
Fluxo de Projecto para Circuitos Mistos

Paulo Flores

PRAXIS XXI - EuroLasic

Actividade 1.1 - Relatório D1.1

INESC

Instituto de Engenharia de Sistemas e Computadores

Índice

1	Introdução	2
2	Fluxo de projecto	2
3	Ferramentas de projecto	5
3.1	Composer	6
3.2	VHDL System Simulator	7
3.3	Design Compiler	7
3.4	Verilog-XL	8
3.5	Cell Ensemble	8
3.6	Diva	9
3.7	Test Compiler	9
4	Extensão para projecto digital/analógico	10
5	Conclusões	11

1 Introdução

Este relatório descreve o ambiente de projecto existente no INESC e a metodologia de projecto a adoptar para o projecto de circuito mistos digitais/analógicos. Esta, tem com base a metodologia usualmente utilizada para o projecto de circuitos digitais sofrendo algumas alterações no fase final de *back-end*, para comportar a inclusão de circuitos analógicos.

Começa-se por apresentar a metodologia que usualmente é adoptada para o projecto de circuitos digitais na secção 2. Uma breve descrição das ferramentas utilizadas encontra-se na secção 3. Algumas considerações a ter na extensão do fluxo de projecto para circuitos mistos é apresentada na secção 4. Finalmente, as conclusões são apresentadas na última secção.

2 Fluxo de projecto

A figura 1 representa uma versão simplificada da metodologia de projecto de circuitos digitais. Esta é geralmente dividida em duas partes, o *front-end* e o *back-end*, que na figura 1 se representam, respectivamente, no lado esquerdo e no lado direito.

O projecto de um circuito digital começa usualmente por uma descrição de alto nível em VHDL. Através de simulação esta descrição é validada e considerada apta para ser sintetizada.

O objectivo da etapa de síntese é produzir um circuito que implementa a funcionalidade apresentada pela descrição VHDL, satisfazendo simultaneamente um conjunto de restrições de área e de atrasos máximos. Este processo é iterativo para serem exploradas diferentes possibilidades para o circuito, cabendo ao projectista seleccionar a solução com o desempenho desejável.

A etapa de síntese lógica pode também ser utilizado para realizar o remapeamento de um circuito numa outra tecnologia. Nestes casos a entrada desta etapa é um esquemático de um circuito numa dada tecnologia. Na

