

Ferramentas Práticas de Baixo Custo para Desenvolvimento de Sistemas baseados em FPGA's

Manuel Almeida, Bruno Pimentel

Universidade de Aveiro/ IEETA, Campus Univ. de Santiago 3810-193, Aveiro, Portugal
manuel.almeida@ieeta.pt, pimentel@ieeta.pt

Resumo

O presente artigo apresenta as características e potencialidades de um novo conjunto de ferramentas práticas de baixo custo, para o desenvolvimento de hardware com base em FPGA's. Muitas placas de prototipagem de hardware existentes no mercado integram uma vasta gama de dispositivos periféricos e interfaces que, podendo nem sequer chegar a ser utilizados, aumentam o seu preço significativamente. Utilizando uma abordagem alternativa, foi desenvolvida uma placa de prototipagem que integra apenas os dispositivos essenciais a uma utilização prática e eficiente da sua funcionalidade e à sua expansibilidade.

A arquitectura base implementada inclui uma FPGA, uma CPLD, uma memória *flash* e um módulo USB. Este último é utilizado para gestão a nível de utilizador e alimentação da placa, a partir de um computador, e também para acoplamento de periféricos. Foi desenvolvido software para gerir a placa, permitindo o envio de bitstreams e a transferência bidireccional de dados de utilizador (com base na memória *flash*.) e para funcionar como periférico de entrada e saída. As ferramentas desenvolvidas podem ser utilizadas na implementação de sistemas que integrem várias placas e técnicas de partição e delegação de tarefas, para dar resposta a problemas concretos de maior complexidade.

Palavras-chave: Computação Reconfigurável, FPGA's, Placa de Prototipagem, Interface USB, Interface Bluetooth.

Abstract

This paper presents the characteristics and potential of a new low-cost set of practical tools for FPGA-based hardware development. Many available hardware prototyping boards include a wide set of peripherals and interfaces which significantly increase their cost and might even not get used once. Using an alternative approach, a prototyping board was developed, integrating solely the devices essential for an easy and efficient use of its functionality and for achieving expansibility.

The base architecture implemented includes an FPGA, a CPLD, a flash memory and a USB module. The latter is utilized for user-level management of the board and as a power source, using a computer, and also for attaching peripherals. Software tools were developed for managing the board, allowing to send bitstreams and two-way user-data transfer (making use of the flash memory). This software can also function as an input/output peripheral. The developed set of tools can be utilized for implementing systems made up of more than one board, integrating partition and task delegation techniques. The goal is to build solutions for specific problems of higher complexity.

Key-words: Reconfigurable Computing, FPGA's, Prototyping Board, USB Interface, Bluetooth Interface.

1 Introdução

Os dispositivos reconfiguráveis constituem uma ferramenta rápida e versátil para prototipagem de hardware no âmbito da indústria, para o desenvolvimento de novas soluções e o auxílio em tarefas computacionalmente intensivas na investigação, e ainda para explorar e testar projectos no processo educativo.

Vários fabricantes têm vindo a construir novas placas de prototipagem baseadas em FPGA's, que comumente integram um elevado número de interfaces e dispositivos periféricos. A aplicabilidade

destas placas é assim alargada em diversas áreas, tais como processamento de imagem, optimização combinatória, multimédia, etc., mas o seu preço aumenta também.

Para tentar dar resposta ao problema do custo, sem impor restrições de aplicabilidade, surgiu a necessidade de desenvolver uma placa de prototipagem baseada numa FPGA, contendo apenas dispositivos essenciais a uma fácil configuração da placa e barramentos de expansão. Deste modo, excluem-se os dispositivos não utilizados, permitindo porém a ligação dos que forem efectivamente necessários.

A secção 2 apresenta a arquitectura da placa de prototipagem que foi desenvolvida (ver figura 1), com

algum detalhe. Na secção 3 é apresentado uma evolução da placa de prototipagem referida na secção 2. A secção seguinte descreve o projecto realizado para a gestão da placa. A secção 5 descreve as ferramentas de *software* criadas para tirar partido da funcionalidade que a placa suporta. A apresentação de algumas tarefas possíveis através desta placa está descrita na secção 6. Por fim algumas conclusões tiradas a partir da realização deste projecto.



