



Autor do Resumo:

Daniel Carlos Guimarães Pedronette
RA: 050269

Artigo:

Graph-based functional test program generation for pipelined processors

P. Mishra and N. Dutt. Graph-based functional test program generation for pipelined processors. In Proceedings of Design Automation and Test in Europe (DATE), pages 182–187, 2004.

A validação e verificação funcional de processadores é reconhecidamente uma tarefa complexa e custosa, representando um dos gargalos nas metodologias de design de processadores. Paralelamente, os requisitos de desempenho para sistemas embutidos têm se tornado cada vez mais exigentes, exigindo a implementação de técnicas de *pipeline*, o que por sua vez torna ainda mais difícil a tarefa de validação.

Sob esta perspectiva, é proposta uma metodologia gráfica de geração de testes funcionais para processadores com *pipeline*. A proposta pretende contribuir, gerando de forma automática um modelo gráfico do processador, gerando os testes funcionais sobre o comportamento do *pipeline* e ainda reduzindo o tempo necessário para geração dos testes.

No primeiro estágio de aplicação da metodologia, a arquitetura do processador é descrita utilizando-se ADL (*Architecture Description Language*). A representação contém informações sobre a estrutura, o comportamento do processador e o mapeamento entre estrutura e comportamento. Estruturalmente, são definidos quatro tipos de componentes: unidades funcionais (ALUs), unidades de armazenamento (registradores), portas e conexões, onde as conexões são definidas por *pipelines* ou barramentos de dados. Sobre o comportamento, as informações abrangem descrição das operações em: *opcode*, operandos e formato da instrução.

Ainda visando a descrição da arquitetura, a abordagem proposta faz uso do conceito de abstração funcional, que consiste no fato de que diferentes arquiteturas fazem uso das mesmas unidades funcionais (uma unidade de *fetch*, por exemplo), mas com parâmetros diferentes. Ou seja, a estrutura de cada unidade funcional é capturada utilizando-se funções genéricas, mas parametrizadas adequadamente para a arquitetura em questão. Uma unidade de *fetch*, por exemplo, possui vários parâmetros, como operações lidas por ciclo, esquema de predição de *branches*, etc.

Baseada na descrição da arquitetura, a estrutura do processador é modelada num grafo $G = (V, E)$. Os vértices representam componentes do processador: unidades funcionais ou de armazenamento e as arestas representam ligações para transferência de dados: via *pipeline* ou barramentos. Cada vértice do grafo contém informações a respeito de suas entradas, saídas, operações suportadas e respectiva duração. Ainda para cada vértice (unidades estruturais) é gerada uma descrição SMV do seu comportamento através de funções genéricas, de acordo com a abordagem de abstração funcional.

A partir da definição do grafo que representa a arquitetura, torna-se necessário definir finalmente uma métrica de cobertura dos testes. A metodologia abordada propõe uma métrica baseada na cobertura funcional do *pipeline*, definindo todas as possibilidades de interação entre as instruções e os estágios do *pipeline*. Um grafo é considerado testado quando todos os seus vértices e arestas estiverem passado pelos quatro estados definidos: *active* (executando ou transferindo dados), *stalled* (devido a *hazards*), *exception* ou *flushed* (devido a *exception* de um nó pai). Assim, o algoritmo de geração de testes percorre o grafo, gerando propriedades de acordo com os estados acima citados, que por sua vez darão origem aos programas de teste.

Foi realizado um estudo de caso, aplicando a metodologia proposta ao processador DLX, num *pipeline* de 5 estágios. Considerando todos os cenários, foram gerados 223 programas de teste em 91 segundos numa máquina Sun UltraSPARC-II trabalhando a 333MHz, com 128M RAM.

A metodologia proposta reduz o tempo necessário para geração de testes e representa uma técnica bastante promissora, no que diz respeito a validação dirigida pela especificação. Atualmente, os testes gerados são aplicados a um simulador estrutural da arquitetura. Os trabalhos futuros incluem a aplicação destes testes na descrição RTL para a validação funcional de processadores com *pipeline*.