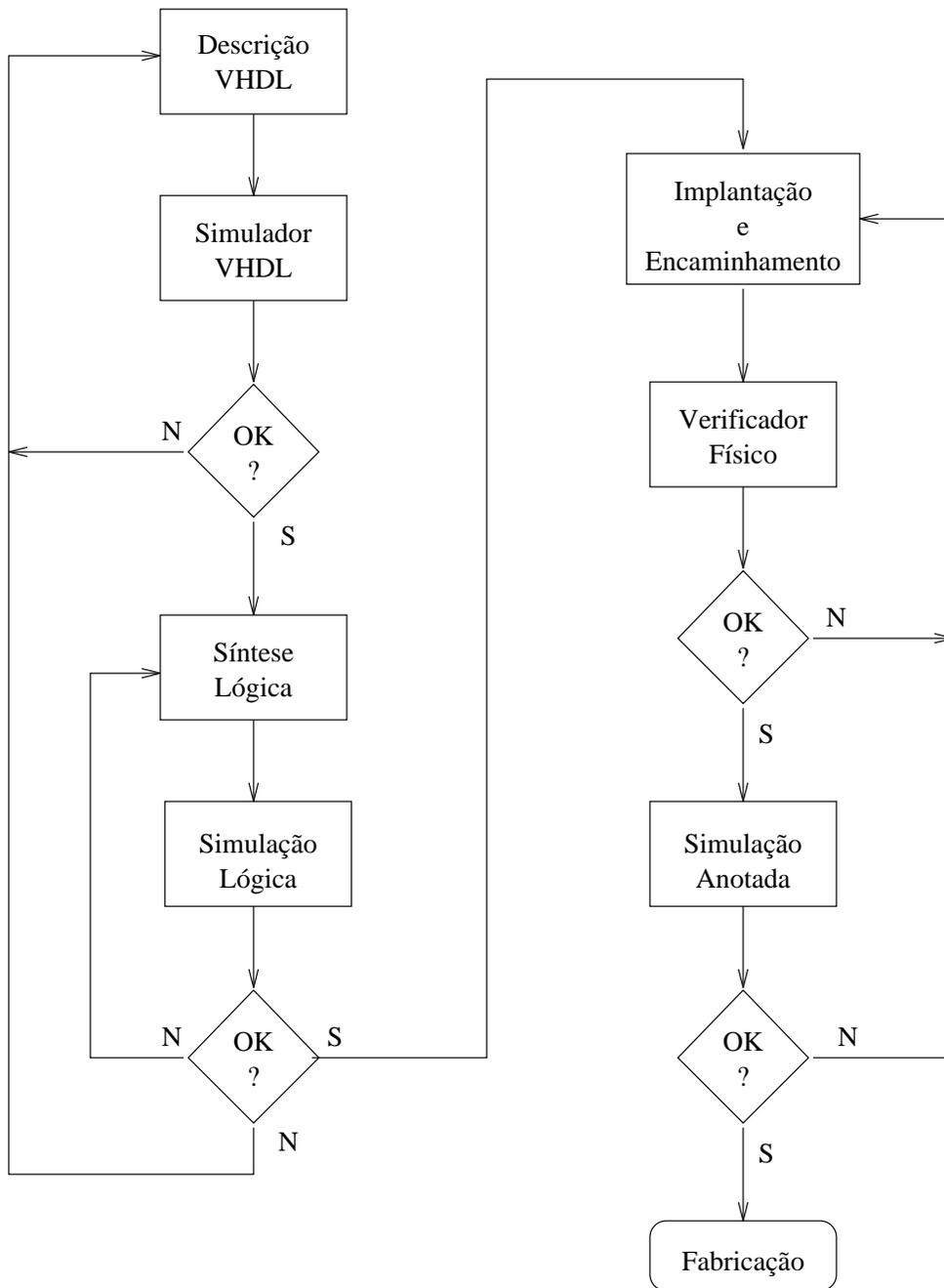


Figura 1: Fluxo de projecto simplificado

saída obtém-se um outro esquemático, do mesmo circuito, mas mapeado na nova tecnologia.

O circuito sintetizado tem que ser simulado ao nível lógico para “verificar” a qualidade do processo de síntese. Esta simulação é realizada utilizando uma biblioteca de células com características específicas da tecnologia, podendo por isso, ser usada para determinar os atrasos nos caminhos críticos do circuito. Note-se que simulação VHDL não requer bibliotecas dependentes da tecnologia, uma vez que a descrição é feita a nível de transferência de registos, e portanto, independente da tecnologia.

Após realizada simulação lógica a próxima etapa é fazer a colocação e encaminhamento do circuito (*Place and Route - P&R*). Esta operação pode ser realizada tendo em conta os tempos de atraso extraídos durante o processo de síntese. Assim, é possível assegurar que os tempos de atraso nos caminhos críticos do circuito não são afectados.

Depois de realizada a colocação e encaminhamento, a implantação do circuito (*layout*) tem que ser verificada perante as regras de desenho da tecnologia, usando-se para isso o verificador de regras de desenho (*Design Rule Check - DRC*). Uma lista de portas deve também ser extraída da implantação para ser confrontada com a lista de portas que foi sintetizada (*Layout Versus Schematic - LVS*).

Finalmente, as capacidades parasitas extraídas da implantação são usadas para criar um ficheiro com anotações temporais que permite realizar uma simulação lógica após a implantação com mais rigor. Esta simulação tem como objectivo verificar se as restrições temporais impostas são observadas na implantação.