Figura 1 – Protótipo da placa desenvolvida

2 Arquitectura da Placa

Os dispositivos que constituem a placa estão organizados segundo a arquitectura apresentada na figura 2.

O componente reconfigurável adoptado foi a FPGA Spartan-3 [1] XC3S400, da Xilinx, baseada na recente tecnologia 90nm, comportando 400 mil portas de sistema, 56Kb de RAM distribuída, 288Kb de blocos de RAM, 16 multiplicadores dedicados e disponibiliza 264 pinos de entrada/saída. Para a sua reconfiguração, foi escolhido o modo paralelo, já que é mais rápido do que o modo série.

A placa integra dois botões: um denominado “*configuration*”, que invoca a *bitstream* de configuração; outro denominado “*program*”, que

configura a FPGA com a *bitstream* de utilizador.

Um outro dispositivo incorporado é o módulo FT245BM da FTDI [2], que é uma interface USB-paralelo (FIFO), sendo compatível com USB 1.1 e USB 2.0. Ligando este módulo a um computador, é possível transferir dados a um máximo de 1 MB/s e, ao mesmo tempo, fazer uso da tensão de 5V fornecida. Assim, um único cabo permite realizar toda a gestão e alimentação da placa, a partir de um computador. Este módulo pode ainda ser utilizado como interface para vários periféricos.

A placa inclui também uma CPLD [3] – um dispositivo que integra várias PLD’s interligadas por uma estrutura programável. Tem como finalidade controlar o processo de configuração da FPGA, utilizando o barramento de endereços e pinos de controlo da memória *flash*.

A memória *flash* incorporada na placa [4] tem uma capacidade de 2 MB, com sectores de 64 KB, exceptuando os últimos quatro. A figura 3 apresenta a divisão lógica desta memória.

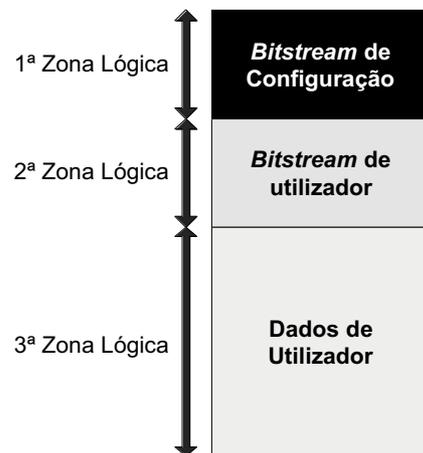


Figura 3 – Divisão lógica da memória *flash*

A primeira zona contém a *bitstream* de configuração que, uma vez carregada na FPGA, permite a gestão da placa, utilizando o *software* que foi desenvolvido e que é abordado na secção 5.

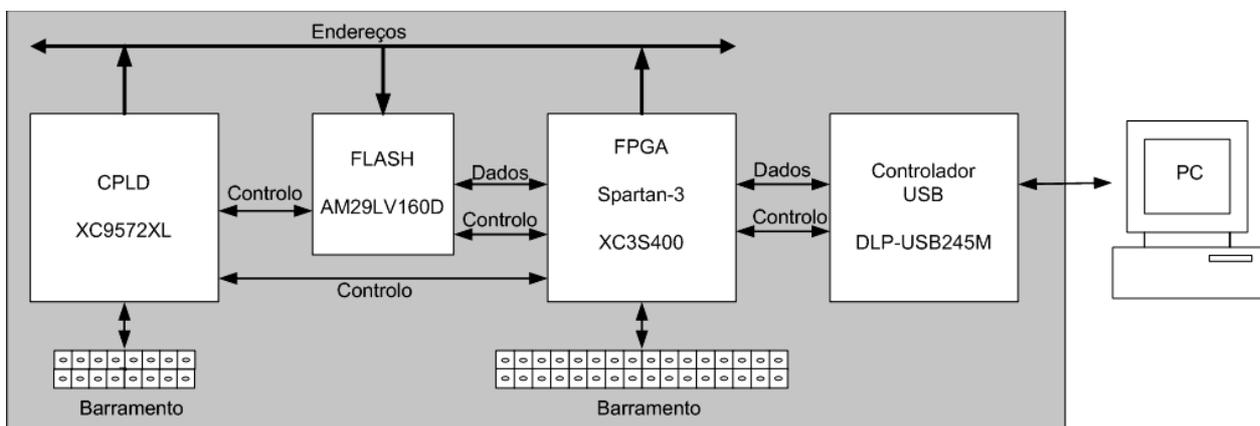


Figura 2 – Arquitectura da placa desenvolvida

A segunda zona lógica tem como finalidade guardar a *bitstream* que o utilizador pretende executar.

