O PROCESSO DE PROJETO DE SISTEMAS DIGITAIS NUM AMBIENTE INTEGRADO DE CAD

por

Flávio Rech Wagner
Carla Maria Dal Sasso Freitas
Lia Goldstein Golendziner

RT nº 024 CPGCC/UFRGS OUT/85

Nota técnica do projeto "BANCO DE DADOS E FERRAMENTAS PARA CAD DE SISTEMAS DIGITAIS"

Trabalho realizado com o apoio do CNPq

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

Av. Osvaldo Aranha, 99
Caixa Postal 1501
90.000 - Porto Alegre - RS - Brasil
Telex (051) 2680 Tel. (0512) 21.8499

Sistemas digitais
CAD/CAM

RESUMO

Este relatório descreve o processo de projeto de sistemas digitais complexos, que é a base de um ambiente integrado de projeto auxiliado por computador. Este processo restringe-se à síntese e validação de sistemas digitais descritos ao nível de portas lógicas ou em níveis mais altos de abstração.

PALAVRAS-CHAVE: sistemas digitais, projeto auxiliado por computador, metodologia de processo.

ABSTRACT

This report describes the process of designing complex digital systems, which is the basis of an integrated computer-aided design environment. This design process is restricted to the synthesis and validation of digital systems described at the logic level or at higher abstraction levels.

KEYWORDS: digital systems, computer-aided design, design methodology.

SUMÁRIO

1. INTRODUÇÃO

O objetivo do projeto "Banco de Dados e Ferramentas para CAD de Sistemas Digitais", em andamento no CPGCC da UFRGS, é definir um ambiente integrado de projeto de sistemas digitais complexos. Para a especificação deste ambiente é necessário planejar (ou modelar) |ENC 83| três elementos chaves de um sistema CAD:

- o processo de projeto;
- o modelo de dados;
- a interface usuário-sistema.

O processo de projeto refletirá uma metodologia de projeto ("top-down" ou "bottom-up", modularidade, etc.), na qual estão embutidas diversas ferramentas.

O modelo de dados |DAT 83| a ser adotado permitirá a representação dos sistemas digitais a serem projetados e compreende a definição de três componentes: 1) tipos de objetos sendo manipulados; 2) operadores a serem aplicados sobre instâncias destes objetos e 3) regras gerais de integridade que restringem os estados válidos de bancos de dados que enquadram-se no modelo em questão.

A interface usuário-sistema refletirá o processo de projeto, permitindo acesso a todas as ferramentas de modo uniforme. Observe-se que cada ferramenta possui sua interface específica com o usuário.

Conforme o exposto, tanto o modelo de dados como a interface usuário-sistema dependem do processo de projeto. É objetivo deste relatório descrever este processo, ou seja, a metodologia e a natureza das ferramentas associadas, sem entrar na especificação detalhada destas. Tanto o modelo de dados como a interface usuário-sistema serão objeto de re-

latórios posteriores.

A médio e longo prazo, será desenvolvido um sistema de banco de dados onde estejam integrados todas as informações relativas ao processo de projeto de um dado sistema digital, de uma forma consistente e eficiente. As diversas ferramentas de projeto estarão integradas no sistema CAD pelo uso compartilhado das informações armazenadas neste banco de dados.

2. SISTEMAS INTEGRADOS DE CAD

Mermet |MER 85| aposta três qualidades que devem estar presentes em um sistema "integrado" de CAD de sistemas digitais:

- deve cobrir todas as fases da vida do produto a a ser projetado;
- deve haver unidade e homogeneidade na forma pela qual o usuário vê às diferentes ferramentas de projeto; e
- deve haver consistência entre as diferentes representações do produto.

A primeira característica implica em que o sistema de CAD cubra desde a fase de especificação do produto até a fase de geração de vetores para o teste dos circuitos fabricados. O modelo de dados a ser proposto tem em vista principalmente a fase de projeto lógico dos sistemas digitais, mas procurar-se-á defini-lo de forma a suportar a integração posterior de ferramentas associadas às demais fases (especificação, projeto físico, teste).

Para garantir a unidade e homogeneidade de visualização do ambiente de CAD pelo usuário será definida uma interface, essencialmente gráfica, que englobe todas as ferra

mentas presentes no ambiente. Em especial, para a fase de projeto lógico, será definido um conjunto de editores gráficos embutidos dentro desta interface comum, e adequados aos diferentes níveis de descrição de sistemas digitais. A descrição destes editores será objeto de relatórios posteriores.

A consistência completa entre as diferentes representações do produto deveria idealmente ser garantida automaticamente pelo sistema de gerência de banco de dados (SGBD) que suporta o ambiente de CAD, o que no entanto ainda é um tema de pesquisa em aberto. Alguma consistência será garantida automaticamente no ambiente proposto já na fase de projeto, devido ao fato dos editores antes mencionados valerem-se de informações geradas nos níveis anteriores de projeto quando da condução do usuário através do projeto num nível posterior.

3. NÍVEIS DE DESCRIÇÃO DE SISTEMAS DIGITAIS

Um sistema digital pode ser descrito em diferentes níveis de abstração |SBN 83|, tais como:

- nível de sistema |MDO 75| (ou PMS, segundo |SBN 83|), no qual o sistema digital é visto como uma interligação de processadores, memórias, controladores, etc;

- nível RT (transferência entre registradores) |BAR 75|, onde o sistema é composto por registradores, somadores, multiplexadores, etc;

- nível lógico, no qual vê-se o sistema como sendo composto unicamente por portas lógicas elementares (AND, OR, NOT, etc.) e, eventualmente, também por flip-flops;

- nível elétrico, onde o sistema é visto como uma interligação de transistores, resistores e capacitores;

- nível do "lay-out" das máscaras para a fabricação de circuitos integrados.

