

AirFryer

Universidade de Aveiro

Dinis Batista, Francisco Ribeiro



AirFryer

Dept. de Eletrónica, Telecomunicações e Informática
Universidade de Aveiro

Dinis Batista, Francisco Ribeiro
(118880) d.batista@ua.pt, (118993) franciscomribeiro05@ua.pt

1 de junho de 2024

Índice

1	Introdução	1
2	Manual de Utilização	2
2.1	Programas	2
2.2	Funcionamento	2
2.3	Esquema da Máquina	3
3	Arquitetura e Implementação	4
3.1	State Machine	4
3.2	Timer	5
3.3	Program Controller	5
4	Validações	7
5	Conclusões e Contribuições	8
5.1	Conclusões	8
5.2	Contribuições dos Autores	8

Capítulo 1

Introdução

Como projeto final da Unidade Curricular de Laboratório de Sistemas Digitais (LSD), escolhemos o projeto da Air Fryer. Para tal, usamos VHSIC Hardware Description Language (VHDL) para simular o seu comportamento na placa FPGA Terasic DE2-115. Haverão 3 secções de displays, uma para a temperatura do programa, e mais duas para os tempos de execução dos diversos programas da Air Fryer. Haverá também um interruptor para ligar a Air Fryer, um para dar início à execução dos programas, outro para abrir e fechar a cuba de cocção, mais 3 interruptores para seleccionar os programas e ainda um último que auxilia nas seleções do programa User. Existem ainda vários LEDs que fornecem ao utilizador informações e que deixam o uso da Air Fryer mais intuitivo. Por fim, há 4 botões que permitem subir e descer a temperatura e tempo de cocção do programa User.

Capítulo 2

Manual de Utilização

2.1 Programas

Programa	Temperatura	Cocção
Default	200°	18 mins
User	Definido pelo utilizador	Definido pelo utilizador
Rissóis	180°	15 mins
Batatas	200°	20 mins
Filetes de peixe	170°	20 mins
Hamburger	170°	20 mins

Tabela 2.1: Programas da Airfryer

2.2 Funcionamento

Para ligar a AirFryer basta mover o interruptor SW[0]. A AirFryer inicializa permitindo ao utilizador seleccionar o programa a ser executado. Para tal, deve colocar o interruptor SW[5..3] na posição desejada. Após a seleção do programa, o utilizador pode iniciar a máquina movendo o interruptor SW[1]. Inicia-se então o preaquecimento. Após o preaquecimento, os LEDG irão acender para avisar o utilizador que deve retirar a cuba e inserir os alimentos. O interruptor SW[2] controla a abertura da cuba: para cima a cuba está aberta e para baixo a cuba está fechada. Depois de fechar a cuba novamente, a máquina começa a contagem do tempo do programa seleccionado. Quando o tempo do programa termina, os LEDG acendem novamente para indicar ao utilizador que deve retirar a comida. O utilizador pode então abrir a cuba movendo o interruptor SW[2] para cima, retirar o alimento e fechar a cuba novamente. A máquina então entra em um período de arrefecimento até atingir os 20 C. Assim que isso aconteça a Air Fryer avisa que o programa terminou através dos LEDs vermelhos e fica em estado de espera para ser desligada. Para desligar a máquina, basta mover o interruptor SW[0].

2.3 Esquema da Máquina

- Switch: SW[0] → On / Off
- Switch: SW[1] → Run
- Switch: SW[2] → Open Oven
- Switch: SW[5..3] → Program Selector
- Switch: SW[6] → User Mode Selector
- Key: Key[0] → TimeUp
- Key: Key[1] → TimeDown
- Key: Key[2] → TempUp
- Key: Key[3] → TempDown
- Led: LEDG[0] → FoodIn
- Led: LEDR[2..0] → Status
- HEX6[6..0] e HEX7[6..0] → Time
- HEX0[6..0] , HEX1[6..0] e HEX2[6..0] → Temperature

