanto, a metodologia TRANCA prefere colocar à disposição do projetista "buffers" que venham a ser utilizados nestes casos, ao invés de estender o uso desta técnica a todas as células de sua biblioteca. Esta iniciativa é viável, até porque as conexões em módulos TRANCA são menos longas.

Com relação à completeza do roteamento nos módulos TRANCA, as construções para otimização de leiaute, constantes na linguagem NILOTRANCA, mostraram-se de extrema utilidade. Tanto que, em cada módulo gerado, não restou nenhuma conexão não implementada, sendo que, nos módulos com 4 e 7 bandas, a única conexão não realizada automaticamente foi efetuada sem problemas via editor de máscaras.

Os anexos 6, 7, 8 e 9, além das figuras 4.10, 4.13, 4.28 e 4.29, documentam a síntese do leiaute em 6 bandas do módulo TREVO.

5.4 Comparação com o Sistema LES

O sistema LES - "Layout Expert System" [LIN 87], voltado para a geração de módulos em lógica aleatória, baseia-se em um estilo de leiaute muito semelhante ao do gerador TRAMO, estruturando o bloco em bandas e realizando as conexões sobre as áreas ativas do circuito.

LES é um sistema especialista que, a partir de uma descrição lógico-estrutural do bloco funcional, gera automaticamente seu leiaute simbólico.

O posicionamento relativo de células é fundamentado na pesquisa de "caminhos sementes". A ordenação dentro de cada banda é determinada a partir da conectividade das células entre si.

A geração de células e o roteamento são realizados após o posicionamento de células, considerando-se a especificação de dimensões, funcionalidade, fios de passagem e posições dos pinos de cada célula.

Também a metodologia de projeto do sistema LES evoca compactação de área com relação ao estilo "Standard Cell" de leiaute. Tanto que em [LIN 87] reporta-se resultados de comparações com leiautes gerados por sistemas "Standard Cell" diferentes, que resultam em reduções de área de até 76%.

O único "benchmark" descrito por completo em [LIN 87] (figura 5.12) foi gerado pela ferramenta TRAMO, para possibilitar a comparação entre sistemas e análise de resultados. O circuito sintetizado trata-se de um flip-flop D, cujo leiaute, com 40 transistores, aparece na figura 5.13.

A área obtida para este pequeno módulo foi de 0.014 mm^2 . As áreas fornecidas em [LIN 87] - também em uma tecnologia de canal de 2 micra, são de 0.011 mm^2 e 0.027 mm^2 para os módulos LES e "Standard Cell" respectivamente. Os tempos de processamento envolvidos foram de 83s em um VAX-750, para o leiaute LES, e de 47s em um PC-AT, para o leiaute TRANCA.

Lembrando que no sistema LES as células são geradas sob medida para síntese de cada módulo, a diferença de desempenho com relação ao gerador TRAMO se dá, justamente, pela baixa taxa de ocupação de trilhas das estruturas de banda do módulo gerado. Para módulos mais complexos, à medida que um maior número de trilhas passa a ser utilizado e que as perdas de área pela geração

estrutural de célula se agravam, os resultados TRANCA tendem a se equiparar e a atingirem níveis superiores.

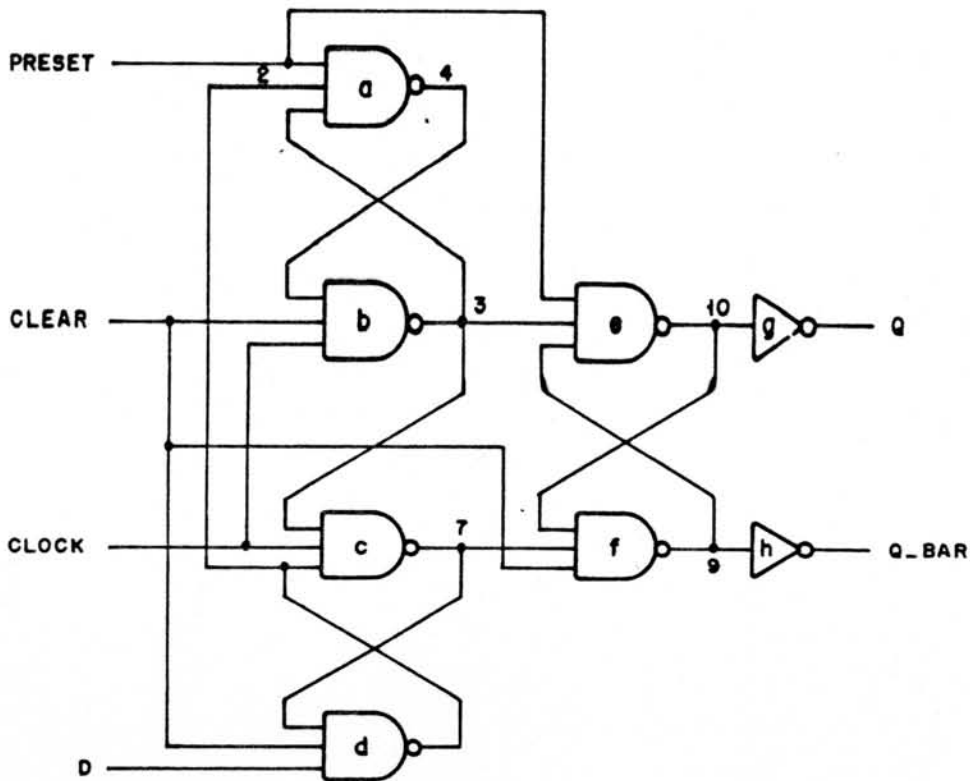


Figura 5.12 - Diagrama lógico do flip-flop D.

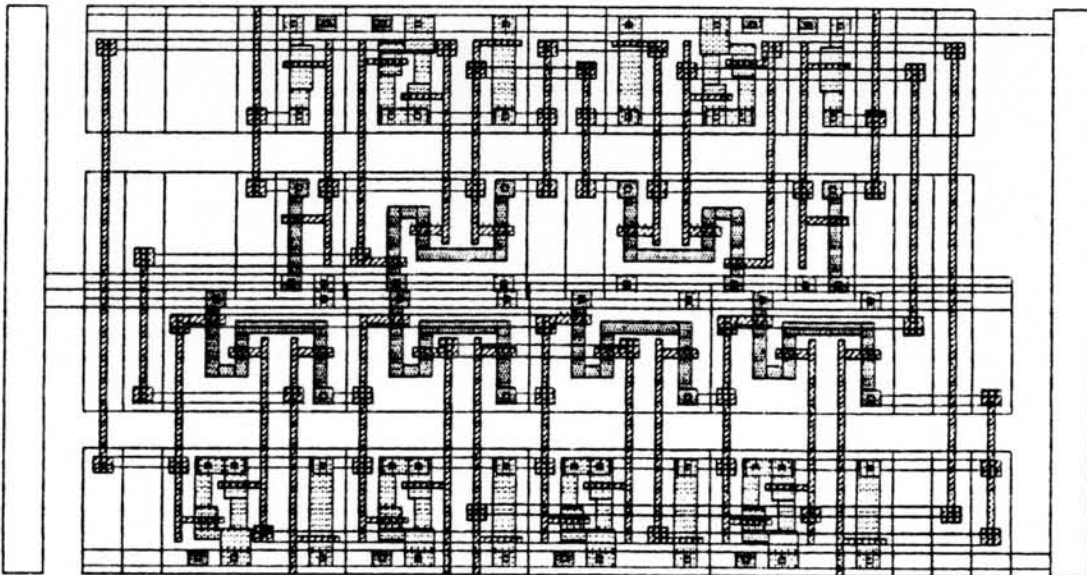


Figura 5.13 - Leiaute do flip-flop D.

No tocante às relações de área com o leiaute "Standard Cell", fica evidente que a supressão dos canais de roteamento e o projeto de células sem super-estimativas

de carga são os principais agentes de compactação dos leiautes TRANCA e LES.

5.5 Propostas para Futuras Otimizações

Das experiências e comparações realizadas e da análise detalhada dos resultados produzidos pelos procedimentos adotados, podemos tanto avaliar a metodologia TRANCA, como propor melhorias às estratégias de síntese de leiaute, configurando assim uma primeira realimentação ao projeto do gerador TRAMO.

No tocante aos procedimentos de composição de células, embora os leiautes sintetizados demonstrem competitividade com leiautes "manuais" e leiautes automáticos gerados por outros sistemas, a análise dos resultados parciais dos subsistemas de posicionamento e de roteamento, bem como da conjunção das soluções destas ferramentas, evidencia pontos frágeis do gerador.

Assim, em se tratando de particionamento, por exemplo, verifica-se a necessidade de aplicar o procedimento a um maior número de partições iniciais diferentes - escolhidas aleatoriamente, e de evoluir para uma função-objetivo que considere, na seleção de células para intercâmbio, a situação dos prováveis pares candidatos aos movimentos futuros. Uma proposta de função-objetivo do tipo "look-ahead" para partição de circuitos é apresentada em [KRI 84].

No que diz respeito ao posicionamento intrabanda, uma maior atenção deve ser dispensada aos critérios de seleção da primeira célula, a qual determina os posicionamentos seguintes. Não é suficiente, por exemplo, o desempate entre células com o mesmo escore pela menor largura - fundamentado no fato de que perdas menores em termos de comprimento de interconexões "provavelmente" ocorram nestas condições. Parece mais razoável avaliar

quantas redes e células ainda não consideradas serão favorecidas, e como, na seleção de uma ou outra célula.

Ainda em termos de posicionamento intrabanda, e também em termos de orientação de células, observa-se que nem sempre a priorização do comprimento de interconexões sobre a possibilidade de realização de conexões por justaposição conduz aos resultados mais interessantes. Melhor seria a avaliação das soluções advindas de um procedimento que priorize um objetivo e depois o outro, e a opção pela melhor delas a nível de roteamento e de área do módulo sintetizado.

Em função das deficiências dos procedimentos de particionamento e posicionamento citadas, talvez valha à pena investir no desenvolvimento de uma ferramenta de otimização (seção 2.2.1.4).

Considerando as estratégias de alocação de células de interconexão, constata-se que dois pontos precisam ser reavaliados: a duplicidade desnecessária e a possibilidade de utilização de células de interconexão coincidentes.

O primeiro problema surge à medida que, quando as células de interconexão intrabanda são posicionadas, se desconhece onde as células de interconexão interbanda serão alocadas. Desta forma, em alguns casos ocorre o posicionamento de células de interconexão adjacentes, ou vizinhas próximas, que implementam a mesma conexão, conforme mostra a figura 5.14a. Para evitar esta perda de área, é interessante que se elimine as células de interconexão intrabanda desnecessárias e se submeta o leiaute, com novos espaços vazios, ao procedimento de compactação já existente no sistema (figura 5.14b).

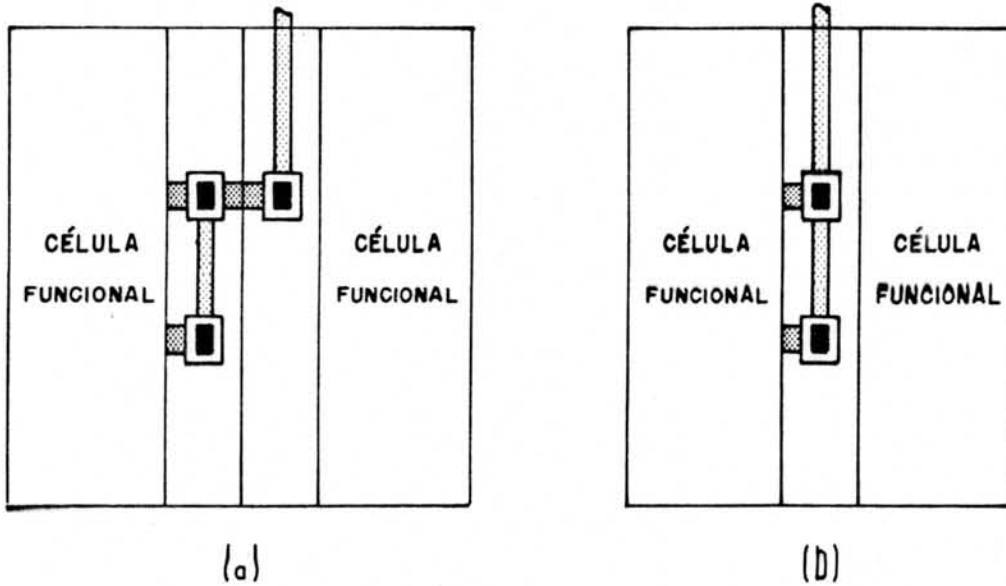


Figura 5.14 - Eliminação de célula de interconexão.

O segundo ponto, por sua vez, deve envolver estratégias bem mais elaboradas para o seu tratamento. A idéia é transformar células de interconexão interbanda em entidades que possam implementar duas conexões distintas simultaneamente, a partir da fusão de células de interconexão dedicadas a uma única rede (figura 5.15). A solução deste problema pode ser encarada tanto do ponto de vista da realização de compactação do leiaute sintetizado, como do ponto de vista da adoção de estratégias de alocação que trabalhem simultaneamente com no mínimo 3 bandas. Observe-se que a complexidade de roteamento intrabanda cresce nesta última prática, pois restrições verticais (seção 2.2.4.3.2) passam a existir. É importante citar que parte do sucesso do sistema LES se dá pelo fato de conexões diferentes poderem ser implementadas em uma mesma coluna simbólica [LIN 87].

A nível do roteamento intrabanda, como já visto na seção 4.4.2, o estudo de critérios mais adequados para a ordenação de redes na estrutura de banda faz-se necessário.

Considerando, ainda, que a impossibilidade de realização de algumas conexões em uma estrutura de banda fixa pode ocorrer, parece promissor o tratamento destes casos via roteamento de propósito geral (seção 2.2.4.3.1).

Esta iniciativa implica, com certeza, na expansão da matriz de roteamento das células envolvidas, a um ponto tal que a sua topologia detalhada venha a ser de conhecimento da ferramenta.

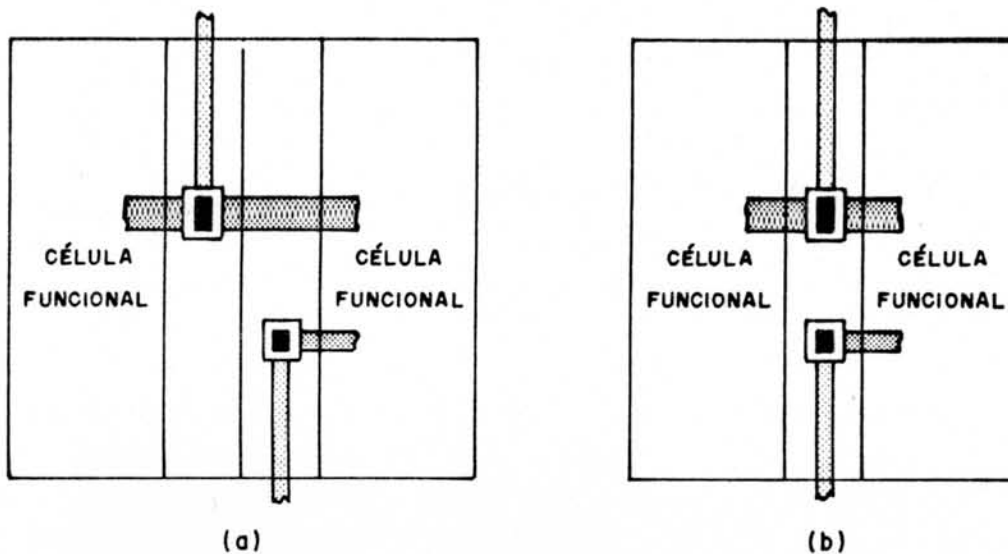


Figura 5.15 - Fusão de células de interconexão.