A terceira e última zona é destinada a um acesso livre por parte do utilizador, transferindo dados de/para o computador através do *software* desenvolvido, e por parte das *bitstreams* que estiverem a ser desenvolvidas. Esta zona pode também ser utilizada para guardar mais 6 *bitstreams* para configuração da FPGA. A selecção da *bitstream* a utilizar é conseguida através de interruptores ligados ao barramento da CPLD, pressionando depois o botão “*program*” para o executar. Por outro lado, a activação de qualquer uma destas *bitstreams* pode ser despoletada pela própria FPGA, num contexto de reconfiguração dinâmica.

Por se tratar de uma memória *flash*, as *bitstreams* e os dados armazenados não se perdem quando se interrompe a alimentação da placa. Deste modo, a placa pode ser utilizada autonomamente, i.e., sem recorrer a um computador, desde que alimentada por uma outra fonte.

Os barramentos de expansão permitem a ligação de inúmeros dispositivos, tais como periféricos e outras placas.

3 Evolução da Arquitectura

Utilizando a arquitectura considerada na secção anterior, desenvolveu-se uma nova placa de prototipagem que consistiu na substituição do módulo USB por um módulo bluetooth.

Esta alteração tem a vantagem de permitir que a comunicação, tanto na transferência de *bitstreams*,

como na de dados de utilizador, seja realizada com uma distância confortável entre a placa e o computador.

Por outro lado, existe a desvantagem da necessidade de uma outra forma de alimentar a placa, como um transformador, por exemplo.

A ligação do módulo bluetooth com a FPGA é feita através do protocolo série, utilizando um *baud rate* de 115200 bps e sequências de 8 bits e sem incorporar controlo de paridade.

Este módulo bluetooth pode também ser usado como periférico, quer nesta nova versão da placa, quer na anterior.

4 Desenvolvimento de Projectos

Nesta secção, é descrito o projecto realizado na ferramenta ISE 7.1 da Xilinx, que permite a gestão da primeira versão da placa, utilizando o módulo bluetooth. O *bitstream* criado neste projecto é designado por *bitstream* de configuração e implementa o esquemático da figura 4.

O bloco DCM (*Digital Clock Manager*) permite fazer um *reset* automático dos restantes blocos. O bloco *clock* divide o relógio por 2^N , sendo N um número inteiro. Este bloco é necessário porque a frequência do oscilador da placa é demasiado alta para ser utilizada na manipulação da memória *flash*.

O bloco *boardconfig* foi especificado em VHDL, de acordo com o diagrama de estados da figura 5, na qual, OP indica a operação (*erase sector*, *read*, *write* ou *write inv*), SA (*Start Address*) é o endereço inicial para a operação indicada, EA (*End Address*) é o eventual