Cada um destes níveis corresponde a objetivos de projeto bem específicos (ver |HAY 78|, p.ex., para uma análise do projeto nos níveis de sistema, RT e lógico). A coexixtência de ferramentas de projeto adequadas a estes diferentes níveis num único ambiente de CAD não significa que o projeto de um determinado circuito deva passar necessariamente por todas estas etapas. Circuitos integrados de baixa complexidade são tradicionalmente projetados diretamente no nível do "lay-out" das máscaras de fabricação, com uma validação no nível elétrico. O projeto a partir de níveis mais abstratos de descrição justifica-se com o crescente aumento da complexidade dos sistemas desejados.

Por outro lado, o projeto a partir de um determinado nível não implica na necessidade de que o projeto passe por todos os níveis abaixo daquele. Isto é especialmente mais verdadeiro no projeto de circuitos VLSI, onde o uso de bibliotecas de células permite que, a partir de uma descrição num alto nível de abstração, obtenha-se diretamente por um processo de síntese automática o "lay-out" das máscaras de fabricação.

No contexto deste trabalho, o modelo de dados a ser proposto cobrirá inicialmente apenas as fases do projeto de sistemas digitais nos níveis de sistema, RT e lógico. Numa etapa posterior espera-se estende-lo para os níveis inferiores.

4. DESCRIÇÕES GRÁFICAS E TEXTUAIS

A descrição de sistemas digitais contém uma forte componente gráfica. Isto é tanto mais verdadeiro quanto mais baixo o nível de descrição do sistema. No nível do "lay-out" das máscaras de fabricação de circuitos integrados, p.ex., a descrição do circuito é essencialmente gráfica. Em todos

os níveis, no entanto, coexistem descrições gráficas e textuais, que se complementam ou são visões distintas de uma mesma função.

Descrições textuais podem ser informais (para efeito de documentação, p.ex.) ou formais, como no caso do uso de linguagens de descrição de hardware |BAR 75|. A partir destas descrições são geradas representações internas dos circuitos que serão utilizados em diferentes ferramentas de projeto, tais como simuladores e programas de síntese automática.

Descrições gráficas são geralmente criadas com a utilização de editores gráficos, orientados para a sintaxe e a semântica dos objetos descritos.

Se estas descrições textuais e gráficas são duas visões distintas de uma mesma realidade, o sistema de CAD deve permitir um projeto a partir de qualquer uma das duas. Ambas descrições devem gerar uma mesma representação interna do objeto, sendo inclusive possível obter-se uma descrição a partir da outra, num processo de extração, tal como na Fig. 1.

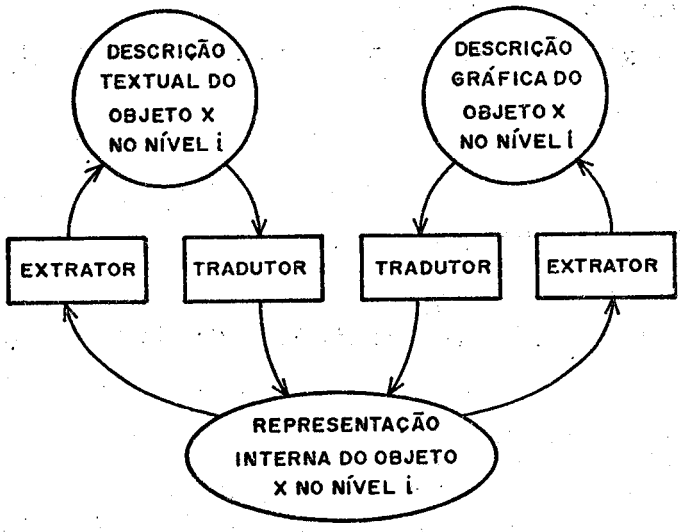

Figura 1. - Descrições textuais e gráficas.

5. DESCRIÇÕES ESTRUTURAIS E COMPORTAMENTAIS

Em qualquer nível de descrição, um objeto é descrito através de dois tipos de informações:

- informação estrutural, i.e., quais são os seus componentes, que entradas e saídas eles possuem, e como estão interligados;

- informação comportamental, i.e., qual é a função associada a cada um destes componentes.

Esta função é descrita através de operações e operandos primitivos característicos do nível de projeto em questão, que se refletem numa linguagem de descrição de hardware especializada para este nível.

Já a informação estrutural pode aparecer na descrição sob uma de duas formas. Ela pode, de um lado, estar embutida na própria linguagem de descrição comportamental antes mencionada, seja implícita ou explicitamente, no caso da linguagem permitir descrições modulares. Um exemplo é a construção "unit" na linguagem KARL |HAR 77|. Por outro lado, alguns sistemas de CAD prevêm uma linguagem específica para a descrição da informação estrutural, como p.ex. a linguagem SL-1 |GEP 75| do sistema SARA. Um outro exemplo é o modelo de redes de agências |WAG 84|, que também separa formalmente as descrições estruturais e comportamentais (ver item 7).