O refinamento do roteamento precisa igualmente ser realizado. Para tanto, basta que se disponha de um procedimento que pesquise a possibilidade de troca de camadas em trechos de conexões, para a minimização do número de contatos utilizados.

O ponto mais polêmico da ferramenta TRAMO talvez seja o procedimento de geração de célula, primeiro devido à dependência de tecnologia, segundo devido a seu caráter de projeto semi-dedicado passível de limitações de implementação física.

A independência de tecnologia pode ser atingida pela manutenção de uma biblioteca simbólica. Nesta abordagem a edição de células deve prever uma "camada" de transparência que promova o alinhamento de trilhas na montagem da biblioteca.

A composição de células pode, então, ser realizada com base nas próprias coordenadas simbólicas de cada célula, ou nas suas coordenadas geométricas (se a

biblioteca for previamente convertida para uma descrição segundo a tecnologia em uso). Os prós e contras da composição simbólica e da geométrica precisam ser considerados. A escolha de uma ou outra abordagem deve ser feita em função dos seguintes aspectos: 1) alinhamento de fronteiras de células para roteamento entre bandas; 2) possibilidades de supressão de trilhas não utilizadas; e, 3) possibilidades de inclusão de trilhas para a completeza do roteamento intrabanda.

Observe-se que, nas abordagens geométrica e simbólica, uma vez definida a topologia de cada célula (detalhada ou não), a maleabilidade destas não pode mais ser explorada durante a composição de células. É justamente este fato que caracteriza projetos baseados em bibliotecas e, portanto, semi-dedicados.

O caráter de projeto dedicado só pode ser atingido via geração estrutural de célula, a partir do momento que a síntese automática de leiaute garanta a utilização somente dos recursos necessários a uma determinada implementação. A geração estrutural de bandas inteiras, ao invés de células individualmente, deve atenuar grandemente a desvantagem de consumo de área característica desta abordagem. A ferramenta TRAGO - "TRANca Gate-matrix generatOr [MOR 89d], vem sendo desenvolvida com o intuito de suprir tais deficiências a nível de geração de módulos TRANCA.

Idealmente, o procedimento de geração de célula deve suportar as três abordagens possíveis, inclusive misturando-as na geração de um mesmo módulo. Imagine-se, por exemplo, que primeiramente se realize a síntese do leiaute de um módulo utilizando a abordagem geométrica. Como, em geral, as bandas próximas às extremidades superior e inferior do módulo são menos largas que as bandas centrais e utilizam um número pequeno de trilhas da estrutura, a regeneração estrutural de tais bandas com menos

trilhas resulta em um novo módulo mais baixo e provavelmente com a mesma largura original. Se, por ventura, em uma determinada banda alguma conexão não pôde ser realizada, a geração estrutural desta banda irá prover o número necessário de trilhas ao roteamento, tentando reduzir sua largura.

Mesmo convivendo com os pontos frágeis da ferramenta TRAMO, verifica-se que o uso da metodologia TRANCA confirma as previsões de compactação de área.

À medida que uma segunda camada de metal venha também a ser utilizada para o roteamento entre bandas, e que a taxa de ocupação de trilhas cresça pela implementação de circuitos mais complexos, tão mais evidentes se tornarão as vantagens da adoção de tal metodologia de concepção de leiaute.

Cabe salientar, no entanto, que o uso de metal 2 para o roteamento não deve implicar no abandono da camada de polissilício. Deve-se lembrar que a transparência vertical do módulo será favorecida se conexões intrabanda e algumas conexões entre bandas adjacentes forem implementadas em polissilício. Conseqüentemente, a melhor opção é dispor de um gerador de módulos que suporte o convívio das duas abordagens de roteamento interbanda.

Sugere-se, então, que o uso de polissilício se limite à formação de superbandas com no máximo 2 bandas, e que estas superbandas envolvam somente as bandas que resultaram menores do particionamento. Portanto, os procedimentos de alocação de células de interconexão do RETRANCA continuam válidos.

Observe-se que o restante do roteamento interbanda, realizado em metal 2, fica extremamente simplificado. A base do roteamento passa a ser uma matriz com as dimensões do próprio módulo, cujas linhas correspondem às trilhas em metal 1 das estruturas de banda

e as colunas correspondem às trilhas verticais em metal 2 (figura 5.16).

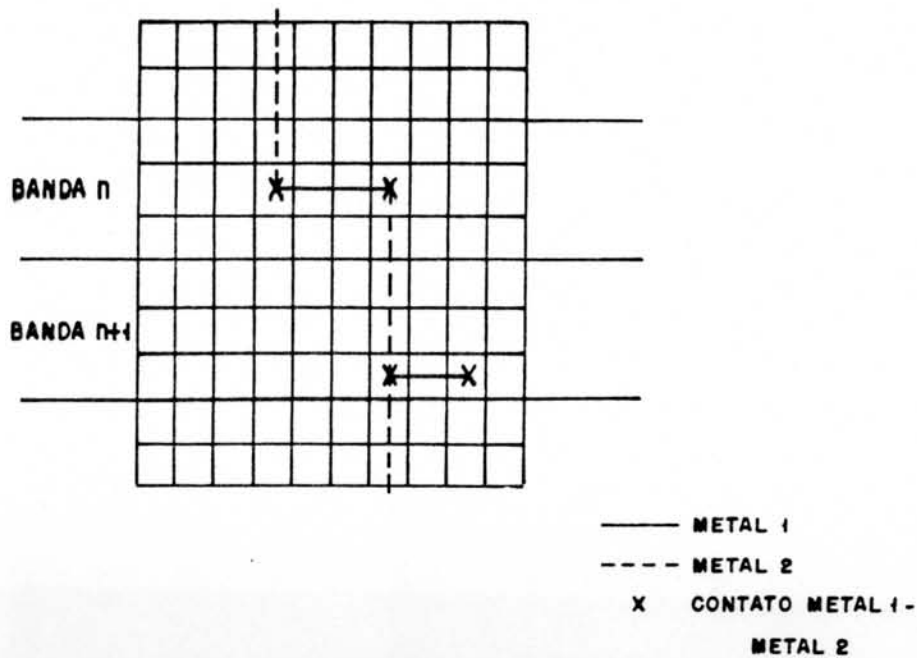
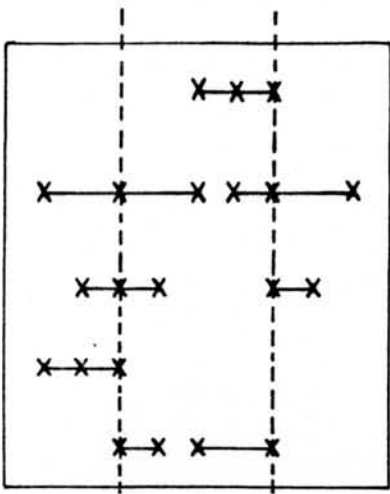


Figura 5.16 - Roteamento com o uso de metal 2.

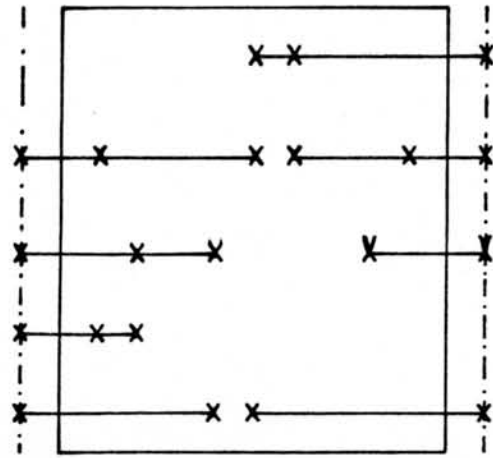
O problema de roteamento vertical se resume, assim, à alocação de colunas desta matriz - em função da localização das células que compartilham cada rede, e à extensão de conexões intrabanda sobre as linhas. A alocação das colunas pode ser realizada de forma semelhante ao tratamento do roteamento de canal (seção 2.2.4.3.2), lembrando que o objetivo maior, neste caso, é a obtenção de conexões não muito longas, ao invés da obtenção de um roteamento com um número reduzido de trilhas verticais. De fato, é necessário que se procure um equilíbrio entre estes dois objetivos.

À medida que cresce significativamente o número de células que se conectam a uma mesma rede, se justifica a utilização de barramentos para a realização de conexões. Em blocos em lógica aleatória mais complexos que os apresentados neste trabalho, a necessidade de uso de barramentos deve surgir em algum momento. Se a camada de metal 2 for utilizada na síntese de leiaute, sugere-se que tais barramentos sejam implementados sobre o módulo, valendo-se do conceito de transparência vertical (figura

5.17a); se somente a camada de polissilício puder ser utilizada para conexões entre bandas, aconselha-se a implementação de barramentos nas extremidades esquerda e/ou direita do bloco funcional (figura 5.17b).



(a) Barramentos sobre o Módulo



(b) Barramentos Laterais

Figura 5.17 - Implementação de barramentos.

A versão atual do gerador TRAMO não suporta a implementação automática de barramentos: ao tentar realizar um número muito grande de conexões que envolvam muitas células, tanto a quantidade de células de interconexão interbanda deve crescer assustadoramente, quanto a saturação da estrutura de banda deve ocorrer em vários pontos do circuito. Propõe-se, desta forma, algumas alterações no gerador para o tratamento destes casos, quais sejam:

a) criação de um atributo de rede do tipo barramento orientado ("west_bus" e "east_bus", por exemplo) e, eventualmente, também ordenado;

b) realização do particionamento desconsiderando as conexões ao barramento, mas mantendo, em cada célula pertinente, uma rede imaginária com a mesma orientação do barramento ao qual ela irá conectar-se;

c) realização do posicionamento intrabanda retomando as conexões ao barramento e considerando-o como qualquer rede orientada;

d) realização da alocação de células de

interconexão interbanda ignorando a rede a ser implementada sob a forma de barramento;

e) realização do roteamento intrabanda, considerando a orientação das conexões ao barramento; e,

f) inclusão de uma rotina que gere o leiaute do barramento devidamente conectado e agregado ao bloco funcional.

Para implementações em metal 2, os itens a, b e c são relevantes, afora as considerações de orientação. A implementação das ligações físicas do barramento, neste caso, pode ser realizada como a de qualquer outra rede do circuito.

Como se pode concluir, o uso do conceito de transparência vertical para a implementação de barramentos configura uma solução extremamente econômica do ponto de vista de área do leiaute.

Uma vez que a avaliação de desempenho elétrico é quem promove o fechamento do ciclo de projeto físico, é de suprema importância que a síntese automática de leiaute seja auxiliada por uma ferramenta de ataque a este problema, e orientada à geração de módulos que cumpram também as especificações elétricas para o circuito.

Considerando que a implementação de leiautes em lógica aleatória resulta, via de regra, na realização de uma mesma conexão utilizando trechos em diferentes camadas, o modelamento das interconexões envolve redes RC bastante elaboradas para a avaliação realística do desempenho elétrico [SAK 83][RUB 83][HOR 83].

A partir dos modelos das redes, e dos valores de capacitância das entradas das células e de resistência de suas saídas - constantes da documentação da biblioteca, viabiliza-se a pesquisa de caminhos críticos e a avaliação de tempos de atraso para os módulos sintetizados. A implementação de uma ferramenta voltada para o tratamento

deste problema permanece como sugestão para futuros trabalhos de pesquisa e desenvolvimento.

6 CONCLUSÃO

As metas traçadas para o desenvolvimento deste trabalho foram plenamente atingidas: além da construção de uma ferramenta para a síntese automática de leiaute, uma primeira avaliação das estratégias e da metodologia envolvidas foi realizada.

O gerador TRAMO demonstrou ser uma ferramenta de grande utilidade prática e uma boa solução inicial para o tratamento da síntese de leiaute de módulos em lógica aleatória. A metodologia TRANCA exercitou a sua eficiência e flexibilidade, mostrando-se ainda mais promissora com a efetiva utilização do conceito de transparência vertical.

As experiências realizadas, apesar do pequeno porte, foram de grande valia, à medida que possibilitaram a formulação de propostas para otimização dos procedimentos de síntese. Dois fatores principais inviabilizaram comparações mais significativas: 1) a não disponibilidade de ferramentas comerciais e de equipamentos que comportem a sua utilização; e, 2) as dificuldades envolvidas na aquisição de "benchmarks" e na elaboração de meios consistentes para a comparação entre sistemas e metodologias de projeto [PRE 87]. A julgar pela mobilização atual da comunidade científica nacional e internacional, essas adversidades devem ser superadas a curto prazo.

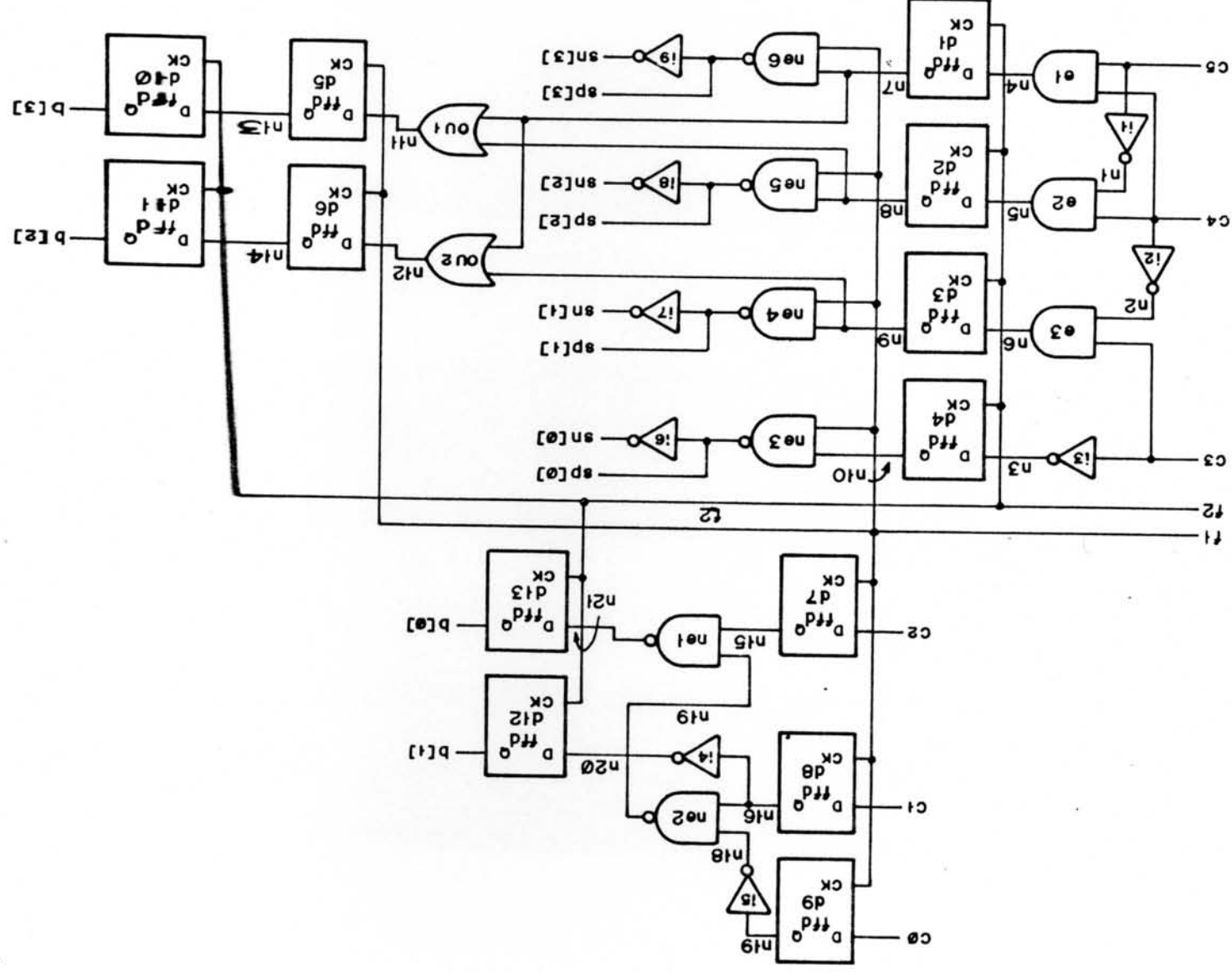
O desenvolvimento deste trabalho não se limitou ao provimento de uma ferramenta de PAC ao GME, investiu também na integração de ferramentas para a construção de um sistema mínimo de projeto físico e no estímulo à retomada do projeto de circuitos pelo grupo.

