

Faculdade de Engenharia da Universidade do Porto



FEUP

Metodologias de projecto de baixo consumo para implementações em FPGA

João Miguel Ramos Meixedo

Tese submetida no Âmbito do
Mestrado Integrado em Engenharia Electrotécnica e de Computadores
Major de Telecomunicações

Orientador: Prof. Dr. António José Duarte Araújo

Julho de 2008

Resumo

Esta dissertação apresenta um trabalho experimental sobre o consumo de potência de circuitos implementados em tecnologia FPGA. Foram identificadas as causas do consumo de potência de circuitos digitais CMOS e descritas as técnicas que ao nível do projecto RTL podem reduzir o consumo de potência. A metodologia implementada para a medição do consumo de potência estática e dinâmica é descrita. Com base nesta metodologia são apresentados os resultados de ensaio de alguns circuitos, envolvendo máquinas de estados e operadores aritméticos, que serviram como casos de estudo. Com as técnicas de redução do consumo de potência mais adequadas para dispositivos FPGA, codificação de estados e utilização de arquitecturas *pipelined*, foram implementadas várias versões destes circuitos. A análise dos resultados obtidos mostrou a efectividade destas técnicas na redução do consumo de potência. A avaliação do consumo por estimação foi também efectuada para os referidos circuitos, tendo permitido aferir a precisão deste processo.

Abstract

This thesis presents an experimental work on the power consumption of circuits implemented in FPGA technology. It was identified the causes of the power consumption digital circuits CMOS and was described the techniques in the project RTL level may reduce the power consumption. The methodology implemented to measure the power consumption of static and dynamic components is described. Based on this methodology are presented the testing results of some circuits, involving finite state machines and arithmetic operators, which served as case studies. With techniques for reducing power consumption, more suitable for FPGA devices, state encoding and pipelined architectures, have been implemented several versions of these circuits. The analysis of the obtained results showed the effectiveness of these techniques in reducing the power consumption. The assessment of consumption by estimation was also made for those circuits, and determined the accuracy of this process.

Agradecimentos

Queria agradecer a todos aqueles que tornaram este trabalho possível. A toda a comunidade da Faculdade de Engenharia da Universidade do Porto, especialmente ao meu orientador, professor António Araújo e ao colega Ricardo Salgado. Quero também agradecer à minha família, sem a qual nunca teria chegado a este ponto, e ainda à minha namorada, Bárbara, por todas as palavras de incentivo.

João Miguel Ramos Meixedo

*“Não podemos dar-nos por vencidos.
Já conhecemos mil processos que não resultam.
Quer dizer que nos encontramos mais perto daquele que resultará.”*

Thomas Edison

Conteúdo

1	Introdução	1
1.1	Consumo de potência	1
1.2	FPGAs	2
1.3	Motivação	5
1.4	Trabalho Realizado	6
1.5	Estrutura da Dissertação	6
2	Consumo de potência em circuitos digitais CMOS	7
2.1	Introdução	7
2.2	Causas do consumo de potência	7
2.2.1	Potência estática	8
2.2.2	Potência de fugas	8
2.2.3	Potência de curto-circuito	9
2.2.4	Potência dinâmica	10
2.3	Redução do consumo de potência ao nível do projecto RTL	10
2.3.1	Análise estática	12
2.3.2	Arquitectura paralela	12
2.3.3	Redução de <i>glitches</i>	12
2.3.4	Arquitectura <i>pipelined</i>	13
2.3.5	Gestão dinâmica do sinal de relógio	14
2.3.6	<i>Clock gating</i>	14
2.3.7	Codificação de estados	15
2.4	Metodologias de projecto para baixo consumo em FPGAs	15
2.5	Estado da arte	16
2.6	Conclusão	17
3	Avaliação do consumo de potência	19
3.1	Introdução	19
3.2	Técnicas de avaliação do consumo de potência	19
3.2.1	Estimação do consumo de potência	20
3.2.2	Medição do consumo de potência	21
3.3	Casos de estudo	23
3.3.1	Circuito <i>md4</i>	24
3.3.2	Circuitos aritméticos	25
3.3.3	Filtros FIR	26
3.4	Conclusão	29

4	Resultados Obtidos	31
4.1	Introdução	31
4.2	Circuitos auxiliares	32
4.3	Circuito <i>md4</i>	32
4.4	Multiplicadores	34
4.5	Divisores	37
4.6	Filtros FIR	38
4.6.1	Implementação de filtros FIR	39
4.6.2	Utilização de filtros FIR numa cadeia de processamento de vídeo	42
4.6.3	Análise da avaliação do consumo de potência	43
4.7	Conclusão	44
5	Conclusão	49
A	Tutorial Xpower	53
A.1	Descrição, validação e síntese do circuito	53
A.2	Simulação <i>post-routing</i>	54
A.3	Estimação	54
	Referências	59

Lista de Figuras

1.1	Arquitectura interna de uma FPGA.	2
1.2	Bloco lógico de uma FPGA.	3
1.3	Ciclo de projecto de circuitos baseados em FPGAs.	4
2.1	Porta lógica CMOS.	9
2.2	Exemplo de uma arquitectura paralela.	12
2.3	Exemplo de uma arquitectura <i>pipelined</i>	13
3.1	Circuito para a medição da corrente consumida pelo núcleo da FPGA.	21
3.2	Visualização do consumo de corrente.	22
3.3	Circuito utilizado na medição real do consumo de potência.	23
3.4	Diagrama da implementação utilizada na medição da corrente.	24
3.5	Multiplicador em forma de <i>array</i>	26
3.6	Divisor em forma de <i>array</i>	27
3.7	Filtro FIR com n andares.	28
3.8	Cadeia de processamento de vídeo, com filtro definido para a detecção de contornos verticais.	28
4.1	Consumo de potência dinâmica dos circuitos multiplicadores.	35
4.2	Potência estática, dinâmica e total do multiplicador de 32 bits.	36
4.3	Ocupação de recursos dos circuitos multiplicadores.	36
4.4	Evolução do número de <i>flip-flops</i> livres no multiplicador de 32 bits.	37
4.5	Consumo de potência dinâmica dos circuitos divisores.	38
4.6	Ocupação de recursos dos circuitos divisores.	39
4.7	Consumo de potência dinâmica dos filtros.	40
4.8	Ocupação de recursos dos filtros expressa em CLBs.	41
4.9	Ocupação de recursos pelos filtros em portas lógicas equivalentes.	41
4.10	Consumo de potência dos filtros FIR utilizados numa cadeia de processamento de vídeo realizando a função de filtro passa-baixo.	43
4.11	Consumo de potência dos filtros FIR utilizados na cadeia de processamento de vídeo para detecção de contornos verticais.	44
4.12	Consumo de potência dinâmica do filtro FIR obtida por medição e por estimação	45
4.13	Consumo de potência estática do filtro FIR obtida através de medição e de estimação.	46
A.1	Janela de processos do ISE onde se pode iniciar a etapa <i>place & route</i>	53
A.2	Janela de processos do ISE onde se pode iniciar a etapa <i>Post-Route Simulation</i>	54
A.3	Janela de propriedades da <i>Post-Route Simulation</i>	55

A.4 Ambiente gráfico do Xpower. 56

Lista de Tabelas

4.1	Consumo de potência dinâmica e ocupação de recursos da FPGA para os circuitos auxiliares utilizados em cada um dos casos de estudo.	32
4.2	Resultados de implementação obtidos para as diferentes codificações de estados da versão inicial do circuito <i>md4</i>	33
4.3	Resultados da implementação para as diferentes codificações de estados da segunda versão do circuito <i>md4</i>	33

Capítulo 1

Introdução

1.1 Consumo de potência

O consumo de potência nos circuitos digitais tem vindo a tornar-se cada vez mais uma questão essencial quer a nível do projecto, quer a nível comercial. Apesar de ser uma questão importante nos equipamentos fixos como computadores de secretária, a economia de potência nos equipamentos portáteis tais como telefones, PDAs, leitores de MP3/MP4 e computadores portáteis torna-se imperativa [1], uma vez que a sua autonomia depende directamente do consumo de energia. A redução do consumo de potência é também desejável no sentido de viabilizar a miniaturização de equipamentos, pois a dissipação de potência pode ser um entrave importante. Com a diminuição do consumo de potência, é possível tornar estes e outros equipamentos autónomos, alimentando-os a partir de pequenas baterias, de que são exemplo as baterias de iões de lítio tornando-os mais pequenos e leves.

As vantagens do baixo consumo de potência nos equipamentos fixos são evidentes no caso dos computadores pessoais, e principalmente nos servidores e *mainframes*, devido à poupança de energia que podem representar. Nos equipamentos fixos a principal vantagem da redução do consumo de potência é a diminuição dos custos com energia, e a diminuição na complexidade e dimensão dos sistemas de alimentação (fontes de alimentação), uma vez que a corrente absorvida diminui, e nos sistemas de arrefecimento tais como ventiladores e radiadores, devido à diminuição da potência dissipada em forma de calor.

No passado, as principais preocupações ao nível do projecto eram o desempenho do circuito e a área por ele ocupado. Os objectivos de um projecto eram tornar o circuito tão rápido quanto possível, de modo a que este fizesse mais operações em menos tempo, e à custa da menor área ocupada, pois quanto mais pequeno fosse o circuito melhor, pois assim o seu custo de produção seria mais reduzido com a vantagem adicional da miniaturização assim conseguida. O consumo de energia era geralmente um requisito

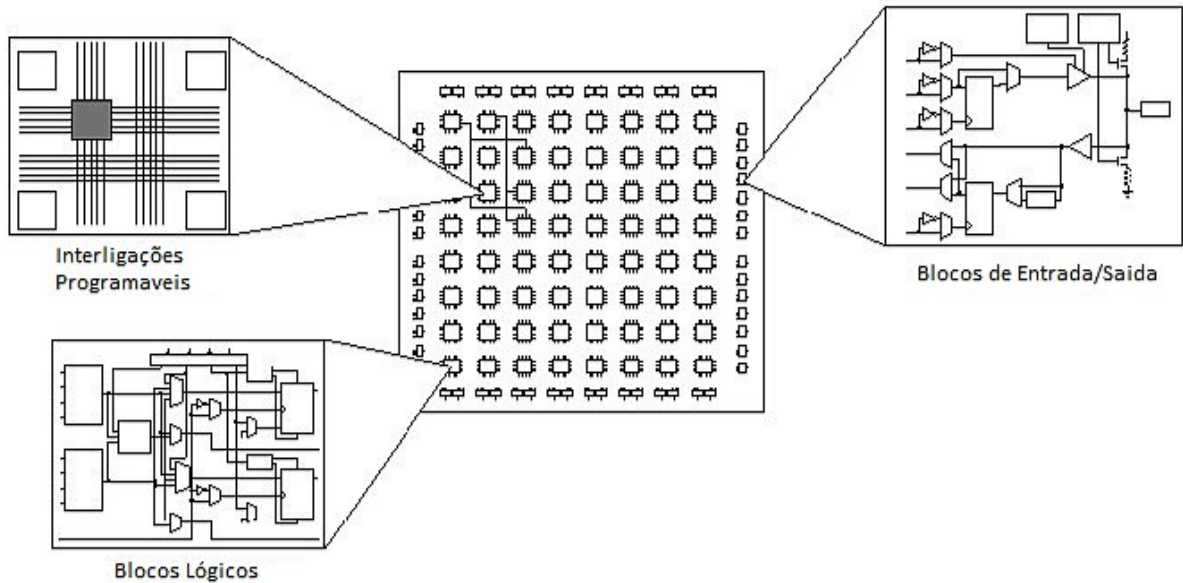


Figura 1.1: Arquitectura interna de uma FPGA.

secundário. Actualmente, devido à emergência de novos produtos e aplicações em que a portabilidade é essencial, a economia de consumo de potência é uma das principais preocupações durante o projecto de um circuito. Desta forma o habitual espaço de soluções passou a ser tridimensional na medida em que devem ser tidos em conta o desempenho, a área de implementação e o consumo de potência do circuito a projectar.

1.2 FPGAs

Este trabalho tem como objectivo explorar metodologias de projecto que permitam a redução do consumo de potência em circuitos implementados em dispositivos *Field-Programmable Gate Array* (FPGA). Estes dispositivos são circuitos integrados digitais que possuem um conjunto de blocos lógicos organizados em forma de matriz e respectivas interligações, que podem ser configurados de modo a criar um circuito digital que se pretenda implementar. Na figura 1.1 é mostrado um exemplo da arquitectura interna de uma FPGA. Os blocos lógicos que compõem as FPGAs têm como principal recurso as denominadas *Look-Up Tables* (LUTs). Estes blocos possuem um certo número de entradas e podem ser configurados de modo a realizar qualquer função lógica com tantas variáveis quantas

as entradas. Existem ainda blocos de entradas e saídas, *Input Output Blocks*, (IOBs), *flip-flops*, blocos de memória RAM, multiplicadores e também processadores completos.

Uma particularidade importante dos blocos lógicos reside no facto de possuírem *flip-flops* nas suas saídas, o que pode permitir implementar arquitecturas do tipo *pipelined* sem aumentar a taxa de ocupação de recursos da FPGA. Esta importante característica foi explorada no presente trabalho e será abordada com mais detalhe nos capítulos seguintes. A figura 1.2 mostra um bloco lógico no qual são visíveis os *flip-flops* presentes nas saídas.

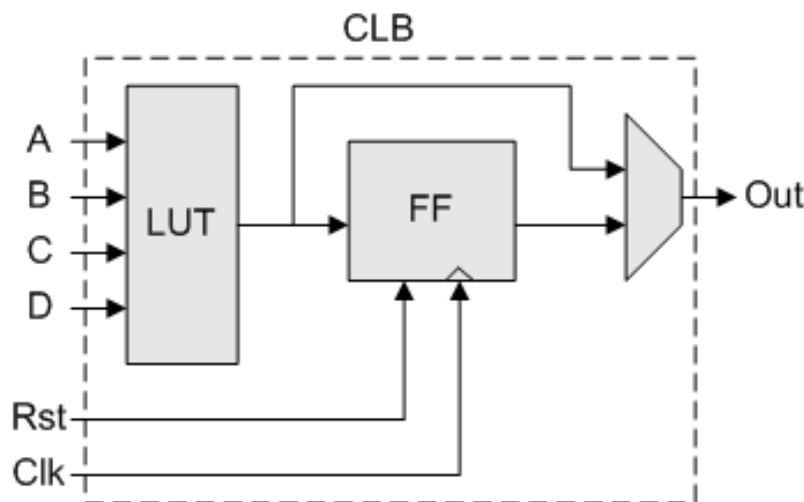


Figura 1.2: Bloco lógico de uma FPGA.

Devido à tecnologia envolvida, o custo de projecto de um ASIC é muito elevado, e estes só se tornam economicamente rentáveis quando produzidos em grandes quantidades, ao contrário das FPGAs, que não têm custos fixos de projecto, embora apresentem um preço unitário mais elevado do que um ASIC que seja produzido em massa. As principais vantagens de uma FPGA face a um ASIC resultam do facto de a duração do ciclo de projecto de um circuito ser muito reduzida, ao contrário do que acontece com um ASIC. Além disso, por serem reconfiguráveis, podem ser implementados diferentes circuitos recorrendo ao mesmo circuito integrado. Deste modo as FPGAs constituem o suporte preferível para realizar prototipagem rápida [2], assim como para implementar circuitos otimizados para aplicações específicas onde se possa tirar partido do paralelismo e de particularidades dos dados a processar. Por outro lado, um circuito implementado numa FPGA tem geralmente um desempenho mais reduzido quando comparado com um ASIC que execute a mesma função [3], uma vez que os atrasos obtidos nos circuitos implementados em FPGA são superiores devido à especificidade da arquitectura interna, a qual inclui recursos para a reconfiguração do dispositivo, e devido às interligações entre blocos lógicos serem maiores. Do mesmo modo o consumo de potência nas FPGAs é também muito elevado em comparação com um ASIC, até cem vezes mais [4, 2], uma vez que existe um maior número de nós e as capacidades associadas aos caminhos de dados, geralmente mais longos, são

também maiores. Torna-se por isso essencial reduzir o consumo de potência nos circuitos baseados em FPGAs, explorando a forma como os circuitos são especificados e projectados.

O ciclo de projecto de um circuito que se queira implementar em FPGA resume-se a três etapas: a descrição do circuito e respectiva validação funcional, a síntese lógica e a implementação na FPGA. O diagrama apresentado na figura 1.3 representa o ciclo de projecto de um circuito baseado na tecnologia FPGA, destacando a necessidade de rever a descrição do circuito se esta não estiver funcionalmente correcta.

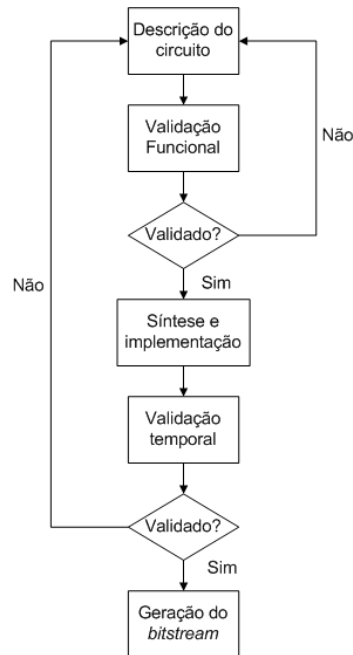


Figura 1.3: Ciclo de projecto de circuitos baseados em FPGAs.

Nas actuais abordagens de projecto a descrição do circuito é feita recorrendo a linguagens de descrição de *hardware*, de que é exemplo a linguagem *Verilog*. Esta linguagem foi estabelecida pelos *standards* do IEEE em 1995 [5], e revista em 2001 [6] e 2005 [7]. O *Verilog* é uma linguagem com uma sintaxe similar à da linguagem de programação C, mas ao contrário desta destina-se a descrever *hardware*. As duas formas básicas de descrever circuitos, na forma de uma descrição estrutural ou na forma de uma descrição comportamental são suportadas pelo *Verilog*. A primeira, descreve a estrutura do circuito à custa da interligação de portas lógicas, ou de módulos criados pelo projectista ou contidos em bibliotecas. A segunda abordagem é centrada na descrição do comportamento do circuito sem a necessidade de descrever a sua constituição, revelando-se a forma mais rápida e produtiva de descrever um circuito digital. Além dos circuitos propriamente ditos, há a considerar ainda descrições não sintetizáveis, com o objectivo de formar "bancadas" de testes (*testbench*) destinando-se a definir os sinais de entrada do circuito a testar e observar os sinais de saída. Deste modo torna-se possível simular o funcionamento do circuito através de uma aplicação de simulação, efectuando assim a sua validação funcional.