Em qualquer destes casos, a integração das ferramentas de projeto num mesmo ambiente de CAD deverá permitir que informações estruturais, que são comuns a diversos níveis de descrição, sejam armazenadas de forma não redundante. As informações estruturais geradas num certo nível devem poder ser utilizadas automaticamente durante a síntese dos níveis inferiores a este. Sob este aspecto, a utilização de uma linguagem ou modelo específico para a infor-

mação estrutural e independente da informação comportamental torna mais simples a integração das ferramentas de projeto.

6. PROCESSO DE SÍNTESE

A síntese de um circuito em qualquer nível, num ambiente integrado de CAD, é feita gerando-se sua descrição, numa linguagem, modelo ou representação interna adequada ao nível, através de um de dois métodos:

- interativamente, a partir da descrição no nível imediatamente superior, utilizando-se um editor (gráfico ou de textos, conforme o tipo de descrição desejada);

- automaticamente, a partir da descrição em algum nível superior, utilizando-se programas de síntese automática |PAR 84|, que são construídos baseados em técnicas heurísticas ou na forma de sistemas especialistas.

No caso de síntese interativa, a descrição resultante deverá necessariamente ser validada por um processo de simulação. Esta simulação poderá ser eventualmente dispensável no caso de síntese automática, embora tal não seja possível em muitos casos, devido à não garantia de correção total dos produtos gerados por programas de síntese.

6.1 Síntese interativa

As figuras 2 e 3 mostram os processos típicos de síntese interativa, respectivamente para os casos de interfaces gráficas ou exclusivamente textuais entre o projetista e o sistema de CAD. As figuras contemplam essencialmente os níveis de descrição considerados na versão inicial do modelo de dados a ser definido.

A síntese através de editores gráficos (figura 2)

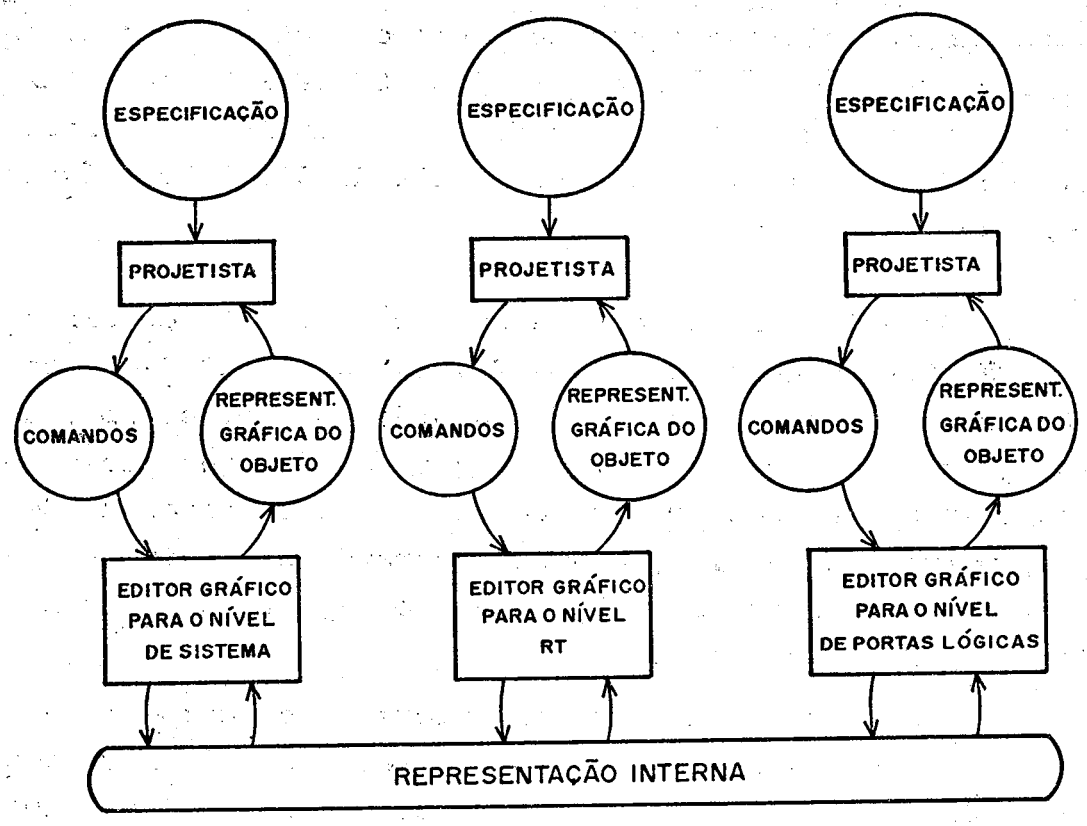

Figura 2.- Processo de síntese interativa através de editores gráficos.

ilustra melhor a integração das ferramentas de projeto no ambiente. Um editor gráfico especializado para o nível de portas lógicas, p.ex., conduzirá o processo de síntese de um dado sistema a partir de informações já existentes a respeito deste sistema, e geradas na etapa anterior do projeto, i.e., na síntese do sistema no nível RT. Estas informações geradas numa etapa do projeto e utilizáveis em etapas subseqüentes são de natureza essencialmente estrutural, facilmente manipuláveis por um editor gráfico. É possível em tal processo de síntese garantir-se de forma automática a consistência entre as informações estruturais de distintos níveis de representação.