No fluxo apresentado na figura 1 destacam-se ainda as seguintes possibilidades de projecto:

- Enquanto a descrição VHDL não apresentar os requerimentos de funcionalidade desejados, deve de ser re-escrita.
- Se a simulação lógica, depois da etapa de síntese, não satisfizer as especificações para o circuito, este pode ser re-sintetizado. Em certos

casos pode mesmo ser necessário alterar a descrição VHDL original. Esta situação pode ser evitada utilizando um estilo de codificação do VHDL orientada para a síntese.

- A etapa de verificação da implantação não deve em principio detectar quaisquer problemas, pelo que neste passo não há necessidade voltar atrás.
- A simulação lógica final, com a anotação da capacidades extraídas, é utilizada para verificar com maior precisão se os tempos de atraso envolvidos no circuito satisfazem as especificações. Caso estas não seja verificadas torna-se necessário repetir a etapa de colocação e encaminhamento, ou mesmo voltar a repetir outras etapas anteriores do projecto. Esta repetição pode ser evitada se for estimado um valor conservador para as capacidades do circuito na simulação lógica após a síntese.

3 Ferramentas de projecto

O ambiente de projecto do INESC é baseado em ferramenta de CAD de elevado desempenho a correrem em *worksations*. O conjunto de ferramentas seleccionadas realizam ambas as tarefas de *front-end* e *back-end*.

O conjunto de ferramentas existentes incluiu: o *Synopsys VHDL System*, para a simulação e síntese de descrições em VHDL; o *Synopsys Test Compiler*, para a inserção de esquemas de teste e geração automática de padrões de teste; o *Cadence Verilog-XL*, para realizar a simulação ao nível lógico; o *Cadence Cell Ensemble*, para a colocação e encaminhamento; e o *Cadence Diva*, para a verificação física do circuito.

A tabela 1 sumariza para cada etapa de projecto as ferramentas utilizadas. Uma breve descrição de cada uma das ferramentas é apresentada nas secções seguintes.

Etapa de Projecto	Ferramenta	Vendedor
Descrição do Circuito	DFII - Composer	Cadence
Simulação Comportamental	VHDL System Simulator	Synopsys
Síntese	Desing Compiler	Synopsys
Simulação Lógica	Verilog-XL	Cadence
Colocação e Encaminhamento	Gate Ensemble	Cadence
Verificação Física	Diva	Cadence
Simulação Anotada	Verilog-XL	Synopsys
Geração de Teste	Test Compiler	Synopsys

Tabela 1: Ferramentas de projecto

3.1 Composer

O editor de esquemáticos integrado no ambiente de projecto da Cadence, *Composer*, foi seleccionado para a etapa de descrição do circuito. A utilização deste editor permite descrever o circuito utilizando simultaneamente vários níveis de representação. É possível descrever simultaneamente blocos em VHDL comportamental ou a nível de transferência de registos e blocos no nível lógico utilizando portas lógicas de uma biblioteca genérica [Composer 94a][Composer 94b].

O editor de esquemáticos Composer permita que a descrição do circuito seja feita de forma hierárquica e descrita a vários níveis de abstracção. Assim, cada bloco do circuito pode ser representado através de um esquemático ou por uma descrição textual em VHDL ou Verilog. Devido à forte integração do Composer no DFII é possível utilizar todas as potencialidades para a gestão dos dados do projecto oferecidas pelo ambiente de projecto da Cadence. Em particular, com utilização da linguagem de extensão, *Skill*, pode-se aceder à base de dados do projecto onde toda a informação é guardada.

3.2 VHDL System Simulator

A simulação funcional da descrição VHDL é feita utilizando o *VHDL System Simulator* (VSS) da Synopsys. Esta simulação é realizada para verificar se a descrição VHDL do circuito satisfaz as especificações originais [VSS 94].