A síntese lógica consiste na interpretação da descrição do circuito, e na geração de uma descrição estrutural baseada em blocos lógicos e respectivas interligações de forma independente da tecnologia alvo. A última etapa do ciclo de projecto consiste na implementação do circuito sintetizado recorrendo aos recursos internos da FPGA. Só aqui surge o circuito real que realiza as funções do circuito especificado na forma de um modelo. A partir da implementação é depois gerado um ficheiro, com o qual é configurada a FPGA de modo a realizar as funções pretendidas.

No contexto deste trabalho de dissertação foi usado o simulador ModelSim [8] para a etapa da validação funcional. Nas etapas de síntese e implementação foram usadas as diferentes ferramentas integradas no ambiente de desenvolvimento ISE [9] da Xilinx. A programação da FPGA é feita com a aplicação iMPACT, que envia as definições de configuração para a FPGA através de um cabo JTAG [10].

1.3 Motivação

Justificou-se na secção 1.1 a importância de reduzir o consumo de potência de circuitos digitais. Na secção 1.2 fez-se uma introdução à utilização de circuitos baseados em tecnologia FPGA, realçando por um lado a importância que actualmente assumem na implementação de aplicações dedicadas e por outro lado o facto de consumirem mais potência que soluções equivalentes baseadas em ASICs. Este aspecto constitui a principal motivação para a realização desta tese. Refira-se ainda que há poucos trabalhos sobre o consumo de potência em FPGAs quando comparado com o número de trabalhos deste género sobre ASICs.

O principal objectivo deste trabalho de dissertação é estudar técnicas e diferentes arquitecturas de *hardware* que ao nível do projecto RTL (*register-transfer level*) permitam diminuir o consumo de energia de circuitos digitais implementados em FPGA. Pretende-se assim estudar de que forma um projectista pode intervir no sentido de realizar uma implementação que satisfaça as especificações e restrições do projecto e que simultaneamente possa reduzir o consumo de potência da solução a encontrar. Pretende-se que a partir deste estudo seja possível conhecer a eficácia quer em termos de redução do consumo de potência, quer em termos de aumento da área de circuito, quer em termos de diminuição do desempenho dos métodos testados.

Por outro lado também se pretende estabelecer um método fiável de avaliar o consumo de potência do circuito contido na FPGA. Uma vez que esta avaliação poderá ser feita por medição ou estimativa, também será possível obter uma comparação dos resultados de cada uma das técnicas.

1.4 Trabalho Realizado

Ao longo do desenvolvimento deste trabalho foram estudadas e aplicadas a circuitos técnicas de projecto com a intenção de diminuir o consumo de potência tais como a codificação de estados em circuitos com máquinas de estados e a arquitectura *pipelined*

Foram ainda estudadas formas de avaliar o consumo de potência, a medição e a estimação. A sua aplicação permitiu estudar e comparar em termos de potência consumida e recursos adicionais gastos, implementações alternativas de circuitos, realizadas de forma a manter o desempenho da implementação original e diminuir o consumo de potência. Foram utilizados circuitos de utilização frequente, e que devido à sua complexidade têm um impacto significativo no consumo de potência do circuito, como casos de estudo.

1.5 Estrutura da Dissertação

Para além desta introdução, este trabalho é composto por mais quatro capítulos. No capítulo 2 são abordadas as causas que originam o consumo total de potência num circuito digital, são igualmente apresentadas várias técnicas conhecidas para a sua diminuição, bem como resultados obtidos por alguns autores.

No terceiro capítulo são descritos os circuitos implementados para realizar os ensaios relativos ao consumo de potência, e ainda os métodos de medida utilizados para obter o consumo de potência dos circuitos ensaiados.

O quarto capítulo apresenta os resultados dos ensaios obtidos. São apresentados e desenvolvidos os consumos de potência e as ocupações de recursos dos circuitos analisados.

Por último as conclusões gerais do trabalho são feitas no quinto capítulo.

Capítulo 2

Consumo de potência em circuitos digitais CMOS

2.1 Introdução

A tecnologia com que são fabricados os circuitos digitais, e que é actualmente a tecnologia dominante no fabrico de circuitos integrados, é a tecnologia CMOS (*Complementary Metal Oxid Semiconductor*). Uma das principais vantagens face às suas concorrentes, e que a torna dominante, é precisamente o seu reduzido consumo de potência.

Neste capítulo é feita uma explicação sobre o consumo de potência em circuitos digitais CMOS e sobre as causas que estão na sua origem. Assim será possível perceber como é possível diminuir este consumo.

Ainda neste capítulo serão apresentadas técnicas de projecto que visam diminuir o consumo de potência de circuitos digitais CMOS.

2.2 Causas do consumo de potência

A potência consumida nos circuitos digitais CMOS resulta da contribuição de várias parcelas [11, 1], tendo cada uma delas uma causa distinta. A equação 2.1 mostra as parcelas em que se decompõe a potência consumida em circuitos CMOS. A primeira parcela, P_E , refere-se à potência estática, P_F refere-se à potência de fugas, P_{CC} refere-se à potência de curto-circuito e por último P_D refere-se à potência dinâmica.

Seguidamente apresentam-se estas componentes com mais detalhe, sendo apontadas as causas que estão na sua origem.

$$P_C = P_E + P_F + P_{CC} + P_D \quad (2.1)$$

2.2.1 Potência estática

A potência estática P_E , é uma parcela da potência consumida que não depende do sinal de relógio, ou seja, é consumida pelo circuito mesmo quando nenhum sinal muda de estado, correspondendo ao regime estacionário. A potência estática está normalmente associada aos circuitos baseados em tecnologia NMOS ou PMOS. As portas lógicas que utilizam esta tecnologia são essencialmente constituídas por uma rede de transístores, e uma resistência de *pull-up*, no caso NMOS, ou *pull-down* no caso PMOS. Quando a rede de transístores está a conduzir existe um caminho directo entre a alimentação positiva e a massa, através da resistência e dos transístores. A potência estática é a potência dissipada nessa mesma resistência e na rede de transístores por efeito de Joule.

A solução adoptada para eliminar este consumo de potência, devido à presença da resistência, foi a introdução da tecnologia CMOS. Esta tecnologia incorpora na mesma porta lógica uma rede NMOS e outra PMOS, complementares, apenas conduzindo corrente eléctrica uma de cada vez, pois têm um funcionamento alternado. A rede PMOS está ligada à alimentação positiva e é responsável por gerar o nível lógico "1" na saída, enquanto a rede NMOS está ligada à massa e é responsável por gerar na saída o nível lógico "0". Assim, nos circuitos baseados na tecnologia CMOS, a componente de potência estática é praticamente nula, pois a resistência das redes NMOS e PMOS é extremamente reduzida quando estão a conduzir e extremamente elevada quando estão ao corte.

2.2.2 Potência de fugas

A potência de fugas, P_F , à semelhança da potência estática não depende do sinal de relógio. Esta componente deve-se à corrente que percorre os transístores de um circuito quando estes estão polarizados na zona de corte [12], podendo o seu valor ser calculado a partir da equação 2.2, em que I_{pn} representa a corrente que flui pelo díodo formado pelo substrato e pela região de difusão quando inversamente polarizado, I_{st} representa a corrente de fugas do transístor, quando este está na zona de corte e V_{DD} representa a tensão de alimentação do circuito [13].

$$P_F = (I_{pn} + I_{st}) \cdot V_{DD} \quad (2.2)$$

O valor desta corrente de fugas aumenta com a diminuição do tamanho dos transístores, pois em transístores com comprimento de canal mais pequeno é mais difícil bloquear a corrente. Apesar de esta componente da corrente ser muito pequena, a soma das correntes de todos os transístores presentes num circuito integrado, que podem ser vários milhões, pode tornar-se significativa. O constante aumento da complexidade e nível de integração dos circuitos aliado à diminuição do tamanho dos transístores, alcançado pela evolução da tecnologia de fabrico, faz com que o peso relativo desta componente da potência se torne cada vez maior face às outras causas do consumo de potência em tecnologia CMOS [14].

2.2.3 Potência de curto-circuito

A potência de curto-circuito, P_{CC} , é causada pelas comutações de nível nas saídas das portas lógicas. O caminho criado directamente entre a alimentação positiva e a massa, quando ambas as redes de transístores, N e P, estão na zona de saturação [12] provoca o aparecimento de uma corrente de curto-circuito, I_{CC} entre os dois pontos, conforme representada na figura 2.1.

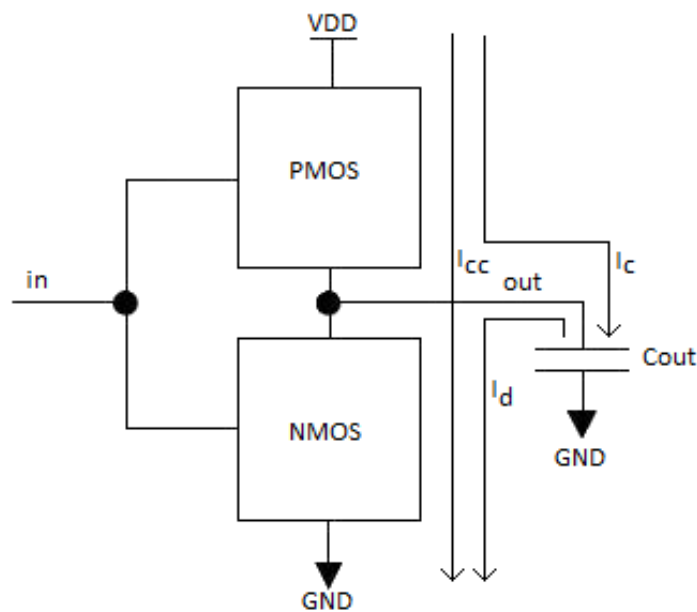


Figura 2.1: Porta lógica CMOS.

A potência de curto-circuito representa uma pequena parcela da potência total, cerca de 10% [15] e o seu valor pode ser calculado a partir da equação 2.3. Nesta equação K representa o número médio de comutações da saída por ciclo de relógio, β representa o factor de ganho dos transístores, o qual depende das características físicas destes, V_{DD} representa a tensão de alimentação, V_T representa a tensão de *threshold*, que é uma constante da tecnologia, f representa a frequência de funcionamento do circuito e τ o tempo de subida e descida do sinal de saída. É de salientar a influência da tensão de alimentação nesta componente da potência consumida, pois esta depende do cubo da primeira. As alterações deste valor, mesmo que pequenas poderão ter um peso significativo no valor da potência de curto-circuito.

$$P_{CC} = K \cdot \frac{\beta}{12} \cdot (V_{DD} - 2V_T)^3 \cdot f \cdot \tau \quad (2.3)$$

2.2.4 Potência dinâmica

A potência dinâmica, P_D é a principal componente da potência consumida pela generalidade dos circuitos implementados em tecnologia CMOS. Esta potência está associada à corrente necessária à carga das capacidades associadas a cada porta lógica e à corrente da respectiva descarga.

Na figura 2.1 estão representados no modelo de uma porta lógica CMOS as correntes de carga e de descarga da capacidade. A porta lógica é composta por uma rede de transístores PMOS que é responsável por carregar a capacidade de carga, e por uma rede de transístores NMOS, responsável pela sua descarga. Esta capacidade representa o equivalente das capacidades associadas aos transístores que formam a porta lógica, as quais dependem essencialmente das dimensões dos transístores, e da capacidade da carga aplicada à saída. Por seu lado, esta depende do tamanho das ligações às quais é aplicado o sinal de saída e do número e dimensões das portas lógicas que recebem este sinal. Em cada transição do nível lógico da saída de "0" para "1", a capacidade é carregada e na transição contrária a capacidade é descarregada. A potência dissipada é devida às correntes existentes durante os processos de carga e descarga da capacidade, ou seja só é dissipada potência nas transições de nível lógico, as quais reflectem a operação do circuito digital. O valor da potência dinâmica é dado pela equação 2.4

$$P_D = K \cdot C_{out} \cdot f \cdot V_{DD}^2 \quad (2.4)$$

onde K representa o número médio de transições de nível lógico por ciclo de relógio, C_{out} representa a capacidade de saída da porta lógica, V_{DD} representa a tensão de alimentação do circuito, a qual tem um peso significativo no cálculo desta potência, e por ultimo f representa a frequência de relógio à qual a porta lógica funciona.

A potência dinâmica é geralmente a componente com maior peso no consumo de potência total, pelo que a sua diminuição permite uma melhoria considerável no consumo total de potência do circuito.

2.3 Redução do consumo de potência ao nível do projecto RTL

As quatro componentes em que se decompõe o consumo de potência podem ser classificadas em dois grupos: as que dependem do sinal de relógio isto é da actividade de comutação do circuito, e as que não dependem do sinal de relógio. A potência estática e a potência de fugas, não dependem do sinal de relógio, e a potência dinâmica e a potência de curto-circuito dependem do sinal de relógio do circuito pois resultam das transições que ocorrem no circuito. Assim, atendendo à existência destas componentes, o consumo de potência pode ser dividido em apenas duas parcelas: potência estática e potência dinâmica.

A partir da compreensão das causas que estão por trás de cada um destes consumos é possível tê-las em consideração durante o projecto de circuitos digitais de modo a diminuí-las. Este é o objectivo deste trabalho pretendendo-se estudar a diminuição do consumo de potência intervindo ao nível do projecto RTL de um circuito.

A potência de fugas, apesar de ser uma componente cada vez mais importante devido à miniaturização dos circuitos, é intrínseca da tecnologia de fabrico dos circuitos. Ao nível do projecto RTL apenas é possível reduzir esta componente diminuindo o número total de portas lógicas, o que também contribui para a diminuição das componentes do consumo. Apesar disso não será viável tentar diminuí-la, pois a grande maioria das técnicas descritas mais à frente necessita de mais recursos, ou seja mais portas lógicas, o que significa mais potência de fugas.

Tal como no caso da potência de curto-circuito, a potência dinâmica depende fundamentalmente da tensão de alimentação do circuito. Este valor é um parâmetro da tecnologia de implementação do circuito, e geralmente é definido tendo em conta as preocupações de economia de potência, usando por isso valores de tensão bastante reduzidos. Sendo assim, a um projectista que especifique um circuito ao nível do projecto RTL não é possível alterar a tensão de alimentação de modo a reduzir o consumo de potência.

Ao nível do projecto RTL pode ser minimizado o número de portas lógicas, por exemplo simplificando o circuito, ou fazendo com que este se torne mais eficiente em termos de consumo de recursos. Outra possibilidade que pode ser explorada, e que à partida produzirá melhores resultados será a minimização do número de transições de sinal ocorridas nas saídas das portas lógicas. Esta minimização pode ser feita à custa da diminuição da frequência de relógio, o que por si só diminuirá também o desempenho do circuito, podendo ser incompatível com os requisitos de certas aplicações.

Uma outra alternativa será eliminar transições desnecessárias resultantes de atrasos diferentes na propagação de diferentes, vulgarmente designados por *glitches*.

A solução a implementar orientada para a diminuição do consumo, passa por utilizar arquitecturas de *hardware* alternativas de modo a minimizar o número de transições, mantendo o desempenho do circuito, tendo ainda em atenção o compromisso entre o consumo de potência conseguido e a área de circuito ocupada. Um caso comum, onde o número de transições pode ser optimizado é em circuitos com máquinas de estados, explorando diferentes formas de codificar os estados.

Conclui-se pois que ao nível do projecto RTL, apenas será possível diminuir o consumo de potência dinâmica. As técnicas descritas daqui para a frente terão como objectivo a diminuição desta componente do consumo de potência.

Descreve-se de seguida vários métodos que podem ser utilizados no sentido de reduzir o consumo de potência intervindo ao nível do projecto RTL, isto é métodos que podem ser postos em prática por quem vai especificar um circuito digital ao nível da transferência entre registos.

2.3.1 Análise estática

A análise estática, ou análise probabilística [16] consiste em determinar estatisticamente quais as combinações dos sinais de entrada mais prováveis, e a partir daí manipular as funções lógicas de maneira a obter um circuito que origine menos comutações. Esta técnica apesar de válida, torna-se impraticável para circuitos de maiores dimensões, ou para os quais não seja possível prever as entradas.

2.3.2 Arquitectura paralela

Uma arquitectura paralela [17, 11, 18] consiste na utilização de duas ou mais unidades funcionais de um circuito, em paralelo, de modo a melhorar o seu desempenho. Desejando manter o desempenho inicial da solução não paralela é possível baixar a frequência do sinal de relógio, e o valor da tensão de alimentação, o que como mostra a equação 2.4, leva a uma redução do consumo de potência. Conforme foi visto, ao nível do projecto RTL apenas é possível intervir na alteração do valor da frequência de relógio, o que neste caso não permite obter qualquer diminuição do consumo, uma vez que o valor da tensão de alimentação se mantém. A figura 2.2 mostra uma representação de uma arquitectura paralela (à direita) e a de uma arquitectura convencional (à esquerda).

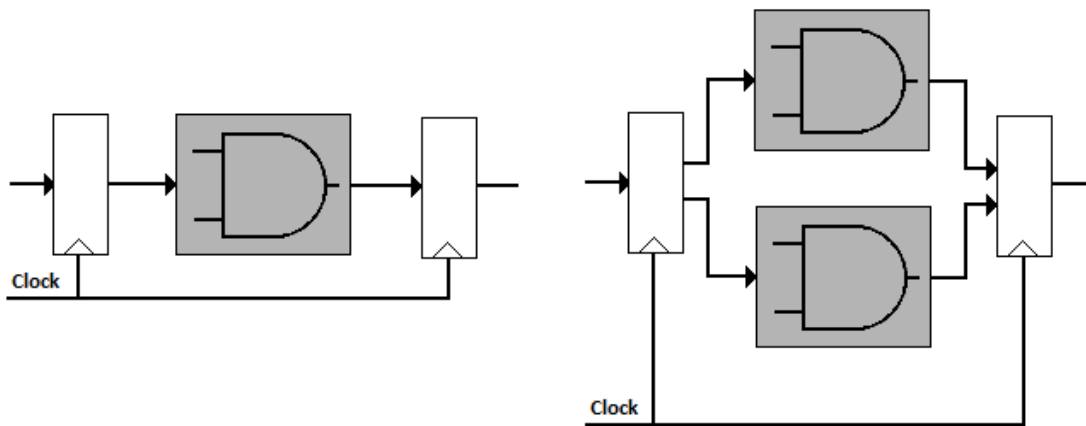


Figura 2.2: Exemplo de uma arquitectura paralela.

Um exemplo da utilização de uma arquitectura paralela são os recentes processadores fabricados pela Intel com vários núcleos de processamento na mesma unidade, que funcionam a frequências mais baixas, consumindo menos energia.

