

# Projeto Final de Laboratório de Sistemas Digitais

Universidade de Aveiro

Olha Buts, André Correia



VERSÃO 1

# Projeto Final de Laboratório de Sistemas Digitais

Dept. de Eletrónica, Telecomunicações e Informática

Universidade de Aveiro

Olha Buts, André Correia  
(112920) o.buts@ua.pt, (87818) amcorreia@ua.pt

30 de maio de 2023

# Conteúdo

<b>1</b>	<b>Introdução</b>	<b>1</b>
<b>2</b>	<b>Desenvolvimento do Sistema Digital</b>	<b>2</b>
2.1	Arquitetura do Sistema . . . . .	2
2.2	Implementação do Sistema . . . . .	4
2.3	Validação do Sistema . . . . .	6
<b>3</b>	<b>Manual de Utilizador</b>	<b>7</b>
<b>4</b>	<b>Conclusões</b>	<b>8</b>

# Lista de Figuras

2.1	Diagrama Lógico do Sistema. . . . .	3
2.2	Abstração do Diagrama Lógico do Sistema. . . . .	3
2.3	Diagrama de Estados da FSM de Controlo. . . . .	4
2.4	Abstração da Zona da FSM de Controlo. . . . .	4
2.5	Diagrama de Estados da FSM Principal. . . . .	5
2.6	Abstração da Zona da FSM Principal. . . . .	5
2.7	Testbench da FSM de Controlo. . . . .	6
2.8	Testbench da FSM Principal. . . . .	6
2.9	Testbench do Top-Level do Sistema. . . . .	6

# Capítulo 1

## Introdução

Os alunos da Unidade Curricular de **L**aboratórios de **S**istemas **D**igitais (*LSD*, código 40333) da **L**icenciatura em **E**ngenharia de **C**omputadores e **I**nmática (*LECI*, código 8316) da **U**niversidade de **A**veiro (*UA*) foram propostos ao desenvolvimento de um Projeto Final que contempla três componentes: desenvolvimento do sistema digital de uma Máquina Automática de Fazer Pão (Projeto Número 8, Versão 2), criação de um relatório do desenvolvimento anteriormente referido, e defesa do projeto perante um Juri.

O sistema digital da Máquina Automática de Fazer Pão deve ser modelado em *Very High Speed Integrated Circuits Hardware Description Language* (*VHSIC-HDL*, ou *VHDL*) e testado numa **F**ield-**P**rogrammable **G**ate **A**rray (*FPGA*). Neste sentido, a máquina desenvolvida apresenta dois modos de operação principal: Fazer Pão Caseiro (Modo 1), ou Fazer Pão Rústico (Modo 2) – sendo que ambos os modos apresentam a possibilidade de adição de um tempo extra inicial, ou adição de um tempo extra final (que aumenta o tempo total de cozedura do pão). Apesar de cada um destes modos ser caracterizado por diferentes parâmetros temporais, ambos partilham a mesma *pipeline*, ou procedimento de ‘fazer pão’ (o amassar da massa, o descanso da massa para levedar, e a cozedura no final).

Relativamente ao documento, este apresenta o relatório do desenvolvimento do sistema digital da Máquina Automática de Fazer Pão (Versão 2 do Projeto 8) de acordo com as competências adquiridas na Unidade Curricular de LSD. Neste sentido, o documento divide-se em quatro componentes, sendo estas a arquitetura do sistema digital (descrição conceptual do sistema), a implementação efetuada para a anterior arquitetura (representação gráfica do sistema digital), os métodos de validação usados (simulações efetuadas sobre a implementação da arquitetura), e por fim, um manual de utilizador da máquina como um todo (em ambiente de desenvolvimento através de uma FPGA) – sendo o foco da arquitetura e da implementação as Máquinas de **E**stados **F**initos (MEF, ou FSM em Inglês).

## Capítulo 2

# Desenvolvimento do Sistema Digital

O desenvolvimento e implementação do sistema digital desta máquina passa por três fases: arquitetura (desenho lógico de todo o funcionamento do sistema), implementação (em VHDL, usando o programa Intel® Quartus® Prime) e uma posterior validação (testes via *testbenches* em VHDL e via uso normal, numa ótica de utilizador).