Esta integração dos dados e conseqüente consistência não é viável apenas numa abordagem "top-down" de síntese, tal como descrita acima. Numa abordagem "bottom-up", um editor gráfico conduzirá a síntese através da interconexão de componentes básicos gerados nos níveis inferiores de descrição, permitindo a ligação apenas através dos sinais de E/S definidos para estes componentes, e garantindo assim a consistência das descrições.

A consistência entre as informações comportamentais associadas aos componentes nos diferentes níveis de descrição é bem mais difícil de se garantir e constitui uma área de pesquisa totalmente em aberto. No contexto deste ambiente integrado de CAD a equivalência entre diferentes descrições de um mesmo sistema digital poderá ser comprovada apenas por via indireta, através de validação por simulação destas descrições.

Na síntese interativa através da geração de descrições textuais em linguagens de descrição de hardware (figura 3), a integração dos dados correspondentes a diferentes níveis já não é tão simples de ser obtida. Esta integração, e portanto alguma forma de consistência automática entre os níveis, será facilitada se as descrições separarem de forma

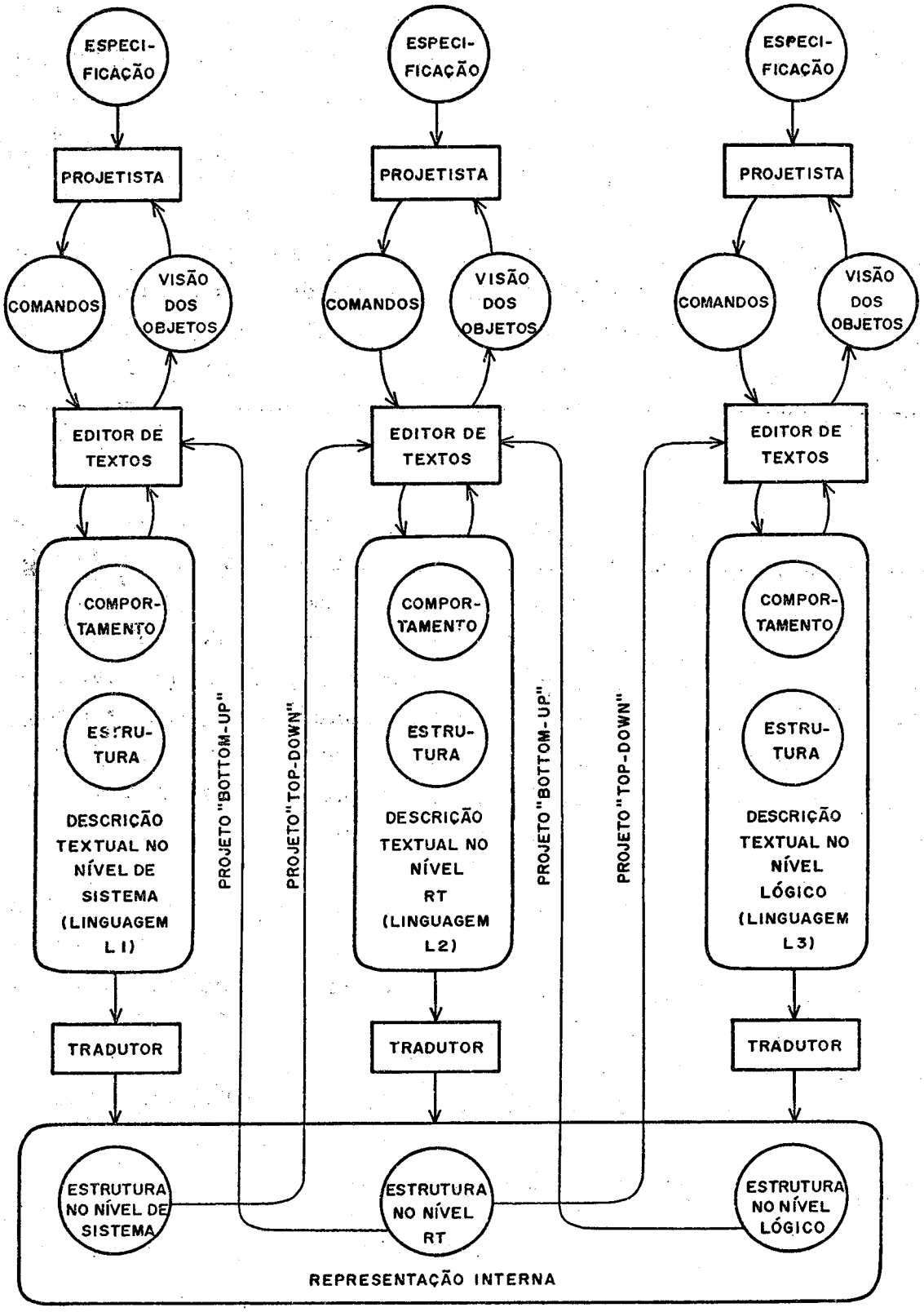

Figura 3. - Processo de síntese interativa através de linguagens de descrição de Hardware.

explícita as informações estruturais e comportamentais, tal como sugerido no item 5. Esta separação explícita, que no caso de síntese através de editores gráficos é um imperativo, será também assumida na síntese através de linguagens de descrição de hardware. Uma modelagem explícita da estrutura modular do sistema se refletirá na descrição textual, através de uma linguagem específica. As linguagens L_1, L_2 e L_3 mencionadas na figura 3 terão assim duas componentes: uma linguagem estrutural e uma comportamental. A linguagem estrutural poderá ser uniforme para todos os níveis de descrição, enquanto uma linguagem comportamental diversa será associada a cada nível.