2.3.3 Redução de *glitches*

Os *glitches* [16] são impulsos de curta duração que resultam de comutações indesejáveis causadas pelos diferentes atrasos que ocorrem no caminho dos sinais que se propagam

por portas lógicas. Por exemplo, se as entradas de uma porta AND num dado instante forem "0" e "1", a saída será "0". Se ambas as entradas trocarem de valor, a saída deve permanecer em "0", mas caso a entrada que estava a "0" comute antes da outra, a saída passará a "1", e só depois com a comutação da outra entrada, tomará o valor correcto. Este tipo de comutações indesejáveis provoca um acréscimo no consumo de potência, uma vez que aumentam a componente de potência dinâmica devido ao acréscimo de comutações inúteis dos sinais. No entanto esta fonte de comutações pode ser diminuída ou mesmo eliminada, equalizando os atrasos dos sinais de entrada das portas lógicas, ou seja igualando os atrasos de todos os sinais. Como esta medida pode ser difícil de implementar, a solução mais simples passa por utilizar registos nas saídas das portas lógicas com um sinal de relógio comum, de modo a eliminar as diferenças entre os tempos de atraso de cada sinal. Esta técnica resulta da implementação do circuito dito *pipelined* embora ela seja geralmente empregue com o objectivo de melhorar o desempenho de um circuito aumentando o débito de produção de resultados.

2.3.4 Arquitectura *pipelined*

Uma arquitectura *pipelined* [19, 17], obtem-se pela divisão por andares de circuitos combinacionais, introduzindo registos síncronos com um sinal de relógio a seguir a cada andar. A figura 2.3 mostra uma representação de uma arquitectura *pipelined* (à direita) e de uma arquitectura não *pipelined* (à esquerda). Assim é possível tratar mais dados em

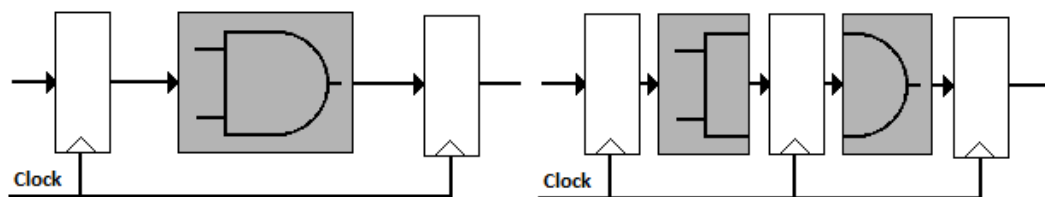


Figura 2.3: Exemplo de uma arquitectura *pipelined*.

menos tempo, pois após a entrada de um dado valor não é necessário esperar que o seu processamento esteja completo para introduzir novos dados. Basta esperar que o primeiro resultado intermédio seja produzido para se poder submeter novos valores às entradas do circuito. O circuito resultante é então mais rápido, podendo assim ser aumentada a sua frequência de funcionamento, o que leva a um aumento do consumo de potência.

No entanto, se a frequência for mantida, e uma vez que a arquitectura *pipelined* impede a propagação dos *glitches* por mais do que um nível, e pode diminuir a capacidade do caminho de dados, uma vez que este se torna mais curto [20], permite diminuir o consumo de potência. Esta arquitectura é particularmente interessante no projecto de circuitos em FPGAs, conforme será justificado na secção 2.4, dado que os blocos funcionais que as compõem incorporam *flip-flops* nas suas saídas, que embora não sejam usados em circuitos combinacionais serão aproveitados para a implementação dos andares de *pipeline*, sem dar origem a um aumento da taxa de ocupação da FPGA pelo circuito resultante. Assim pode ser possível diminuir o consumo de potência sem aumentar a utilização de recursos. Por outro lado, uma vez que os *flip-flops* serão síncronos com o sinal de relógio, o consumo provocado pelo aumento da árvore de distribuição de relógio e conseqüente aumento da capacidade, vai conduzir a um aumento do consumo de potência.

2.3.5 Gestão dinâmica do sinal de relógio

O consumo de potência de um circuito digital depende em grande parte da frequência do sinal de relógio. É sabido que a simples diminuição do seu valor se torna por vezes impraticável devido à inerente diminuição de desempenho. Porém, em grande parte das aplicações é aceitável diminuir o desempenho sem conseqüências no funcionamento. São exemplos disso circuitos que utilizem taxas de entrada de dados variáveis. Nessas circunstâncias é possível e desejável gerir dinamicamente o sinal de relógio do circuito [21] de acordo com as necessidades de desempenho, e conseqüentemente diminuir a potência consumida. A gestão dinâmica consiste na diminuição da frequência do sinal de relógio quando não for necessário um desempenho tão elevado, e o aumento da frequência quando necessário. A variação da frequência deve ser feita com valores pré-definidos, de modo a mais facilmente ser possível a adaptação a dispositivos de entrada e saída, bem como a melhor gestão de relações temporais.

2.3.6 *Clock gating*

O *clock gating* [22] consiste em desligar o sinal de relógio de secções de um circuito que momentaneamente não são necessárias, de modo a eliminar as comutações que ocorram nessa região. Esta solução tem o mesmo princípio da gestão dinâmica do sinal de relógio, com a diferença de que esta é praticada a nível local, ao contrário da anterior que é global. Trata-se de uma solução do tipo liga/desliga, ao contrário da anterior em que o sinal de relógio pode adoptar vários valores de frequência em diferentes intervalos de tempo. O *clock gating* deve ser utilizado tendo em consideração o atraso adicional no sinal de relógio imposto pela *latch* ou porta lógica usada para o desligar, podendo dar origem a *glitches*.

2.3.7 Codificação de estados

A codificação de estados usada na especificação de máquinas de estados pode em determinadas situações ter influência no consumo de potência de um circuito [23, 24]. Nos circuitos baseados em máquinas de estados em que existam muitas trocas de estado ou muitos estados, a potência consumida pela máquina de estados pode ser significativa. Para diminuir o consumo, a codificação dos estados deve ser escolhida de modo a minimizar as trocas de bits entre estados consecutivos. Existem códigos especialmente concebidos para esse efeito, sendo exemplo disso os códigos *one-hot* e *Gray*, que têm a particularidade de apresentarem códigos consecutivos que diferem apenas em dois bits, ou em um bit, respectivamente. Portanto, em circuitos com máquinas de estados e onde estas representem parte importante dos circuitos, será possível minimizar o consumo de potência explorando diferentes formas de codificação de estados.

2.4 Metodologias de projecto para baixo consumo em FPGAs

Os métodos apresentados na secção anterior visam reduzir o número de comutações do nível de sinal à custa de um aumento da área de circuito ocupada. A vantagem conseguida por cada técnica em termos de consumo de potência tem pois de ser ponderada em função da penalização de área acrescentada para a sua implementação. Faz parte deste trabalho de dissertação avaliar a eficácia das principais técnicas descritas em circuitos implementados em FPGAs.

As FPGAs são circuitos que devido à sua reconfigurabilidade apresentam consumo de potência mais elevado do que circuitos equivalentes implementados em tecnologia ASIC. O consumo de potência estática em circuitos do tipo FPGA tem um peso mais significativo do que aquele que tem nos circuitos ASICs. Esta diferença deve-se fundamentalmente à necessidade de utilizar mais transístores para realizar um mesmo bloco numa FPGA do que aqueles que seriam necessários num ASIC, o que se traduz em maiores correntes de fugas.

A generalidade das FPGAs disponíveis actualmente no mercado possuem uma arquitectura que foi desenvolvida sem ter como objectivo uma solução dedicada à implementação de circuitos de muito baixo consumo. Há contudo actualmente trabalhos de desenvolvimento com vista a fabricar circuitos FPGA com recursos que viabilizam a implementação de circuitos para aplicações onde o consumo de potência seja fundamental. O autor de [25] utilizou uma tensão de alimentação mais baixa e efectuou melhoramentos na configuração das células de memória com o objectivo de baixar o consumo de potência de uma FPGA implementada numa tecnologia de CMOS de 90 nm, duplo V_T e óxido triplo. Foi obtida com esta implementação uma redução do consumo de potência de 99% e 46% das componentes estática e dinâmica respectivamente, quando comparada com uma Spartan 3 equivalente. Ainda acerca de FPGAs orientadas para baixo consumo, o autor de [26] apresenta um estudo sobre arquitecturas de FPGA e sobre diferentes optimizações que

podem ser feitas nas gerações futuras de FPGAs. Estão também em curso trabalhos que visam otimizar a fase de implementação ao nível das ferramentas informáticas quanto ao consumo de potência [27].

No caso concreto de circuitos implementados em FPGAs será necessário tirar partido das características de *hardware* particulares deste tipo de circuitos. Uma das características que pode ser aproveitada é a presença de *flip-flops* à saída de cada bloco lógico e que muitas vezes não são utilizados. Uma forma de os rentabilizar é utilizando arquitecturas do tipo *pipelined*, que além das vantagens já referidas em termos de redução do consumo de potência, apresenta neste caso a vantagem de aproveitar recursos existentes no circuito. Torna-se assim possível diminuir o consumo sem aumentar a necessidade de recursos. A utilização de arquitecturas *pipeline* é uma das técnicas mais vantajosas para circuitos implementados em FPGAs.

Das técnicas de diminuição de consumo de potência apresentadas, algumas são mais apropriadas do que outras para determinados tipos de circuitos, e simultaneamente, algumas serão mais eficazes do que outras do ponto de vista quantitativo. Parte deste trabalho consiste em qualificar e quantificar os resultados obtidos para várias formas de circuitos implementado recorrendo a algumas técnicas expostas.

2.5 Estado da arte

Comparativamente aos trabalhos sobre o consumo de potência em circuitos do tipo ASIC, existem poucos trabalhos semelhantes sobre circuitos implementados em FPGAs.

De entre as técnicas de redução do consumo de potência, as mais promissoras em termos de resultados, segundo o autor de [20], serão a codificação de estados e a arquitectura *pipelined*, podendo estas ser combinadas entre si, ou com outras já referidas.

No que respeita a máquinas de estados é dito em [23] que é possível obter uma redução de 57% no consumo de energia, com a escolha adequada da codificação de estados. É referido também que para máquinas de estados com poucos estados, tipicamente menos de oito, o tipo de codificação que leva a um menor consumo é a codificação binária, e que no caso de máquinas de estados maiores, mais que 16 estados, a codificação *one-hot* apresenta melhores resultados. Ainda no mesmo artigo é mencionado que a codificação *two-hot*, não traz qualquer benefício, e que por vezes uma codificação binária orientada para a simplificação, composta por códigos que originam funções de saída mais simples pode trazer vantagens face à codificação binária.

Num outro artigo [24] é sugerida a decomposição de máquinas de estados com muitos estados em máquinas mais pequenas, com baixa actividade entre elas. Por exemplo um circuito que durante grande parte do tempo se encontre apenas num grupo pequeno de estados, pode ser dividido em dois mais pequenos. Com a aplicação desta técnica, segundo os autores do estudo, é possível reduzir o consumo de potência de um circuito que esteja nestas condições em cerca de 46%.

Uma arquitectura *pipelined*, reduz a quantidade de *glitches* no caminho de dados e diminui a carga capacitiva do mesmo. São referidos em [28] dois exemplos da utilização desta técnica aplicada a circuitos multiplicadores. Num deles, implementado na FPGA XC3050 foi obtida uma redução de 33% no consumo. No outro caso, implementado na FPGA XC4005 a redução no consumo de potência foi de 58%.

Ainda recorrendo à arquitectura *pipelined* o autor de [19] conseguiu uma redução de 81% de consumo de potência através da utilização de uma arquitectura *pipelined* com vários andares, aplicada a um circuito multiplicador de 64 bits. Este resultado foi obtido com a FPGA *Stratix EP1S40F780C5* fabricada pela Altera.

Em [16] é descrito como a partir da redução dos *glitches* é diminuído o consumo de potência. O autor atingiu uma redução de 92% no consumo através da equalização dos atrasos. O autor utilizou vários tipos de circuitos tais como somadores, multiplicadores e divisores.

2.6 Conclusão

Apresentou-se neste capítulo as causas do consumo de potência em circuitos digitais CMOS. Pela importância que assume mostrou-se como se pode proceder ao nível do projecto RTL para diminuir a componente dinâmica do consumo de potência. Foi também salientado que os circuitos baseados em FPGAs gozam de particularidades específicas de *hardware* que podem permitir tirar partido da utilização de *pipelined*, sem que para isso sejam gastos blocos lógicos face à solução não *pipelined*.

Ao contrário de circuitos baseados em ASICs, não existem muitos trabalhos sobre o consumo de potência em circuitos baseados em FPGAs. De facto as FPGAs são uma tecnologia relativamente recente tendo ganho relevância pelas possibilidades ao nível de reconfiguração e da implementação de *hardware* dedicado para aplicações específicas, e que só à pouco despertou a merecida importância quanto ao consumo de potência destes circuitos.

Capítulo 3

Avaliação do consumo de potência

3.1 Introdução

No capítulo anterior concluiu-se que várias técnicas podem ser usadas no sentido de reduzir o consumo de potência em circuitos implementados em FPGAs. Em particular durante o projecto RTL de um circuito pode actuar-se no sentido de diminuir o consumo de potência considerando opções arquitecturais, de que é exemplo a utilização de *pipelining*, que reduzem o consumo de potência dinâmica à custa de uma penalização tolerável ao nível de recursos de implementação.

Para verificar experimentalmente a eficácia é necessário avaliar o consumo de potência de circuitos que as usam. Este capítulo descreve como estimar a potência consumida por um circuito usando a ferramenta Xpower [29], integrada no ambiente de desenvolvimento ISE da Xilinx. A medição do consumo de potência é depois apresentada no contexto de uma metodologia experimental que visa medir a potência efectivamente consumida por um circuito digital.

A terminar este capítulo descrevem-se os circuitos que serviram de base à verificação experimental das técnicas de redução do consumo de potência abordadas.

3.2 Técnicas de avaliação do consumo de potência

É fundamental para a realização deste trabalho a capacidade de avaliar a potência consumida por um circuito. Sem esta capacidade é impossível tirar qualquer tipo de conclusões quanto à eficácia de qualquer técnica de redução do consumo de potência. É ainda importante conseguir avaliar as componentes, da potência consumida, dinâmica e estática, de modo a concluir se a redução da componente dinâmica é conseguido sem o aumento excessivo da componente estática resultante da modificações arquitecturais efectuadas.

O consumo de potência pode ser conhecido através de dois métodos: estimação por *software*, e medição real. A medição envolve a implementação do circuito em questão, em conjunto com circuitos complementares, permitindo tirar conclusões mais fiáveis e precisas pois assenta em resultados reais. Por outro lado, a estimação permite obter resultados mais facilmente do que através da medição real, mas com um grau de incerteza maior.

3.2.1 Estimação do consumo de potência

Ao nível do projecto RTL a avaliação do consumo de potência por estimação torna-se apropriada pois é feita por ferramentas ao mesmo nível das ferramentas utilizadas na fase de projecto. Além disso, seria totalmente inconveniente, na fase de projecto construir o circuito para daí medir o seu consumo de potência. Neste contexto, a avaliação por estimação é a alternativa mais simples e rápida de se conhecer a potência absorvida por um circuito digital. Sem qualquer montagem laboratorial e apenas com recurso a ferramentas de desenvolvimento, é possível estimar o consumo de potência.

Neste trabalho a estimação da potência foi obtida recorrendo à ferramenta Xpower [29] disponibilizada pela Xilinx. O Xpower permite estimar o consumo de potência estático e dinâmico, apresentando os resultados parciais, relativos ao núcleo e aos *pads* da FPGA, e totais, quer de corrente que percorre o circuito, quer do consumo de potência associada.

O Xpower realiza a estimação do consumo de potência de circuitos implementados em FPGAs com base no conhecimento das características internas da FPGA em causa, correntes de fugas e capacidades de cada um dos nós internos, e ainda da taxa de comutação associada a cada um destes nós. A taxa de comutação, ou taxa de actividade de cada um dos nós, depende do circuito e de todos os valores actuais e passados das entradas do mesmo. Se as entradas do circuito forem sucessivamente constantes a taxa de comutação será em princípio nula, enquanto que se estas variarem, à partida a actividade do circuito será maior.

As taxas de actividade podem ser fornecidas à ferramenta de duas formas. Por inserção manual ou através de um ficheiro de simulação que contenha essa informação (ficheiro *.vcd). A introdução manual apenas se torna viável para circuitos de muito pequena escala, uma vez que é impossível conhecer em tempo útil a actividade de cada um dos milhares de nós presentes num circuito maior. No entanto, o Xpower permite a introdução manual desta informação, no seu ambiente gráfico sob a forma de frequências de comutação.

O recurso a um ficheiro de simulação é mais apropriado para as dimensões dos circuitos mais usuais. Este ficheiro pode ser produzido pela ferramenta de simulação, durante uma simulação *post-routing*. Esta simulação, tal como o nome indica é feita depois do processo de síntese e implementação, após a qual, está já definido o conteúdo e a configuração da FPGA de modo a produzir o circuito desejado.

A construção do *testbench* para esta simulação deve ser feita tendo em conta que deve ser reproduzido fielmente o funcionamento real do circuito, pois só assim as taxas de actividade obtidas serão verdadeiras. É necessário ter ainda em conta que quanto maior

for a duração da simulação, maior o volume e teoricamente a diversidade de vectores de entrada, pelo que maior será o realismo e a fiabilidade da simulação.

Neste trabalho, à semelhança das simulações feitas para a validação funcional das descrições dos circuitos, as simulações *pós-routing* foram feitas com o simulador ModelSim. O apêndice A apresenta um guia de utilização do Xpower para utilizadores do ISE da Xilinx. Pretende-se ilustrar a sua utilização detalhando aspectos de ordem prática.

3.2.2 Medição do consumo de potência

A medição da potência, P , consumida por um circuito pode ser obtida através do conhecimento simultâneo da intensidade de corrente, I , que percorre o circuito e da tensão, U que o alimenta, sendo dada por $P = U \cdot I$.

Neste trabalho o consumo de potência dos circuitos implementados foi obtida através da medição da corrente consumida pelo núcleo da FPGA. Para esse efeito foi usada uma placa de prototipagem (Digilent Spartan-3 Starter Kit Board) [30] equipada com uma Spartan-3 XC3S200 [31] da Xilinx, a qual foi alterada para permitir medir a corrente consumida pelos circuitos implementados. A alteração consistiu na remoção do regulador de tensão (FAN1112) [32], que alimenta o núcleo da FPGA. A alimentação deste (1,2V) foi então feita externamente, tornando assim possível medir o consumo de corrente. Foi ainda colocada em série uma resistência de muito baixo valor (no caso 1Ω) de modo a ser possível observar a evolução da corrente num osciloscópio. A medição da corrente foi feita utilizando um multímetro digital com a função de medição *true RMS*. A figura 3.1 mostra a montagem final do circuito de medição e a figura 3.2 apresenta a visualização do

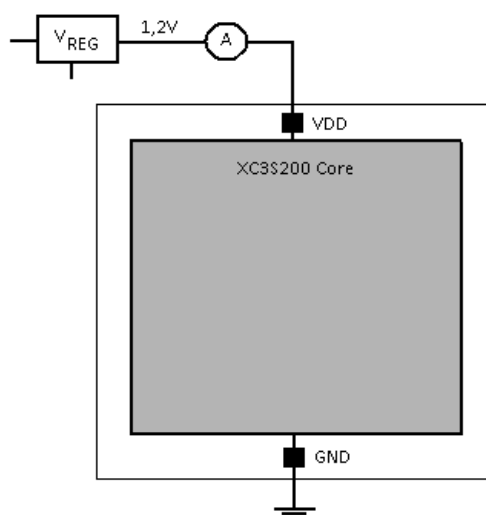


Figura 3.1: Circuito para a medição da corrente consumida pelo núcleo da FPGA.

consumo de corrente do núcleo da FPGA utilizando um osciloscópio digital evidenciando-se o carácter dinâmico do circuito através das variações evidenciadas pela corrente consumida.