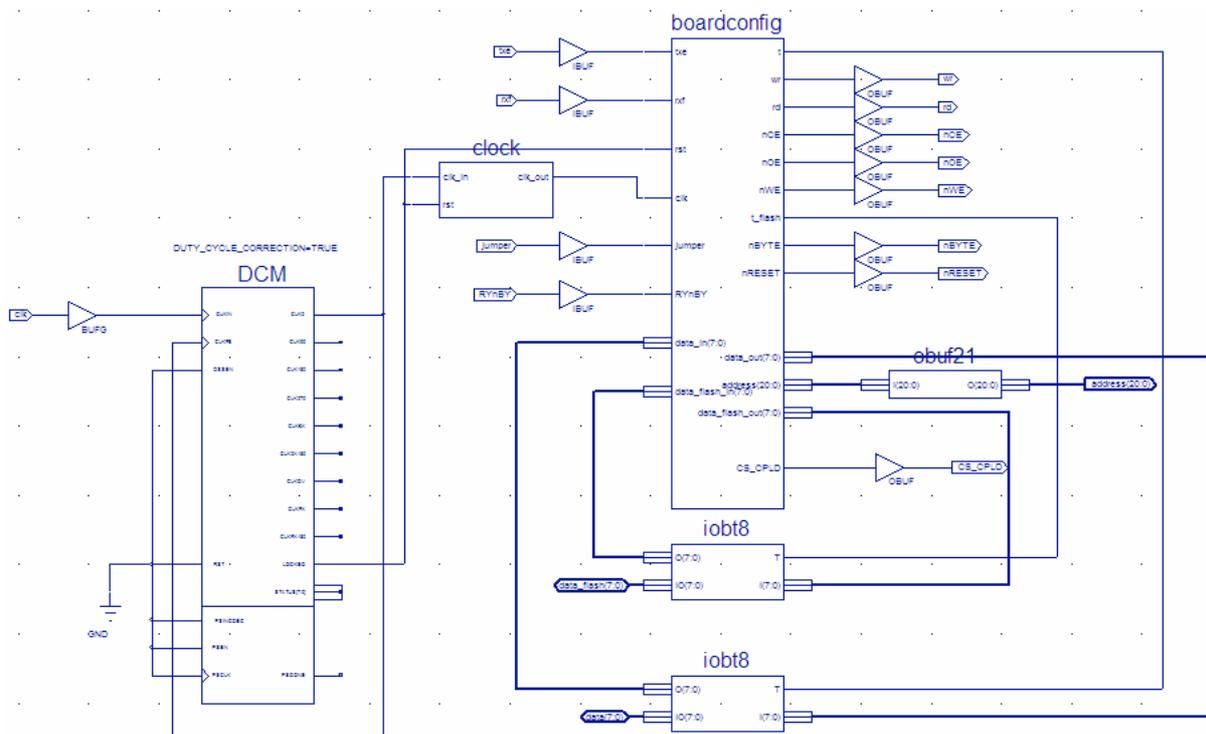


Figura 4 – Esquemático com a interligação dos blocos constituintes

endereço final, LFA (*Last Flash Address*) é o último endereço da memória *flash*, UBSA (*User Bitstream Starting Address*) é o endereço da memória *flash* no qual é armazenada a *bitstream* de utilizador, PROTECTED indica se o mecanismo de protecção da primeira zona lógica está activo ou não e OK é a mensagem de controlo que valida os parâmetros submetidos e confirma o sincronismo no final.

O mecanismo de protecção referido acima pretende proteger contra escrita na primeira zona lógica da memória *flash*, por parte de utilizadores menos experientes. Em caso desta zona ser acedida e corrompida, é necessário voltar a armazenar o *bitstream* de configuração, operação que passa a ser possível apenas através da porta JTAG.

5 Ferramentas de Software

Para facilmente se tirar o máximo partido das potencialidades desta placa de prototipagem, foi desenvolvida a aplicação PBM (*Prototyping Board Manager*). Este *software* apresenta uma interface simples e amigável (ver figura 6) e vem acompanhado de um manual de utilizador que está disponível através dos menús, em formato PDF e também *online*, nas línguas portuguesa e inglesa [5]. A partir de um computador e utilizando uma ligação USB, esta aplicação permite, entre outras funcionalidades úteis, enviar *bitstreams* com as quais se queira configurar a FPGA, transferir dados de um ficheiro para a memória *flash* da placa e vice-versa e apagar sectores dessa mesma memória. A figura 7 apresenta a árvore de menús de acesso às funcionalidades disponibilizadas por esta aplicação.

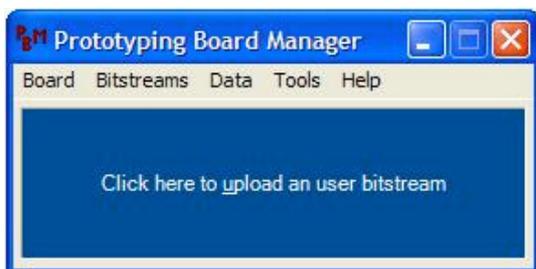


Figura 6 – Janela principal do PBM

Antes de o utilizador poder fazer uso destas funcionalidades, é necessário pressionar o botão “*configuration*” para que a *bitstream* gravada na primeira zona lógica da memória *flash* (aquando da construção da placa) seja carregada para a FPGA. Esta *bitstream* implementa uma máquina de estados, cuja descrição é apresentada na figura 5, de acordo com o protocolo também seguido pelo PBM. As funcionalidades disponibilizadas pela aplicação são construídas a partir das operações básicas que este protocolo suporta: apagar um sector, ler de um intervalo de endereços e escrever uma sequência de *bytes*. Uma variante da operação de escrita compreende uma prévia

inversão da ordem dos *bits* dentro de cada *byte*. Esta variante é utilizada na transferência de *bitstreams*, já que a configuração de uma FPGA implica a realização prévia desta transformação.