À medida que as deficiências de ferramentas para o projeto em silício sejam superadas, a preocupação com a automação do projeto de sistema deve imperar. Para evoluir no sentido de aprimorar os recursos para síntese é necessário que se invista no desenvolvimento de ferramentas

de projeto visando a testabilidade, de síntese lógica e de mapeamento na tecnologia, como já vem ocorrendo no PGCC. Este é o caminho para o combate à complexidade crescente de circuitos VLSI.

ANEXO 1 - EXEMPLO DE DESCRIÇÃO NILOTRANCA

A1.1 Esquemático



A1.2 Descrição Textual

agency conversor.1.1;

level=NILO;

interface

```

        north in f1,
                f2'      : terminal;

    ordered west in c0,
                    c1,
                    c2'      : terminal;

    ordered east in c3,
                    c4,
                    c5'      : terminal;

    ordered south out b[3:0] : terminal;

    ordered north out sn[3:0],
                    sp[3:0]' : terminal;

```

behavior

gate

```

i1 {in=1, region=1, strip=2} : not;
i2 {in=1}                    : not;
i3 {in=1}                    : not;
i4 {in=1}                    : not;
i5 {in=1}                    : not;
i6 {in=1}                    : not;
i7 {in=1}                    : not;
i8 {in=1}                    : not;
i9 {in=1, region=2, mirrored} : not;
e1 {in=2}                    : and;
e2 {in=2}                    : and;
e3 {in=2}                    : and;
ne1 {in=2, strip=1, ordering=7} : nand;
ne2 {in=2}                    : nand;
ne3 {in=2}                    : nand;
ne4 {in=2}                    : nand;
ne5 {in=2, non_mirrored}      : nand;
ne6 {in=2}                    : nand;
ou1 {in=2}                    : or;
ou2 {in=2}                    : or;
d1 {in=2, out=2}              : ffd;
d2 {in=2, out=2}              : ffd;
d3 {in=2, out=2}              : ffd;
d4 {in=2, out=2}              : ffd;
d5 {in=2, out=2}              : ffd;
d6 {in=2, out=2}              : ffd;
d7 {in=2, out=2}              : ffd;
d8 {in=2, out=2}              : ffd;
d9 {in=2, out=2}              : ffd;
d10 {in=2, out=2}             : ffd;
d11 {in=2, out=2}             : ffd;
d12 {in=2, out=2}             : ffd;
d13 {in=2, out=2}             : ffd;

```

node

```

n1 (weight=4) = i1.2, e2.2;
n2            = i2.2, e3.2;
n3            = i3.2, d4.1;
n4            = e1.3, d1.1;
n5            = e2.3, d2.1;
n6            = e3.3, d3.1;

```

```

n7          = d1.3,ou1.2,ou2.2,ne6.2;
n8          = d2.3,ou1.1,ne5.2;
n9          = d3.3,ou2.1,ne4.2;
n10         = d4.3,ne3.2;
n11         = ou1.3,d5.1;
n12         = ou2.3,d6.1;
n13         = d5.3,d10.1;
n14(track=P0) = d6.3,d11.1;
n15         = d7.3,ne1.2;
n16         = d8.3,ne2.2,i4.1;
n17         = d9.3,i5.1;
n18         = i5.2,ne2.1;
n19         = ne2.3,ne1.1;
n20         = i4.2,d12.1;
n21         = ne1.3,d13.1;

```

equivalence

```

c0          = d9.1;
c1          = d8.1;
c2          = d7.1;
c3          = i3.1,e3.1;
c4          (weight=4,
            track=P4) = i2.1,e2.1,e1.1;
c5          = i1.1,e1.2;
f1          = d5.2,d6.2,d7.2,d8.2,
            d9.2,ne3.1,ne4.1,
            ne5.1,ne6.1;
f2          = d1.2,d2.2,d3.2,d4.2,
            d10.2,d11.2,d12.2,
            d13.2;
b[0]        = d13.3;
b[1]        = d12.3;
b[2]        = d11.3;
b[3]        (track=P7) = d10.3;
sn[0]       = i6.2;
sn[1]       = i7.2;
sn[2]       = i8.2;
sn[3]       = i9.2;
sp[0]       = ne3.3,i6.1;
sp[1]       = ne4.3,i7.1;
sp[2]       = ne5.3,i8.1;
sp[3]       = ne6.3,i9.1;

```

end;

ANEXO 2 - DOCUMENTAÇÃO DE UMA CÉLULA DA BIBLIOTECA

A) Nome: **XOR2.CEL**

B) Características:

B.1) Tecnologia: CMOS

B.2) Regras: "French MPC Design Rules for Two Metal CMOS process" (17 de Novembro de 1986)

B.3) Transparência da célula: P5

B.4) Largura e Área:

I) Largura: 71 λ sII) Área: 7526 λ s quadrados

B.5) Características Elétricas:

I) Consumo: 1.01E-10 Watts

II) Atraso Intrínseco: 1.4 ns

III) Capacitância de Entrada:

Entrada	Capacitância
E1	0.02971 pF
E2	0.04640 pF

IV) Resistência de Saída:

Resistência de Pull-up: 3.37 KOhms

Resistência de Pull-down: 3.22 KOhms

V) Fan-out: 16 ns/pF

VI) Largura de Pulso Mínimo: 1.4 ns

B.6) Topologia de Entrada/Saída:

Entrada	Posição X	Conectibilidade	
		Esquerda	Direita
E1	2	P9	P5+7
E2	46	P5	P5+7

Saída	Linhas
S	6

C) Descrição: Trata-se de uma célula ou-exclusivo estática de duas entradas.

D) Status: [não-checada]

[simulada]

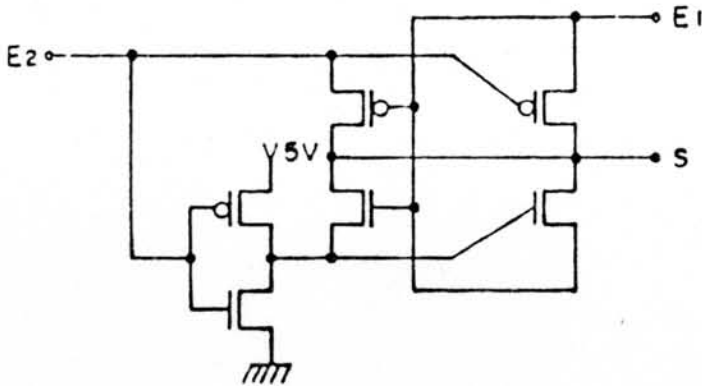
E) NÚMERO DA CÉLULA : 100

F) NÚMERO DE TRANSISTORES : 6

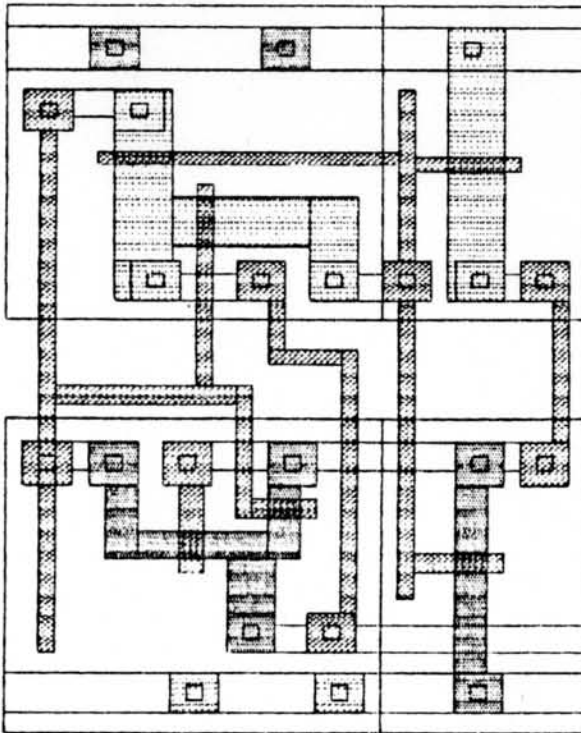
G) DIAGRAMA LÓGICO :



H) DIAGRAMA ELÉTRICO :



I) Leiaute e Matriz de Roteamento



Vcc	2	2	3
P 7	1	0	2
P 5	0	0	2
P 3	0	0	2
P 9	0	3	2
P 1	0	0	2
P 0	0	0	2
P 8	0	3	2
P 2	0	0	2
P 4	0	0	2
P 6	0	2	1
Gnd	2	2	3
	E 1	E 2	S

LEGENDA :

- 0 - USO E PASSAGEM PERMITIDOS
- 1 - USO PERMITIDO, PASSAGEM PROIBIDA
- 2 - USO PROIBIDO, PASSAGEM PERMITIDA
- 3 - USO E PASSAGEM PROIBIDOS

ANEXO 3 - ALGORITMO GERAL DO PARTICIONADOR**A3.1 Procedimento Principal**

```
início;
declarações {
    S é percentual de tolerância de equilíbrio;
    NB é número de bandas;
    determina_partição(n) é rotina;
    RA é relação de aspecto;
};
procedimento {
    1. entra dados;
    2. inicializa estruturas;
    3. faz S = 1;
    4. calcula NB;
    5. determina_partição(NB);
    6. se (RA não atendida)
        então 6.1. reinicializa estruturas;
            6.2. reavalia NB;
            6.3. volta ao passo 5;
        senão;
    7. para S = 0,1 até 0,9 passo 0,1
        7.1. determina_partição(NB)
        7.2. avalia solução;
        7.3. próximo S;
    8. determina melhor solução;
    9. sai dados;
};
fim;
```

A3.2 Rotina determina_partição(n)

```
início;
declarações {
    NR é número de regiões;
    particiona(m) é rotina;
};
```

```

    i é variável de contagem;
    n é parâmetro externo;
};

procedimento {
    1. calcula NR;
    2. trata ordenação da interface;
    3. cria redes imaginárias para partição
       horizontal;
    4. particiona(NR);
    5. cria redes imaginárias para partição
       vertical;
    6. para i = 1 até NR
        6.1. particiona(n);
        6.2. próximo i;
    7. retorna;
};

fim;

```

A3.3 Rotina particiona(m)

```

início;
declarações {
    j é variável de contagem;
    m é parâmetro externo;
};

procedimento {
    1. para j = 1 até m-1
        1.1. determina equilíbrio;
        1.2. gera partição inicial;
        1.3. calcula e ordena custos de células;
        1.4. forma pares de ganho máximo;
        1.5. seleciona pares para intercâmbio;
        1.6. verifica equilíbrio;
        1.7. troca células;
        1.8. realiza os passo de 1.3 a 1.8
            enquanto houver par com ganho positivo;
        1.9. realiza transferências individuais;

```

```
    1.10.altera pesos das redes imaginárias;  
    1.11.dissocia banda gerada;  
    1.12.próximo j;  
2. retorna;  
};  
fim;
```


ANEXO 4 - ALGORITMO GERAL DO POSICIONADOR INTRABANDA**A4.1 Procedimento Principal**

início;

declarações {

 EOL é escore de orientação oeste/leste;

 i é variável de contagem;

 NR é número de regiões;

 ELR é escore de orientação de rede sem restrição;

 ENS é escore de ordenação norte/sul;

 EVZ é escore de vizinhança;

 EGL é escore global;

 desempata_escores() é rotina;

 NC é número total de células do circuito;

};

procedimento {

 1. entra dados;

 2. inicializa estruturas;

 3. calcula limites de ordenação;

 4. centraliza bandas;

 5. calcula apontadores de posicionamento;

 6. calcula EOL;

 7. para i = 1 até NR

 7.1. se (NR > 1) calcula ELR;

 7.2. seleciona banda;

 7.3. calcula ENS;

 7.4. calcula EVZ;

 7.5. calcula EGL;

 7.6. se (empate EGL) desempata_escores();

 7.7. move célula selecionada;

 7.8. atualiza apontador de posicionamento
 da banda;

 7.9. realiza os passos de 7.2 a 7.9
 enquanto houver banda incompleta;

 7.10. próximo i;

```

8. para i = 1 até NC
  8.1. calcula coordenada média dos pinos
      funcionalmente equivalentes;
  8.2. avalia comprimento de conexões para
      configuração não espelhada e espelhada;
  8.3. adota melhor configuração, se houver;
  8.4. próximo i;
9. para i = 1 até NC
  9.1. se (configuração não determinada)
      então 9.1.1. avalia conexões com a
              interface não orientada;
              9.1.2. adota melhor configuração,
                  se houver;
              9.1.3. se (configuração não determinada)
                      então
                          9.1.3.1. avalia possibilidades
                              de justaposição;
                          9.1.3.2. adota melhor
                              configuração;
                      senão;
              senão;
  9.2. próximo i;
10. sai dados;
};
fim;

```

A4.2 Rotina desempata_escores()

```

início;
declarações { };
procedimento {
  1. avalia comprimentos das redes terminais;
  2. se (empate comprimento máximo)
      então 2.1. avalia comprimentos das redes não
              terminais não pertencentes à interface
              do módulo;

```



```
2.2. se (empate comprimento máximo)
    então 2.2.1. avalia largura das células
            que permanecem candidatas;
    2.2.2. se (empate largura mínima)
        então
            2.2.2.1 seleciona pri-
                meira célula;
            senão;
        senão 2.2.3. seleciona célula com
                comprimento máximo;
    senão 2.3. seleciona célula com comprimento máximo;
3. retorna;
};
fim;
```


ANEXO 5 - ALGORITMO GERAL DO ROTEADOR**A5.1 Procedimento Principal**

```
início;
declarações {
    i é variável de contagem;
    NB é número de bandas;
    realiza roteamento_global() é rotina;
    roteia_redes_intrabanda() é rotina;
};
procedimento {
    1. entra dados;
    2. inicializa estruturas;
    3. para i = 1 até NB
        3.1. assinala saídas das células;
        3.2. realiza roteamento imposto;
        3.3. realiza conexões por justaposição;
        3.4. procura caminhos diretos;
        3.5. assinala entradas das células;
        3.6. aloca células de interconexão
            intrabanda;
        3.7. próximo i;
    4. orienta interface livre de restrições;
    5. prepara interface norte/sul;
    6. recentraliza bandas;
    7. realiza_roteamento_global();
    8. para i = 1 até NB
        8.1. roteia_redes_intrabanda();
        8.2. próximo i;
    9. roteia interface oeste/leste;
    10.gera saída simbólica;
    11.gera leiaute;
};
fim;
```