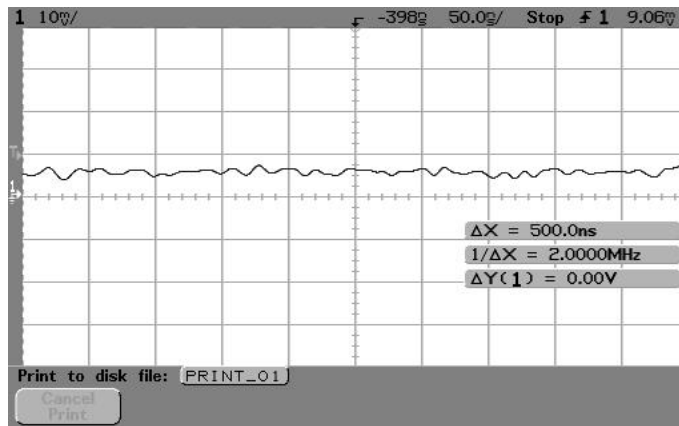


Figura 3.2: Visualização do consumo de corrente.

A medição correcta do consumo de potência de um apenas pode ser feita se o circuito estiver em funcionamento, isto é, se o circuito realizar alguma operação sobre dados que vão variando nas suas entradas. Por outro lado deve-se limitar o número de entradas e saídas do circuito de modo a minimizar as influências externas sobre esse circuito, tais como capacidades de carga. Assim, torna-se necessário gerar internamente os vectores que excitam as entradas do circuito em questão. Para esse propósito foi usado um módulo que implementa um *Linear Feedback Shift Register* (LFSR), o qual permite gerar sequências de valores pseudo-aleatórios. As entradas do circuito em teste passam a ser geradas pelo LFSR, eliminando a necessidade de utilizar circuitos externos. Este módulo foi gerado a partir do COREGEN [9], integrado no ISE o qual permite gerar circuitos rápida e facilmente através da escolha dos parâmetros necessários. Em termos de saídas de um circuito a implementar na FPGA, a ferramenta de síntese XST obriga a que o circuito tenha pelo menos uma saída. Para garantir esta saída e assim reduzir o número de entradas e saídas do circuito ao mínimo, foi implementado um outro módulo que gera um bit da paridade de todas as saídas efectivas do circuito em teste.

Um aspecto não menos importante prende-se com o facto de ser desejável medir distintamente os consumos de potência em regime dinâmico e em regime estático. Este problema foi solucionado com a introdução de um sinal de habilitação de relógio (*clock enable*), que habilita o relógio à entrada do circuito, permitindo ligar ou desligar o relógio de todo o circuito. Este sinal é directamente controlado por um dos oito interruptores que equipam a placa de ensaio utilizada. Assim, a medição da potência consumida com o *clock enable* activo representa o consumo total de potência, enquanto que o consumo obtido com o *clock enable* inactivo representa apenas o consumo estático. O consumo dinâmico é então obtido pela diferença das duas medições.

A figura 3.3 apresenta a implementação feita para a medição do consumo de potência. A placa de circuito impresso produzida para esse efeito foi ligada à placa de ensaio através

de um dos conectores de expansão. A alimentação para o regulador externo que fornece os 1,2V é retirada deste conector.

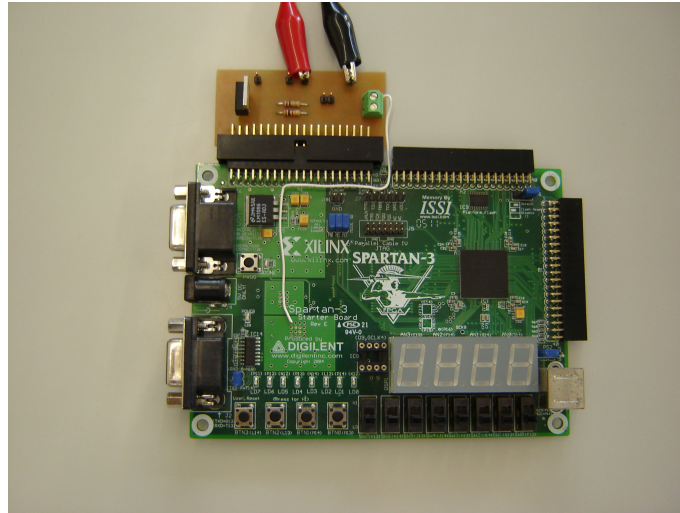


Figura 3.3: Circuito utilizado na medição real do consumo de potência.

Outro facto que deve ser levado em consideração, é o consumo de potência dos circuitos adicionais, tais como o LFSR e o gerador de paridade, implementados na FPGA. Efectivamente, quando se mede a potência consumida por um circuito em ensaio, está-se a quantificar não só a potência consumida por esse circuito mas também a parcela consumida pelos circuitos auxiliares do ensaio. A solução adoptada para eliminar esta parcela da potência medida consistiu na medição da potência consumida pelos circuitos auxiliares, tendo para isso feita a sua implementação isoladamente. Mais uma vez o consumo real pode ser obtido pela diferença entre o valor medido e o consumo dos circuitos auxiliares.

A figura 3.4 mostra um diagrama da implementação utilizada na medição real. O bloco central CUT (*Circuit Under Test*) representa o circuito em relação ao qual se pretende medir o consumo. Está exemplificado na figura com duas entradas (A e B) cujos valores são fornecidos pelo LFSR. Está ainda representado o *clock enable*, o LFSR que gera os vectores de entrada do circuito e o *array* de portas do tipo XOR para obter o bit de paridade.

3.3 Casos de estudo

Este trabalho de dissertação consistiu na aplicação de técnicas de projecto orientadas para o baixo consumo de circuitos implementados em FPGAs. As secções anteriores apresentam formas de avaliação do consumo de potência e, em particular na infraestrutura para a medição da potência consumida. Esta secção apresenta os circuitos usados em ensaios e que permitem investigar a eficácia das técnicas mencionadas.

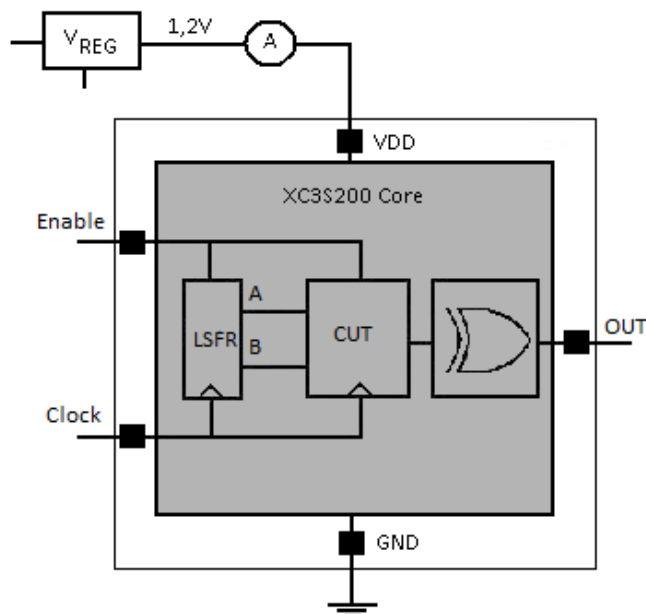


Figura 3.4: Diagrama da implementação utilizada na medição da corrente.

3.3.1 Circuito md_4

Um dos circuitos utilizados neste trabalho, para experimentar as técnicas de redução do consumo de potência, resultou da implementação em *hardware* do algoritmo " md_4 " [33], utilizado na encriptação de dados.

Este circuito é composto por uma máquina de estados finitos que controla uma sequência de operações lógicas e aritméticas efectuadas sobre um conjunto de dados de 32 bits. A máquina de estados do circuito original possui cinquenta e cinco estados. Em quarenta e oito deles é feita uma adição algébrica entre três valores. São eles o valor armazenado num registo, o resultado de uma operação lógica entre os valores armazenados em três outros registos, e uma constante. Esta constante é igual a zero nos primeiros 16 estados.

Uma vez que o circuito executa todas estas operações e atendendo às inúmeras transições de estados, é de esperar que o seu consumo de potência dependa da codificação de estados. Assim sendo, optou-se por considerar este circuito neste trabalho. A descrição original deste circuito foi feita sem ter em conta quaisquer preocupações quanto ao consumo de potência, e a codificação da máquina de estados foi feita com codificação binária com sequência de estados praticamente aleatória.

Como foi mostrado no capítulo 2, uma redução significativa do consumo de potência pode ser obtida através de uma escolha apropriada da codificação da máquina de estados. Nesta tarefa, a ferramenta de síntese *XST* facilita o trabalho de projecto, pois permite ao utilizador escolher como parâmetro da síntese a optimização das máquinas de estados e

qual a codificação de estados que se pretende usar. Assim, foram sintetizadas sete versões do circuito "md4", cada uma delas com uma codificação de estados diferente, para assim avaliar o impacto em termos de potência consumida.

A FPGA utilizada para este circuito foi uma Virtex5 XC5VLX50T fabricada pela *Xilinx*. A placa de prototipagem preparada para a medição da potência não foi utilizada uma vez que a Spartan-3 XC3S200 não possui recursos suficientes para albergar o circuito "md4". Por esta razão apenas foi possível obter o consumo de potência das várias implementações deste circuito por estimação. Todos os resultados foram obtidos para situações de funcionamento com o mesmo desempenho do circuito original, e a mesma interface com o exterior.

Num segundo conjunto de ensaios o circuito foi otimizado distribuindo as duas adições executadas em cada estado, por dois novos estados. O resultado desta modificação foi o equivalente à introdução de um andar de registos entre cada soma, que como foi explicado no capítulo 3, é conhecido como *pipelining*. A máquina de estados deste novo circuito é composta por cento e dois estados, pelo que a codificação dos estados é uma questão a ter ainda mais em conta. Foram sintetizadas sete novas versões do circuito, utilizando esta diferente arquitectura, e mais uma vez foi feita a estimação do consumo de potência.

3.3.2 Circuitos aritméticos

Os circuitos aritméticos formam uma classe de circuitos conhecida pelo seu elevado consumo de potência e vários trabalhos têm sido feitos neste campo. A escolha destes circuitos durante neste trabalho deve-se á sua grande importância em aplicações nas áreas de processamento de sinal e imagem, onde as operações aritméticas são frequentes. Além disso, permitem a consideração de arquitecturas *pipelined*, que como foi justificado, podem conduzir à redução do consumo de potência. A juntar a isto, a presença de *flip-flops* nos CLBs das FPGAs livres em circuitos de natureza combinacional, permitem a implementação de arquitecturas *pipelined* sem custo adicional. Como forma de aproveitar estes factos foram utilizados circuitos multiplicadores e divisores combinacionais em forma de *array*. Estes circuitos consomem uma grande quantidade de potência dinâmica devido à propagação de *glitches*, dado que no seu funcionamento normal existe a propagação de *carry* por todos os *full-adders* que compõem a estrutura. Outra questão que não pode ser deixada de lado, é o facto de a ocorrência de *glitches* ser tanto maior quanto maior a dimensão do circuito, uma vez que os atrasos de propagação se vão acumulando.

Na figura 3.5 está representada a estrutura de um multiplicador de $n \times n$ bits. Este multiplicador é composto por n^2 *full-adders* e n^2 portas AND. A introdução de registos entre andares consecutivos bloqueia a propagação de *glitches* de andar para andar, o que como foi visto no capítulo 2 permite a diminuição do consumo de potência.

Foram consideradas 4 versões de circuitos multiplicadores, de 4, 8, 16 e 32 bits, de modo a estudar a influência da dimensão do circuito na eficácia da arquitectura *pipelined*. A descrição do circuito baseou-se na descrição funcional de um *fulladder*, e na descrição

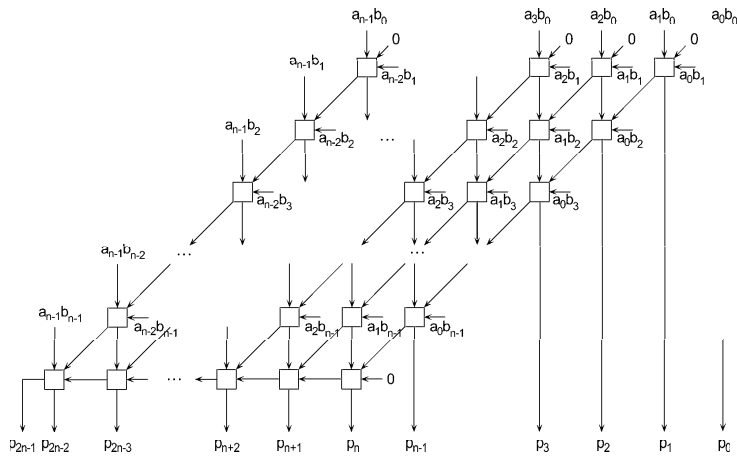


Figura 3.5: Multiplicador em forma de *array*.

estrutural da matriz que forma o multiplicador. Para cada uma das dimensões do circuito foram feitas diferentes versões, cada uma com um número diferente de andares de *pipeline*. O circuito foi implementado na FPGA Spartan-3 XC3S200 utilizando a placa de prototipagem equipada com a infraestrutura (secção 3.2) para a medição da potência. O projecto final de cada um dos circuitos inclui o LFSR para gerar os vectores de entrada, assim como o *array* de portas do tipo XOR que gera o bit de paridade a partir da saída do multiplicador.

Com o objectivo de avaliar a precisão de estimação do Xpower, o consumo de potência foi obtido quer por estimação, quer por medição real. Em qualquer dos casos o circuito final apenas implementado possui três ligações com o exterior. O sinal de relógio, a entrada de habilitação de relógio e a saída de paridade. Foram obtidos os consumos de potência dinâmica e estática do núcleo da FPGA, em ambos os casos, bem como os recursos ocupados por cada um deles.

A mesma estratégia foi seguida para os circuitos divisores. Foram descritos circuitos divisores de 4, 8, 16 e 32 bits, e para cada um foram consideradas diferentes versões, cada uma com um número diferente de andares de *pipeline*. A figura 3.6 mostra a arquitectura de um divisor em forma de *array*, tal como os que foram implementados durante deste trabalho. É de salientar as semelhanças desta arquitectura face à arquitectura do circuito multiplicador, tornando simples a tarefa de transformação em soluções *pipelined*.

Os circuitos divisores assumem uma importância ainda maior do que os multiplicadores, uma vez que a generalidade das FPGAs, ao contrário dos multiplicadores, não possui internamente circuitos divisores, nem as ferramentas de síntese inferem tais operadores.

3.3.3 Filtros FIR

Os filtros digitais constituem uma das aplicações práticas dos circuitos aritméticos e são empregues com frequência em diversas aplicações. Um desses filtros é o filtro FIR (*Finite*

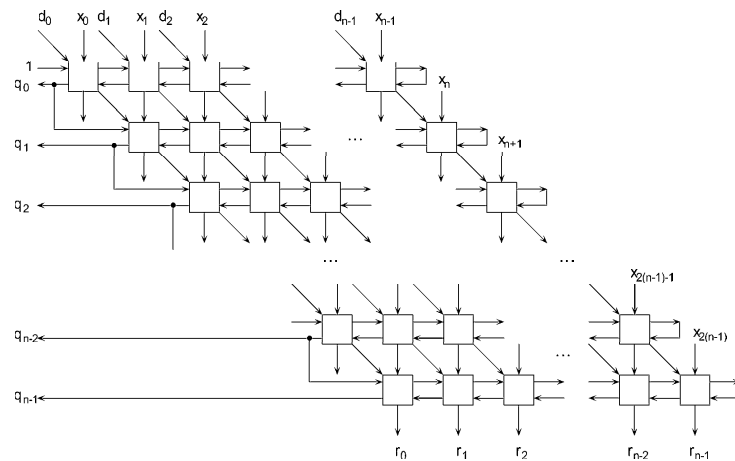


Figura 3.6: Divisor em forma de *array*.

Impulse Response) que quando aplicado a uma imagem, por exemplo, permite transformar um pixel à custa de uma relação de dependência de outros pixels próximos.

Uma vez que a implementação destes filtros se baseia em circuitos aritméticos, sofrem dos mesmos problemas de consumo de potência. É portanto importante estudar os consumos de potência destes circuitos, não só pelas razões apontadas na secção anterior para os circuitos aritméticos, mas também porque será possível utilizar e testar um filtro FIR numa aplicação real, inserido numa cadeia de processamento de vídeo [34], e assim comparar a potência consumida pelo filtro no contexto desta aplicação, com a potência medida usando o LFSR para gerar valores de entrada artificiais. A principal vantagem desta experiência é a utilização de dados reais e de funcionalidades reais, e não apenas circuitos modelo com entradas aleatórias.

A figura 3.7 mostra a arquitectura de um filtro FIR com n andares, em que $x(k)$ e $y(k)$ representam os vectores de entrada e saída de ordem k , e h representa os coeficientes do filtro. Esta estrutura corresponde à implementação da equação 3.1.

$$Y(n) = \sum_{i=0}^p h_i \cdot x \cdot (n - i) \quad (3.1)$$

Os filtros implementados possuem 8 andares, e têm como coeficientes, constantes de 8 bits representadas em complemento para dois. A implementação dos filtros foi feita usando duas abordagens diferentes. Numa delas foram utilizados os multiplicadores combinacionais descritos aquando da implementação dos circuitos multiplicadores, e na outra foram utilizados os multiplicadores disponíveis na Spartan-3 XC3S200 utilizada. A partir destas duas implementações foram criadas outras quatro, cada uma com um número diferente de andares de *pipeline*.

A figura 3.8 mostra a infraestrutura laboratorial com a cadeia de processamento em funcionamento.