O simulador VSS está de acordo com a norma IEEE-1076 de VHDL e encontra-se integrado no ambiente de síntese da Synopsys. Os resultados de simulação podem ser visualizados directamente na representação esquemática de alto nível do circuito ou num visualizador de formas de onda. Este simulador apresenta também um interface gráfico que auxilia a detecção de erros. As capacidades de executar o código VHDL passo a passo, observar sinais e colocar pontos de paragem, simplificam grandemente a tarefa de detecção e correcção de erros da funcionalidade do circuito.

3.3 Design Compiler

As etapas de síntese lógica e o mapeamento tecnológico são realizadas pela ferramenta de síntese *Desing Compiler* (DC) da Synopsys. Esta ferramenta aceita uma descrição hierárquica do circuito, quer numa linguagem de descrição de hardware, como VHDL ou Verilog, quer no formato EDIF. Uma descrição mista utilizando os vários formatos anteriores é também suportada. Apesar de, nem toda a linguagem VHDL ser suportada, a Synopsys afirma que o Desing Compiler é a ferramenta comercial de síntese que sintetiza o maior subconjunto da linguagem VHDL [DC 94].

O Desing Compiler é uma ferramenta interactiva que optimiza o circuito baseados nas especificação fornecidas pelo utilizador/projectista. A existência de um analisador temporal incorporado na própria ferramenta permite impor restrições temporais complexas e analisar eficazmente o comportamento temporal do circuito sintetizado.

3.4 Verilog-XL

O passo de simulação lógica é realizada utilizando o simulador *Verilog-XL* da Cadence. Este passo tem como objectivo verificar que o circuito está de acordo com as especificações tendo em conta os atrasos específicos da tecnologia no qual o circuito foi mapeado. O Simulador Verilog-XL aceita descrições em Verilog desde do nível comportamental até ao nível interruptor, mas no fluxo de projecto descrito apenas realizará simulações ao nível lógico. A principal vantagem do simulador Verilog-XL é o seu elevado desempenho para circuitos de grande dimensão e a existência de muitas bibliotecas de modelos certificadas pelas grandes fábricas de circuitos integrados [Verilog 94].

A capacidade de ler um ficheiro com anotações temporais em SDF (*Standard Delay Format*) faz com que este simulador seja também utilizados para a simulação do circuito após a implantação. Assim, o circuito pode ser simulado com atrasos mais realistas resultantes da extracção do valor das capacidades parasitas existentes na implantação [SDF 94].

3.5 Cell Ensemble

A ferramenta usada para a colocação e encaminhamento é o *Cell Ensemble* da Cadence. Esta ferramenta está fortemente integrado no ambiente de projecto do DFII e tem acesso directo a base de dados do projecto. A interface de utilizador para os “motores” de colocação e de encaminhamento desta ferramenta são feitas através do planeador hierárquico de colocação, *Preview* [CE 94] [Preview 93].

O Cell Ensemble é capaz de fazer a colocação e encaminhamento de circuitos com células normalizadas (*standard cells*) ou com um misto de células normalizadas e blocos desenhados pelo projectista. As tarefas realizadas pela ferramenta podem ser feitas de forma automática ou interactivamente, mas em geral o encaminhamento detalhado no interior das células e feito em modo automático. O Cell Ensemble suporta circuito com

2 ou 3 níveis de metal.

A utilização do Preview permite estimar os efeitos físicos da implantação antes de realizar a colocação e o encaminhamento do circuito. Esta potencialidade minimiza riscos de se ter de realizar várias iterações no processo de colocação e encaminhamento, o que reduz os custos e tempo de projecto de um circuito.

3.6 Diva

A etapa de verificação física do circuito é realizada pelo sistema *Diva* da Cadence. O Diva é constituído por um conjunto de ferramentas de verificação que podem ser utilizadas interactivamente para detectar e corrigir erros na implantação do circuito [Diva 94].