A5.2 Rotina realiza roteamento global()

```
início;
declarações {
    j é variável de contagem;
    NRD é número de redes divididas;
    k é variável de contagem;
};
procedimento {
    1. seleciona par de bandas;
    2. pesquisa locais para células de interconexão
       interbanda;
    3. ordena locais preferenciais;
    4. escolhe sentido de deslocamento para as
       células da banda de referência;
    5. para j = 1 até NRD
        5.1. aloca célula de interconexão para
            rede j;
        5.2. para k = j + 1 até NRD
            5.2.1. pesquisa locais preferenciais
                para a rede k a partir do local
                da rede j;
            5.2.2. próximo k;
        5.3. ordena locais preferenciais a partir da
            rede j+1;
        5.4. próximo j;
    6. forma superbanda;
    7. compacta superbanda;
    8. recentraliza bandas;
    9. realiza os passos de 1 a 9
       enquanto houver par de bandas não roteadas;
    10. retorna;
};
fim;
```

A5.3 Rotina roteia_redes_intrabanda()

início;

declarações {

 r é variável de contagem;

 N é número de nodos na banda;

};
procedimento {

1. para r = 1 até N

 1.1. procura próximo nodo da banda que
 compartilha a mesma rede de r;

 1.2. se (nodo existe)

 então 1.2.1. procura trilha de maior
 prioridade para roteamento;

 1.2.2. se (trilha encontrada)

 então 1.2.2.1.roteia;

 senão 1.2.2.1.registra falha;

 senão;

 1.3. próximo r;

2. retorna;

};

fim;

ANEXO 6 - DESCRIÇÃO DE UM MULTIPLICADOR COMBINACIONAL DE
4 BITS

agency TREVO.1.1;

level=NILO;

```
interface
    in      x[3:0]:      terminal;
    in      y[3:0]:      terminal;
    out     s[7:0]:      terminal;
```

behavior
gate

```
d3 (in=2) : and;
f51 (in=2) : xor;
f52 (in=2) : xor;
f53 (in=2) : and;
f54 (in=2) : and;
f55 (in=2) : and;
f56 (in=3) : nor;
f57 (in=1) : not;
d2 (in=2) : and;
f41 (in=2) : xor;
f42 (in=2) : xor;
f43 (in=2) : and;
f44 (in=2) : and;
f45 (in=2) : and;
f46 (in=3) : nor;
f47 (in=1) : not;
c3 (in=2) : and;
c21 (in=2) : xor;
c22 (in=2) : xor;
c23 (in=2) : and;
g21 (in=3) : nand;
g22 (in=2) : nand;
g23 (in=1) : not;
g24 (in=3) : nand;
h21 (in=2,strip=3) : xor;
h22 (in=2,strip=3) : and;
c2 (in=2) : and;
d1 (in=2) : and;
f31 (in=2) : xor;
f32 (in=2) : xor;
f33 (in=2) : and;
f34 (in=2) : and;
f35 (in=2) : and;
f36 (in=3) : nor;
f37 (in=1) : not;
b3 (in=2) : and;
c11 (in=2) : xor;
c12 (in=2) : xor;
c13 (in=2) : and;
g11 (in=2) : nand;
g12 (in=2) : nand;
g13 (in=1) : not;
f21 (in=2) : xor;
f22 (in=2) : xor;
f23 (in=2) : and;
f24 (in=2) : and;
f25 (in=2) : and;
f26 (in=3) : nor;
f27 (in=1) : not;
f11 (in=2) : xor;
f12 (in=2) : xor;
f13 (in=2) : and;
f14 (in=2) : and;
f15 (in=2) : and;
f16 (in=3) : nor;
```

```

f17 (in=1, ordering=8): not;
a3 (in=2): and;
c1 (in=2): and;
d0 (in=2): and;
b2 (in=2): and;
c01 (in=2): xor;
C03 (in=2): and;
h11 (in=2): xor;
h12 (in=2): and;
f01 (in=2): xor;
f02 (in=2): xor;
f03 (in=2): and;
f04 (in=2): and;
f05 (in=2): and;
f06 (in=3): nor;
f07 (in=1): not;
c0 (in=2): and;
b1 (in=2): and;
a2 (in=2): and;
b0 (in=2): and;
a1 (in=2): and;
h01 (in=2): xor;
h02 (in=2): and;
a0 (in=2): and;

```

node

```

elf5 = d3.3, f51.1, f53.2, f54.1;
e2f5 = f47.2, f51.2, f54.2, f55.1;
cif5 = g24.4, f52.2, f53.1, f55.2;
elf4 = d2.3, f41.1, f43.2, f44.1;
e2f4 = c3.3, f41.2, f44.2, f45.1;
cif4 = h22.3, f42.2, f43.1, f45.2;
sf4 = f42.3, c21.1, c23.1;
elh2 = c2.3, h21.1, h22.1;
e2h2 = d1.3, h21.2, h22.2;
sh2 = h21.3, f31.1, f33.2, f34.1;
e2f3 = b3.3, f31.2, f34.2, f35.1;
cif3 = f17.2, f32.2, f33.1, f35.2;
cof3 = f37.2, c21.2, c23.2;
sf3 = f32.3, c11.1, c13.1;
cic2 = g12.3, c22.2;
gc2 = c23.3, g23.1;
pc2 = c21.3, g21.3, g22.2, c22.1;
pcl = c11.3, c12.1, g13.1, g11.1,
g21.2;
g0g2 = c13.3, g22.1;
p0g2 = c03.3, c12.2, g11.2, g21.1;
e2c1 = f27.2, c11.2, c13.2;
elf1 = d0.3, f11.1, f13.2, f14.1;
e2f1 = b2.3, f11.2, f14.2, f15.1;
cif1 (track=P4) = c1.3, f12.2, f13.1, f15.2;
sf1 (track=P6) = f12.3, f21.1, f23.2, f24.1;
e2f2 = a3.3, f21.2, f24.2, f25.1;
cif2 = f07.2, f22.2, f23.1, f25.2;
sf2 = f22.3, c01.1, c03.1;
e2c0 = h12.3, c01.2, c03.2;
elf0 = b1.3, f01.1, f03.2, f04.1;
e2f0 = a2.3, f01.2, f04.2, f05.1;
cif0 = c0.3, f02.2, f03.1, f05.2;
sf0 = f02.3, h11.1, h12.1;
e2h1 = h02.3, h11.2, h12.2;
elh0 = b0.3, h01.1, h02.1;
e2h0 = a1.3, h01.2, h02.2;
if51 = f51.3, f52.1;
if52 = f53.3, f56.1;
if53 = f54.3, f56.2;
if54 = f55.3, f56.3;
if55 = f56.4, f57.1;
if41 = f41.3, f42.1;
if42 = f43.3, f46.1;
if43 = f44.3, f46.2;
if44 = f45.3, f46.3;

```



```

if45      = f46.4, f47.1;
if31      = f31.3, f32.1;
if32      = f33.3, f36.1;
if33      = f34.3, f36.2;
if34      = f35.3, f36.3;
if35      = f36.4, f37.1;
if21      = f21.3, f22.1;
if22      = f23.3, f26.1;
if23      = f24.3, f26.2;
if24      = f25.3, f26.3;
if25      = f26.4, f27.1;
if11      = f11.3, f12.1;
if12      = f13.3, f16.1;
if13      = f14.3, f16.2;
if14      = f15.3, f16.3;
if15      = f16.4, f17.1;
if01      = f01.3, f02.1;
if02      = f03.3, f06.1;
if03      = f04.3, f06.2;
if04      = f05.3, f06.3;
if05      = f06.4, f07.1;
ilg1      = g11.3, g12.1;
i2g1      = g13.2, g12.2;
ilg2      = g21.4, g24.1;
i2g2      = g22.3, g24.2;
i3g2      = g23.2, g24.3;

```

equivalence

```

x[0]      = d0.1, c0.1, b0.1, a0.1;
x[1]      = d1.1, c1.1, b1.1, a1.1;
x[2]      = d2.1, c2.1, b2.1, a2.1;
x[3]      = d3.1, c3.1, b3.1, a3.1;
y[0]      = a3.2, a2.2, a1.2, a0.2;
y[1]      = b3.2, b2.2, b1.2, b0.2;
y[2]      = c3.2, c2.2, c1.2, c0.2;
y[3]      = d3.2, d2.2, d1.2, d0.2;
s[0]      = a0.3;
s[1]      = h01.3;
s[2]      = h11.3;
s[3]      = c01.3;
s[4]      = c12.3;
s[5]      = c22.3;
s[6]      = f52.3;
s[7]      = f57.2;

```

end;

ANEXO 7 - RELATÓRIO DE PARTICIONAMENTO

**Particao do circuito C:\TRANCA\ARQ\COM\TREVO
em 6 bandas e 1 regioes**

Numero de celulas = 79, redes = 87

s = 70, TMAX = 71, FAT.FORMA = 0.980

TAM.REG. = 4237

Banda 1 [677]
 Regiao 1 [677]
 Lista de banda
 f45, f46, f47, f44, f43, f41, f57, f56, f55, f54, f53,
 f52, f51

Banda 2 [720]
 Regiao 1 [720]
 Lista de banda
 c13, c11, g13, g12, g11, c12, g24, g23, g22, g21, c23,
 c22, c21, f42

Banda 3 [710]
 Regiao 1 [710]
 Lista de banda
 f36, f35, f34, f33, f32, f31, c2, c3, d3, d2, f37,
 h22, h21

Banda 4 [678]
 Regiao 1 [678]
 Lista de banda
 c01, a3, c03, f17, f27, f26, f25, f24, f23, f22, f21,
 b3, d1

Banda 5 [724]
 Regiao 1 [724]
 Lista de banda
 a1, a0, c0, a2, b2, d0, c1, f16, f15, f14, f13, f11,
 f12

Banda 6 [728]
 Regiao 1 [728]
 Lista de bloco
 f07, h12, h11, f01, f02, f03, f04, f05, f06, b1, b0,
 h01, h02

Rede	Regiao.Banda
x[0]	1.5, 1.6
x[1]	1.4, 1.5, 1.6
x[2]	1.3, 1.5
x[3]	1.3, 1.4
y[0]	1.4, 1.5
y[1]	1.4, 1.5, 1.6
y[2]	1.3, 1.5
y[3]	1.3, 1.4, 1.5
s[0]	1.5
s[1]	1.6
s[2]	1.6
s[3]	1.4

s[4]	1.2	
s[5]	1.2	
s[6]	1.1	
s[7]	1.1	
e1f5	1.3, 1.1	
e2f5	1.1	
cif5	1.2, 1.1	
e1f4	1.3, 1.1	
e2f4	1.3, 1.1	
cif4	1.3, 1.2, 1.1	
sf4	1.2	
elh2	1.3	
e2h2	1.4, 1.3	
sh2	1.3	
e2f3	1.4, 1.3	
cif3	1.4, 1.3	
cof3	1.3, 1.2	
sf3	1.3, 1.2	
cic2	1.2	
gc2	1.2	
pc2	1.2	
pc1	1.2	
g0g2	1.2	
p0g2	1.4, 1.2	
e2c1	1.4, 1.2	
e1f1	1.5	
e2f1	1.5	
cif1	1.5	
sf1	1.5, 1.4	
e2f2	1.4	
cif2	1.6, 1.4	
sf2	1.4	
e2c0	1.6, 1.4	
e1f0	1.6	
e2f0	1.5, 1.6	
cif0	1.5, 1.6	
sf0	1.6	
e2h1	1.6	
elh0	1.6	
e2h0	1.5, 1.6	
if51	1.1	
if52	1.1	
if53	1.1	
if54	1.1	
if55	1.1	
if41	1.1, 1.2	
if42	1.1	
if43	1.1	
if44	1.1	
if45	1.1	
if31	1.3	
if32	1.3	
if33	1.3	
if34	1.3	
if35	1.3	
if21	1.4	
if22	1.4	
if23	1.4	
if24	1.4	
if25	1.4	
if11	1.5	
if12	1.5	
if13	1.5	
if14	1.5	
if15	1.5, 1.4	
if01	1.6	
if02	1.6	
if03	1.6	
if04	1.6	
if05	1.6	
ilg1	1.2	
i2g1	1.2	

i1g2 1.2
 i2g2 1.2
 i3g2 1.2

Total de orientacoes atendidas = 0 de 0

Celulas de Interconexao na Banda	1 =	6+	3+	9	[893]
Celulas de Interconexao na Banda	2 =	14+	3+	8	[1020]
Celulas de Interconexao na Banda	3 =	17+	6+	5	[1046]
Celulas de Interconexao na Banda	4 =	19+	6+	5	[1038]
Celulas de Interconexao na Banda	5 =	18+	6+	4	[1060]
Celulas de Interconexao na Banda	6 =	8+	4+	5	[932]

Legenda: ent_ban + sai_iso + n_abut

Ordenacao da interface Norte atendida

Ordenacao da interface Sul atendida

Ordenacao da interface Leste atendida

Ordenacao da interface Oeste atendida

Total de ordenacoes atendidas = 0 de 0

Area do circuito = 660540
 Aumento largura = 50.106 %
 Relacao de aspecto = 0.556

ANEXO 8 - RELATÓRIO DE POSICIONAMENTO INTRABANDA

Posicionamento do circuito C:\TRANCA\ARQ\COM\TREVO

Banda 1

celula=	f57	coord_simbolica=	25	st_rotac=	espelhada
celula=	f56	coord_simbolica=	46	st_rotac=	espelhada
celula=	f55	coord_simbolica=	98	st_rotac=	espelhada
celula=	f54	coord_simbolica=	151	st_rotac=	espelhada
celula=	f53	coord_simbolica=	204	st_rotac=	espelhada
celula=	f51	coord_simbolica=	257	st_rotac=	normal
celula=	f52	coord_simbolica=	328	st_rotac=	normal
celula=	f47	coord_simbolica=	399	st_rotac=	espelhada
celula=	f46	coord_simbolica=	420	st_rotac=	espelhada
celula=	f45	coord_simbolica=	472	st_rotac=	espelhada
celula=	f44	coord_simbolica=	525	st_rotac=	espelhada
celula=	f41	coord_simbolica=	578	st_rotac=	normal
celula=	f43	coord_simbolica=	649	st_rotac=	espelhada

Banda 2

celula=	g13	coord_simbolica=	4	st_rotac=	normal
celula=	g12	coord_simbolica=	25	st_rotac=	normal
celula=	g11	coord_simbolica=	60	st_rotac=	espelhada
celula=	c22	coord_simbolica=	95	st_rotac=	espelhada
celula=	c12	coord_simbolica=	166	st_rotac=	espelhada
celula=	c21	coord_simbolica=	237	st_rotac=	espelhada
celula=	c23	coord_simbolica=	308	st_rotac=	normal
celula=	g23	coord_simbolica=	361	st_rotac=	normal
celula=	g24	coord_simbolica=	382	st_rotac=	espelhada
celula=	g21	coord_simbolica=	438	st_rotac=	espelhada
celula=	g22	coord_simbolica=	494	st_rotac=	espelhada
celula=	f42	coord_simbolica=	529	st_rotac=	espelhada
celula=	c11	coord_simbolica=	600	st_rotac=	espelhada
celula=	c13	coord_simbolica=	671	st_rotac=	espelhada