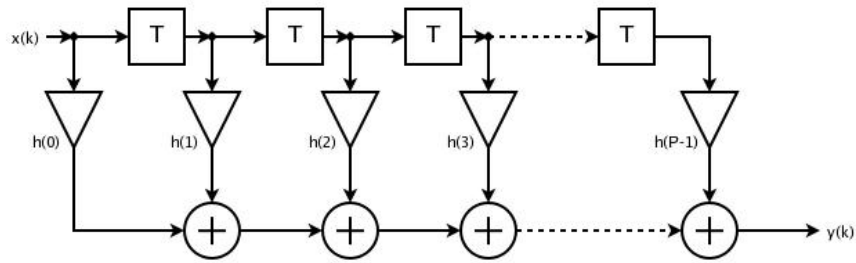


Figura 3.7: Filtro FIR com n andares.

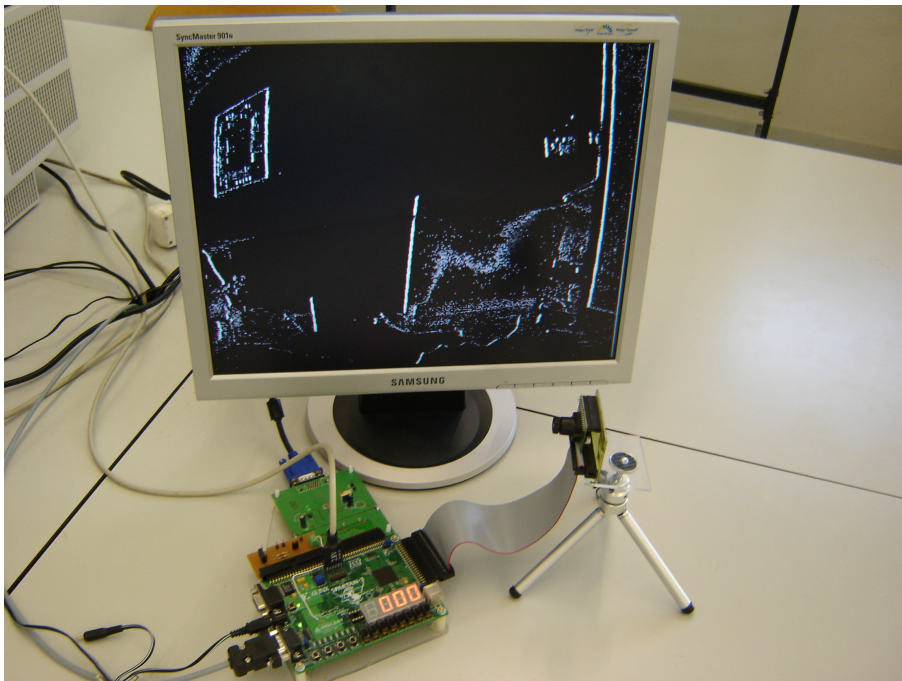


Figura 3.8: Cadeia de processamento de vídeo, com filtro definido para a detecção de contornos verticais.

3.4 Conclusão

Este capítulo descreve as formas de avaliação do consumo de potência praticadas durante a realização deste trabalho. Foram descritos ainda os circuitos auxiliares necessários para a correcta medição do consumo de potência, tais como os LFSRs e os geradores de paridade.

Os circuitos descritos na secção 3.3, serviram para apurar a eficácia das técnicas de redução do consumo de potência utilizadas. Foram utilizadas várias versões dos circuitos e para cada uma destas foram medidas as componentes dinâmica e estática da potência, com o objectivo de verificar se a diminuição da potência dinâmica era feita à custa do aumento da potência estática. A estimação do consumo de potência por software foi feita para determinar a precisão das estimativas.

Capítulo 4

Resultados Obtidos

4.1 Introdução

No capítulo anterior foi proposta uma metodologia experimental que permite avaliar o consumo de potência de circuitos digitais implementados em tecnologia FPGA. Os circuitos descritos foram implementados e ensaiados, utilizando para cada um várias versões, cada uma com diferentes codificações de estados ou andares de *pipeline*, conforme o tipo de circuito em questão. As várias versões de cada circuito foram implementadas tendo em vista manter o desempenho do circuito inicial, sendo o objectivo das várias versões, tornar possível a comparação do consumo de potência, em condições de funcionamento equivalentes. A síntese dos circuitos foi feita tendo como objectivo o desempenho do mesmo. As entradas e saídas dos circuitos ensaiados foram artificialmente reduzidas ao mínimo, utilizando um LFSR implementado na própria FPGA para gerar as entradas do circuito em estudo e um gerador de paridade que gera uma única saída, a partir das saídas efectivas do circuito, necessária para a correcta síntese do circuito. O motivo da minimização do número de entradas e saídas dos circuitos implementados na FPGA foi o de minimizar o efeito que as cargas capacitivas associadas aos *pads* desta pudesse ter no consumo de potência. Sendo as únicas ligações com o exterior a entrada do sinal de relógio e a respectiva entrada de habilitação e apenas uma saída representando a paridade da saída efectiva do circuito.

A secção 4.2 apresenta os resultados da implementação dos circuitos auxiliares constituídos pelo gerador de entradas aleatórias e gerador de paridade.

A avaliação do consumo de potência dos circuitos implementados foi feita recorrendo à estimação por *software* e à medição real tal como descrito no capítulo 3. Os resultados do consumo de potência obtidos são apresentados e analisados neste capítulo.

4.2 Circuitos auxiliares

Tal como referido na secção 3.2, foram utilizados circuitos auxiliares para permitir a medição do consumo de potência dos circuitos utilizados como casos de estudo segundo a metodologia proposta. O consumo de potência medido reflecte não só o consumo do circuito em teste, mas também o consumo do LFSR e do gerador de paridade. Como forma de conhecer o consumo de potência efectivo dos circuitos em estudo, os circuitos auxiliares foram caracterizados quanto ao consumo de potência, e quanto à ocupação de recursos na FPGA, por forma a subtrair estes valores aos correspondentes nos circuitos completos implementados.

A tabela 4.1 caracteriza os circuitos auxiliares quanto ao consumo de potência e quanto à ocupação de recursos. Cada um dos circuitos é referenciado através do circuito em que foi utilizado. Por exemplo, o circuito referenciado na tabela como "Multiplicador 16 bits", refere-se ao LFSR de 32 bits, 16 para cada uma das entradas, e um gerador de paridade com 32 entradas.

Circuito	Potência dinâmica (mW)	Área (CLBs)
Multiplicador 4 bits	0,324	118
Multiplicador 8 bits	0,300	141
Multiplicador 16 bits	0,312	182
Multiplicador 32 bits	0,276	256
Divisor 4 bits	1,044	120
Divisor 8 bits	1,164	151
Divisor 16 bits	1,092	200
Divisor 32 bits	1,224	290
Filtro FIR	0,852	152

Tabela 4.1: Consumo de potência dinâmica e ocupação de recursos da FPGA para os circuitos auxiliares utilizados em cada um dos casos de estudo.

4.3 Circuito *md4*

As diferentes versões do circuito *md4*, cujas funcionalidades e implementações foram descritas no capítulo 3, foram sintetizadas e implementadas. Seguidamente foram feitas as simulações *post-routing*, necessárias à estimação do consumo de potência. Não foram feitas quaisquer medições, pois a placa utilizada no âmbito deste trabalho está equipada com uma FPGA que não possui recursos suficientes para implementar este circuito. Como tal usou-se o Xpower para obter os consumos do circuito

Na tabela 4.2 estão apresentados os resultados da estimação do consumo de potência obtidos através do Xpower, bem como a área ocupada pelo circuito, para cada uma das codificações de estados disponibilizadas pelo XST: *User*, *One-hot*, *Compact*, *Sequential*, *Gray*, *Jonhson*, *Speed1*. As designações usadas na tabela são as mesmas utilizadas pela

ferramenta. Nas duas colunas da direita são apresentadas as variações percentuais do

Codificação	Potência dinâmica (mW)	Potência estática (mW)	Área (CLBs)	Δ Potência (%)	Δ Área (%)
<i>User</i>	96,21	398,86	7022	–	–
<i>One-hot</i>	85,41	398,96	6966	-11,23	-0,80
<i>Compact</i>	96,59	398,78	6966	+0,39	-0,80
<i>Sequential</i>	92,70	398,73	7056	-3,64	+0,48
<i>Gray</i>	87,31	398,53	7108	-9,25	+1,22
<i>Jonhson</i>	86,72	398,51	7311	-9,86	+4,12
<i>Speed1</i>	83,00	398,38	7232	-13,73	+2,99

Tabela 4.2: Resultados de implementação obtidos para as diferentes codificações de estados da versão inicial do circuito md_4 .

consumo de potência e da área ocupada relativamente à implementação inicial, referida na tabela como *User*. Nestas duas colunas um valor negativo significa uma diminuição do valor, potência ou área, ao passo que um valor positivo significa um aumento. A codificação denominada por *User* é a codificação usada inicialmente na descrição do circuito, que como foi referido, não fez uso de quaisquer critérios ou manipulações arquitecturais com o objectivo de baixar o consumo de potência.

A tabela 4.3 apresenta os resultados das implementações da segunda versão do circuito md_4 , mais uma vez para cada uma das diferentes codificações de estado. Os resultados estão dispostos tal como na tabela 4.2, pelo que explicação da organização, continua a ser válida para este caso. Porém a comparação de consumo do potência e da ocupação de área contida nas duas colunas da direita, foi feita em relação à implementação inicial do circuito, com a codificação de estados *User*. Optou-se por este critério, pois esta versão do circuito resultou da arquitectura do circuito inicial.

Codificação	Potência dinâmica (mW)	Potência estática (mW)	Área (CLBs)	Δ Potência (%)	Δ Área (%)
<i>User</i>	72,75	397,98	7649	-24,38	+9,56
<i>One-hot</i>	58,68	397,47	7441	-31,30	+6,81
<i>Compact</i>	63,37	397,64	7441	-34,38	+6,81
<i>Sequential</i>	63,37	397,64	7682	-31,,64	+8,87
<i>Gray</i>	58,54	397,47	7951	-32,95	+11,86
<i>Jonhson</i>	73,13	398,33	7347	-15,71	+0,49
<i>Speed1</i>	41,19	396,85	7785	-50,37	+7,69

Tabela 4.3: Resultados da implementação para as diferentes codificações de estados da segunda versão do circuito md_4 .

Pela análise dos resultados expressos nas tabelas apresentadas anteriormente, verifica-se que a codificação de estados mais eficiente no que respeita ao consumo de potência é a codificação *Speed1* [9], seguida de perto da codificação *One-hot*. A redução do consumo de

potência alcançadas por estas codificações para a versão inicial do circuito foi de 13,73% e de 11,23% respectivamente. Quanto à área ocupada pelo circuito, a codificação *Speed1* levou a um aumento de 2,99%, ao passo que no circuito com a codificação *One-hot* se verificou uma diminuição de 0,8%. A redução no consumo médio de potência das implementações da segunda versão do circuito face à primeira foi de 33,53%, para um aumento médio da área de circuito de 7,44%. A maior redução do consumo de potência, 50,37% foi obtida, com a conjugação da codificação de estados *Speed1* e da distribuição das somas por dois estados, com uma penalização da área ocupada pelo circuito de 7,44%. Em ambos os casos o consumo de potência estática manteve-se aproximadamente constante, apresentando uma leve variação desprezável face à variação observada no consumo de potência dinâmica. Deste modo foi obtida uma redução do consumo total de potência.

Estes resultados mostram a influência da codificação de estados no consumo de potência de um circuito. O caso do circuito *md4*, com uma máquina de estados caracterizada por possuir algumas dezenas de estados, mostra que a codificação de estados permite reduzir o consumo de potência do circuito em causa em cerca de 14%. A transformação aplicada à versão inicial do circuito, resultando numa máquina de estados com cerca de o dobro dos estados, permitiu ainda melhorar os resultados ao conseguir baixar o consumo de potência dinâmica para metade do gasto originalmente à custa de um acréscimo de cerca 7% na área ocupada pelo circuito.

4.4 Multiplicadores

A avaliação do consumo de potência dos circuitos multiplicadores foi realizada utilizando os dois processos, estimação e medição, descritos no capítulo 3. Desta forma poderá ser feita uma comparação entre os valores medidos e os valores estimados, podendo assim ser aferida a precisão da estimação.

A figura 4.1 apresenta a evolução do consumo de potência dinâmica medida para cada uma das versões do multiplicador (4, 8, 16 e 32 bits) em função do aumento do número de andares de *pipeline*, considerados em cada versão.

A análise de figura 4.1 permite verificar aspectos importantes. Em primeiro lugar, evidencia-se a acentuada diminuição do consumo de potência dinâmica do circuito multiplicador de 32 bits com o aumento do número de andares de *pipeline* usados. Por outro lado, é de salientar que nas versões de menores dimensões, os resultados foram diferentes. As diminuições no consumo foram menos expressivas, e no caso da versão mais reduzida (4 bits) verificou-se mesmo um ligeiro aumento. A explicação para esta diferença reside na dimensão dos circuitos. Por norma, circuitos maiores originam mais *glitches* e a sua propagação é feita por uma maior área. Assim, o consumo de potência dinâmica devido às *glitches* é também maior, e a sua redução origina resultados mais expressivos. No caso do aumento do consumo verificado nestas medições relativamente ao circuito de 4 bits, este deve-se ao reduzido consumo de potência originado por *glitches*, aliado ao aumento

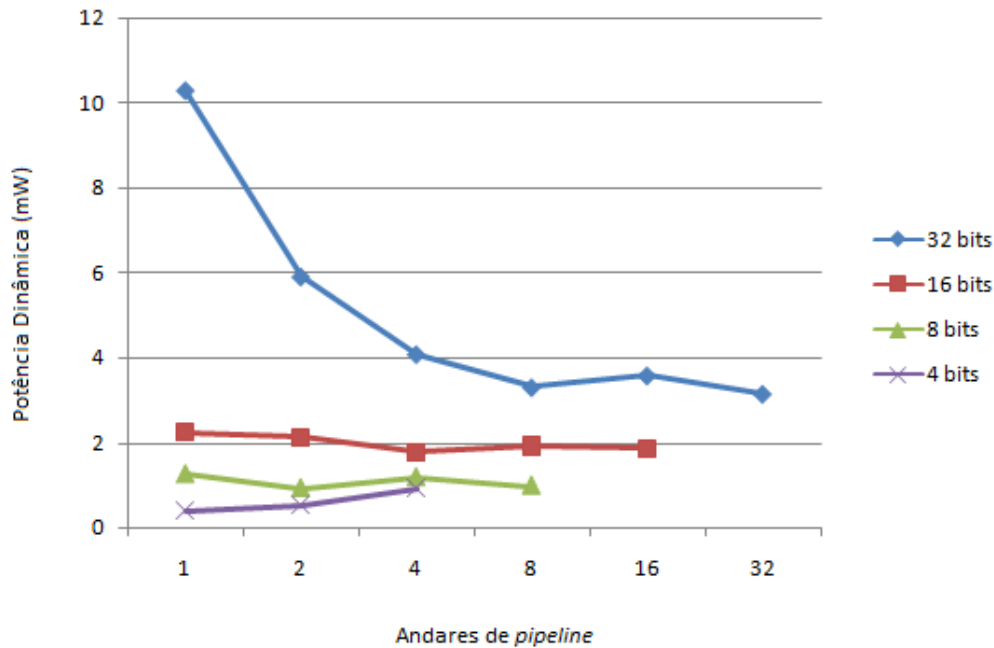


Figura 4.1: Consumo de potência dinâmica dos circuitos multiplicadores.

do consumo devido aos circuitos auxiliares utilizados para implementar o *pipeline*, tais como *flip-flops* e árvore de distribuição do sinal de relógio. Quanto à potência estática, não se verificou um aumento significativo do seu consumo, face à diminuição do consumo de potência dinâmica observada. A figura 4.2 são apresentadas as potências estática, dinâmica e total do circuito multiplicador de 32 bits em função do número de andares de *pipeline*. Verifica-se que o consumo de potência estática se mantém praticamente constante, e que devido à diminuição obtida na potência dinâmica, o consumo de potência total diminui.

Como foi visto, a redução do consumo de potência de um circuito é normalmente feita sacrificando o desempenho ou aumentando a utilização de recursos, traduzindo-se esta na utilização de mais blocos lógicos no caso das FPGAs. Uma técnica de redução do consumo de potência só é praticável se a sua implementação no circuito não tiver um consumo exagerado de recursos. A figura 4.3 apresenta a evolução da ocupação de recursos por cada versão implementada dos multiplicadores, expressa em blocos lógicos.

A ocupação de recursos dos circuitos multiplicadores mantém-se praticamente constante à medida que o número de andares de *pipeline* vai aumentando. Isto mostra as vantagens associadas à utilização desta técnica em circuitos baseados em FPGAs, tal como foi evidenciado no capítulo 3, devido aos *flip-flops* livres existentes nos blocos lógicos ocupados pelo circuito multiplicador. A figura 4.4 mostra a evolução da utilização dos *flip-flops* livres no circuito multiplicador combinacional de 32 bits, em função do aumento

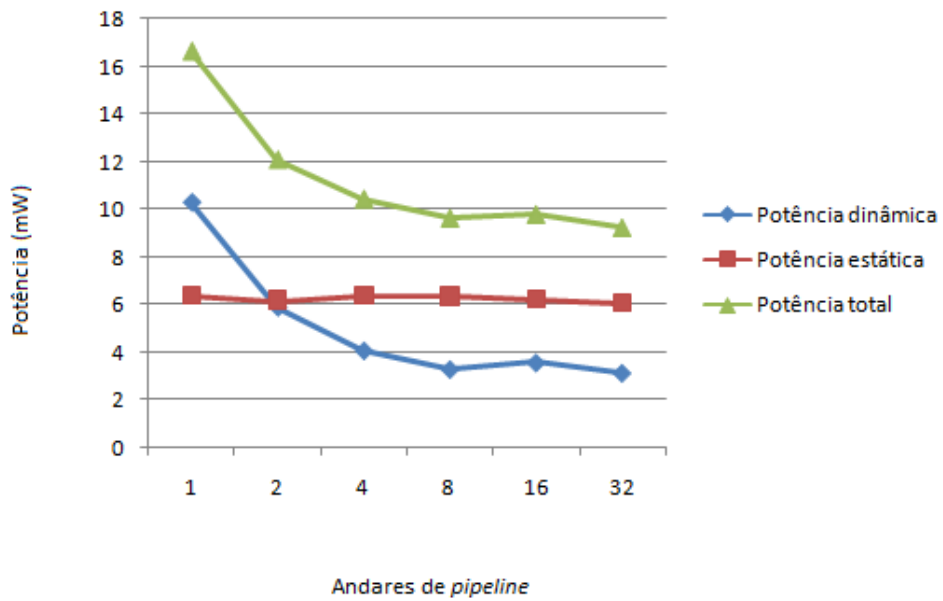


Figura 4.2: Potência estática, dinâmica e total do multiplicador de 32 bits.