A figura 3 mostra uma estrutura rígida na qual se associa necessariamente uma linguagem a um nível. Cada nível será definido no modelo de dados como um conjunto de objetos e operações. Esta definição procurará estabelecer um conjunto que seja o mais típico possível para cada nível. A representação interna, na figura 3, contém descrições estruturais e comportamentais. Mostra-se, explicitamente, apenas a descrição estrutural de cada nível porque a passagem de um nível para outro se dá através desta descrição.

Muitas linguagens de descrição de hardware pretendem descrever uma realidade que é, no entanto, mais ampla do que a definição restrita a ser dada a estes níveis. Um exemplo é a construção "process", típica do nível de sistema, existente na linguagem ISPS |BAR 81|, pensada primeiramente para o nível RT. Outro exemplo é a construção "delay", típica do nível lógico, encontrada na linguagem KARL |HAR 77|, também pensada primeiramente para o nível RT.

O ambiente de CAD aqui descrito não faz restrições à inclusão de ferramentas de projeto baseadas em linguagens que não se encaixam perfeitamente nos níveis definidos. A definição dos níveis condiciona o modelo de dados e, portanto, a representação interna dos sistemas, mas não im-

pede que o tradutor de uma determinada linguagem gere informações que irão preencher diferentes níveis da representação interna adotada.

A abordagem aqui definida uniformiza o processo de síntese interativa, pois a separação explícita entre estrutura e comportamento aparece tanto na síntese via editores gráficos como na síntese através de linguagens de descrição de hardware. A homogeneidade na visão que o projetista tem das diferentes ferramentas de projeto é um dos requisitos a serem preenchidos num ambiente integrado de CAD (ver item 2) . Além disto esta uniformidade simplifica a definição e implementação do banco de dados que suportará o ambiente. Por último, ela força descrições modulares dos sistemas, contribuindo para a qualidade dos projetos.

6.2 Simulação

O processo de síntese interativa, seja ela gráfica ou textual, supõe uma validação do projeto via simulação. O ambiente de CAD suportará uma série de simuladores especializados para cada nível de descrição, tal como visto na figura 4.

Esta figura mostra a visão que o usuário tem das ferramentas de simulação. Cada simulador (para o nível de sistema, para o nível RT e para o nível lógico) opera sobre a representação interna do sistema no nível para o qual ele é especializado, não enxergando as informações referentes aos demais níveis. Na implementação do ambiente, estas representações internas serão visões diferentes do modelo de dados que abrange todos os níveis de descrição.

O ambiente poderá suportar adicionalmente um simulador multi-nível. Tal tipo de ferramenta tem sido preconi-

zado (p.ex. |RAM 80|) como forma de se diminuir a complexidade da tarefa de simulação detalhada de sistemas de grande porte, permitindo que ela seja feita considerando-se diferentes módulos do sistema descritos em diferentes níveis. Sem que se entre aqui em considerações a respeito da coexistência de algoritmos de simulação diversos dentro de um único simulador, esta ferramenta é viável no ambiente aqui proposto em função da separação explícita entre estrutura e comportamento, da natureza necessariamente modular das descrições geradas e da existência de um modelo de dados que leva em conta esta modularidade e as representações em níveis diversos de descrição.