Banda 3

celula=	f37	coord_simbolica=	9	st_rotac=	normal
celula=	f36	coord_simbolica=	30	st_rotac=	espelhada
celula=	f35	coord_simbolica=	82	st_rotac=	espelhada
celula=	f34	coord_simbolica=	135	st_rotac=	espelhada
celula=	f33	coord_simbolica=	188	st_rotac=	espelhada
celula=	f31	coord_simbolica=	241	st_rotac=	normal
celula=	d3	coord_simbolica=	312	st_rotac=	espelhada
celula=	f32	coord_simbolica=	365	st_rotac=	normal
celula=	c3	coord_simbolica=	436	st_rotac=	normal
celula=	c2	coord_simbolica=	489	st_rotac=	normal
celula=	d2	coord_simbolica=	542	st_rotac=	espelhada
celula=	h21	coord_simbolica=	595	st_rotac=	espelhada
celula=	h22	coord_simbolica=	666	st_rotac=	espelhada

Banda 4

celula=	f25	coord_simbolica=	25	st_rotac=	normal
celula=	a3	coord_simbolica=	78	st_rotac=	normal
celula=	b3	coord_simbolica=	131	st_rotac=	normal
celula=	f26	coord_simbolica=	184	st_rotac=	normal
celula=	f23	coord_simbolica=	236	st_rotac=	espelhada
celula=	f24	coord_simbolica=	289	st_rotac=	espelhada
celula=	f21	coord_simbolica=	342	st_rotac=	normal
celula=	f17	coord_simbolica=	413	st_rotac=	espelhada
celula=	f22	coord_simbolica=	434	st_rotac=	normal
celula=	d1	coord_simbolica=	505	st_rotac=	normal
celula=	f27	coord_simbolica=	558	st_rotac=	normal

celula= c03 coord_simbolica= 579 st_rotac=espelhada
 celula= c01 coord_simbolica= 632 st_rotac=normal

Banda 5

celula= a0 coord_simbolica= 2 st_rotac=espelhada
 celula= a1 coord_simbolica= 55 st_rotac=normal
 celula= a2 coord_simbolica= 108 st_rotac=normal
 celula= c0 coord_simbolica= 161 st_rotac=normal
 celula= b2 coord_simbolica= 214 st_rotac=normal
 celula= c1 coord_simbolica= 267 st_rotac=normal
 celula= f12 coord_simbolica= 320 st_rotac=espelhada
 celula= f11 coord_simbolica= 391 st_rotac=espelhada
 celula= d0 coord_simbolica= 462 st_rotac=espelhada
 celula= f15 coord_simbolica= 515 st_rotac=normal
 celula= f16 coord_simbolica= 568 st_rotac=espelhada
 celula= f14 coord_simbolica= 620 st_rotac=espelhada
 celula= f13 coord_simbolica= 673 st_rotac=espelhada

Banda 6

celula= f07 coord_simbolica= 0 st_rotac=normal
 celula= f06 coord_simbolica= 21 st_rotac=espelhada
 celula= f03 coord_simbolica= 73 st_rotac=espelhada
 celula= f04 coord_simbolica= 126 st_rotac=espelhada
 celula= f05 coord_simbolica= 179 st_rotac=espelhada
 celula= f01 coord_simbolica= 232 st_rotac=normal
 celula= f02 coord_simbolica= 303 st_rotac=normal
 celula= b1 coord_simbolica= 374 st_rotac=espelhada
 celula= b0 coord_simbolica= 427 st_rotac=normal
 celula= h01 coord_simbolica= 480 st_rotac=normal
 celula= h02 coord_simbolica= 551 st_rotac=normal
 celula= h12 coord_simbolica= 604 st_rotac=normal
 celula= h11 coord_simbolica= 657 st_rotac=normal

ANEXO 9 - RELATÓRIO DE ROTEAMENTO

Roteamento da Banda 1 do Circuito C:\TRANCA\ARQ\COM\TREVO

Topo VCC. Ord_tri VCC P7 P5 P3 P9 P1 P0 P8 P2 P4 P6 GND.

Largura Banda=1099

Celula Tipo Largura Coord Nodos

	INT	12	87	s[7]								
f57	not	21	99	&s[7]	&s[7]	*if55						
f56	nor	52	120	&if55	&if55	*if53	*if52	*if54				
f55	and	53	172	&if54	&if54	*e2f5	*cif5					
	INT	12	225	if53								
f54	and	53	237	&if53	&if53	*e2f5	*elf5					
	INT	12	290	if52								
f53	and	53	302	&if52	&if52	*cif5	*elf5					
f51	xor	71	355	*elf5	*e2f5	&if51						
f52	xor	71	426	*if51	*cif5	&s[6]						
	INT	12	497	s[6]								
	EXT	12	509	elf5								
	INT	12	521	e2f5								
	EXT	12	533	cif5								
	VAZ	10	545									
f47	not	21	555	&e2f5	&e2f5	*if45						
f46	nor	52	576	&if45	&if45	*if43	*if42	*if44				
f45	and	53	628	&if44	&if44	*e2f4	*cif4					
	VAZ	38	681									
	EXT	12	719	e2f4								
	EXT	12	731	cif4								
	VAZ	18	743									
	EXT	12	761	if41								
	VAZ	2	773									
	INT	12	775	if43								
f44	and	53	787	&if43	&if43	*e2f4	*elf4					
f41	xor	71	840	*e2f4	*elf4	&if41						
	INT	12	911	if41								
	INT	12	923	if42								
f43	and	53	935	&if42	&if42	*elf4	*cif4					
	VAZ	94	988									
	EXT	12	1082	elf4								

Acesso a Interface Oeste-Leste:

Oeste	Leste
VCC	VCC

GND GND

Redes Nao Roteadas:

Roteamento da Banda 2 do Circuito C:\TRANCA\ARQ\COM\TREVO

Topo GND. Ord_tri GND P6 P4 P2 P8 P0 P1 P9 P3 P5 P7 VCC.

Largura Banda=1099

Celula Tipo Largura Coord Nodos

Celula	Tipo	Largura	Coord	Nodos
	EXT	12	45	cof3
g13	NOT	21	57	*pc1 &i2g1 &i2g1
g12	nand	35	78	*i2g1 *ilg1 &cic2 &cic2
	INT	12	113	cic2
	INT	12	125	ilg1
g11	nand	35	137	&ilg1 &ilg1 *pc1 *p0g2
	INT	12	172	s[5]
c22	xor	71	184	&s[5] *cic2 *pc2
	INT	12	255	s[4]
c12	xor	71	267	&s[4] *p0g2 *pc1
	INT	12	338	pc2
c21	xor	71	350	&pc2 *cof3 *sf4
	VAZ	2	421	
	EXT	12	423	elf5
c23	and	53	435	*cof3 *sf4 &gc2 &gc2
g23	not	21	488	*gc2 &i3g2 &i3g2
	EXT	12	509	elf5
	INT	12	521	i3g2
	EXT	12	533	cif5
	INT	12	545	cif5
	VAZ	14	557	
	EXT	12	571	sf3
	EXT	12	583	e2f4
g24	nand	56	595	&cif5 &cif5 *i3g2 *i2g2 *ilg2
g21	nand	56	651	&ilg2 &ilg2 *p0g2 *pc1 *pc2
	INT	12	707	i2g2
	EXT	12	719	e2f4
	EXT	12	731	cif4
	VAZ	6	743	
	EXT	12	749	elf4
	EXT	12	761	if41
g22	nand	35	773	&i2g2 &i2g2 *pc2 *g0g2
f42	xor	71	808	&sf4 *if41 *cif4
	INT	12	879	pc1
c11	xor	71	891	&pc1 *e2c1 *sf3
	INT	12	962	g0g2
c13	and	53	974	&g0g2 &g0g2 *sf3 *e2c1
	VAZ	19	1027	
	EXT	12	1046	cif4
	EXT	12	1058	p0g2
	EXT	12	1070	e2c1
	EXT	12	1082	elf4

Acesso a Interface Oeste-Leste:

Oeste Leste
GND GND

s[5]
s[4]

VCC VCC

Redes Nao Roteadas:

Roteamento da Banda 3 do Circuito C:\TRANCA\ARQ\COM\TREVO

Topo VCC. Ord_tri VCC P7 P5 P3 P9 P1 P0 P8 P2 P4 P6 GND.

Largura Banda=1099

Celula Tipo Largura Coord Nodos

```

f37  not      21    24  *if35 &cof3 &cof3
     EXT      12    45  cof3
     INT      12    57  cof3
     INT      12    69  if35
f36  nor      52    81  &if35 &if35 *if33 *if32 *if34
f35  and      53   133 &if34 &if34 *e2f3 *cif3
     INT      12   186  if33
f34  and      53   198 &if33 &if33 *e2f3 *sh2
     INT      12   251  if32
f33  and      53   263 &if32 &if32 *cif3 *sh2
f31  xor      71   316 *e2f3 *sh2 &if31
     EXT      12   387  e2f3
     EXT      12   399  cif3
     INT      12   411  elf5
     EXT      12   423  elf5
     EXT      12   435  x[3]
d3   and      53   447  &elf5 &elf5 *x[3] *y[3]
f32  xor      71   500 *if31 *cif3 &sf3
     EXT      12   571  sf3
     EXT      12   583  e2f4
     INT      12   595  sf3
     EXT      12   607  y[2]
c3   and      53   619 *x[3] *y[2] &e2f4 &e2f4
     INT      12   672  e2f4
c2   and      53   684 *y[2] *x[2] &elh2 &elh2
     INT      12   737  elh2
     EXT      12   749  elf4
     INT      12   761  elf4
d2   and      53   773 &elf4 &elf4 *x[2] *y[3]
     VAZ      24   826
     EXT      12   850  y[3]
     EXT      12   862  x[2]
     INT      12   874  sh2
h21  xor      71   886 &sh2 *e2h2 *elh2
     INT      12   957  cif4
h22  and      53   969 &cif4 &cif4 *elh2 *e2h2
     EXT      12  1022  e2h2
     EXT      12  1034  p0g2
     EXT      12  1046  cif4
     EXT      12  1058  p0g2
     EXT      12  1070  e2c1
     EXT      12  1082  e2c1

```

Acesso a Interface Oeste-Leste:

Oeste Leste
VCC VCC

y[3]

GND GND

Roteamento da Banda 4 do Circuito C:\TRANCA\ARQ\COM\TREVO

Topo GND. Ord_tri GND P6 P4 P2 P8 P0 P1 P9 P3 P5 P7 VCC.

Largura Banda=1099

Celula Tipo Largura Coord Nodos

Celula	Tipo	Largura	Coord	Nodos
f25	EXT	12	27	y[0]
	and	53	39	*cif2 *e2f2 &if24 &if24
	INT	12	92	if24
a3	and	53	104	*y[0] *x[3] &e2f2 &e2f2
	INT	12	157	e2f2
b3	and	53	169	*x[3] *y[1] &e2f3 &e2f3
	INT	12	222	e2f3
f26	nor	52	234	*if24 *if22 *if23 &if25 &if25
	INT	12	286	if25
	EXT	12	298	y[1]
	EXT	12	310	cif2
	INT	12	322	if22
f23	and	53	334	&if22 &if22 *cif2 *sf1
	EXT	12	387	e2f3
	EXT	12	399	cif3
	INT	12	411	if23
	EXT	12	423	sf1
	EXT	12	435	x[3]
f24	and	53	447	&if23 &if23 *sf1 *e2f2
f21	xor	71	500	*sf1 *e2f2 &if21
	INT	12	571	cif3
	EXT	12	583	y[2]
	EXT	12	595	if15
	EXT	12	607	y[2]
	VAZ	5	619	
f17	not	21	624	&cif3 &cif3 *if15
f22	xor	71	645	*if21 *cif2 &sf2
d1	and	53	716	*x[1] *y[3] &e2h2 &e2h2
	INT	12	769	e2h2
f27	not	21	781	*if25 &e2c1 &e2c1
	INT	12	802	e2c1
	EXT	12	814	x[1]
	EXT	12	826	y[3]
	EXT	12	838	x[2]
	EXT	12	850	y[3]
	EXT	12	862	x[2]
	INT	12	874	p0g2
c03	and	53	886	&p0g2 &p0g2 *sf2 *e2c0
c01	xor	71	939	*sf2 *e2c0 &s[3]
	INT	12	1010	s[3]
	EXT	12	1022	e2h2
	EXT	12	1034	p0g2
	EXT	12	1046	e2c0
	VAZ	24	1058	
	EXT	12	1082	e2c1

Acesso a Interface Oeste-Leste:

Oeste Leste
GND GND

y[0] s[3]
x[3]

y[0]

Roteamento da Banda 5 do Circuito C:\TRANCA\ARQ\COM\TREVO

Topo VCC. Ord_tri VCC P7 P5 P3 P9 P1 P0 P8 P2 P4 P6 GND.

Largura Banda=1099

Celula Tipo Largura Coord Nodos

	EXT	12	27	y[0]					
	VAZ	6	39						
	EXT	12	45	cif2					
	EXT	12	57	x[0]					
	EXT	12	69	x[1]					
	EXT	12	81	e2h0					
	INT	12	93	s[0]					
a0	and	53	105	&s[0]	&s[0]	*y[0]	*x[0]		
a1	and	53	158	*y[0]	*x[1]	&e2h0	&e2h0		
	INT	12	211	e2h0					
a2	and	53	223	*y[0]	*x[2]	&e2f0	&e2f0		
	INT	12	276	e2f0					
	VAZ	10	288						
	EXT	12	298	y[1]					
	EXT	12	310	cif2					
	VAZ	6	322						
c0	EXT	12	328	e2f0					
	and	53	340	*x[0]	*y[2]	&cif0	&cif0		
	INT	12	393	cif0					
	EXT	12	405	cif0					
	VAZ	6	417						
	EXT	12	423	sf1					
	EXT	12	435	y[1]					
b2	and	53	447	*x[2]	*y[1]	&e2f1	&e2f1		
	INT	12	500	e2f1					
c1	and	53	512	*x[1]	*y[2]	&cif1	&cif1		
	INT	12	565	cif1					
	VAZ	6	577						
	EXT	12	583	y[2]					
	EXT	12	595	if15					
	INT	12	607	sf1					
f12	xor	71	619	&sf1	*cif1	*if11			
f11	xor	71	690	&if11	*e2f1	*elf1			
d0	and	53	761	&elf1	&elf1	*y[3]	*x[0]		
	EXT	12	814	x[1]					
	EXT	12	826	y[3]					
	EXT	12	838	x[2]					
f15	and	53	850	*cif1	*e2f1	&if14	&if14		
	INT	12	903	if14					
	INT	12	915	if15					
f16	nor	52	927	&if15	&if15	*if14	*if12	*if13	
	EXT	12	979	e2c0					
f14	and	53	991	&if13	&if13	*e2f1	*elf1		
	VAZ	2	1044						
	EXT	12	1046	e2c0					
	INT	12	1058	if12					
f13	and	53	1070	&if12	&if12	*cif1	*elf1		

Acesso a Interface Oeste-Leste:

Oeste Leste
VCC VCC
Y[0]

s[0]
y[2]

x[2]

Legenda

INT - celula de interconexao intrabanda
 EXT - celula de interconexao interbanda
 VAZ = espaco vazio
 * - entrada de celula funcional
 & - saida de celula funcional

