

# Implementação em FPGA de um ordenador numérico recursivo com interface gráfica

Nuno Dias, Sérgio Tafula

**Resumo** – Este artigo, desenvolvido no âmbito da cadeira de Sistemas Digitais Reconfiguráveis, descreve a implementação em *Field Programmable Gate Arrays* (FPGA) de um pequeno ordenador numérico recursivo que permite uma interacção com o utilizador através de um teclado, para introdução dos valores a ordenar, e de um monitor, para apresentação dos resultados.

A linguagem de descrição de hardware utilizada (VHDL) não permite a implementação de algoritmos recursivos, o que obrigou à utilização de uma técnica baseada numa *Hierarchical Finite State Machine* (HFSM) conjugada com um algoritmo [1] de construção de uma árvore binária que visa a ordenação dos números introduzidos. A interacção com o utilizador foi feita recorrendo a dois módulos VHDL de nomes VGA [2] e PS2 que controlam, respectivamente, o monitor VGA e o teclado PS2. Todos estes algoritmos e interfaces são sintetizados (em hardware) na FPGA.

Pretende-se descrever exclusivamente a interligação e o controlo de vários sistemas já existentes, entre eles o módulo que implementa a HFSM, o processo que comanda esse módulo e os módulos VGA e PS2.

**Abstract** – This paper demonstrates results of the project proposed within the discipline on reconfigurable digital systems. The objective was to implement a recursive data sorting algorithm in hardware which would communicate with a keyboard, to allow the user to input the initial data to be sorted, and a VGA monitor to output both the original unsorted data and the same data after the sorting algorithm. Since the used hardware description language (VHDL in particular) does not allow implementing recursive algorithms, a technique based on hierarchical finite state machines has been applied. The algorithm itself constructs a binary tree for further sorting using known methods, and then sorts the data on the basis of the binary tree. The work done intends to reassemble existing blocks (such as that needed for controlling a VGA monitor, entering data from a keyboard, supporting execution of recursive algorithms, etc.) and to complete a project which makes it possible to implement the considered sorting algorithm in an FPGA-based digital system. The FPGA synthesized hardware includes all the necessary components to implement the desired interfaces and algorithms.

**Palavras chave** – SDR, ordenação, FPGA, DETIUA-S3, HFSM, VGA, PS2, recursividade.

## I. INTRODUÇÃO

Com a crescente proliferação dos *Programmable Logic Devices* (PLDs) torna-se cada vez mais fácil projectar e implementar sistemas digitais complexos recorrendo à

descrição de hardware, como se de software se tratasse. Os PLDs representam uma alternativa cada vez mais eficaz aos *Application Specific Integrated Circuits* (ASICs) pois podem ser utilizados numa ampla gama de aplicações práticas. Como tal, verifica-se que o uso de FPGAs para implementação de tarefas complexas tem vindo a crescer significativamente, especialmente devido à elevada performance e grande flexibilidade.

A disciplina de Sistemas Digitais Reconfiguráveis [3] [4] é uma cadeira opcional do 5º ano do Mestrado Integrado em Engenharia Electrónica e Telecomunicações (MIEET) e é leccionada no Departamento de Electrónica, Telecomunicações e Informática da Universidade de Aveiro (DETI-UA). O seu principal objectivo é introduzir aos alunos os sistemas digitais reconfiguráveis (FPGAs) e uma linguagem de descrição destes (VHDL<sup>1</sup>). Esta disciplina conta com uma forte componente prática, no âmbito da qual foi desenvolvido este trabalho com o objectivo primário de permitir melhor testar e compreender alguns dos conceitos apresentados na disciplina.

Como base de desenvolvimento de todo o projecto está uma placa de protótipo autónoma desenvolvida no DETI. Esta placa, apresentada na figura 1, denomina-se DETIUA-S3 [5] e usa uma FPGA Xilinx da série Spartan-3. Usando esta placa e os módulos VHDL de nome HFSM, VGA e PS2, construiu-se um sistema que ordena recursivamente uma série de números, e que interage com o utilizador através de uma interface gráfica.

Salientamos que o uso de circuitos reconfiguráveis que su-

<sup>1</sup>VHDL - VHSIC Hardware Description Language, VHSIC - Very High Speed Integrated Circuits.