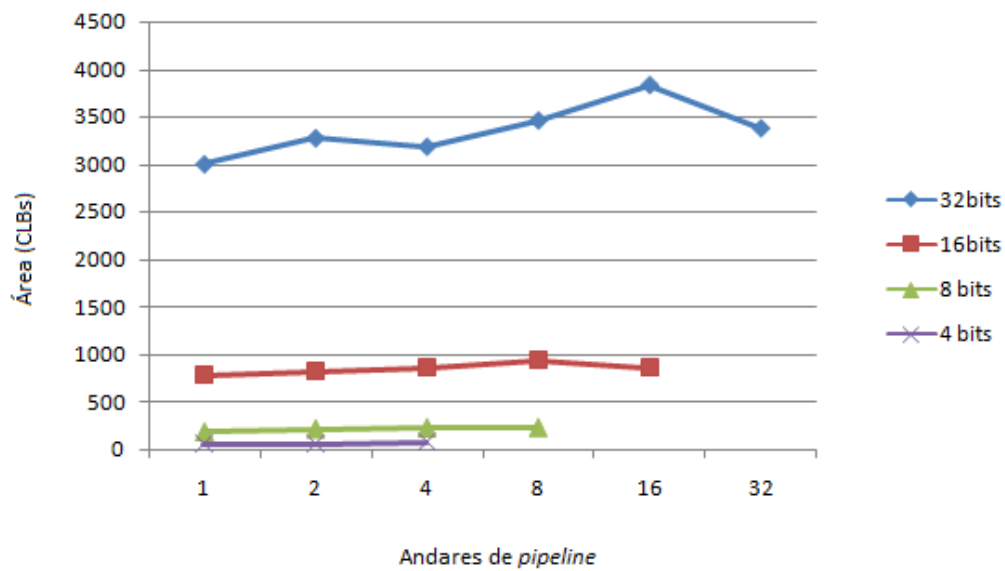


Figura 4.3: Ocupação de recursos dos circuitos multiplicadores.

do número de andares de *pipeline*. O decréscimo do número de *flip-flops* livres, significa que estes vão sendo aproveitados para implementar os andares de *pipeline*. No caso do multiplicador de 32 andares observa-se um valor negativo, ficando a dever-se à utilização de *flip-flops* extra para os implementar.

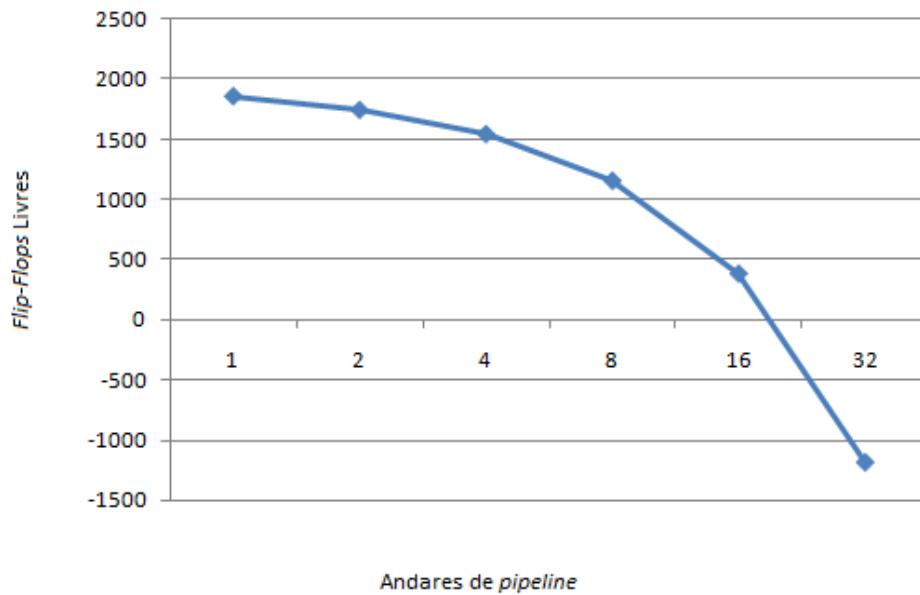


Figura 4.4: Evolução do número de *flip-flops* livres no multiplicador de 32 bits.

4.5 Divisores

O trabalho realizado para os circuitos multiplicadores e o respectivo procedimento foram repetidos para os circuitos divisores. Tal como nos multiplicadores foram feitas medições e estimativas do consumo de potência das várias versões dos circuitos considerando a aplicação de *pipelining*.

A figura 4.5 apresenta a evolução do consumo de potência dinâmica obtida por medição para cada uma das versões do divisor (4, 8, 16 e 32 bits) em função do número de andares de *pipeline*.

Mais uma vez se verifica que a arquitectura *pipelined* pode ser utilizada com êxito na redução do consumo de potência. Neste caso a redução do consumo obtida foi ainda mais expressiva do que no caso dos circuitos multiplicadores, sendo mais uma vez mais pronunciada nos circuitos de maior dimensão. Esta maior redução da dissipação de potência dinâmica, deve-se ao maior número de *glitches* que ocorrem e são propagados por este circuito. Note-se que no caso dos circuitos de 32 bits com 32 andares de *pipeline*, nos quais

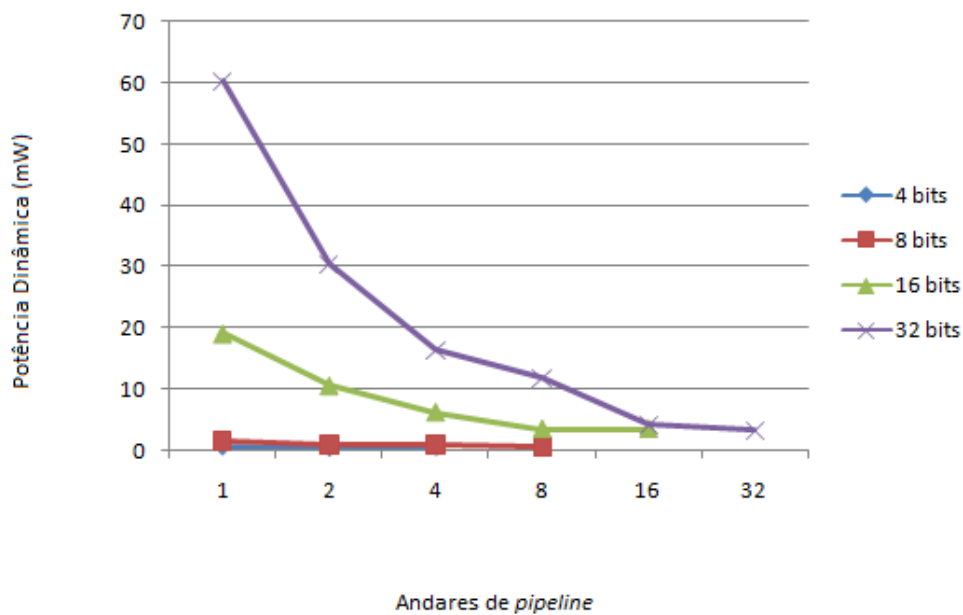


Figura 4.5: Consumo de potência dinâmica dos circuitos divisores.

a propagação de *glitches* é bastante reduzida o consumo de potência dinâmica é baixo, quando comparado com o consumo de versões de menor dimensão. Verifica-se ainda que nestas, os consumos dos circuitos são aproximadamente iguais, e que no caso das versões combinacionais, o circuito divisor consome cerca de seis vezes mais do que o circuito multiplicador. Mais uma vez o aumento do consumo de potência estática não foi significativo, quando comparado com a diminuição do consumo de potência dinâmica, permitindo assim a diminuição do consumo de potência total.

No que se refere à ocupação de recursos da FPGA, a evolução apresentada pelos dois tipos de circuitos é semelhante. A evolução da ocupação de recursos por parte dos circuitos divisores é apresentada na figura 4.6. Esta semelhança deve-se às arquitecturas utilizadas, as quais são compostas por um array de blocos de configuração idêntica. Tal como no caso dos multiplicadores, a aplicação da arquitectura *pipelined* aos circuitos divisores, tira partido dos *flip-flops* livres resultantes das implementações circuitos combinacionais. Assim, o custo de implementação da arquitectura *pipelined* em termos de recursos ocupados tem um efeito reduzido.

4.6 Filtros FIR

Foram considerados Filtros FIR com 8 andares, e com uma entrada e coeficientes de 8 bits. Foram implementadas duas versões deste circuito. Uma delas utilizando os blocos

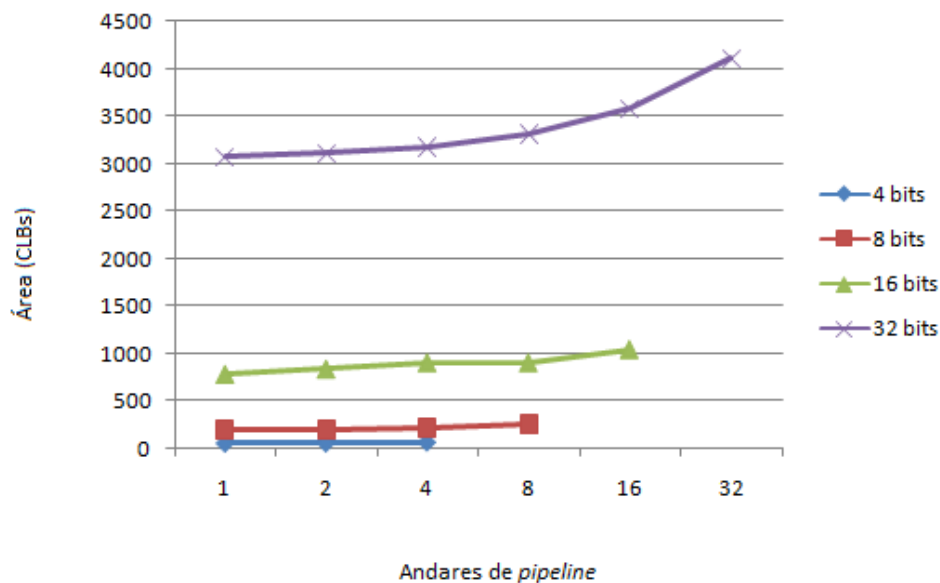


Figura 4.6: Ocupação de recursos dos circuitos divisores.

multiplicadores de 18 bits presentes na FPGA, e outra utilizando multiplicadores de 8 bits implementados a partir de blocos lógicos. A cada uma foram ainda adicionados andares e *pipeline*, colocados ao nível do filtro, entre os somadores.

4.6.1 Implementação de filtros FIR

A estratégia seguida nos casos anteriores voltou a ser aplicada no caso do estudo dos filtros FIR. O resultado das medições do consumo de potência dinâmica destes circuitos é apresentado na figura 4.7.

A análise destes resultados mostra que o consumo de potência dinâmica diminui com o aumento do número de andares de *pipeline*.

Outro aspecto que se não se pode ignorar é a diferença obtida entre os circuitos que utilizam os blocos multiplicadores presentes na FPGA utilizada, e os circuitos que utilizam os multiplicadores combinacionais de 8 bits. Os multiplicadores presentes na FPGA, à semelhança de outros blocos presentes neste e noutros modelos de FPGAs são semelhantes aos circuitos da tecnologia ASIC, não usam as extensas ligações entre blocos lógicos, pelo que têm um consumo de potência dinâmica e estática mais baixo, o que explica a redução do consumo de um circuito que os utilize. À semelhança do que foi observado nas secções anteriores relativamente a outros circuitos utilizados como casos de estudo, o aumento do consumo de potência estática observado nos circuitos com mais andares de *pipeline* não é

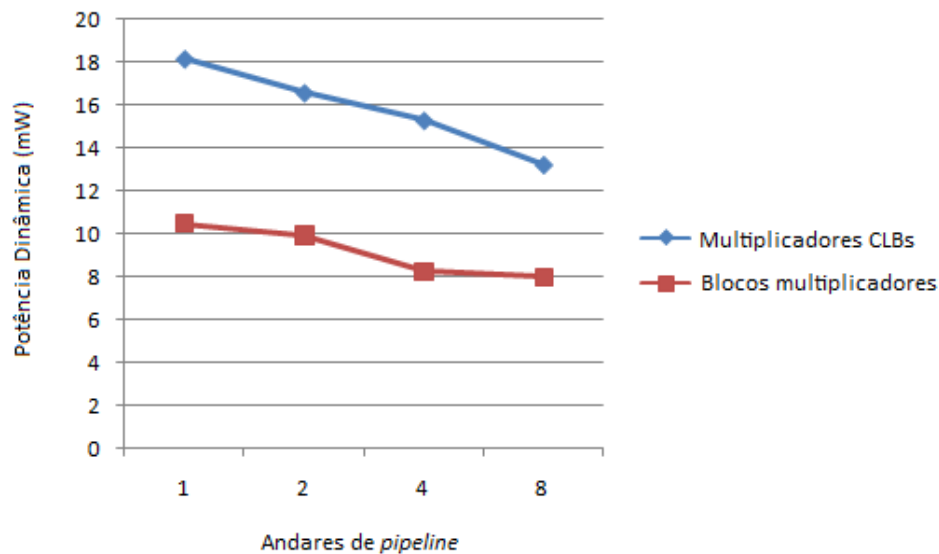


Figura 4.7: Consumo de potência dinâmica dos filtros.

significativo face à diminuição do consumo de potência dinâmica conseguida, resultando numa diminuição do consumo total de potência.

A figura 4.8 mostra a evolução da ocupação de recursos da FPGA, expressa em blocos lógicos, para as duas versões do filtro FIR, com o aumento do número de andares de *pipeline*. Também neste caso se verifica que o número de blocos lógicos utilizados pelos circuitos se manteve praticamente constante para as diferentes versões dos dois circuitos. No entanto existe uma diferença significativa entre os dois filtros. O filtro implementado à custa dos blocos multiplicadores presentes na FPGA necessitou de menos de metade dos blocos lógicos que o outro filtro utilizou. Este aspecto deve ser tido em consideração, uma vez que a utilização dos multiplicadores, ou outro qualquer bloco presente na FPGA, além de diminuir o consumo de potência, liberta blocos lógicos que assim podem ser utilizados para outros fins. Por outro lado, a implementação destes filtros, desperdiça parte de cada um dos multiplicadores, uma vez que estes são de 18 bits e o filtro utiliza multiplicadores de apenas 8 bits. Este desperdício pode ser quantificado numericamente, se for feita uma comparação entre a ocupação de recursos dos dois circuitos, expressa em portas lógicas equivalentes, uma vez que a contagem destas é feita considerando a utilização de multiplicadores de 18 bits e não de 8. tal como é mostrado na figura 4.9.

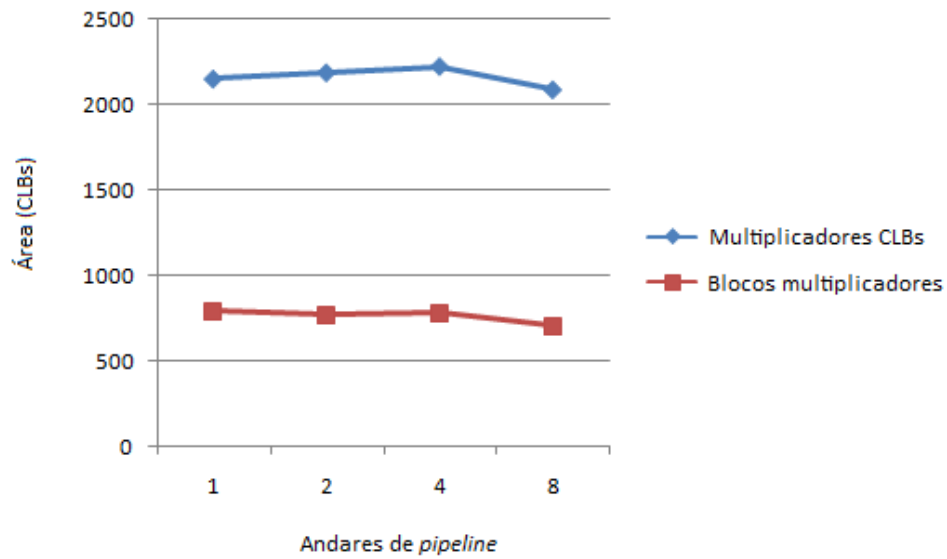


Figura 4.8: Ocupação de recursos dos filtros expressa em CLBs.

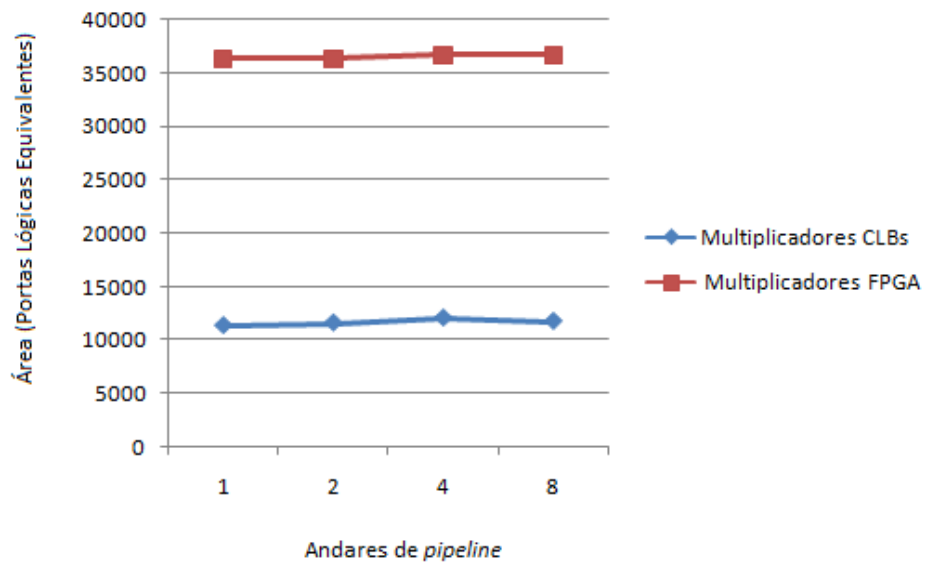


Figura 4.9: Ocupação de recursos pelos filtros em portas lógicas equivalentes.

4.6.2 Utilização de filtros FIR numa cadeia de processamento de vídeo

A utilização dos filtros FIR numa cadeia de processamento de vídeo permite estudar o consumo de potência do circuito inserido numa aplicação real, com ligações entre blocos lógicos potencialmente mais longas e com entradas de dados reais, neste caso imagens captadas por uma câmara digital.

O consumo do filtro foi obtido medindo o consumo total da cadeia de processamento de vídeo com o filtro incluído, subtraindo-lhe o valor do consumo da cadeia medido sem o filtro. Neste caso não foi possível utilizar a abordagem anterior para obter separadamente os valores da potência estática e dinâmica, pois uma vez que este circuito é mais complexo que os anteriores e utiliza circuitos externos síncronos com o mesmo sinal de relógio global, a utilização do *gated clock* que permitia desligar o sinal de relógio trazia problemas de funcionamento. Assim sendo, esta abordagem foi simplificada, referindo-se as medições efectuadas para estes circuitos ao consumo total do filtro. As imagens foram captadas num ambiente com diversidade de cor, mas com pouco movimento de modo manter as condições das várias medições tão estáveis quanto possível. Nestas condições não existe a aleatoriedade dos dados a operar provocada nos ensaios realizados antes desta aplicação, sendo de esperar um menor consumo de potência porque a actividade de comutação do circuito é menor.