Informacoes do Modulo Gerado

Area do Circuito = 695082
 Relacao de Aspecto = 0.528

Interface Norte/Sul

Rede	Banda	Celula	CoordX	Pino
x[0]	6	b0	666	1
x[1]	6	b1	613	2
y[1]	6	b1	613	1
s[1]	6	INTERC	790	
s[2]	6	INTERC	1074	
s[6]	1	INTERC	497	
s[7]	1	INTERC	87	

Interface Oeste/Leste

Rede	Banda	CoordX	Trilha
x[2]	5	21	P2
x[3]	4	21	P3
y[0]	4	21	P9
y[2]	5	21	P1
y[3]	3	1126	P3
s[0]	5	21	P9
s[3]	4	1126	P9
s[4]	2	21	P1
s[5]	2	21	P0

ANEXO 10 - EXEMPLO DE UTILIZAÇÃO DA INTERFACE HOMEM-MÁQUINA

A10.1 Descrição do Circuito

Arquivos	Celulas	ciRcuito	Sintese	Verificacao	saida
-----------------	---------	----------	---------	-------------	-------

Directorio					
Alterar dir					
Circuito MODEM					
Limpa Tela					
OS shell	Alt-D				
Fin	Alt-F				

CONTROLA.NIL	CONVERSO.NIL	DEC.NIL	DEC_INST.NIL	FFLOPD.NIL	
GAMA.NIL	GOMA.NIL	LOG.NIL	LOGI.NIL	MODEM.NIL	
MULPA2.NIL	TESTE.NIL	TREVO.NIL	ULA_4B.NIL		

PROJETO

TRANCA

Concepcao de circuitos integrados Standard Cell com transparencia

 Universidade Federal do Rio Grande do Sul
 Grupo de Microeletronica

F1 - HELP	GERENCIAMENTO DE FERRAMENTAS	TRANCA - GME - UFRGS
-----------	------------------------------	----------------------

Arquivos	Celulas	ciRcuito	Sintese	Verificacao	saida
-----------------	---------	-----------------	---------	-------------	-------

Textual	Alt-E				
Grafico					
Compilacao					

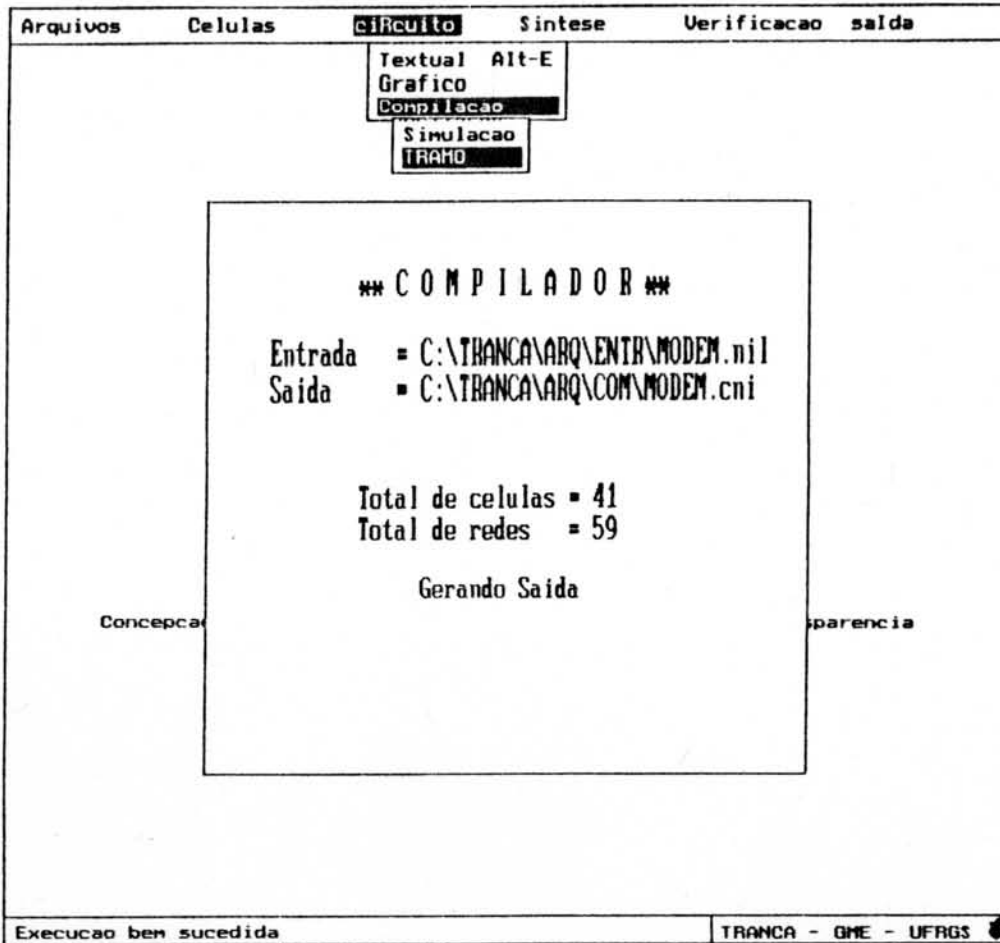
PROJETO

TRANCA

Concepcao de circuitos integrados Standard Cell com transparencia

 Universidade Federal do Rio Grande do Sul
 Grupo de Microeletronica

F1 - HELP	GERENCIAMENTO DE FERRAMENTAS	TRANCA - GME - UFRGS
-----------	------------------------------	----------------------

A10.2 Compilação da DescriçãoA10.3 Declaração dos Parâmetros de Síntese

A10.4 Particionamento

Arquivos Celulas ciRcuito **Sintese** Verificacao salda

TRANCA

- Parametros F7
- Particionamento F8**
- posicionamento F9
- Roteamento Alt F7
- relatorio Alt F8
- Exibicao Alt F9

```

** PARTI CION A D O R **
Selecao por area                Nro regioes=1

Nro bandas =5
Fat forma =1.000000 Acresc area=0.000000
Entrada   = C:\TRANCA\ARQ\COM\MODEM.cni
Saida    = C:\TRANCA\ARQ\COM\MODEM.pos
Relatorio = C:\TRANCA\ARQ\REL\MODEM.rpa

                nb S Area Interf
Concepcao Gerando Particao : 5 30 559.636 0
parencia Melhor Solucao : 5 30 559.636 0
Particao gerada = 1 5 5
Gera Saida
    
```

Execucao bem sucedida TRANCA - GME - UFRGS

A10.5 Exibição do Particionamento

Arquivos Celulas ciRcuito **Sintese** Verificacao salda

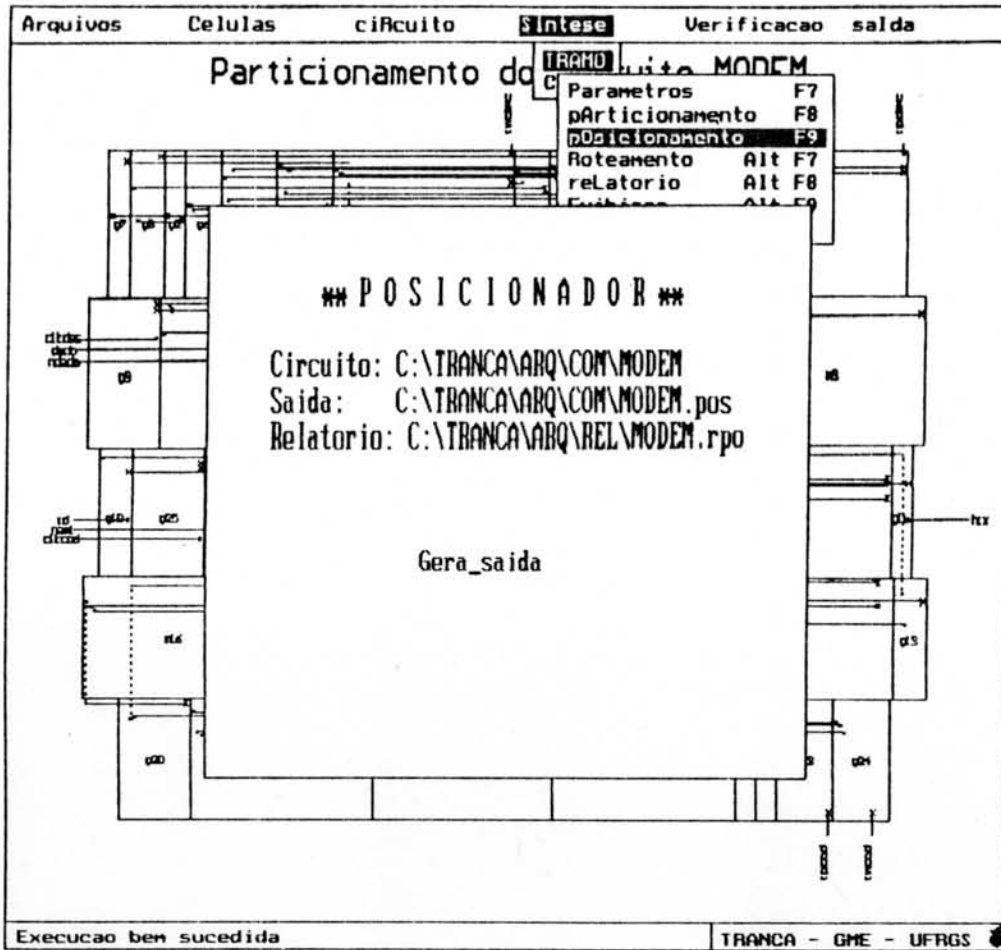
TRANCA

- Parametros F7
- Particionamento F8**
- posicionamento F9
- Roteamento Alt F7
- relatorio Alt F8
- Exibicao Alt F9
- Sintese Automatica

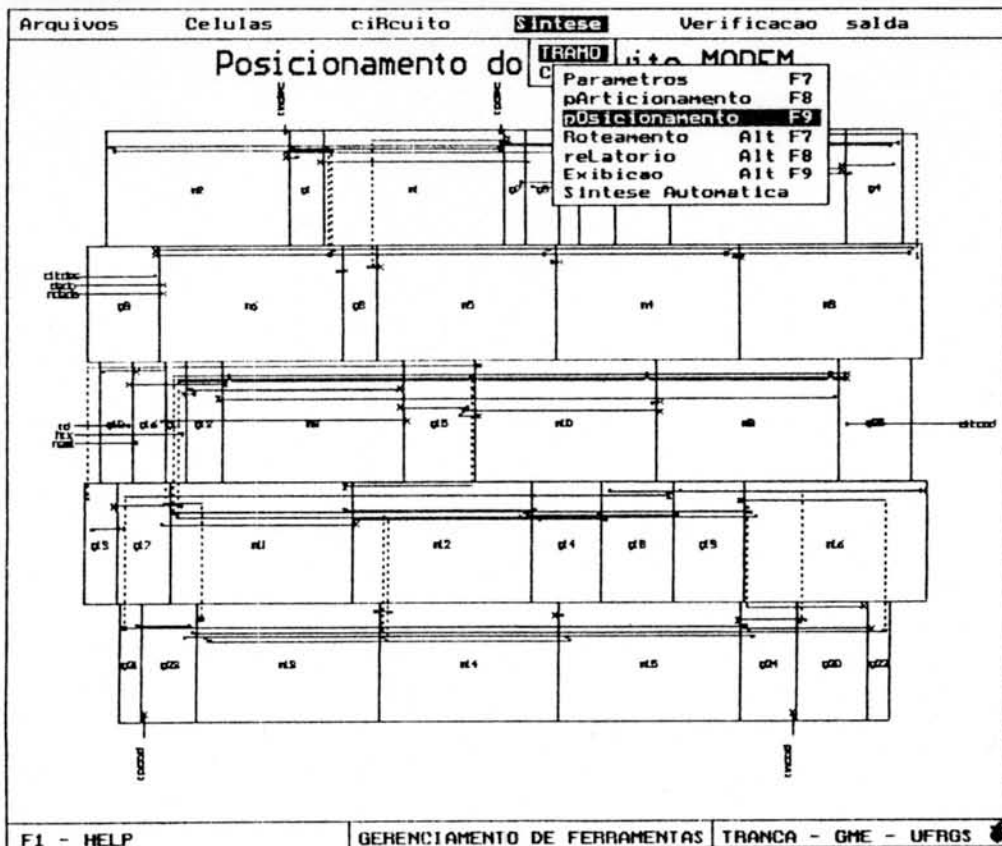
Particionamento do Circuito MODEM

F1 - HELP GERENCIAMENTO DE FERRAMENTAS TRANCA - GME - UFRGS

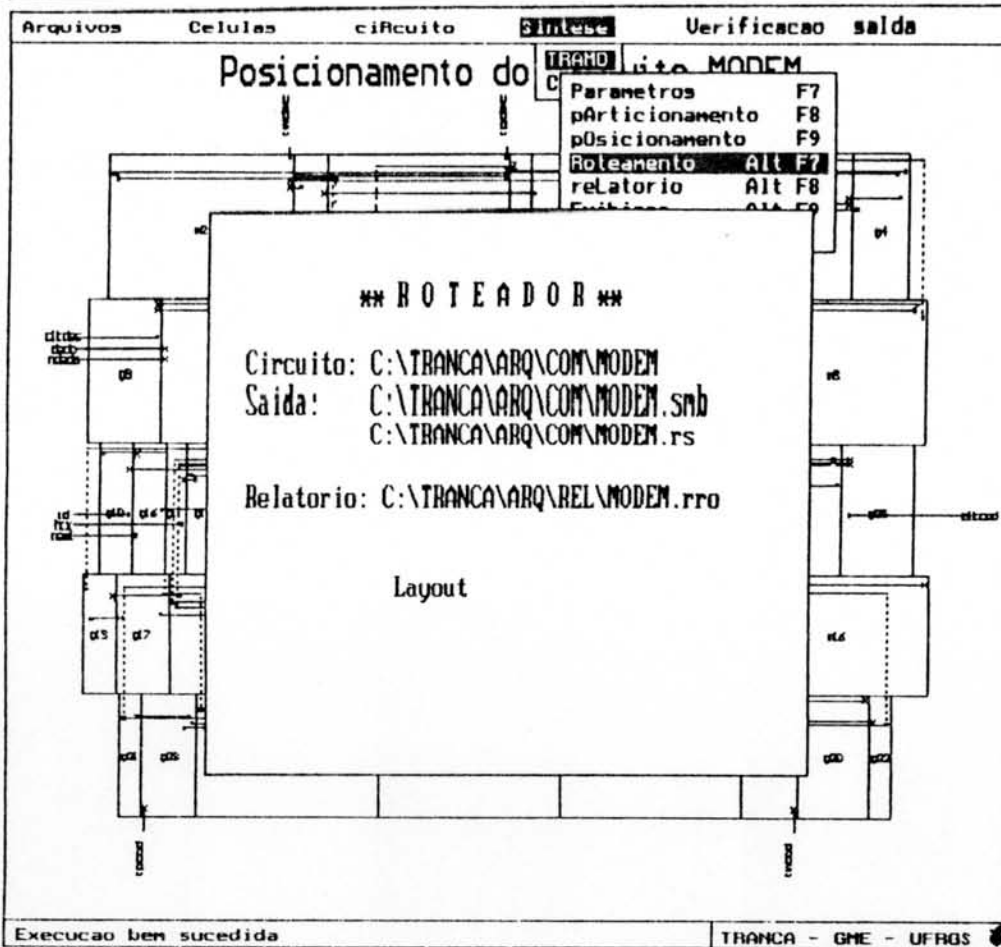
A10.6 Posicionamento



A10.7 Exibição do Posicionamento



A10.8 Roteamento



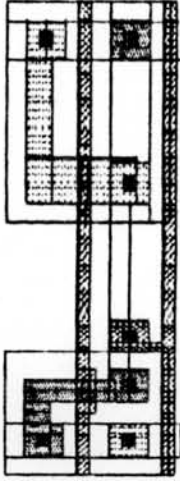
A10.9 Exibição do Leiaute



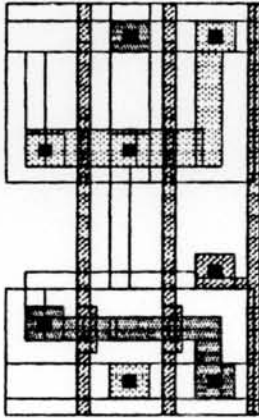
ANEXO 11 - BIBLIOTECA DE "STANDARD CELLS" CONVENCIONAIS