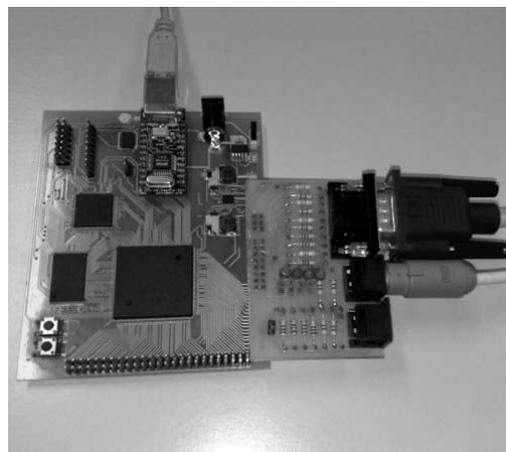


Figura 1 - Imagem da placa DETIUA-S3

portam interacção com monitores VGA [2], permitindo a visualização dos resultados e dos dados intermédios num ecrã, facilita a depuração do hardware sintetizado em FPGA e aumenta as potencialidades de interacção com o utilizador. Complementando a visualização num monitor com um módulo que interage com um teclado PS2, consegue obter-se uma completa interactividade, onde o utilizador pode inserir dados, alterar sinais, e visualizar resultados.

A implementação em hardware de uma Máquina de estados Finitos Hierárquica (HFSM) [6] permite o uso de recursividade, levando a uma reutilização de hardware. Nas máquinas de estados convencionais, todos os estados são considerados como estando ao mesmo nível, não sendo capturadas as semelhanças existentes entre estados. Pelo contrário, numa HFSM essas semelhanças são capturadas, organizando os estados segundo uma hierarquia. Os estados de níveis mais baixos contêm assim as características dos estados de níveis superiores. Numa HFSM os diversos estados vão sendo guardados numa memória do tipo pilha, criando uma tabela hierárquica dos estados pelos quais se passou até chegar ao estado actual. Torna-se assim possível que vários estados usem os mesmos recursos, sem que se perca o contexto da execução do algoritmo.

Começar-se-á por uma breve visão geral do circuito, após a qual se abordarão individualmente os diversos blocos que o constituem.

## II. VISÃO GERAL

Uma imagem de todo o hardware do sistema desenvolvido está apresentada na figura 2.

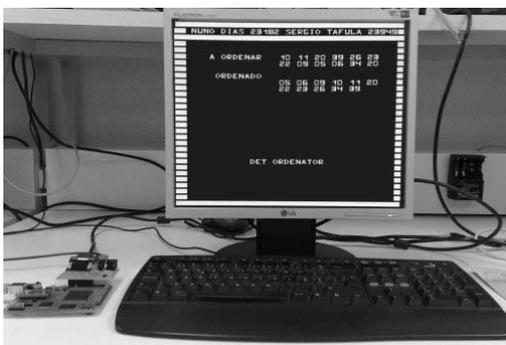


Figura 2 - Imagem de todo o hardware do sistema

O sistema desenvolvido na FPGA, apresentado no esquema da figura 3, é constituído por um subsistema de interacção com o monitor e teclado (secção III), um subsistema que implementa uma HFSM (secção IV), e um módulo de interligação e controlo destes (secção V).

O subsistema de interligação e controlo está materializado no bloco “Ordenador” apresentado na figura 7. Este bloco recolhe a informação proveniente do teclado através do bloco PS2 (figura 5) e escreve nas memórias da figura 4 que definem o que aparece no monitor VGA. O bloco “Ordenador” tem também a função de controlar a HFSM (figura 6) e a árvore binária de modo a que as sequências de números fornecidas pelo utilizador sejam ordenadas.

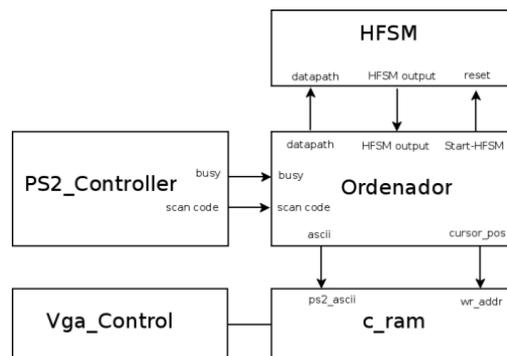


Figura 3 - Esquema do sistema completo.

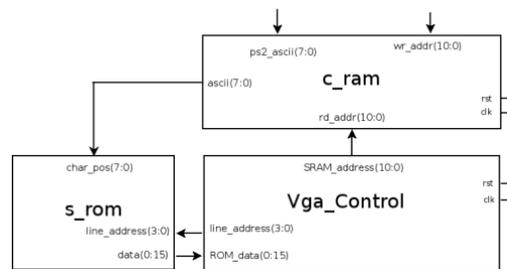


Figura 4 - Blocos do módulo VGA.