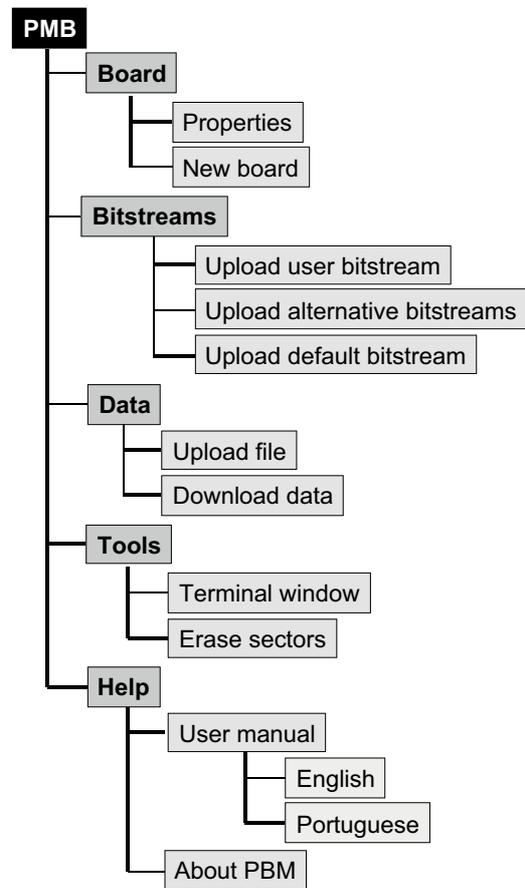


Figura 7 – Árvore de menús da aplicação PBM

A tabela 1 apresenta os tempos médios necessários para a realização de algumas tarefas de referência. Note-se que cada sector tem 64 KB, que os tempos de escrita incluem o tempo de apagar os sectores envolvidos e que a escrita de uma *bitstream* envolve 4 sectores.

Tarefa	Tempo médio (s)
Apagar sector	0,7
Ler sector	0,4
Escrever sector	1,5
Escrever <i>bitstream</i>	5,5

Tabela 1. Tempos médios para tarefas de referência

O PBM inclui ainda uma janela de terminal para a troca de *bytes* entre o utilizador e a placa de prototipagem, em tempo de execução, através da ligação USB. Esta ferramenta (ver figura 8) constitui, assim, um periférico de entrada e saída integrado, ideal para monitorar, testar e integrar projectos em fase de desenvolvimento.

Para não restringir a utilização do PBM a uma única placa de prototipagem, foram estipuladas algumas directrizes para o desenvolvimento de novas placas que garantem a compatibilidade com esta aplicação. O conjunto destas directrizes está organizado em 3 âmbitos: arquitectura da placa, *bitstream* de configuração e ficheiro PBS (*Prototyping Board Specifications*). O primeiro aborda a construção da placa e as propriedades do *hardware* utilizado; o segundo reporta-se à *bitstream* responsável pela comunicação com o PBM; o terceiro diz respeito ao ficheiro que conterá informação acerca da placa, para que a aplicação possa gerir as transferências de *bitstreams* e dados.