A figura 4.10 apresenta o consumo de potência de duas versões do filtro, uma utilizando os blocos multiplicadores presentes na FPGA e a outra utilizando os multiplicadores implementados à custa de CLBs, cada uma usando vários níveis de andares de *pipeline* colocados entre os circuitos somadores. Os coeficientes destes filtros foram definidos de modo a realizar um filtro do tipo passa baixo. Para este efeito os coeficientes usados foram: 4, 28, 28, 28, 28, 28, 28 e 4.

Apesar dos consumos de potência obtidos para os filtros aplicados à cadeia de processamento de vídeo reflectirem os consumos dinâmicos e estáticos, estes são muito reduzidos, quando comparados com os consumos obtidos para os mesmos filtros com coeficientes e entradas aleatórias. O facto de os coeficientes serem constantes reduz por si só o consumo dinâmico, uma vez que assim diminuem-se as comutações de nível lógico. Por outro lado, uma imagem real não tem geralmente variações de cor abruptas, como aquelas que são aleatoriamente provocadas pelos geradores de vectores de entrada usados nos restantes ensaios. Desta forma a entrada do filtro pode manter-se praticamente constante, ou com poucas variações, o que também leva a uma redução do consumo de potência dinâmica.

A figura 4.11 apresenta o consumo de potência das duas versões do filtro, com os dois tipos de multiplicadores, e mais uma vez em função dos níveis de andares de *pipeline*. Os coeficientes dos filtros foram definidos para efectuar a detecção de contornos verticais, isto é, são salientados contornos verticais dos objectos presentes na imagem. Os coeficientes aplicados para este efeito foram: 64, -58, 0, 0, 0, 0 e 0.

A diminuição do consumo de potência neste caso foi ainda maior. A explicação para

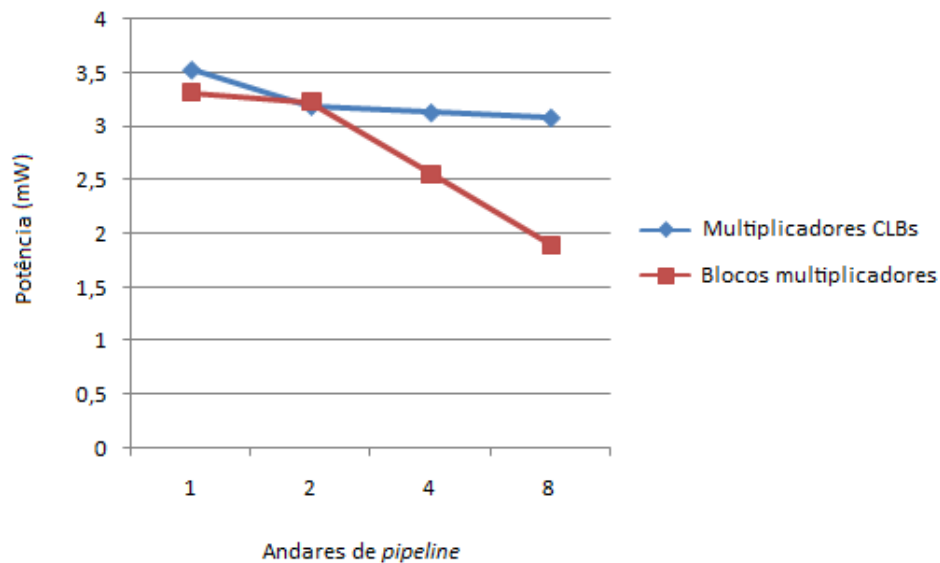


Figura 4.10: Consumo de potência dos filtros FIR utilizados numa cadeia de processamento de vídeo realizando a função de filtro passa-baixo.

esta redução atribui-se à quantidade de coeficientes com o valor zero, pois os multiplicadores associados a estes coeficientes não apresentam actividade, não contribuindo para o consumo de potência dinâmica.

4.6.3 Análise da avaliação do consumo de potência

É sabido que os resultados obtidos por estimação estão afectados por uma imprecisão que pode ser significativa [26]. Os resultados obtidos através da medição e da estimação, permitem aferir a grau de precisão do valor do consumo de potência estimado face ao valor real medido. O erro associado à medição é pequeno e pode ser desprezado face ao erro que pode afectar a estimativa [35]. Assim o valor medido pode ser tomado como o valor real do consumo de potência, ou seja, considerado o valor de referência.

A figura 4.12 apresenta a evolução do consumo de potência dinâmica obtido por estimação e por medição para as várias versões do filtro FIR implementado apenas à custa de CLBs. À semelhança da figura 4.12, a figura 4.13 apresenta o consumo de potência estática obtido por estimação e por medição para os mesmos circuitos.

As estimativas do consumo de potência dinâmica apresentadas na figura 4.12 são em média cerca de 24 % inferiores aos valores medidos. No caso do consumo de potência

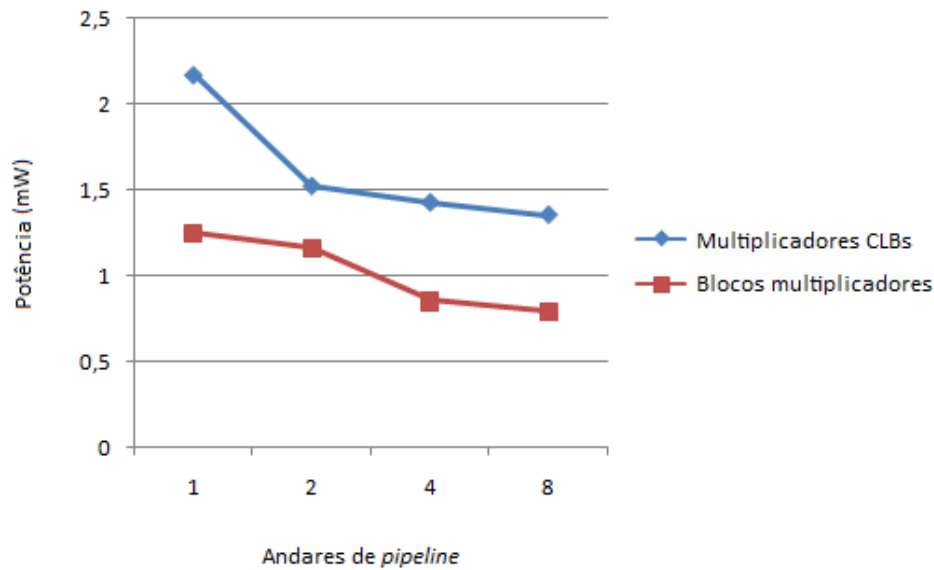


Figura 4.11: Consumo de potência dos filtros FIR utilizados na cadeia de processamento de vídeo para detecção de contornos verticais.

estática foi observado um erro cometido por excesso de cerca de 55%. Embora não se pretenda com isto fazer uma generalização, os erros cometidos nas estimativas da potência consumida pelos filtros, aqui apresentados, são semelhantes aos erros cometidos nas estimativas feitas para os outros circuitos apresentados. Apesar da imprecisão observada nos valores obtidos por estimação, a utilização do Xpower continua a ser justificável, devido às vantagens que apresenta face ao processo de medição real, devido nomeadamente à facilidade com que podem ser obtidos resultados durante a fase de projecto. Apesar do erro pelo qual estão afectadas as estimativas, as conclusões que possam ser tiradas quanto à diminuição do consumo de potência através da sua análise mantêm-se válidas.

4.7 Conclusão

Neste capítulo foram apresentados e analisados os resultados da avaliação em termos do consumo de potência e da ocupação de recursos da FPGA dos circuitos utilizados como casos de estudo. Nestes circuitos foram usadas técnicas de redução do consumo de potência ao nível do projecto RTL. Mais concretamente, foram exploradas a codificação de estados e a arquitectura *pipelined*. Devido ao facto de se ter usado circuitos auxiliares, começou-se por apresentar a caracterização destes circuitos utilizados para permitir a medição do

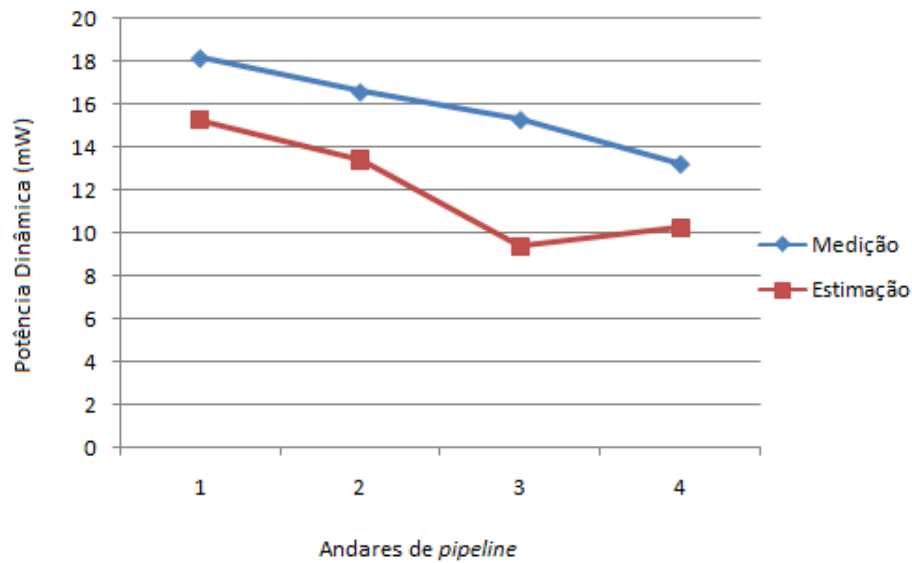


Figura 4.12: Consumo de potência dinâmica do filtro FIR obtida por medição e por estimação

consumo de potência dos circuitos em estudo em condições de funcionamento normais e com o mínimo de influência de circuitos externos à FPGA.

O estudo do circuito *md4* mostrou a influência que a codificação de estados pode ter no consumo de potência dinâmica de um circuito. Os ensaios realizados consistiram na implementação do circuito utilizando várias codificações de estados, verificando-se que a maior redução do consumo de potência foi obtida com a utilização da codificação *Speed1*. Foram ainda implementadas novas versões do circuito caracterizadas por possuírem mais estados, e por as operações feitas num dos estados anteriores, serem agora feitas em dois, tendo sido alcançada uma maior diminuição do consumo de potência face à implementação inicial.

A aplicação de arquiteturas *pipelined* aos circuitos multiplicadores revelou-se, vantajosa quanto à redução do consumo de potência. Esta redução depende da dimensão do circuito, sendo tanto maior, quanto maior o circuito. Na versão com menores dimensões foi observado um ligeiro aumento da potência uma vez que a redução obtida foi inferior ao consumo adicional introduzido pela inserção dos registos de *pipeline*. Foi ainda observada para o multiplicador de 32 bits, a evolução da utilização dos *flip-flops* que são deixados livres na versão combinacional do circuito e que são utilizados pelos registos das versões *pipelined*. O estudo dos circuitos divisores seguiu os mesmos passos do estudo dos multiplicadores. Os resultados obtidos para estes circuitos reflectem o que foi concluído para

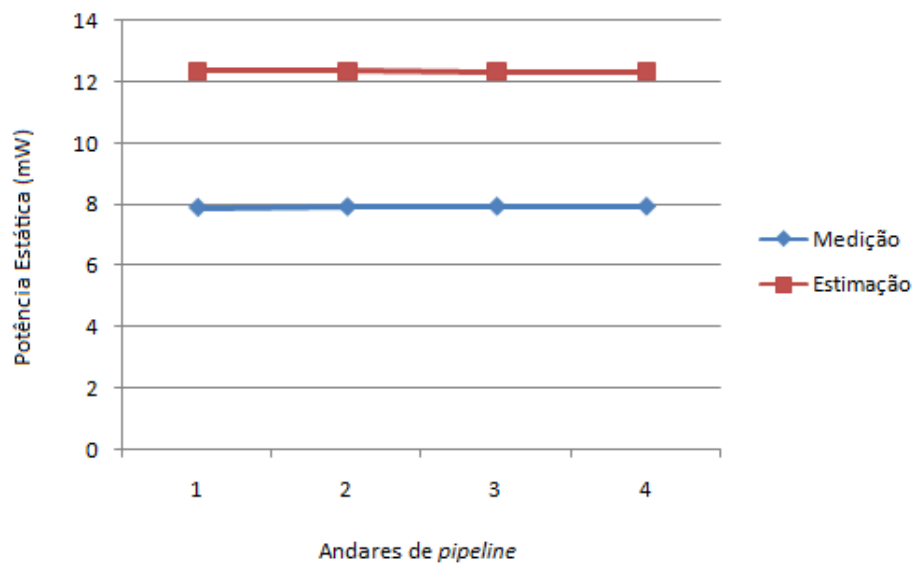


Figura 4.13: Consumo de potência estática do filtro FIR obtida através de medição e de estimação.

os circuitos multiplicadores.

Os filtros do tipo FIR foram aqui abordados sobretudo devido à sua frequente utilização, e também devido à possibilidade de os utilizar integrados numa aplicação real, utilizando dados reais. Neste contexto foram seguidas duas abordagens distintas. Numa delas a implementação dos filtros recorreu aos blocos multiplicadores presentes na FPGA e na outra utilizam-se multiplicadores realizados à custa de CLBs. Os resultados obtidos mostram a economia de potência que se pode obter com a utilização de circuitos específicos presentes na FPGA. Concluiu-se ainda, a partir do consumo de potência do filtro quando inserido na cadeia de processamento de vídeo, que a utilização de dados gerados de forma aleatória leva a um valor do consumo de potência mais elevado do que aquele que é observado na implementação real. Este é um aspecto significativo que merece ser realçado, pois a medição da potência pode conduzir a valores por excesso, tal como acontece neste caso. Conclui-se daqui que é fundamental possuir informação acerca dos dados a operar por um circuito para que as medições da potência consumida sejam o mais fiáveis possível.

A par da redução do consumo de potência, a utilização de recursos da FPGA também foi avaliada neste capítulo. Foram verificados ligeiros aumentos da utilização dos recursos pelos circuitos com *pipeline*, tal como era de esperar, o que se explica pela necessidade de *flip-flops* e de interligações adicionais entre blocos lógicos.

Nos vários ensaios realizados, verificou-se que a componente estática da potência, se

mantém praticamente constante, apesar dos aumentos da utilização de recursos. Para todos os ensaios, ponderando ambas as componentes da potência, estática e dinâmica, obtem-se uma redução do consumo total.

Capítulo 5

Conclusão

Neste trabalho de dissertação foi estudado o consumo de potência em circuitos implementados em tecnologia FPGA, e formas de, ao nível de projecto RTL, o diminuir.

Foi mostrado que o consumo de potência em circuitos digitais CMOS se divide em quatro componentes: potência dinâmica, potência de curto-circuito, potência estática e potência de fugas. E que estas podem classificar-se em apenas dois grupos: as de natureza estática, ou seja independentes do sinal de relógio, e as de natureza dinâmica, que dependem do sinal de relógio. Quanto a dispositivos do tipo FPGA, foi referido, que devido a serem sistemas reconfiguráveis, apresentam um elevado consumo de potência, tendo a componente estática um peso mais significativo do que em circuitos semelhantes implementados na forma de tecnologia ASIC. Foi também concluído que ao nível do projecto RTL apenas é possível diminuir a componente dinâmica do consumo de potência.

Após uma análise de diversas técnicas de projecto orientadas para a redução do consumo de potência, verificou-se que a mais promissora para aplicar a circuitos implementados em dispositivos FPGA, é a baseada em arquitecturas *pipelined*. Apesar de o *pipeline* ser uma técnica que permite aumentar o desempenho de um circuito, pode ser usada também como forma de conseguir reduzir o consumo de potência desse circuito. Os *flip-flops* presentes nas saídas dos blocos lógicos de que compõem as FPGAs, e que ficam livres em circuitos de natureza combinacional, podem ser usados para implementar arquitecturas *pipelined* sem ocupar recursos adicionais. Outra técnica explorada foi a codificação de estados, em circuitos com máquinas de estados finitos, sobretudo pela facilidade com que pode ser aplicada e pelos resultados que permite obter.

No capítulo 3 foram apresentadas formas de avaliar o consumo de potência por estimação com base na ferramenta Xpower, e por medição real através da observação da corrente consumida pelo circuito. Com o objectivo de permitir a medição do valor de potência consumida por um circuito foi proposta uma metodologia experimental. A abordagem utilizada neste método consistiu na utilização de uma montagem baseada na placa

de prototipagem, modificada de modo a permitir a medição da corrente consumida pelo núcleo da FPGA, e através deste valor conhecer a potência. O funcionamento do circuito em termos de componente dinâmica, durante a medição e a minimização das entradas e saídas foi garantido com a utilização de circuitos auxiliares. As entradas do circuito em teste foram geradas através de um LFSR, que permite gerar sequências aleatórias, implementado na própria FPGA. As saídas efectivas do circuito foram encaminhadas para um gerador de paridade que assim permitiu a implementação do circuito ensaiar com apenas uma saída. Estas intervenções justificam-se pelas consequências que resultariam se fosse usado um elevado número de entradas e saídas do circuito, devido às cargas introduzidas pela utilização dos *pads* da FPGA. A medição de ambas as componentes da potência, dinâmica e estática, foi conseguida, com a utilização de um sinal de habilitação do sinal de relógio. Bloqueando a actividade do sinal de relógio em todo o circuito, a potência medida é apenas a componente estática. O consumo dinâmico, foi obtido pela diferença entre o consumo total e o consumo estático. Foram ainda descritos no capítulo 3, os circuitos que foram utilizados como casos de estudo neste trabalho.

A codificação de estados foi ensaiada num dos casos de estudo, o circuito *md4*, composto por uma máquina de estados com cerca de 50 estados e vários circuitos lógicos e aritméticos. O circuito foi implementado várias vezes, utilizando diferentes codificações de estados. Numa nova realização do circuito, as operações aritméticas feitas em cada estado foram divididas por dois estados, elevando para cerca do dobro o número de estados necessários. Os valores do consumo de potência, obtidos por estimação, revelaram que a codificação que levou à maior diminuição consumo de potência dinâmica, foi a codificação *Speed1*, e que a versão do circuito com mais estados, implementada também com esta forma de codificação de estados, permitiu uma diminuição de cerca de 50% do consumo de potência dinâmica à custa de um aumento de cerca de 7% da utilização de recursos. A variação do consumo estático não foi significativa, tendo sido por isso conseguida a diminuição do consumo total de potência.