### 2.1 Arquitetura do Sistema

Nesta secção aborda-se a estrutura do sistema digital através de uma descrição conceptual da lógica que gerou o produto final (em ambiente de desenvolvimento por via de FPGA). Neste sentido, a arquitetura aplicada neste projeto divide-se em duas zonas que estão intrinsecamente interligadas:

- Zona de controlo do sistema, responsável maioritariamente pelos *inputs* – por exemplo:
  - Conjunto (físico / *hardware*) de *keys*.
  - Conjunto de *switches*.
  - Comportamento (lógico) de *start/stop*.
  - Comportamento de *reset* do sistema.
- Zona de controlo do procedimento de 'fazer pão', responsável pela maior parte do *output*, e por toda a funcionalidade de 'fazer pão', – por exemplo:
  - Comportamento de cada etapa do processo de amassar, levedar e cozer o pão.
  - Output do estado atual da máquina, assim como de informações ao utilizador, através de componentes físicos tais como o *Liquid-Crystal Display* (LCD) e os *7-Segment Displays* da FPGA.

Estas duas zonas de controlo são ambas compostas por elementos lógicos *standard*, assim como por controladores customizados (as FSM). Estas FSM caracterizam-se por serem comunicantes, o que possibilita comportamentos que interligam simultaneamente a lógica das duas zonas de controlo – destacando-se o caso da possibilidade de adicionar um tempo extra de cozedura (no final do processo de ‘fazer pão’).

Na Figura 2.1 visualiza-se o diagrama lógico completo do sistema, estando a azul destacada a zona de controlo do sistema (denominada doravante por FSM de Controlo), e a verde destacada a zona de controlo do procedimento de ‘fazer pão’ (doravante denominada por FSM Principal).

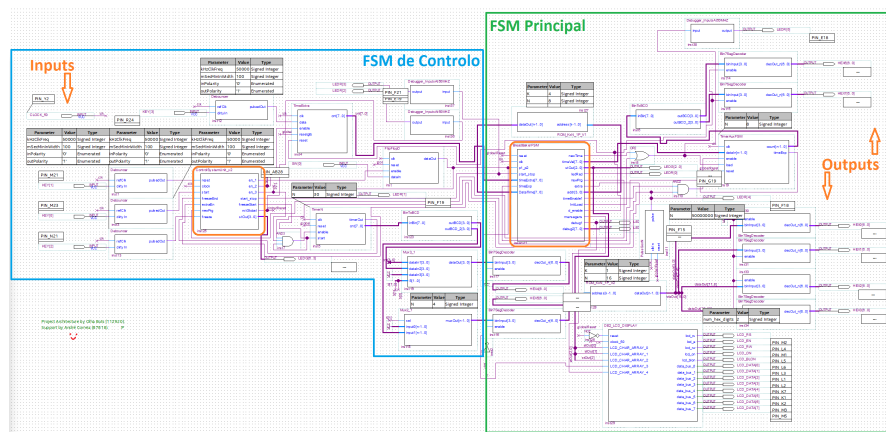


Figura 2.1: Diagrama Lógico do Sistema.

Assim, simplifica-se o circuito lógico através da abstração representada pela Figura 2.2 onde se destaca, conceptualmente, o sistema. Nesta Figura pode-se identificar que o sistema é constituído por duas FSM principais que controlam logicamente o conjunto de *inputs* e *outputs* fornecidos pela FPGA. A FSM de Controlo pode ser descrita pela responsabilidade dos mecanismos de *start/stop*, *reset*, temporizadores (ligados à lógica de adição de tempos extra no início, ou no final), e pela seleção do modo de programa (tipo de pão Caseiro, ou Rústico). A FSM Principal é reponsável por toda a *pipeline* de ‘fazer pão’ – isto é, os estados de amassar, levedar e cozer.

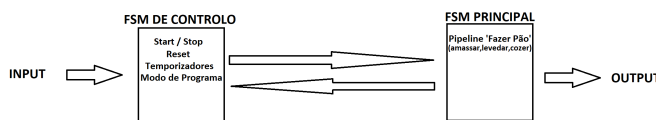


Figura 2.2: Abstração do Diagrama Lógico do Sistema.