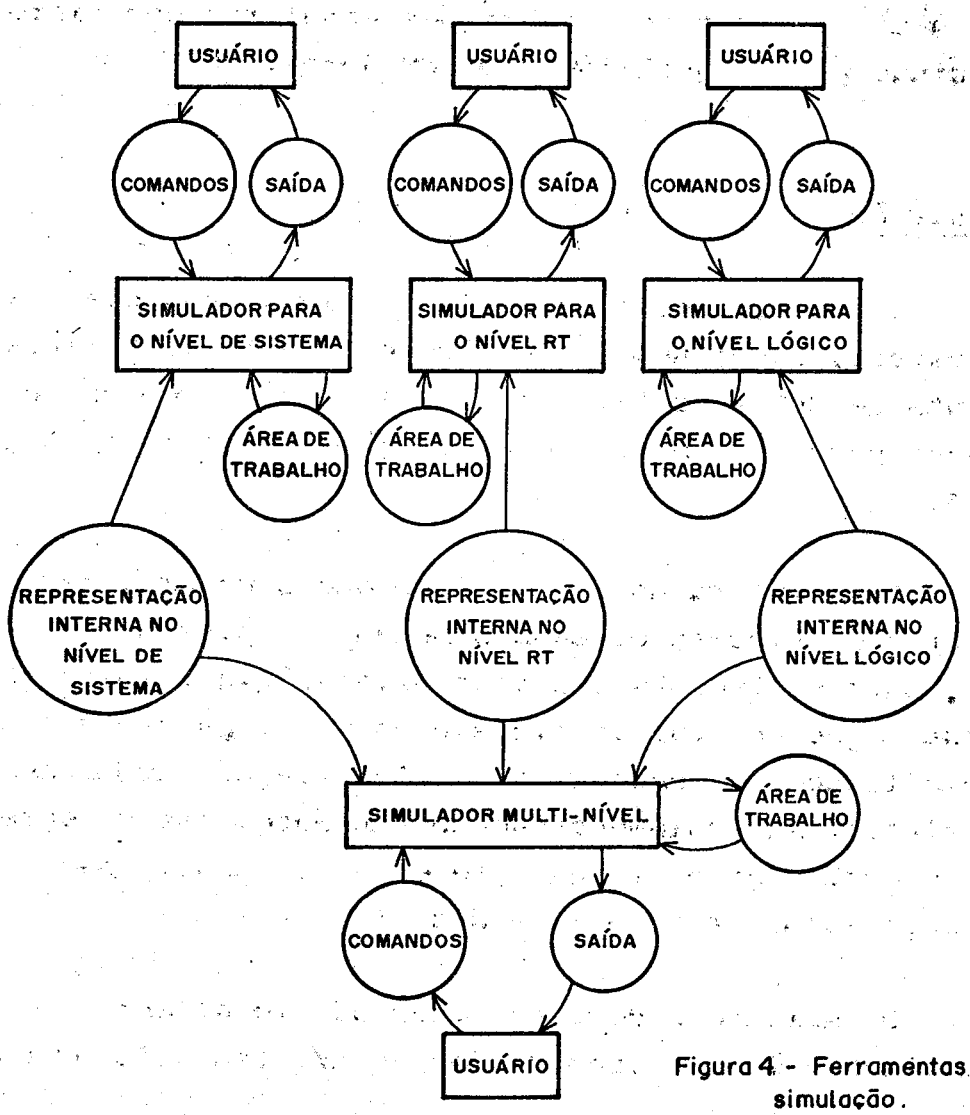

Figura 4 - Ferramentas de simulação.

Cada simulador tem a si associada uma área de trabalho onde são armazenadas informações dinâmicas a respeito do sistema, próprias do processo de simulação (listas de avaliação de elementos, listas de eventos, etc.). Um tratamento uniforme das áreas de trabalho dos diferentes simuladores será estudado, o que será especialmente útil para a simulação multi-nível.

6.3 Síntese automática

Programas de síntese automática têm sido preconizados nos últimos anos especialmente dentro do contexto de concepção de circuitos integrados |REI 84|. Genericamente falando, a síntese automática implica na geração por um programa de uma descrição do sistema digital num nível i, a partir de uma descrição num nível superior j. No contexto de circuitos integrados, três tipos de programas de síntese automática merecem referência especial:

- geração de uma descrição estrutural no nível RT a partir de uma descrição comportamental (eventualmente no próprio nível RT |COS 84|);
- geração de uma descrição no nível lógico a partir de uma descrição no nível RT; e
- geração de uma descrição no nível do "lay-out" das máscaras de fabricação dos circuitos integrados a partir de uma descrição no nível RT.

Em todos estes três casos, há uma divisão explícita do sistema digital em duas partes: o bloco operacional e o bloco de controle. No primeiro tipo de programa de síntese automática mencionado, o resultado da síntese é uma especificação separada para cada um destes blocos. Nos outros dois casos, ferramentas diversas são previstas para a síntese de cada bloco , sendo cada ferramenta apropriada para uma

dada organização-alvo para o bloco (p. ex. síntese do bloco de controle com PLA's, ou com ROM's, ou microprogramada, ou com lógica aleatória).

O modelo de dados a ser definido para o ambiente de CAD aqui descrito levará em consideração esta separação entre o bloco de controle e o bloco operacional no nível RT e nos níveis abaixo deste, o que será visível para algumas ferramentas de projeto, tais como os programas de síntese automática (ver figura 5). Para outras ferramentas esta separação poderá ser transparente.

Como a figura mostra, se a saída de um programa de síntese automática for uma descrição do circuito no nível do "lay-out" das máscaras de fabricação, esta descrição resultante não estará em princípio integrada na base de dados que suporta o ambiente de CAD, já que este nível não está previsto na versão inicial do sistema.

7. REDES DE AGÊNCIAS

O ambiente de CAD suportará ferramentas de projeto, essencialmente editores gráficos e simuladores, que utilizarão explicitamente o modelo de rede de agência [WAG 84]. Tal modelo, essencialmente estrutural, permite a representação de sistemas digitais em todos os níveis a serem considerados no modelo de dados a ser definido para este ambiente. É possível realizar-se então o projeto completo de um circuito digital, desde o nível de sistema até o nível lógico, utilizando-se sempre o mesmo modelo, desde que se associe às agências em cada nível uma descrição comportamental em alguma linguagem de descrição de hardware. Serão desenvolvidos tanto um simulador genérico de redes de agências |WAG 85| como simuladores baseados neste modelo, mas otimizados para cada nível de des