Por representarem uma classe importante de circuitos, dada a frequência com que são usados em aplicações específicas em FPGAs, os circuitos multiplicadores e divisores foram usados neste trabalho. Concretamente, os circuitos multiplicadores e divisores foram explorados no sentido de se implementarem com arquitecturas *pipelined*. Estes circuitos implementados inicialmente como combinacionais, sofreram modificações de modo a inserir andares de *pipeline*. Foram considerados diferentes dimensões e com diferentes níveis de *pipeline*. O consumo de potência destes circuitos foi avaliado utilizando a metodologia proposta. Os resultados obtidos mostram que o uso de arquitecturas do tipo *pipeline* permitem diminuir significativamente o consumo de potência dinâmica em circuitos de grandes dimensões. Em circuitos mais pequenos a diminuição é menos expressiva ou mesmo inexistente. A variação verificada em termos de potência estática não foi significativa, pelo que no total, o consumo de potência foi diminuído. Verificou-se ainda que as arquitecturas do tipo *pipeline* permitem diminuir o consumo de potência dinâmica, à custa de um ligeiro

aumento da utilização de recursos da FPGA.

A utilização de filtros do tipo FIR além de comprovar a eficácia das arquitecturas *pipelined*, permitiu verificar a influência da utilização de blocos multiplicadores presentes na FPGA. Além disso, a actividade do filtro quando inserido numa aplicação real, a operar com dados reais conduz a um consumo de potência que é inferior ao medido quando o filtro opera com entradas geradas aleatoriamente. Os resultados do consumo de potência obtidos mostram que os circuitos presentes na FPGA são mais eficientes, relativamente ao consumo de energia, quando comparados com circuitos implementados à custa de blocos lógicos, uma vez que os primeiros não sofrem dos problemas associados à reconfigurabilidade. O consumo de potência do filtro inserido na cadeia de processamento de vídeo, a funcionar com imagens reais, mostrou o erro que pode ser cometido ao usar dados aleatórios, uma vez que o consumo neste caso foi significativamente inferior ao obtido com a utilização de dados aleatórios. Mais uma vez, as diminuições conseguidas ao nível do consumo dinâmico, não afectaram significativamente o consumo estático, pelo que o consumo de potência total foi diminuído.

A precisão das estimativas obtidas com o Xpower foi analisada com base nos resultados recolhidos para os filtros. Após a análise verifica-se que as estimativas podem conter erros grosseiros, no entanto a sua utilização justifica-se pela facilidade e rapidez com que permite obter resultados durante a fase de projecto.

Anexo A

Tutorial Xpower

O Xpower é uma ferramenta informática que permite estimar o consumo de potência de circuitos implementados em FPGAs da Xilinx. Neste apêndice é descrito o procedimento para efectuar a estimação do consumo de potência.

A.1 Descrição, validação e síntese do circuito

A estimação do consumo de potência é feita com base na síntese da descrição do circuito. O circuito é normalmente descrito, utilizando uma linguagem de descrição de *hardware*, por exemplo o *Verilog*.

A aplicação utilizada para o desenvolvimento dos circuitos baseados em FPGAs da Xilinx é o ISE [9]. A descrição do circuito deve ser devidamente validada e em seguida deve ser feita a síntese e a implementação.

A síntese e implementação são feitas no ambiente de desenvolvimento da Xilinx, seleccionando a opção `Shyntesis / Implementation` no menu `sources for:` e seleccionando a função `Place & Route` no menu `Implementation Design`. Tal como na figura A.1, na qual é apresentada a janela de processos do ambiente de desenvolvimento ISE.

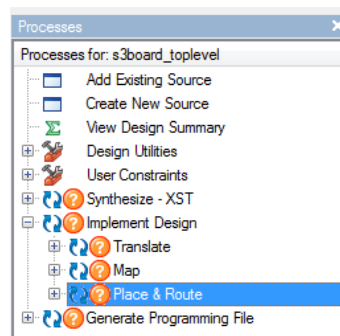


Figura A.1: Janela de processos do ISE onde se pode iniciar a etapa *place & route*.

A.2 Simulação *post-routing*

A simulação *Post-Routing* é uma simulação feita tendo em consideração a implementação do circuito na FPGA, e não apenas o seu modelo abstracto. Esta simulação permite gerar o ficheiro (*.vcd) que contem as indicações das taxas de actividade de cada um dos nós do circuito, definidas a durante a simulação. É feita com a ferramenta de simulação, seleccionando a opção **Simulate Post-Place & Route Model** no menu **Sources for:** e activando a opção **Post-Route Simulation**. A figura A.2 apresenta a janela de processos do ISE, com a opção **Post-Route Simulation**. A bancada de testes utilizada deve ser definida de forma a reproduzir o mais fielmente possível o funcionamento do circuito. Só assim se pode obter resultados realistas.

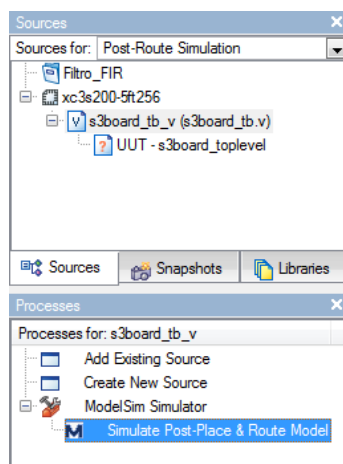


Figura A.2: Janela de processos do ISE onde se pode iniciar a etapa *Post-Route Simulation*.

Para que seja gerado o ficheiro *.VCD, é necessário activar a opção **Generate VCD File** na janela de propriedades da *Post-Route Simulation*, tal como mostra a figura A.3. Ainda nesta janela é possível seleccionar o tempo de simulação. Deve ser tido em atenção que quanto maior for o tempo de simulação, em princípio mais realistas seram os dados obtidos. Por outro lado, uma vez que a esta simulação é um processo computacionalmente pesado, um tempo de simulação do funcionamento do circuito exagerado pode levar que a duração da simulação se torne impraticável.

A.3 Estimação

Após a simulação, seleccionando novamente a opção **Synthesis/Implementation** no menu **sources for:**, e abrindo o menu **Implementation Design, Place & Route**, está disponível a opção **Analyze Power (Xpower)**, que permite abrir a aplicação Xpower.

Depois de aberta a aplicação, deve ser indicada a localização do ficheiro *.VCD. Este pode ser seleccionado através do menu **File, Open Simulation File**. Em seguida a estimativa do consumo de potência é apresentada tal como mostra a figura A.4. Do

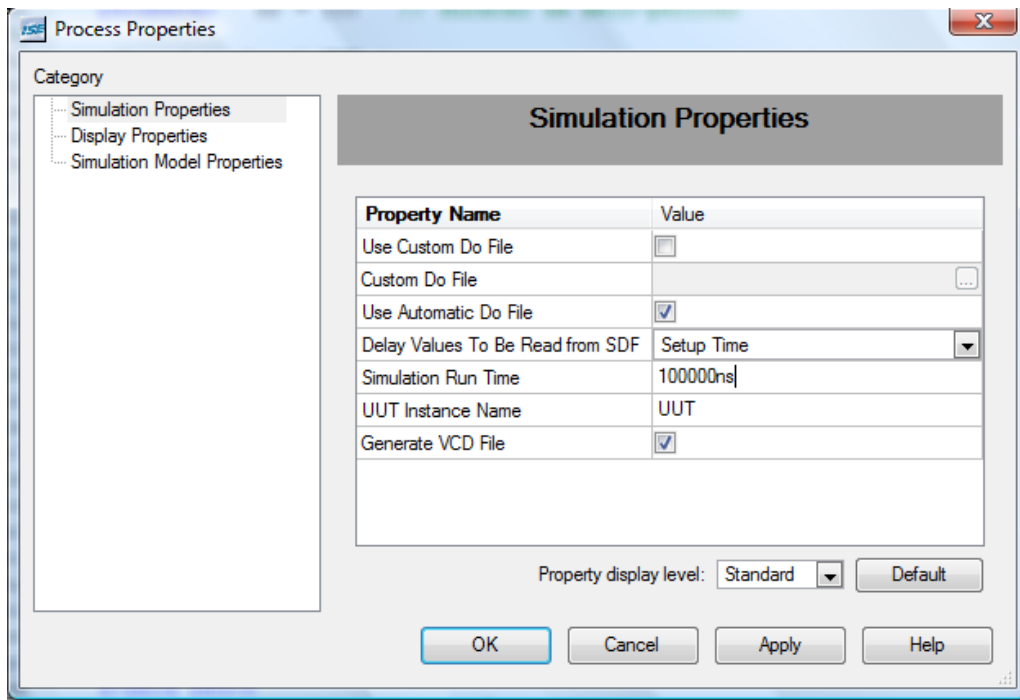


Figura A.3: Janela de propriedades da *Post-Route Simulation*.

lado esquerdo da janela da aplicação são apresentados os consumos de potência estática, dinâmica e total. Do lado esquerdo é possível observar características como frequência ou corrente dos sinais de cada um dos nós, que podem ser seleccionados abrindo **Data Views**.

O resultado obtido pode ser guardado sob a forma de ficheiro. Para isso basta abrir o menu **File** e seleccionar a opção **Save Settings As**.

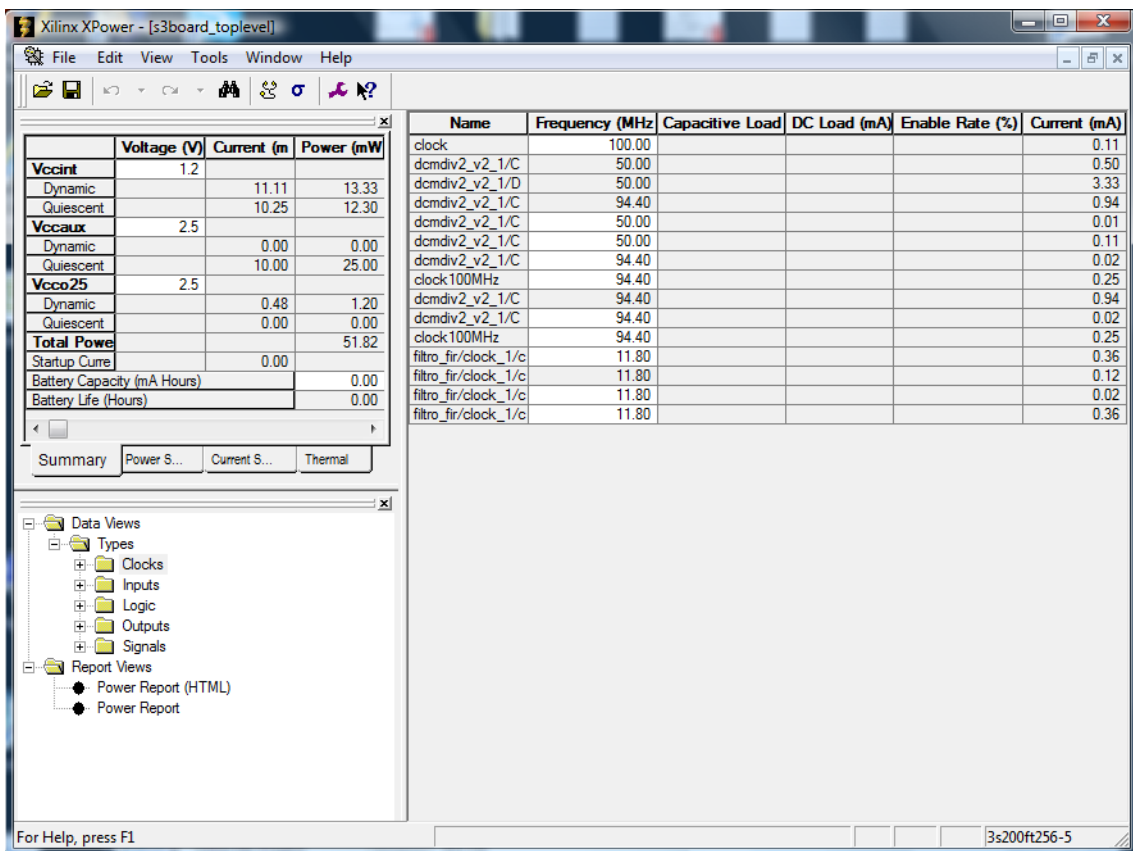


Figura A.4: Ambiente gráfico do Xpower.

Referências

- [1] Paul J.M. Havinga and Gerard J.M. Smit. Design techniques for low-power systems. *J. Syst. Archit.*, 46(1):1–21, 2000.
- [2] Andrés David García García, Luis Fernando González Pérez, and Reynaldo Felix Acuña. Power consumption management on FPGAs. In *CONIELECOMP '05: Proceedings of the 15th International Conference on Electronics, Communications and Computers*, pages 240–245, Washington, DC, USA, 2005. IEEE Computer Society.
- [3] Amara Amara, Frédéric Amiel, and Thomas Ea. FPGA vs. ASIC for low power applications. *Microelectronics Journal*, (37):669–677, 2006.
- [4] Eric Kusse and Jan Rabaey. Low-energy embedded FPGA structures. In *Proceedings of the 1998 International Symposium on Low Power Electronics and Design (ISLPED'98)*, pages 155–160, New York, NY, USA, 1998. ACM.
- [5] The Institute of Electrical and Electronics Engineers, New York. *IEEE Standard Verilog hardware description language*, September 1995. IEEE Standard 1364-1995.
- [6] The Institute of Electrical and Electronics Engineers, New York. *IEEE Standard Verilog hardware description language*, September 2001. IEEE Standard 1364-2001.
- [7] The Institute of Electrical and Electronics Engineers, New York. *IEEE Standard Verilog hardware description language*, September 2005. IEEE Standard 1364-2005.
- [8] MentorGraphics. *ModelSim Reference Manual*, 2007.
- [9] Xilinx. *Development System Reference Guide 9.1i*.
- [10] The Institute of Electrical and Electronics Engineers, New York. *IEEE Standard JTAG Test Access Port and Boundary-Scan Architecture*, 1990. IEEE Standard 1149.1-1990.
- [11] Anantha P. Chandrakasan, Samuel Sheng, and Robert W. Brodersen. Low power CMOS digital design. In *IEEE Journal of Solid-State Circuits*, volume 27, pages 473–484, 1992.
- [12] Adel S. Sedra and Kenneth C. Smith. *Microelectronic circuits, 2nd ed.* Holt, Rinehart & Winston, Austin, TX, USA, 1987.
- [13] K Roy, S Mukhopadhyay, and H Mahmoodi-Meimand. Leakage current mechanisms and leakage reduction techniques in deep-submicrometer CMOS circuits. *Proceedings of the IEEE*, 91(2), 2003.

- [14] James Kao, Siva Narendra, and Anantha Chandrakasan. Subthreshold leakage modeling and reduction techniques. In *Proceedings of the 2002 IEEE/ACM International Conference on Computer-Aided Design (ICCAD '02)*, pages 141–148, New York, NY, USA, 2002. ACM.
- [15] Li Shang, Alireza S. Kaviani, and Kusuma Bathala. Dynamic power consumption in Virtex-II FPGA family. In *Proceedings of the 2002 ACM/SIGDA Tenth International Symposium on Field-Programmable Gate Arrays (FPGA '02)*, pages 157–164, New York, NY, USA, 2002. ACM.
- [16] Nathaniel H. Rollins. Reducing power in FPGA designs through glitch reduction. Master's thesis, Brigham Young University, 2007.
- [17] Anantha P. Chandrakasan and Robert W. Brodersen. Minimizing power consumption in digital CMOS circuits. In *Proceedings of the IEEE*, volume 83, pages 498–523, 1995.
- [18] Sanjiv Kumar Mangal, Raghavendra B. Deshmukh, Rahul M. Badghare, and R. M. Patrikar. FPGA implementation of low power parallel multiplier. In *Proceedings of 20th International Conference on VLSI Design*, 2007.
- [19] Steven J.E. Wilton, Su-Shin Ang, and Wayne Luk. The impact of pipelining on energy per operation in field-programmable gate arrays. *Lecture notes in computer science*, 3203:719–728, 2004.
- [20] Gustavo Sutter and Eduardo Boemo. Experiments in low power FPGA design. *Latin America Applied Research*, 37(1):99–104, 2007.
- [21] Ian Brynjolfson and Zeljko Zilic. Dynamic clock management for low power applications in FPGAs. In *Proceedings of the Custom Integrated Circuits Conference*, pages 39–142, Orlando, FL, USA, May 2000.
- [22] Gustavo Sutter, Elias Todorovich, and Eduardo Boemo. Design of power aware FPGA-based systems. In *Jornadas de Computación Reconfigurable y Aplicaciones*, September 2004.
- [23] Gustavo Sutter, Elias Todorovich, Lopez Buedo, and Eduardo Boemo. Low-power FSMs in FPGA: Encoding alternatives. *Lecture Notes in Computer Science*, 2451:363–370, 2002.
- [24] Gustavo Sutter, Elias Todorovich, Lopez Buedo, and Eduardo Boemo. FSM decomposition for low power in FPGA. *Lecture Notes in Computer Science*, 2438:350–359, 2002.
- [25] Tim Tuan, Sean Kao, Arif Rahman, Satyaki Das, and Steve Trimberger. A 90nm low-power fpga for battery-powered applications. In *FPGA '06: Proceedings of the 2006 ACM/SIGDA 14th international symposium on Field programmable gate arrays*, pages 3–11, New York, NY, USA, 2006. ACM.
- [26] Tim Tuan and Steve Trimberger. The power of FPGA architectures, the present and future of low-power FPGA design. *Xcell Journal*, Second Quarter:12–15, 2007.
- [27] Subodh Gupta and Jason Andreson. Optimizing FPGA power with ISE design tools. *Xcell Journal*, Second Quarter:16–19, 2007.

- [28] Eduardo Boemo, Guillermo Rivera, Lopez buedo, and Juan Meneses. Some notes on power management on FPGA-based systems. *Lecture Notes in Computer Science*, 975:149–157, 1995.
- [29] Xilinx. *Xpower Tutorial: FPGA Design*, 2002.
- [30] Xilinx. *Spartan-3 Starter Kit Board User Guide*, 2005. <http://www.digilentinc.com>.
- [31] Xilinx. *Datasheet Spartan3*, 2006.
- [32] Fairchild. *Datasheet FAN1112*, 2001.
- [33] R. Rivest. The md4 message-digest algorithm, 1992.
- [34] António J. Araújo and José C. Alves. A project driven digital design course using FPGAs. In *Proceedings of the 19th. International Conference of the European Association for Education in Electrical and Information Engineering (EAEEIE'2008)*, Tallinn, Estonia, June 2008.
- [35] Deming Chen, Jason Cong, and Yiping Fan. Low-power high-level synthesis for FPGA architectures. In *Low Power Electronics and Design*, pages 134–139, 1, August 2003.