## 2.2 Implementação do Sistema

Nesta secção ilustram-se alguns aspetos da implementação da arquitetura anteriormente apresentada, tais como os diagramas de estados das FSM de Controlo e FSM Principal.

Deste modo, começa-se por analisar o diagrama de estados da FSM de Controlo (ver Figura 2.3) onde se destaca a responsabilidade de iniciar a *pipeline* da FSM Principal, assim como a responsabilidade de pausar todo o procedimento – seja no momento de amassar, levedar, ou amassar o pão. Além disto, a Figura 2.3 demonstra o controlo dos tempos extras (inicial, ou final).

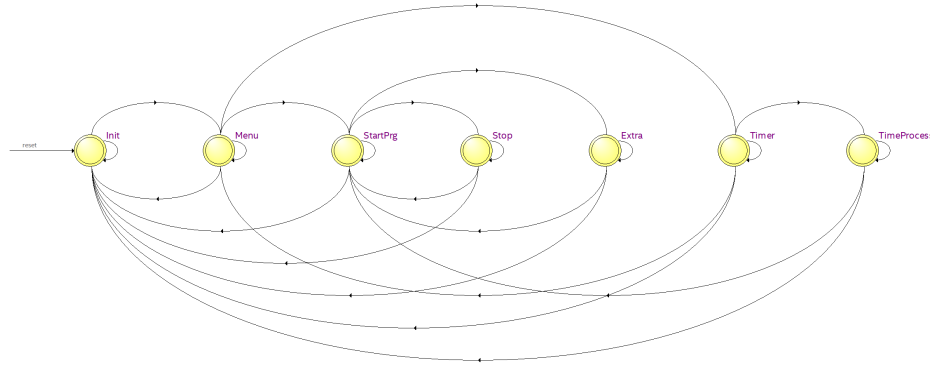


Figura 2.3: Diagrama de Estados da FSM de Controlo.

O conjunto de lógica que suporta esta FSM é constituído por entradas (*keys* e *switches*), por *debouncers*, por *Flip-Flops* e pelos temporizadores responsáveis pelos tempos extra inicial e final (ver a abstração apresentada na Figura 2.4).

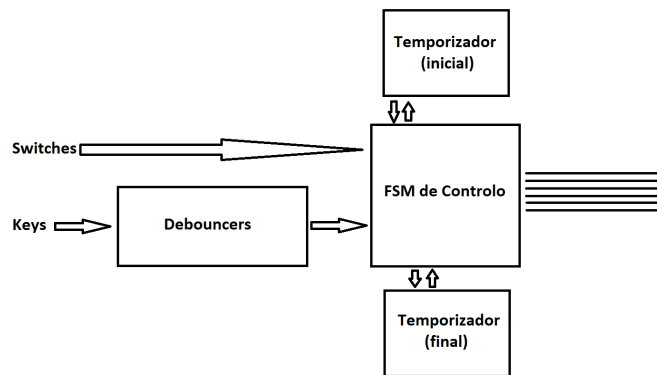


Figura 2.4: Abstração da Zona da FSM de Controlo.



Relativamente à FSM Principal, apesar desta conter uma interface que alimenta alguns sinais de input da FSM de Controlo (por exemplo, o sinal "*newPrq*" que indica que terminou o seu ciclo), denota-se uma responsabilidade quase total pela pipeline de amassar, levedar, e cozer o pão (ver Figura 2.5) – onde a diferença entre o pão caseiro e rústico é no tempo de amassar (10s/15s, respetivamente) e de levedar (04s/08s); cozer (10s/10s) – sendo que os tempos específicos para cada tipo de pão (ou modo de operação) são alimentados por uma o sinal *Read Only Memory* (ROM).