O verificador de regras geométricas do Diva detecta violações das restrições de desenho impostas pela tecnologia na implantação. Este verificador comporta uma grande leque de tecnologias e metodologias de implantação (desde desenho personalizado até à implantação automática). Os extractores do Diva operam em circuitos com ou sem hierarquia para produzirem uma lista de portas lógicas e um conjunto de atrasos nas interligações destas. O comparador da implantação com o esquemático do Diva verifica se o circuito em ambos os níveis estão concordantes em termos de ligações, dispositivos e parâmetros. As situações de erro podem ser indicadas e verificadas em ambos os níveis de representação do circuito. O verificador de regras eléctricas (*electrical rules checker - ERC*) do Diva verifica a conectividade da lista de portas e destaca as problemas eléctricos existentes, tais como: ligações, redes ou portas lógicas não ligadas.

3.7 Test Compiler

A inserção de estruturas de pesquisa para teste (*scan path*) e a geração automática de vectores de teste, para testar o circuito após fabricação, é realizada pelo *Test Compiler* da Synopsys. Esta capacidade está integrada

no Design Compiler, permitindo assim a optimização do circuito já com as estruturas de teste incluídas [TC 94].

O Test Compiler é uma ferramenta de síntese de teste que automatiza também o processo de projecto para a testabilidade do circuito (*Design for testability - DFT*) e a geração automática de vectores de teste (*Automatic Test Pattern Generation - ATPG*). Esta ferramenta suporta três metodologias comuns de inserção de estruturas de pesquisa: *flip-flops* multiplexados, *flip-flops* LSSD ou *latches* LSSD, e a metodologia de pesquisa de periferia (*boundary scan*) da norma IEEE 1149.1 JTAG. Um verificador de regras interno, certifica-se que o circuito satisfaz os requisitos necessários para a metodologia de teste escolhida e reporta todas as violações que possam reduzir a cobertura de faltas. Os vectores gerados automaticamente podem ser escritos no formato normalizado TSSI para serem fornecidos ao equipamento de teste.

4 Extensão para projecto digital/analógico

Para que a metodologia de projecto apresentada suporte completamente o projecto de circuitos mistos é necessário adicionar duas etapas extras ao fluxo apresentado: introdução da descrição da parte analógico do circuito e a simulação mista digital/analógica do circuito. A criação destas novas duas etapas implica a utilização de duas novas ferramentas: o editor de implantações e o simulador analógico.

O *Virtuoso Layout Editor* e o editor de implantações da Cadence. O Virtuoso concede ao projectista a capacidade de desenhar e editar polígonos, colocar células dentro de outras células para obter uma representação hierárquica, criar células parametrizáveis, e criar e editar circuitos analógicos ao mais baixo nível [Virtuoso 93].

No ambiente de projecto da Cadence a simulação mista é realizada no ambiente de simulação do *Analog Artist*. Esta ferramenta permite ao projectista realizar simulações puramente analógicas ou mistas utilizando sem-

pre o mesmo esquema de interacção com o utilizador [Artist 94].

O ambiente misto de simulação do Analog Artist usa:

- os simuladores Spectre, Cadence Spice ou HSpice para as partes analógicas do circuito;
- o simulador Verilog-XL para a partes digitais do circuito;

O utilizador controla ambos os simuladores num ambiente e interface de utilização comum. Os dois simuladores, analógico e digital, comunicam entre si de forma transparente para o utilizador, usando as facilidades existentes no UNIX para comunicação entre processos através de *sockets*. Desta forma é também possível correr cada um dos simuladores em máquinas diferentes. A figura 2, adoptada de [Artist 94], identifica a relação existente entre os diferentes programas.