Altura da Biblioteca: 81 lambdas

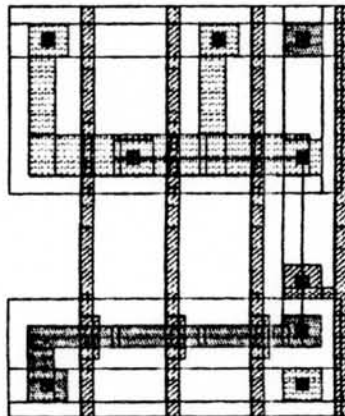
A11.1 Célula: not.cel
Largura: 28 lambdas
Leiaute:



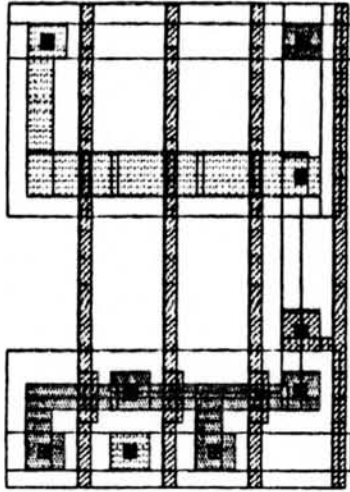
A11.2 Célula: nand2.cel
Largura: 41 lambdas
Leiaute:



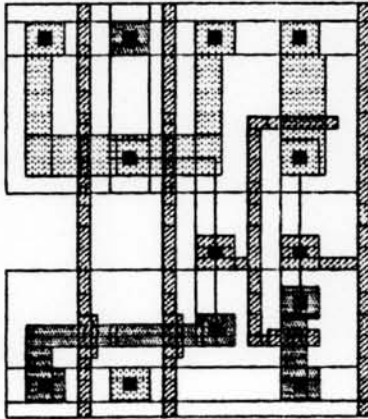
A11.3 Célula: nand3.cel
Largura: 54 lambdas
Leiaute:



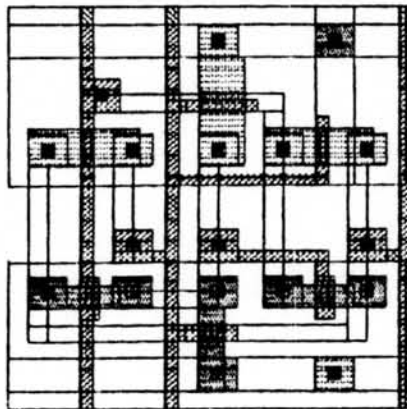
A11.4 Célula: **nor3.cel**
 Largura: 54 lambdas
 Leiaute:



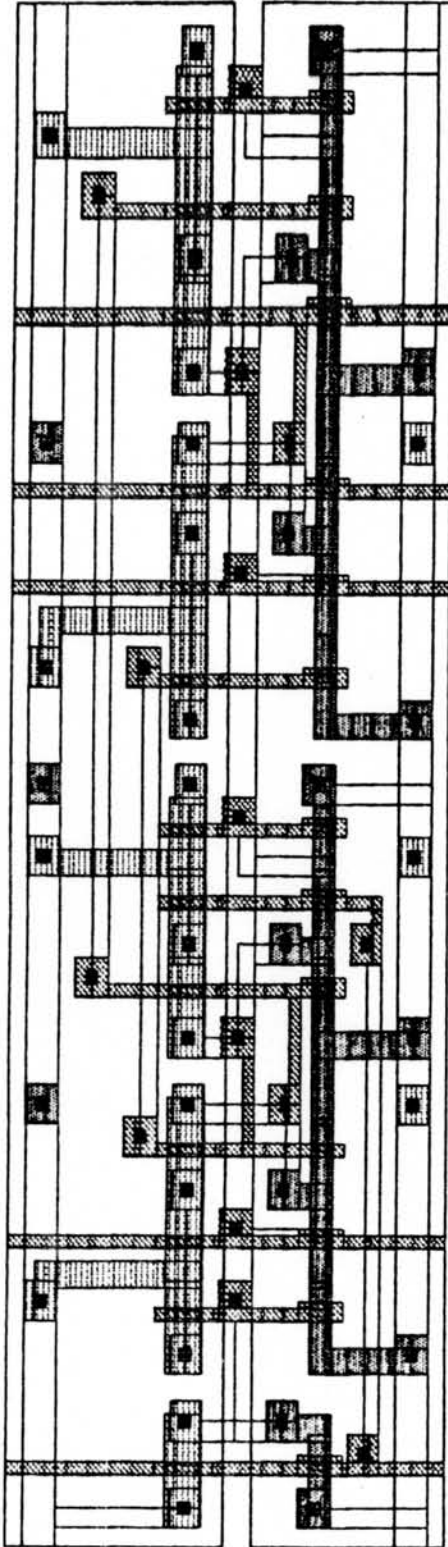
A11.5 Célula: **and2.cel**
 Largura: 58 lambdas
 Leiaute:



A11.6 Célula: **xor.cel**
 Largura: 64 lambdas
 Leiaute:



A11.7 Célula: ffdms.cel
Largura: 223 lambdas
Leiaute:



BIBLIOGRAFIA

- [AKE 81] AKERS, S. B. On the Use of the Linear Assignment Algorithm in Module Placement. In: DESIGN AUTOMATION CONFERENCE, 18, Nashville, June 29, 30 - July 1, 1981. Proceedings. New York, IEEE, 1981, p. 137-43.
- [ANC 82] ANCEAU, F. & REIS, R. A. L. Complex Integrated Circuit Design Strategy. IEEE Journal of Solid State Circuits, New York, 17(3): 459-64, June 1982.
- [AUD 87] AUDE, J. S. et al. Sistema de Geração Automática de Layout com Acelerador de Roteamento. In: CONGRESSO DA SOCIEDADE BRASILEIRA DE COMPUTAÇÃO, 8, Rio de Janeiro, 17-22 Julho, 1988. Anais. Rio de Janeiro, SBC, 1988. p. 245-61.
- [BAG 89] BAGGIO, A. Uma Interface de Gerenciamento para o Projeto TRANCA. Porto Alegre, Departamento de Informática/UFRGS, Agosto 1989. Projeto de Diplomação.
- [BUR 83a] BURSTEIN, M. & PELAVIN, R. Hierarchical Channel Router. In: DESIGN AUTOMATION CONFERENCE, 20, Miami Beach, June 27-29, 1983. Proceedings. New York, IEEE, 1983, p. 591-7.
- [BUR 83b] BURSTEIN, M. & PELAVIN, R. Hierarchical Channel Router. The VLSI Journal, 1: 21-38, 1983.
- [CAL 88] CALAZANS, N. L. V. et al. A Logic Simulator for an Integrated Environment of Digital Hardware Design. In: CONGRESSO DA SOCIEDADE BRASILEIRA DE MICROELETRÔNICA, 3, São Paulo, 12-14 Julho, 1988. Anais. São Paulo, SBMICRO, 1988. p. 385-95.

- [CAR 87] CARRO, L. et al. Desenvolvimento e Caracterização de uma Biblioteca de Standard Cells. Porto Alegre, Pós-Graduação em Ciência da Computação, Setembro 1987. RP 81.
- [CAR 88] CARRO, L. et al. GROM: Gerador de ROMs parametrizável. In: CONGRESSO DA SOCIEDADE BRASILEIRA DE MICROELETRÔNICA, 3, São Paulo, 12-14 Julho, 1988. Anais. São Paulo, SBMICRO, 1988. p. 471-81.
- [CAR 89] CARRO, L. Gerador Parametrizável de Partes Operativas CMOS. Porto Alegre, Pós-Graduação em Ciência da Computação/UFRGS, 1989. Dissertação de Mestrado.
- [CHE 88] CHEN, R. R. W. Graph Theory in the Context of VLSI and CAD. In: INTERNATIONAL SYMPOSIUM ON CIRCUITS AND SYSTEMS, Helsinki, 7-9, June 1988. Proceedings. Piscataway, IEEE, 1988. Vol. 2, p. 1305-12.
- [CHO 88] CHO, H. G & KYUNG, C. M. An $O(n)$ -Time Standard Cell Placement Algorithm using Constrained Multi-Stage Graph Mode. In: INTERNATIONAL SYMPOSIUM ON CIRCUITS AND SYSTEMS, Helsinki, 7-9 June, 1988. Proceedings. Piscataway, IEEE, 1988. Vol 2, p. 1687-90.
- [CON 88] CONG, J. & PREAS, B. A New Algorithm for Standard Cell Global Routing. In: INTERNATIONAL SYMPOSIUM ON CIRCUITS AND SYSTEMS, Helsinki, 7-9 June, 1988. Proceedings. Piscataway, IEEE, 1988. Vol. 1, p. 176-9.

- [DEM 87] De MAN, H. Evolution of CAD tools towards third generation custom VLSI design. Révue de Physique Appliquée, 22(1):31-45, Janvier 1987.
- [DEM 89] De MICHELI, G. Synthesis Systems for Digital Design. In: CONGRESSO DA SOCIEDADE BRASILEIRA DE MICROELETRÔNICA, 4, Porto Alegre, 12-14 Julho, 1989. Anais. Porto Alegre, SBMICRO, 1989. Vol. 2, p. 481-85.
- [DEU 76] DEUTSCH, D. N. A "Dogleg" Channel Router. In: DESIGN AUTOMATION CONFERENCE, 13, San Francisco, June 28-30, 1976. Proceedings. New York, IEEE, 1976. p. 425-33.
- [DOS 90] DOSSA, M. K. Estudo e Projeto de um Conversor A/D Integrado para Aplicação em um Digitalizador de Imagens em Tempo Real. Porto Alegre, Pós-Graduação em Ciência da Computação/UFRGS, 1990. Dissertação de Mestrado.
- [FID 82] FIDUCCIA, C. M. & MATHEYSES, R. M. A Linear-Time Heuristic for Improving Network Partitions. In: DESIGN AUTOMATION CONFERENCE, 19, Las Vegas, June 14-16, 1982. Proceedings. New York, IEEE, 1982. p. 175-81.
- [GAJ 83] GAJSKI, D. & KUHN, R. H. New VLSI Tools. IEEE Computer, New York, 16(12): 11-4, Dec. 1983.
- [GAL 89] GALLO, E. L. et al. GALOPA II: Gerador Automático de Layouts de PLAs. In: SIMPÓSIO BRASILEIRO DE CONCEPÇÃO DE CIRCUITOS INTEGRADOS, 4, Rio de Janeiro, 12-14 Abril, 1989. Anais. Rio de Janeiro, SBC, 1989. p. 154-63.

- [GOM 88] GOMES, R. F. DARC: Um Verificador de Regras de Projeto de CIs utilizando Programação em Lógica. In: SIMPÓSIO BRASILEIRO DE CONCEPÇÃO DE CIRCUITOS INTEGRADOS, 3, Gramado, 13-15 Abril, 1988. Anais. Porto Alegre, SBC, 1988. p. 85-94.
- [GOM 90] GOMES, R. F. Um Compilador de Silício voltado à Execução de Operações Paralelas. Porto Alegre, Pós-Graduação em Ciência da Computação/UFRGS, 1990. Dissertação de Mestrado.
- [HAS 71] HASHIMOTO, A. & STEVENS, J. Wire Routing by Optimizing Channel Assignment within Large Apertures. In DESIGN AUTOMATION WORKSHOP, 8, Atlantic City, June 28-30, 1971. Proceedings. New York, ACM, 1971. p. 155-63.
- [HAS 82] HASSET, J. E. Automated Layout in ASHLAR: An Approach to the Problems of "General Cell" Layout for VLSI. In: DESIGN AUTOMATION CONFERENCE, 19, Las Vegas, 14-16 June, 1982. Proceedings. New York, ACM/IEEE, 1982. p. 777-83.
- [HIG 69] HIGHTOWER, D. W. A Solution to the Line Routing Problem on a Continuous Plane. In: DESIGN AUTOMATION WORKSHOP, 6, 1969. Proceedings. p. 1-24.
- [HOR 83] HOROWITZ, M. Timing Models for MOS Circuits. Stanford, Stanford University, December 1983. Technical Report No. SEL83-003.

- [HSU 87] HSU, D. et al. The ChipCompiler, An Automated Standard Cell/Macrocell Physical Design Tool. In: CUSTOM INTEGRATED CIRCUITS CONFERENCE, Portland, May 1987. Proceedings. New York, IEEE, 1987. p. 488-91.
- [KAM 82] KAMBE, T. et al. A Placement Algorithm for Polycell LSI and its Evaluation. In: DESIGN AUTOMATION CONFERENCE, 19, Las Vegas, June 14-16, 1982. Proceedings. New York, IEEE, 1982. p. 655-61.
- [KAN 83] KANG, S. Linear Ordering and Application to Placement. In: DESIGN AUTOMATION CONFERENCE, 20, Miami-Beach, June 27-29, 1983. Proceedings. New York, IEEE, 1988. p. 457-64.
- [KER 70] KERNIGHAN, B. W. & LIN, S. An Efficient Heuristic Procedure for Partitioning Graphs. Bell System Technical Journal, New York, 49: 291-308, February 1970.
- [KER 73] KERNIGHAN, B. W. et al. An Optimum Channel-Routing Algorithm for Polycell Layouts of Integrated Circuits. In: DESIGN AUTOMATION WORKSHOP, 10, Portland, June 17-19, 1973. Proceedings. New York, IEEE, 1973. p.50-59.
- [KER 78] KERNIGHAN, B. W & RITCHIE, D. M. C - A Linguagem de Programação. Rio de Janeiro, Campus, 1986.
- [KOL 85] KOLLARITSCH, P.W. & WESTE, N. H. E. TOPOLOGIZER: An Expert System Translator of Transistor Connectivity to Symbolic Cell Layout. IEEE Journal of Solid State Circuits, New York, 20(3): 799-804, June 1985.