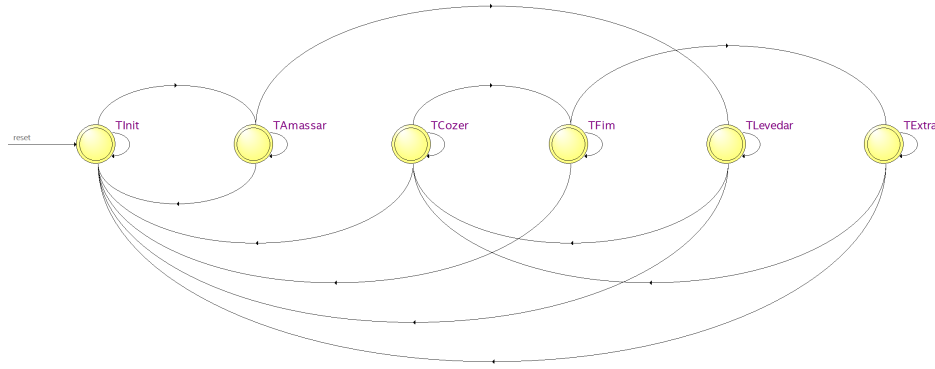


Figura 2.5: Diagrama de Estados da FSM Principal.

Os dispositivos lógicos que constituem esta zona da responsabilidade da FSM Principal são (entre outros): decodificadores de binário para **Binary-Coded Decimal** (BCD), decodificadores de binário para 7-Segment, ROMS, e contadores (ver a abstração apresentada na Figura 2.6).

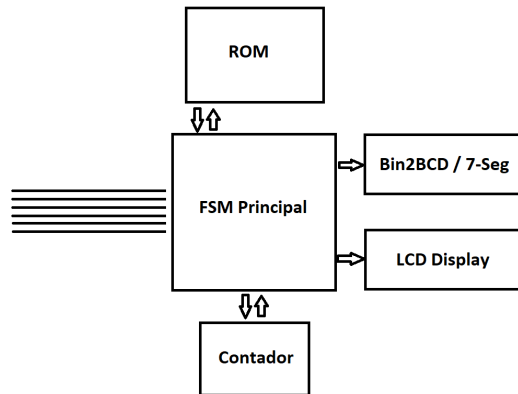


Figura 2.6: Abstração da Zona da FSM Principal.

Contudo, o sistema contém outra lógica de suporte fulcral que não está mencionada em nenhuma das duas abstrações anteriormente apresentadas, tal como geradores de pulsos.

## 2.3 Validação do Sistema

Esta última secção do Desenvolvimento do Sistema Digital (capítulo 2) tem como objetivo abordar os procedimentos implementados para validação do sistema – com foco sobre a validação das FSM usando *testbenches* desenhadas em VHDL. Contudo, também se aplicou uma rotina de interação com a máquina na ótica do utilizador (num formato de desenvolvimento em FPGA) em cada nova versão da implementação.

Neste sentido, verifica-se na Figura 2.7 a correta transição de estados da FSM de Controlo quando aplicados diferentes valores nos seus inputs. É de se notar que esta FSM de Controlo tem como principais blocos lógicos *standard* dois temporizadores (que são responsáveis pelo tempo extra no início, ou no final), assim como um sinal proveniente da FSM Principal (que indica se o ciclo da *pipeline* terminou). Deste modo, a manipulação destes valores de acordo com comportamentos esperados – e outros aleatórios (*edge-cases* potencialmente problemáticos) – permite validar o bloco lógico como corretamente funcional.

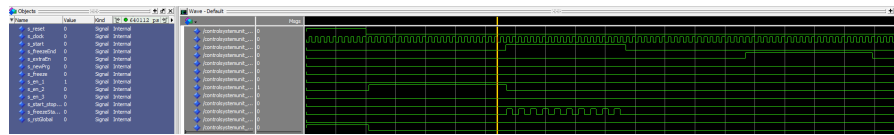


Figura 2.7: Testbench da FSM de Controlo.

Relativamente à FSM Principal, como se verifica na Figura 2.8, aplicando a mesma metodologia de manipulação das entradas, obtem-se o comportamento esperado (correto) do bloco lógico.

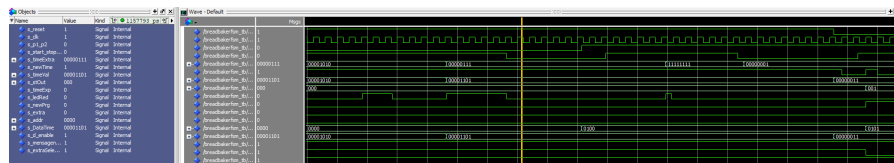


Figura 2.8: Testbench da FSM Principal.