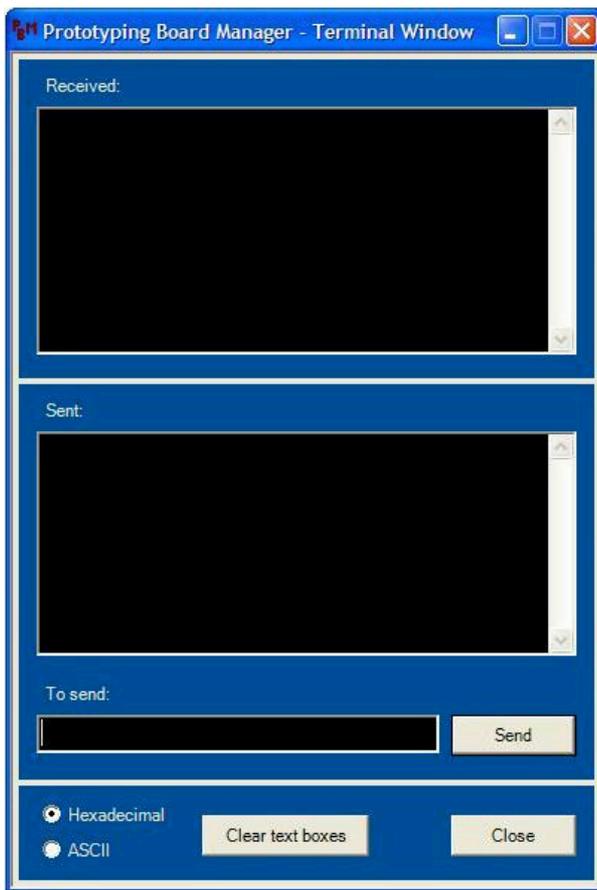


Figura 8 – Interface do periférico integrado para entrada e saída de dados

Relativamente à arquitectura, as placas têm de incluir:

- um controlador USB que utilize o driver FTD2XX.dll, no qual o ‘Product id’ tem de tomar um valor único para esse modelo;
- uma memória *flash* com endereçada ao byte , verificando as seguintes condições:
 - A memória *flash* tem de estar logicamente dividida em 3 zonas, pela seguinte ordem: zona da *bitstream* de configuração, zona da *bitstream* de utilizador e zona de dados de utilizador;

- As zonas lógicas têm de ser conjuntos consecutivos de sectores exclusivos;
- Cada uma das duas primeiras zonas tem de ter uma dimensão suficiente para guardar uma *bitstream* (sem cabeçalho) para a FPGA da placa;
- uma forma de configurar a FPGA tanto com a *bitstream* armazenada na primeira zona lógica, como com a armazenada na segunda;
- um mecanismo de protecção contra escrita na zona da *bitstream* de configuração, disponibilizando um sinal de 1 bit para indicar se a protecção está activa ou não, acessível para a FPGA (para ser utilizado pela *bitstream* de configuração).

Relativamente à *bitstream* de configuração, há que respeitar o seguinte:

- Tem de implementar a funcionalidade da máquina de estados descrita na figura 5;
- Não é responsável por descartar os cabeçalhos das *bitstreams* transferidas (o PBM fá-lo antes de transferir);
- É responsável por inverter a ordem dos bits em cada byte das *bitstreams* transferidas (ao recebê-las ou ao carregá-las para a FPGA);
- Tem de ser transferida para a primeira zona lógica da memória *flash*, antes de poder ser acedida via PBM.

Por último, um ficheiro com as características da placa tem de ser criado, tendo em conta o seguinte:

- Tem de ser um ficheiro de texto não formatado com codificação de 8 bits, composto por uma sequência bem definida de campos;
- Cada campo é composto por uma linha de identificador, seguida de uma ou mais linhas de valor;
- A extensão do ficheiro deve ser “.pbs” para uma identificação fácil e coerente.

O conjunto de campos que têm de constar e as regras de representação consta do manual de utilizador

6 Potencialidades

Para que o utilizador possa facilmente tirar partido dos dispositivos que integram a placa, serão disponibilizadas macros reutilizáveis para a sua manipulação. No caso do módulo USB e Bluetooth, o recurso a estas macros poderá então ser combinado com a utilização da janela de terminal do PBM, para interagir com os projectos do utilizador. No caso da memória *flash*, as macros permitirão o livre acesso à terceira zona lógica, protegendo as duas primeiras.

Serão criadas pequenas placas acopláveis à já desenvolvida, incorporando alguns periféricos mais frequentemente utilizados, tais como rato, teclado, monitor VGA, Bluetooth e LCD’s, e as respectivas macros de manipulação.