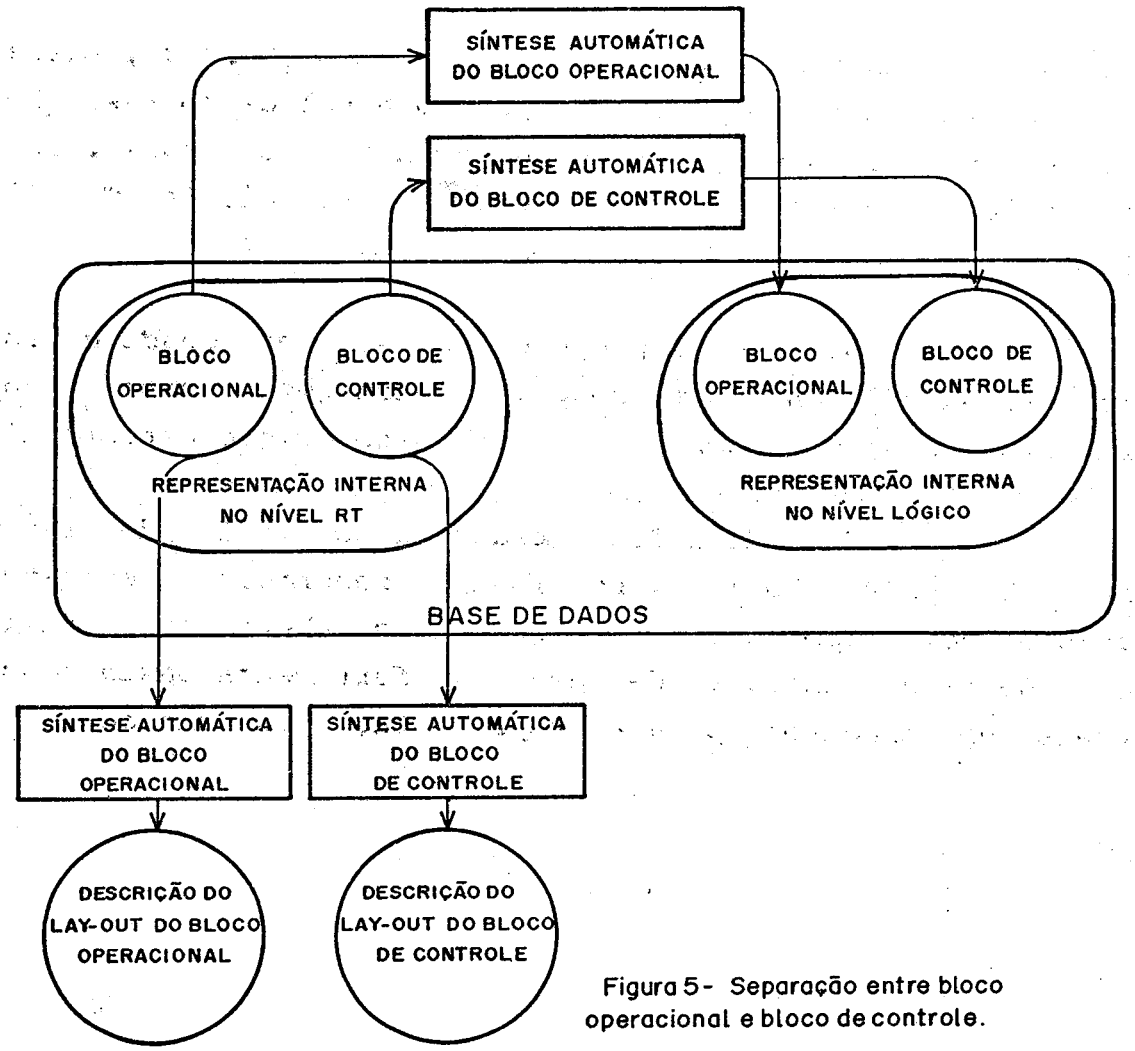

Figura 5- Separação entre bloco operacional e bloco de controle.

crição.

O uso destas ferramentas será uma alternativa de projeto em relação ao uso de editores gráficos, linguagens e simuladores específicos para cada nível.

8. CONCLUSÕES

Foram comentados todos os aspectos do processo de projeto cujas ferramentas estão sendo definidas e serão implementadas no ambiente integrado de CAD de sistemas digitais do CPGCC. As ferramentas planejadas para esta fase são: um editor gráfico para redes de agências e canais, um

editor gráfico para uma linguagem RT (KARL |HAR 77|, possivelmente), um editor gráfico para o nível de portas lógicas, um simulador de redes de agências e canais, tradutores e simuladores para linguagens nos níveis RT e lógico (p.ex. KARL e EPILOG) e um simulador lógico.

O processo de projeto pode ser empregado de forma "top-down" on "bottom-up". A passagem de um nível para o outro se dá sempre a partir da descrição estrutural.

O modelo de dados abrangerá os níveis de sistema, RT e portas lógicas, permitindo descrições estruturais e comportamentais. O sistema é aberto na medida em que para a inclusão de uma nova ferramenta basta construir um tradutor para a representação interna adotada.

Agradecemos a Lúcio Fernando Ribeiro Machado e Flávio Moreira de Oliveira, que participaram de muitas discussões nas quais se definiram as bases do processo de projeto descrito neste trabalho.

BIBLIOGRAFIA

[BAR 75] BARBACCI, M. A Comparison of Register Transfer Languages for Describing Computers and Digital Systems. IEEE Transaction on Computers. C-24(2): 137-150, Feb. 1975.

[BAR 81] BARBACCI, M.R. Instruction Set Processor Specifications (ISPS): The Notation and its Applications. IEEE Transactions on Computers, C-30(1): 24-40, jan. 1980.

[COS 84] COSTA, A.C.R. Caracterização dos Conhecimentos e da Arquitetura de um Sistema Especialista em Projeto Lógico de Circuitos Digitais. Porto Alegre, CPGCC/UFRGS, 1984. (Relatório Técnico nº 007/84)

[DAT 83] DATE, C.J. An Introduction to Database Systems - Volume II. Reading, Addison-Wesley 1983.

[ENC 83] ENCARNAÇÃO, J. & SCHLECHTENDAHL, E.G. Computer Aided Design - Fundamentals and System Architectures. Berlin, Springer-Verlag, 1983.

[GEP 75] GARDNER, R., G. ESTRIN e H. POTASH. A Structural Modeling Language for Architecture of Computer Systems. Proceedings of the International Symposium on Computer Hardware Description Languages and their Applications. New York, September 1975. p. 161-171.