Por fim, indica-se a Figura 2.9 onde demonstra o teste efetuado ao sistema como um todo (*Top-Level*). Nesta Figura, verifica-se o correto comportamento da Máquina Automática de Fazer Pão – exatamente como analisado manualmente na FPGA.

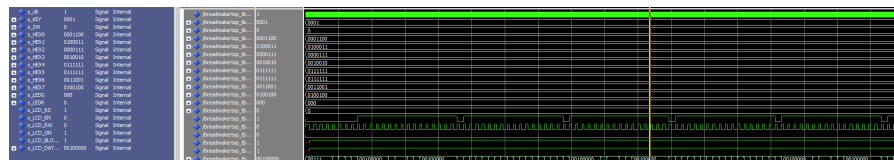


Figura 2.9: Testbench do Top-Level do Sistema.

## Capítulo 3

# Manual de Utilizador

Tomando os passos de preparação da FPGA como garantidos – conectar (USB e *Power Delivery*), e submeter o projeto para a FPGA – então o utilizador, após ligar o dispositivo, é recebido por um menu de utilizador.

- SW[0] - Permite alterar o modo de operação (entre Pão Caseiro e Pão Rústico).
- KEY[0] - Inicia o processo de 'fazer pão' (*start*).
- KEY[2] - Permite adicionar um tempo extra de atraso ao iniciar o processo de 'fazer pão' (necessário depois premir KEY[0] para iniciar).

Depois de iniciar o processo de 'fazer pão', o utilizador pode pausar esta *pipeline* em qualquer etapa do processo, premindo o KEY[0] (*stop*).

Com isto, no final deste processo de 'fazer pão', o utilizador é recebido por uma mensagem que permite adicionar um tempo extra de cozedura. Para isto, terá de premir o KEY[3] para incrementar o temporizador (que tem um alcance limitado e cíclico), sendo necessário premir o KEY[0] (*start/stop*) para iniciar este processo de aquecimento extra.

É de se notar que, sempre que a máquina acaba a sua *pipeline*, o utilizador tem a opção de um aquecimento extra (quantas vezes assim pretender) – sendo que para quebrar este ciclo, o utilizador tem de ter o temporizador a 0 e premir o KEY[0] (*start/stop*).

Por fim, o utilizador em qualquer momento pode fazer um *reset* à máquina premindo o KEY[1]. Esta ação irá levar a máquina para o estado do menu (inicial) de utilizador.

## Capítulo 4

# Conclusões

A arquitetura da Máquina Automática de Fazer Pão e a posterior implementação através de Máquinas de Estados Finitos Comunicantes, e toda a envolvente lógica digital, demonstrou atingir um nível de complexidade que necessita de procedimentos de desenvolvimento bem definidos desde o início do projeto.

Deste modo, é de importância realçar a necessidade de estratégias de desenvolvimento faseadas e de mecanismos de controlo, tais como versões de projeto, assim como metodologia em todas as etapas do projeto. Por se prever este nível de complexidade, esta equipa preparou um repositório na plataforma GitHub onde se encontram múltiplas versões do Projeto, documentação de suporte ao desenvolvimento, versões do presente relatório, e outro material usado (como *drivers* do LCD).

Além disto, canais de comunicação bem definidos, assim como o estabelecimento de um líder para cada componente do Projeto Final são condições necessárias, e aplicadas neste projeto, para ver o produto final com sucesso e a qualidade esperada pela equipa. Deste modo, destaca-se a *Lead-Architect* Olha Buts que tomou controlo da arquitetura e implementação do sistema digital.

Em suma, este projeto decorreu como esperado. Com isto, o nível de dificuldade superou ligeiramente as expectativas iniciais, mas toda a preparação prévia de organização do trabalho da equipa (parágrafos anteriormente mencionados) foi fulcral para ver este projeto sucedido. Portanto, o Projeto Final de LSD é sem dúvida um elemento de avaliação necessário para garantir a consolidação dos conhecimentos lecionados nesta UC, mas também para garantir que os alunos têm mecanismos de trabalho em equipa e uma ótica orientada aos objetivos.

Assim, a percentagem de trabalho é:

- **50%:** Olha Buts (112920) - o.buts@ua.pt
- **50%:** André Correia (87818) - amcorreia@ua.pt