Para dar resposta a necessidades mais exigentes, será implementado um conjunto de tarefas tais como:

- a) Integração de várias placas equivalentes à que foi já desenvolvida, das quais uma fica responsável pela delegação de tarefas e encaminhamento dos dados necessários, permitindo a partição de problemas que se revelem difíceis de solucionar numa única FPGA/placa de prototipagem;
- b) Desenvolvimento de uma placa orientada à resolução de problemas combinatórios, cujos problemas são expressos na forma de matrizes [6,7], incorporando uma memória para armazenar a matriz, uma FPGA para operar sobre as colunas e uma outra FPGA para operar sobre as linhas;
- c) Técnicas utilitárias, tais como partição de problemas e compressão/descompressão de dados, como as descritas em [8].

7 Conclusões

Circuitos baseados em FPGA's têm mostrado grande utilidade na criação de soluções *hardware* para muitas aplicações práticas. Para auxiliar o desenvolvimento de novas soluções, foi criada uma placa de prototipagem que apresenta grande expansibilidade, versatilidade de utilização e suporte a funcionalidades úteis.

Conseguiu-se obter uma placa de baixo custo, através da exclusão de dispositivos e interfaces não fundamentais, podendo porém ser acoplados, quando necessário.

A zona de dados de utilizador da memória *flash* permite armazenar mais 6 *bitstreams*, com as quais se pode configurar a FPGA, recorrendo a uma expansão com interruptores ou a reconfiguração dinâmica.

Um módulo USB permite gerir e alimentar, simultaneamente, a placa desenvolvida, a partir de um computador.

A introdução do módulo Bluetooth na placa permite uma nova abordagem quanto à sua utilização, quer substituindo o módulo USB quer usado como periférico externo à placa.

Foi criada uma aplicação denominada PBM, que apresenta uma interface amigável para gerir a placa de prototipagem, transferindo *bitstreams* e dados de/para a memória *flash* e que incorpora ainda um periférico de entrada e saída para monitorar e testar soluções em desenvolvimento, através da troca de *bytes* em tempo de execução.

Toda a funcionalidade da aplicação desenvolvida apresenta resultados temporais cativantes,

nomeadamente os da transferência de uma *bitstream*, que demora aproximadamente 5,5 segundos.

A aplicação PBM foi desenvolvida para ser compatível com novas placas de prototipagem. Para tal, há que respeitar algumas restrições base e criar um ficheiro do qual conste a concretização das características parametrizadas.

É possível usar os barramentos de expansão para acoplar várias placas iguais à que foi apresentada e/ou orientadas a tarefas concretas, recorrendo a uma gestão eventualmente centralizada numa das placas.

Agradecimentos

Os autores agradecem ao Prof. Catedrático Valery Sklyarov pela valiosa orientação e acompanhamento, imprescindíveis à realização deste trabalho.

Referências

- [1] Spartan-3 FPGA Family, 2003 [online] <http://www.xilinx.com>
- [2] DLP-USB245M User Manual [online] <http://www.dlpdesign.com>.
- [3] CPLD XC9572XL High Performance, 2004 [online] <http://direct.xilinx.com/bvdocs/publications/ds057.pdf>
- [4] AM29LV160D Data Sheet [online] <http://www.amd.com>.
- [5] Conteúdos para a disciplina de Sistemas Digitais Reconfiguráveis [online] <http://elearning.ua.pt>, Departamento de Electrónicas e Telecomunicações, 1º Semestre, (login: alunosdr, password: sistemas).
- [6] V.SKLYAROV, I.SKLIAROVA, P.ALMEIDA, M.ALMEIDA, "High-Level Design Tools for FPGA-Based Combinatorial Accelerators", Proceedings of the 13th International Conference on Field Programmable Logic and Applications – FPL'03, Lisbon, Portugal, September, 2003, Springer, 2003, pp. 976-979.
- [7] V. SKLYAROV, I. SKLIAROVA, B. PIMENTEL, J. ARRAIS, "Hardware/Software Implementation of FPGA-targeted Matrix-Oriented SAT Solvers", Proceedings of the 14th International Conference on Field-Programmable Logic and its Applications – FPL'04, Antuérpia, Bélgica, Setembro de 2004, pp. 922-926.
- [8] V. SKLYAROV, I. SKLIAROVA, B. PIMENTEL, "Using Compression/Decompression Technique for FPGA targeted Matrix-Oriented SAT solvers", Proceedings of the XX Conference on Design of Circuits and Integrated Systems - DCIS'05, Lisboa, Portugal, Novembro de 2005, Ref. 1d.1.