## III. SISTEMA ENTRADA/SAÍDA

A descrição exaustiva do sistema de exibição de dados no monitor [2] não faz parte do âmbito deste artigo. Contudo, é importante referir que este sistema implementa uma interface que consiste numa memória de nome C RAM, apresentada na figura 4, onde são escritos os caracteres a apresentar no monitor através do barramento “ps2\_ascii”. Esta memória contém todo o mapa de caracteres e a sua manipulação não é completamente assíncrona, existindo importantes constrangimentos de sincronização. A posição onde os caracteres serão exibidos no monitor é definida pelo endereço “wr\_addr”.

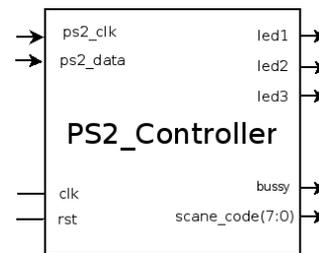


Figura 5 - Bloco PS2.

Do bloco da figura 5, que implementa o interface PS2 com o teclado, importa reter que é ele que deposita os códigos provenientes do teclado no barramento “scan\_code”, e que actualiza a flag “busy” que indica se a informação deste barramento é ou não válida.

## IV. SISTEMA DE ORDENAÇÃO COM HFSM

A HFSM implementada no bloco da figura 6 serve de base ao sistema recursivo de ordenação, essencialmente porque as saídas desta máquina de estados “output\_to\_datapath”

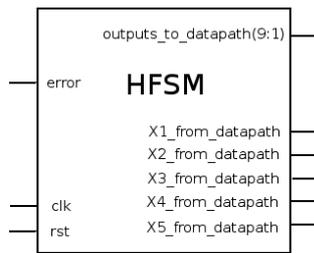


Figura 6 - Bloco que implementa a HFSM.

comandam o processo de construção da árvore binária e, conseqüentemente, a ordenação dos números. As saídas da HFSM são obviamente dependentes da sequência de alterações das suas entradas "X1...X5". Estas, por sua vez, dependem da informação a ordenar.

Dentro do bloco "Ordenador" (figura 7) existe um processo, uma memória RAM, uma ROM e uma stack, destinadas a interagir com a HFSM. Estes componentes, apesar de estarem incluídos no bloco da figura 7, fazem parte do sistema de ordenação.

A cada reinicialização, o sistema de ordenação transforma os números desordenados depositados no array "ROM" do bloco "Ordenador" num conjunto de números ordenados depositados nesse mesmo array.

## V. CONTROLO

Como já foi referido, a essência deste trabalho foi construir um só sistema que aglomerasse funcionalidades de vários subsistemas. O bloco "Ordenador", da figura 7, une o subsistema de ordenação, o subsistema VGA e o subsistema PS2.

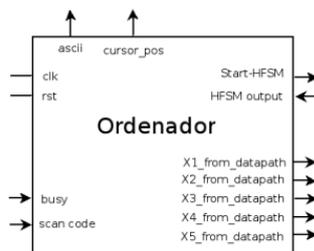


Figura 7 - Bloco de interligação e controlo.

Este bloco capta os dados do teclado, providencia a exibição destes antes de serem ordenados, comanda o subsistema de ordenação e a exibição dos dados antes e depois de ordenados.

O algoritmo de leitura de caracteres do teclado baseia-se num processo com sensibilidade á flag "busy". Este processo implementa uma máquina de estados "circular" em que uma transição negativa do sinal busy implica a leitura e tradução do código proveniente do teclado, o armazenamento do número lido e o salto para o próximo estado da máquina. Este processo é circular na medida em que, após a leitura do último número, retorna ao primeiro.

O bloco de controlo é responsável pela validação dos caracteres provenientes do teclado, ou seja, se o utilizador tentar escrever numa determinada posição do monitor um carácter que conduza à introdução de um valor que não esteja

entre 0 e 39, o carácter é descartado e o cursor permanece na mesma posição.

Sempre que uma sequência completa de números é lida do teclado, o sistema de ordenação é reiniciado. Este processo é comandado pelo sinal "start\_ord" do bloco "Ordenador" e está ligado ao reset do bloco HFSM, pelo que a sua alteração afecta ambos componentes do subsistema de ordenação.

O mecanismo de exibição dos números a ordenar e ordenados, baseia-se numa máquina de estados que ciclicamente rescreve todos os caracteres na memória do módulo VGA. Esta máquina é implementada num processo com sensibilidade ao relógio do sistema. A uma cadência múltipla da frequência de relógio é incrementado o estado da máquina e escrito alternadamente o carácter e o posição deste no monitor. Por motivos de sincronização entre a velocidade de leitura do teclado e a velocidade de refrescamento dos caracteres no monitor, os números ordenados e a ordenar são armazenados também em sinais dentro do módulo ordenador.

Os números são exibidos no monitor sempre que tenham algum significado no contexto, ou seja, o sistema de exibição quando não tem um carácter válido para mostrar numa determinada posição não exhibe nada. Isto é conseguido testando o conteúdo de todos os valores antes destes serem exibidos.