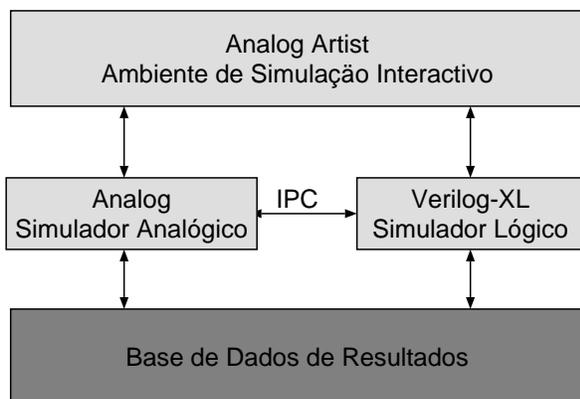


Figura 2: Relação entre os diferentes programas

5 Conclusões

Neste relatório apresentamos o fluxo de projecto para circuitos mistos, digitais/analógicos. Este fluxo é baseado em linguagens de descrição de *hardware* e em síntese lógica. As linguagens de descrição de *hardware* permitem que a descrição do circuito seja mais rápida e mais simples de documentar e alterar. A síntese lógica gera automaticamente uma lista de

portas optimizadas a partir da descrição do circuito usando uma linguagem de *hardware*. A fase de *back-end* esta ajustada para o projecto de circuitos em células normalizadas (*standard cells*).

As ferramentas de projecto seleccionadas para o ambiente de projecto do INESC foram também brevemente descritas. Este conjunto de ferramentas foi escolhido de acordo com o fluxo de projecto apresentado e formam um ambiente de projecto de elevado desempenho a correr em *worksations*. Todas as ferramenta que fazem parte do ambiente de projecto estão integradas no DFII, tirando por isso vantagens das facilidades fornecidas pelo infra-estrutura para a gestão dos dados e do fluxo de projecto.

O uso de formatos e linguagens normalizadas para representar o circuito facilita a troca de informação entra as várias ferramentas e contribuí para um ambiente de projecto aberto a outras ferramentas.

Para o projecto de circuitos mistos trona-se necessário alterar o fluxo de projecto tradicional usado para circuito digitais, de forma a comportar o projecto dos componentes analógicos do circuito. Foi explicado como suportar o projecto de circuitos mistos realizando uma extensão à metodologia existente. Todas as ferramentas mantêm a função que desempenhavam mas novas ferramentas são utilizadas para fornecer a funcionalidade adicional necessária ao projecto de circuitos mistos.

Referências

- [Artist 94] Cadence Design Systems. *Analog Artist Mixed-Signal Simulation Reference*, 4.3 edition, March 1994.
- [CE 94] Cadence Design Systems. *Preview Cell Ensemble Reference*, 4.3 edition, March 1994.
- [Composer 94a] Cadence Design Systems. *Composer Reference Manual*, 4.3 edition, March 1994.

- [Composer 94b] Cadence Design Systems. *Composer User Guide*, 4.3 edition, March 1994.
- [DC 94] Synopsys, Inc. *Design Compiler Family Reference*, April 1994.
- [DFII 93] Cadence Design Systems. *Design Framework User Guide*, September 1993.
- [Diva 94] Cadence Design Systems. *Diva Interactive Verification Reference Manual*, 4.3 edition, March 1994.
- [Preview 93] Cadence Design Systems. *Preview User Guide*, 4.2.2 edition, September 1993.
- [SDF 94] Cadence Design Systems. *SDF Annotator User Guide*, 1.2 edition, June 1994.
- [TC 94] Synopsys, Inc. *Test Compiler Reference*, April 1994.
- [Verilog 94] Cadence Design Systems. *Verilog-XL Reference Manual*, 2.0 edition, March 1994.
- [Virtuoso 93] Cadence Design Systems. *Virtuoso Layout Editor User Guide*, 4.2.2 edition, June 1993.
- [VSS 94] Synopsys, Inc. *VSS Expert Software Tools*, 1994.