- [KOZ 83] KOZAWA, T. et al. Automatic Placement Algorithms for High Packing density VLSI. In: DESIGN AUTOMATION CONFERENCE, 20, 1983. Proceedings. New York, IEEE, 1983. p. 175-81.
- [KRI 84] KRISHNAMURTHY, B. An Improved Min-Cut Algorithm for Partitioning VLSI Networks. IEEE Transactions on Computers, New York, 33(5): 438-46, May 1984.
- [LEE 61] LEE, C. An Algorithm for Path Connections and its Applications. IRE Transactions on Electronic Computers, VEC-10: 346-65, Sept. 1961.
- [LIE 85] LIESENBERG, H. K. D. A Layout Module for a Silicon Compiler. Newcastle-Upon-Tyne, University of Newcastle-Upon-Tyne, 1985.
- [LIN 87] LIN, Y. L. S. & GAJSKI, D. D. LES: A Layout Expert System. In: DESIGN AUTOMATION CONFERENCE, 24, Miami Beach, 1987. Proceedings. New York, ACM/IEEE, 1987. p. 672-8.
- [LUB 88a] LUBASZEWSKI, M. S. et al. Uma Ferramenta para Projeto de Circuitos Standard Cell. In: SIMPÓSIO BRASILEIRO DE CONCEPÇÃO DE CIRCUITOS INTEGRADOS, 3, Gramado, 13-15 Abril, 1988. Anais. Porto Alegre, SBC, 1988. p. 173-82.
- [LUB 88b] LUBASZEWSKI, M. S. O Módulo de Posicionamento Relativo do Projeto TRANCA. Porto Alegre, Pós-Graduação em Ciência da Computação/UFRGS, 1988. 55 p. Trabalho Individual.

- [LUB 88c] LUBASZEWSKI, M. S. et al. A Biblioteca de Standard Cells do Projeto TRANCA. In: CONGRESSO DA SOCIEDADE BRASILEIRA DE MICROELETRÔNICA, 3, São Paulo, 12-14 Julho, 1988. Anais. São Paulo, SBMICRO, 1988. p. 331-41.
- [LUB 88d] LUBASZEWSKI, M. S. O Módulo de Posicionamento Relativo do Projeto TRANCA. In: JORNADAS ARGENTINAS DE INFORMÁTICA E INVESTIGACIÓN OPERATIVA, 17, Buenos Aires, Set. 1988. Anales. Buenos Aires, SADIO, 1988. Tomo I, p.47-66.
- [LUB 89a] LUBASZEWSKI, M. S. et al. A Random Logic Generator using the TRANCA Methodology. In: CONGRESSO DA SOCIEDADE BRASILEIRA DE MICROELETRÔNICA, 4, Porto Alegre, 12-14 Julho, 1989. Anais. Porto Alegre, SBMICRO, 1989. Vol. 1, p. 67-78.
- [LUB 89b] LUBASZEWSKI, M. S. et al. POTRANCA: O Subsistema de Posicionamento do Gerador de Módulos TRANCA. In: CONGRESSO DA SOCIEDADE BRASILEIRA DE COMPUTAÇÃO, 9, Uberlândia, 16-21 Julho, 1989. Anais. Uberlândia, SBC, 1989. p. 237-51.
- [MAR 89] MARCHIORO, G. F. & CARRO, L. Editor Simbólico para Circuitos Integrados. In: CONGRESSO DA SOCIEDADE BRASILEIRA DE MICROELETRÔNICA, 4, Porto Alegre, 12-14 Julho, 1989. Anais. Porto Alegre, SBMICRO, 1989. Vol. 2, p. 837-48.
- [MIK 68] MIKAMI, K. & TABUCHI K. A Computer Program for Optimal Routing of Printed Circuit Connectors. IFIPS Proc., Vol. H47: 1475-8, 1968.

- [MOR 88] MORAES, F. G. et al. Biblioteca Standard Cells do Projeto TRANCA. Porto Alegre, Pós-Graduação em Ciência da Computação/UFRGS, Agosto 1988, RP 92.
- [MOR 89a] MORAES, F.G. & REIS, R. Síntese Automática de Memórias RAM utilizando Layout Simbólico. In: SIMPÓSIO BRASILEIRO DE CONCEPÇÃO DE CIRCUITOS INTEGRADOS, 4, Rio de Janeiro, 12-14 Abril, 1989. Anais. Rio de Janeiro, SBC, 1989, p. 143-53.
- [MOR 89b] MORAES, F. G. & REIS, R. Síntese Automática de Células Utilizando Estratégia Gate-Matrix. In: CONGRESSO DA SOCIEDADE BRASILEIRA DE MICROELETRÔNICA, 4, Porto Alegre, 12-14 Julho, 1989. Anais. Porto Alegre, SBMICRO, 1989. Vol. 2, p. 227-38 .
- [MOR 89c] MORAES, F. G. & REIS, R. EXTRALO - Um Extrator Lógico. In: SEMINÁRIO INTERNO DE MICROELETRÔNICA, 5, Tramandaí, 17-18 Novembro, 1989. Anais. Porto Alegre, GME/UFRGS, 1989. p. 99-102.
- [MOR 89d] MORAES, F. G. & REIS, R. TRAGO - TRANCA Gate-Matrix Generator. In: SEMINÁRIO INTERNO DE MICROELETRÔNICA, 5, Tramandaí, 17-18 Novembro, 1989. Anais. Porto Alegre, GME/UFRGS, 1989. p. 103-106.
- [MUR 80] MURAI, S. et al. The Effects of the Initial Placement Techniques on the Final Placement Results - Constructive vs. Top-Down Techniques. In: INTERNATIONAL CONFERENCE ON CIRCUITS AND COMPUTERS, Port Chester, 1-3 October, 1980. Proceedings. New York, IEEE, 1980. p. 80-2.

- [OBR 82] OBREBSKA, M. Études Comparative de différentes méthodes de conception des Parties de Controle des microprocesseurs. Grenoble, Institut Polytechnique de Grenoble, 1982.
- [PER 89a] PEREIRA, C. E. et al. Estudo de Caso: Vantagens e Desvantagens de um Projeto com Ferramentas de PAC não Integradas. In: SIMPÓSIO BRASILEIRO DE CONCEPÇÃO DE CIRCUITOS INTEGRADOS, 4, Rio de Janeiro, 12-14 Abril, 1989. Anais. Rio de Janeiro, SBC, 1989. p. 95-104
- [PER 89b] PEREIRA, C. E. & BARONE, D. A. C. SCHAROP - Um Roteador Detalhado de Canal. In: CONGRESSO DA SOCIEDADE BRASILEIRA DE MICROELETRÔNICA, 4, Porto Alegre, 12-14 Julho, 1989. Anais. Porto Alegre, SBMICRO, 1989. Vol. 2, p. 849-59.
- [PER 90] PEREIRA, C. E. AMARGO: Ambiente Hierárquico de Roteamento de Circuitos Integrados. Porto Alegre, Pós-Graduação em Ciência da Computação/UFRGS, 1990. Dissertação de Mestrado.
- [PIG 88] PIGUET, C. et al. ALLADIN: A CMOS Gate Matrix Layout System. In: INTERNATIONAL SYMPOSIUM ON CIRCUITS AND SYSTEMS, Helsinki, 7-9 June, 1988. Proceedings. Piscataway, IEEE, 1988. Vol. 3, p. 2427-30.
- [PRE 87] PREAS, B. Benchmarks for Cell-Based Layout Systems. In: DESIGN AUTOMATION CONFERENCE, 24, Miami Beach, 1987. Proceedings. New York, ACM/IEEE, 1987. p. 319-20.
- [PRE 88] PREAS, B. T. & LORENZETTI, M. J. Physical Design Automation of VLSI Systems. Menlo Park, Benjamin/Cummings, 1988.

- [REE 85] REED, J. et al. A New Symbolic Channel Router: YACR2. IEEE Transactions on Computer-Aided Design, New York, 4(3): 208-19, July 1985.
- [REI 83] REIS, R. A. L. TESS: Evaluator Topologique Predictif pour la génération automatique des plans de masse de circuits VLSI. Grenoble, Institut Polytechnique de Grenoble, 1983.
- [REI 85] REIS, R. A. L. Estratégias Básicas para a Concepção Automática do Layout de Circuitos em Lógica Aleatória. In: SIMPÓSIO BRASILEIRO DE CONCEPÇÃO DE CIRCUITOS INTEGRADOS, 2, Porto Alegre, 20-27 Julho, 1985. Anais. Porto Alegre, SBC, 1985. p. 223-36.
- [REI 87] REIS, R. A. L. A New Standard Cell CAD Methodology. In: CUSTOM INTEGRATED CIRCUITS CONFERENCE, Portland, 4-7 May, 1987. Proceedings. New York, IEEE, 1987. p. 385-8.
- [REI 88] REIS, R. A. L. et al. An Efficient Design Methodology for Standard Cell Circuits. In: INTERNATIONAL SYMPOSIUM ON CIRCUITS AND SYSTEMS, Helsinki, 7-9 June, 1988. Proceedings. Piscataway, IEEE, 1988. Vol. 2, p. 1213-6.
- [REI 89a] REIS, R. A. L. & GOMES, R. F. Planejamento Topológico de Circuitos VLSI. In: CONGRESSO DA SOCIEDADE BRASILEIRA DE MICROELETRÔNICA, 4, Porto Alegre, 12-14 Julho, 1989. Anais. Porto Alegre, SBMICRO, 1989. Vol. 2, p. 827-36.

- [REI 89b] REIS, A. I. et al. MODEM - Desenvolvimento de um ASIC para MODEMS de Banda Base. In: CONGRESSO DA SOCIEDADE BRASILEIRA DE MICROELETRÔNICA, 4, Porto Alegre, 12-14 Julho, 1989. Anais. Porto Alegre, SBMICRO, 1989. Vol. 2, p. 617-26.
- [RIV 82] RIVEST, R. L. & FIDUCCIA, C. M. A "Greedy" Channel Router. In: DESIGN AUTOMATION CONFERENCE, 19, Las Vegas, June 14-16, 1982. Proceedings. New York, IEEE, 1982. p. 418-24.
- [ROD 86] RODRIGUEZ, J. L. T. Um Posicionador para Células Padrão em Layouts de Circuitos Integrados. In: CONGRESSO DA SOCIEDADE BRASILEIRA DE MICROELETRÔNICA, 1, Campinas, Julho 1986. Anais. Campinas, SBMICRO, 1986. p. 320-8.
- [RUB 83] RUBISTEIN, J. et al. Signal Delay in RC Tree Networks. IEEE Transactions on Computer Aided Design, New York, 2(3): 202-11, July 1983.
- [SAK 83] SAKURAI, T. Approximation of Wiring Delay in MOSFET LSI. IEEE Journal of Solid-State Circuits, New York, 8(4): 418-26, Aug. 1983.
- [SCH 72a] SCHWEIKERT, D. G. & KERNIGHAN, B. W. A Proper Model for the Partitioning of Electrical Circuits. In: DESIGN AUTOMATION WORKSHOP, 9, June 1972. Proceedings. p. 57-62.
- [SCH 72b] SCHULER, D. M. & ULRICH, E. G. Clustering and Linear Placement. In: DESIGN AUTOMATION WORKSHOP, 9, June 1972. Proceedings. p. 50-6.
- [SEC 85] SECHEN, C. & SANGIOVANNI-VINCENTELLI, A. The TimberWolf Placement and Routing Package. IEEE Journal of Solid State Circuits, New York, 20(2): 510-22, April 1985.

- [SEC 88] SECHEN, C. & CHEN, D. An Improved Objective Function for Mincut Circuit Partitioning. In: INTERNATIONAL CONFERENCE ON COMPUTER-AIDED DESIGN, Santa Clara, November 1988. Proceedings. New York, IEEE, 1988. p. 502-5.
- [SHI 80] SHIRAIISHI, H. & HIROSE, R. Efficient Placement and Routing Techniques for Master Slice LSI. In: DESIGN AUTOMATION CONFERENCE, 17, Minneapolis, June 23-25, 1980. Proceedings. New York, IEEE, 1980. p. 458-64.
- [SHR 88] SHRAGOWITZ, E. et al. Algorithms for physical design of "sea-of-gates" chips. IEEE Transactions on Computer-Aided Design, New York, 20(7): 382-97, Sept. 1988.
- [SIL 87] SILVA JÚNIOR, J. L. et al. Concepção de um CI para Controle Industrial. In: CONGRESSO DA SOCIEDADE BRASILEIRA DE COMPUTAÇÃO, 7, Salvador, 11-19 Julho, 1987. Anais. Salvador, SBC, 1987. p. 417-27.
- [SOM 89] SOMENZI, F. et al. Biblioteca de Células do Projeto TRANCA em Regras do 2o. PMU. Porto Alegre, Pós-Graduação em Ciência da Computação/UFRGS, Janeiro 1989. RP 106.
- [SOU 81] SOUKUP, J. Circuit Layout. Proceedings of IEEE, New York, 69(10): 1281-304, Oct. 1981.
- [STE 89a] STEMMER, M. A. & REIS, R. A. L. EXTRIBO - Um Extrator Hierárquico de Circuitos. In: SIMPÓSIO BRASILEIRO DE CONCEPÇÃO DE CIRCUITOS INTEGRADOS, 4, Rio de Janeiro, 12-14 Abril, 1989. Anais. Rio de Janeiro, SBC, 1989. p.1-9.

- [STE 89b] STEMMER, M. A. & REIS, R. A. L. Editor de Máscaras EMA2 - Interface Gráfica do Extrator. Porto Alegre, Pós-Graduação em Ciência de Computação/UFRGS, 1989. Trabalho Individual.
- [SUN 87] SUNTER, S. Designing a CMOS Standard Cell Library. In: CUSTOM INTEGRATED CIRCUITS CONFERENCE, Portland, 4-7 May, 1987. Proceedings. New York, IEEE, 1987. p. 237-40.
- [WAG 87a] WAGNER, F. R. et al. Linguagens de Descrição de Hardware para suporte à Integração do Processo de Projeto em AMPLO. Porto Alegre, Pós-Graduação em Ciência da Computação/UFRGS, Março 1987. RP 65.
- [WAG 87b] WAGNER, F. R. & FREITAS, C. M. D. S. NILO - Uma Linguagem para a Descrição de Hardware no nível de Portas Lógicas. Porto Alegre, Pós-Graduação em Ciência da Computação/UFRGS, Março 1987. RP 66.
- [WAG 88] WAGNER, F. R. AMPLO - Um Ambiente Integrado para Projeto de Circuitos VLSI. In: CONGRESSO DA SOCIEDADE BRASILEIRA DE MICROELETRÔNICA, 3, São Paulo, 12-14 Julho, 1988. Anais. São Paulo, SBMICRO, 1988. p. 437-46.
- [WAG 89] WAGNER, F. R. Um Ambiente Integrado para a Simulação de Sistemas Digitais. In: SIMPÓSIO BRASILEIRO DE CONCEPÇÃO DE CIRCUITOS INTEGRADOS, 4, Rio de Janeiro, 12-14 Abril, 1989. Anais. Rio de Janeiro, SBC, 1989. p. 74-82.
- [WES 85] WESTE, N. H. & ESHRAGUIAN, K. Principles of CMOS VLSI Design. Reading, Addison-Wesley, 1985.

- [YAM 88] YAMADA, M. & LIU, C. L. An Analytical Method for Optimal Module Orientation. In: INTERNATIONAL SYMPOSIUM ON CIRCUITS AND SYSTEMS, Helsinki, 7-9 June, 1988. Proceedings. Piscataway, IEEE, 1988. V. 2, p.1679-82.

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

Geração automática de lógica aleatória
utilizando a metodologia TRANCA

Dissertação apresentada aos Srs.



Prof. Dr. Flávio Rech Wagner



Prof. Dr. Ricardo A. da L. Reis




Prof. Dr. Altamiro A. Suzim



Prof. Dr. Jean Bodinaud

Visto e permitida a impressão.
Porto Alegre, .30./05.../90..


v Prof. Ricardo A. da L. Reis,
orientador


v Prof. Ricardo A. da L. Reis,
Coordenador do Curso de Pós-
-Graduação em Ciência da Com
putação.