[HAR 77] HARTENSTEIN, R. Fundamentals of Structured Hardware Design. Amsterdam, North-Holland, 1977.

[HAY 78] HAYES, J.P. Computer Architeture and Organization. New York, McGraw-Hill, Inc., 1978.

[MDO 75] Mac DOUGALL, M.H. System Level Simulation. In: BREUER, M. (ed). Digital System Design Automation: Languages, Simulation and Data Base. Pitman Publ. Ltd., 1975.

[MER 85] MERMET, J. Several Steps towards a Circuits Integrated CAD System: CASCADE. Grenoble, IMAG-ARTEMIS. Rapport de Recherche nº 515. Março 1985.

[PAR 84] PARKER, A. Automated Synthesis of Digital Systems. IEEE Design & Test, 1(4): 75-81, nov. 1984.

[RAM 80] RAMMIG, F.J. Entwurf, Beschreibung und Implementation von Systemen mit Hilfe der nebenläufigen Programmiersprache CAP. In: G. Zimmermann (ed). GI-NTG Fachtagung: Struktur und Betrieb von Rechensystemen, Berlin, Springer-Verlag, 1980. p.205-216

[REI 84] REIS, R.A.L., A.A. SUZIM e E. ZYSMAN. Nova Etapa na Concepção de Circuitos Integrados: Compiladores de Silício. Simpósio Brasileiro de Microeletrônica, IV, 1984.

[SBN 83] SIEWIOREK, D., C.G. BELL e A. NEWELL. Computer Structures: Principles and Examples. New York, McGraw-Hill, Inc. 1983.

[WAG 84] WAGNER, F.R. Modelamento de Processos Digitais com Redes de Instâncias. Porto Alegre, CPGCC-UFRGS, 1984. (Relatório Técnico nº 006/84)

[WAG 85] WAGNER, F.R. Simulação de Sistemas Modelados como Redes de Agências. Porto Alegre, CPGCC-UFRGS, 1985. (Relatório Técnico, em elaboração)

RELATÓRIOS TÉCNICOS MAIS RECENTES

RT-005: PRONDZYNSKI, P. r.
Arquiteturas para videos graficos com varredura fixa.
Porto alegre, CPGCC/UFRGS, nov/83.

RT-006: WAGNER, F. R.
Modelamento de processos digitais com redes de instanci
Porto Alegre, CPGCC/UFRGS, mar/84.

RT-007: COSTA, A. C. R.
Caracterizacao dos conhecimentos e da arquitetura de um sistema especialista em projeto logico de circuitos digitais.
Porto Alegre, CPGCC/UFRGS, jun/84.

RT-008: FREITAS, C. M. D. S.
Programacao grafica interativa com o PGE/UFRGS.
Porto Alegre, CPGCC/UFRGS, jun/84.

RT-009: FREITAS, C. M. D. S.
Descricao do pacote grafico PGE/UFRGS.
Porto Alegre, CPGCC/UFRGS, jul/84.

RT-010: SAYAO, M. & TOSCANI, S. S.
Sistema multiprogramavel HP2100S - manual de referencia.
Porto Alegre, CPGCC/UFRGS, out/84.

RT-011: SAYAO, M. & TOSCANI, S. S.
Sistema multiprogramavel HP2100S - manual de usuario.
Porto Alegre, CPGCC/UFRGS, out/84.

RT-012: WAGNER, F. R.
Basic techniques for gate level simulation - a tutorial.
Porto Alegre, CPGCC/UFRGS, out/84.

RT-013: WAGNER, F. R.
Hazard detection in logic simulation.
Porto Alegre, CPGCC/UFRGS, nov/84.

RT-014: COSTA, A. C. R.
Clause machines.
Porto Alegre, CPGCC/UFRGS, jan/85.

RT-015: COSTA, A. C. R.
Especificacao das tarefas do sistema especialista em projeto logico de circuitos digitais.
Porto Alegre, CPGCC/UFRGS, jan/85.

RT-016: TOSCANI, L. V. & outros.
Laboratorio de Matematica Computacional - manual do usuario.
Porto Alegre, CPGCC/UFRGS, dez/84.

RT-017: TOSCANI, L. V. & outros.
Laboratorio de Matematica Computacional - manual de programas.
Porto Alegre, CPGCC/UFRGS, dez/84.

RT-018: COSTA, A. C. R.
Introducao aos sistemas especialistas e descricao informal do sistema muPROSPECTOR.
Porto Alegre, CPGCC/UFRGS, mai/85.

RT-019: FRIEDRICH, L. F. & COSTA, A. C. R.
Descricao da implementacao do montador MC68000 escrito em Pascal Sequencial.
Porto Alegre, CPGCC/UFRGS, jun/85.

RT-020: COSTA, A. C. R.
Processando linguagens naturais em PROLOG - Parte 1: Formalismo gramatical basico.
Porto Alegre, CPGCC/UFRGS, jul/85.

RT-021: WAGNER, F. R.
Algoritmos de simulacao de hardware no nivel RT.
Porto Alegre, CPGCC/UFRGS, set/85.

RT-022: WAGNER, F. R.
On the properties of event oriented logic simulation according to significant timing models.
Porto Alegre, CPGCC/UFRGS, set/85.

RT-023: TAZZA, M.
Performance evaluation using a net theory based model.
Porto Alegre, CPGCC/UFRGS, set/85.