## VI. RESULTADOS

Construiu-se um sistema ordenador que usa os módulos e arquitectura supra citada e apresenta o interface gráfico apresentado na figura 8. Este interface exhibe os caracteres válidos, introduzidos através do teclado, nas linhas ao lado do texto "A Ordenar" e exhibe estes mesmos dados ordenados nas linhas ao lado do texto "Ordenado".

O sistema de exibição não mostra os zeros adicionados pelo algoritmo de ordenação no fim da sequência ordenada. Estes zeros são resultado da repetição de números na entrada.

Os caracteres válidos vão aparecendo no monitor á medida que são introduzidos no teclado. No caso de introdução de um carácter não válido, nada acontece.

No final da introdução de uma sequência completa de números para ordenar, qualquer tecla serve para limpar o resultado anterior e em seguida, qualquer tecla serve para mostrar a nova sequência ordenada.

Foram identificados problemas na exibição de caracteres em algumas posições do ecrã. Detectamos que os caracteres escritos na parte inferior do ecrã não são visíveis e detectamos também que a escrita em algumas posições da parte superior altera a informação que aparece nas extremidades do ecrã.

O código html deste projecto pode ser consultado na página [www.detordenator.pt.vu](http://www.detordenator.pt.vu) [7]

## VII. CONCLUSÃO

Com este projecto demonstramos que é viável a integração num só sistema dos vários módulos disponíveis.

Verificamos que este ordenador complementado com uma interface gráfica apresenta mais-valias importantes



Figura 8 - Imagem do monitor

no campo da usabilidade. Este acréscimo de funcionalidade pode ser aproveitado para uma demonstração mais eficiente, interactiva e pedagógica do ordenador recursivo construído com uma HFSM e uma árvore binária.

A integração de todos estes blocos num só sistema foi uma tarefa árdua. Os pormenores de adaptação de níveis lógicos e de sincronismo dos diferentes módulos, blocos e processos apresentaram-se de uma importância extrema e foram a chave para a condução do projecto a bom termo. Existem, contudo, problemas identificados para os quais não obtivemos soluções.

O maior dos problemas identificados prende-se com a escrita de caracteres no monitor. Recorrendo ao algoritmo de escrita na memória "C\_RAM" apresentado na secção V, verificamos que a apresentação dos dados no monitor é ligeiramente instável, obrigando-nos a escolher por tentativa e erro as posições usadas para exibir dados de forma a obter um resultado visualmente aceitável.

Julgamos que a instabilidade detectada poderá dever-se a alguma escrita síncrona incorrecta na RAM do módulo VGA ou a alguma incompatibilidade entre o módulo VGA e a placa usada.

Este sistema, na configuração aqui apresentada, tem uma função meramente pedagógica. A tecnologia usada permite ordenar sequências de tamanho apenas limitado pela memória disponível e a uma velocidade só superável pelo uso de um ASIC de elevada performance, que implemente em hardware as funcionalidades deste sistema. Neste contexto, ordenar 12 números de 0 a 39 é uma amostra do potencial do sistema, pelo que aumentar a sequência e a gama de entrada é uma linha de evolução óbvia do sistema visado.

#### REFERÊNCIAS

- [1] Valery Sklyarov, "Fpga-based implementation of recursive algorithms", *Microprocessors and Microsystems Issue on FPGAs : Applications and Designs*, vol. 28, no. 5-6, 2004.
- [2] Iouliia Skliarova, "Desenvolvimento de circuitos reconfiguráveis que interagem com um monitor vga", *RevDETI-UA*, vol. 4, no. 5, SETEMBRO de 2005.
- [3] Valery Sklyarov, "Página pessoal", [www.ieeta.pt/~skl](http://www.ieeta.pt/~skl), Fevereiro de 2008, Sítio na Internet do professor Valery Sklyarov.
- [4] Valery Sklyarov, "Sdr", [http://www.ieeta.pt/~skl/SDR\\_V.html](http://www.ieeta.pt/~skl/SDR_V.html),

Fevereiro de 2008, Sítio na Internet da cadeira de Sistemas Digitais Reprogramáveis.

- [5] Manuel Almeida, *Manual de Utilizador Placa DETUA-S3*, Universidade de Aveiro.
- [6] Valery Sklyarov, "Hierarchical finite state machines and their use for digital control", *IEEE Transactions on VLSI Systems*, vol. 7, no. 2, 2005.
- [7] Sérgio Tafula, Nuno Dias, "Sdr", <http://www.detordenator.pt.vu>, Fevereiro de 2008, Sítio na Internet do projecto de Sistemas Digitais Reconfiguráveis - DetOrdenator.