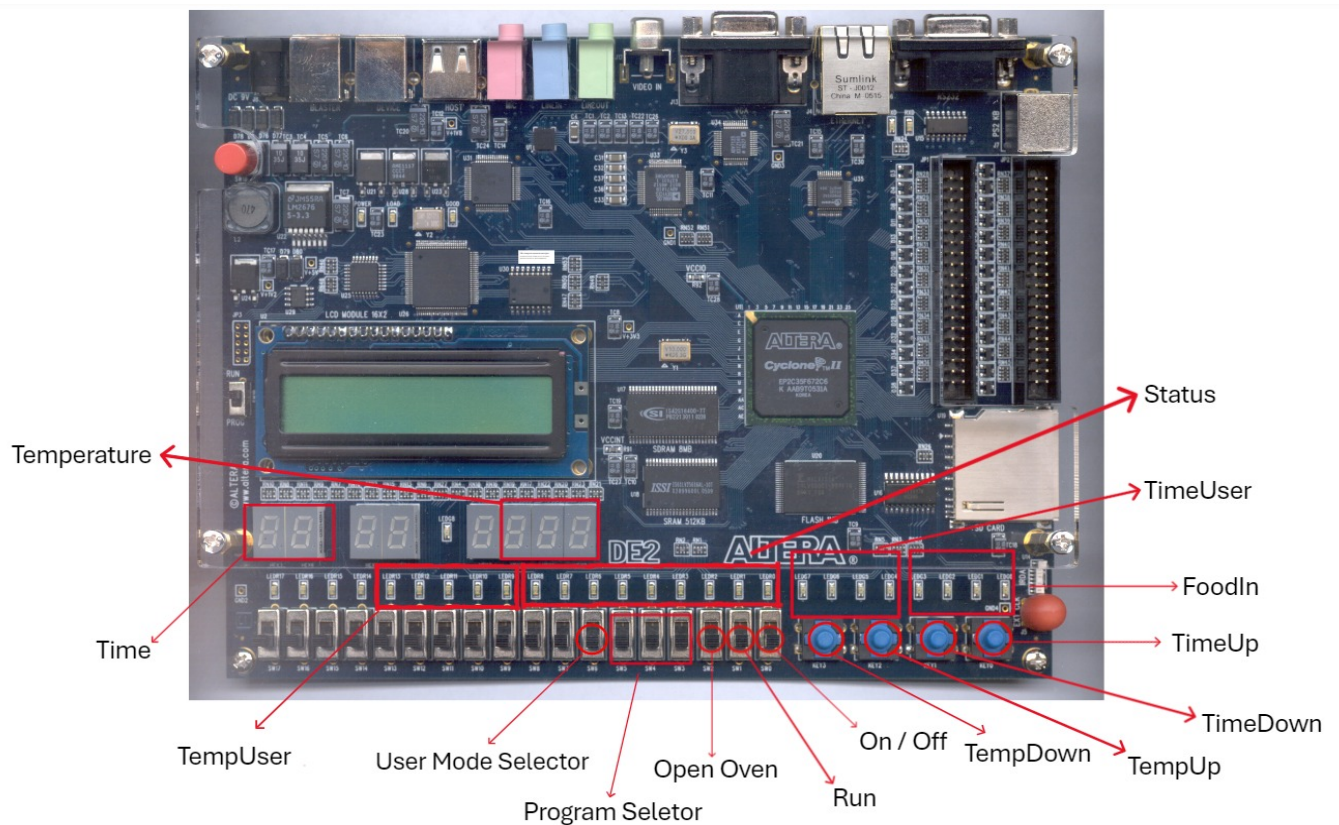


Figura 2.1: Ilustração do Esquema da Máquina

Capítulo 3

Arquitetura e implementação

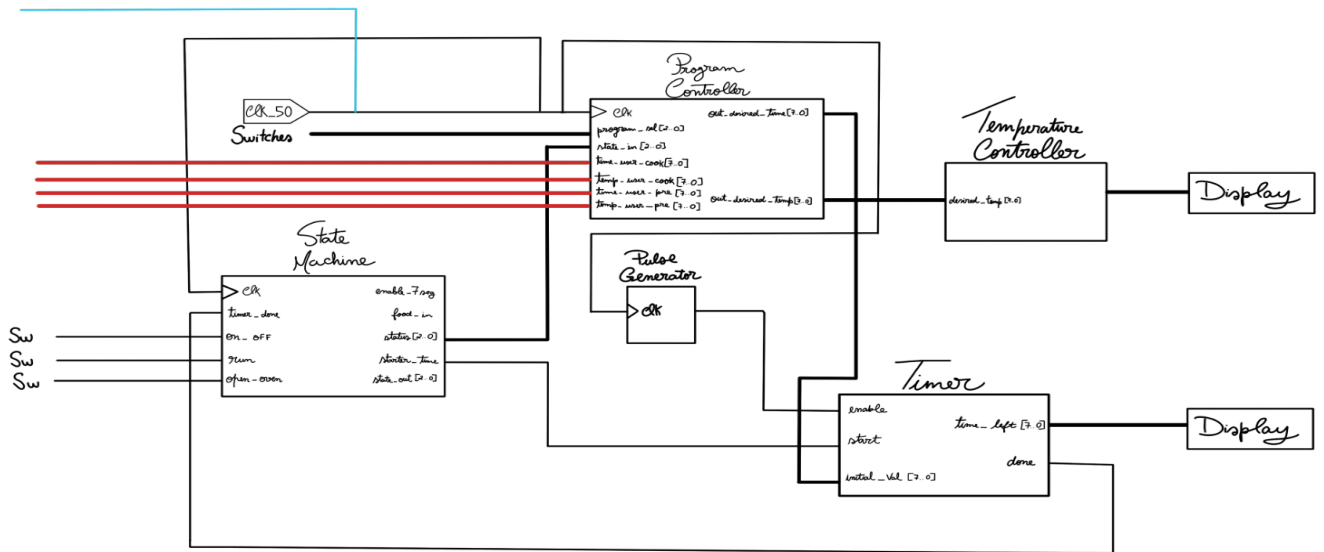


Figura 3.1: Diagrama de Blocos do Projeto

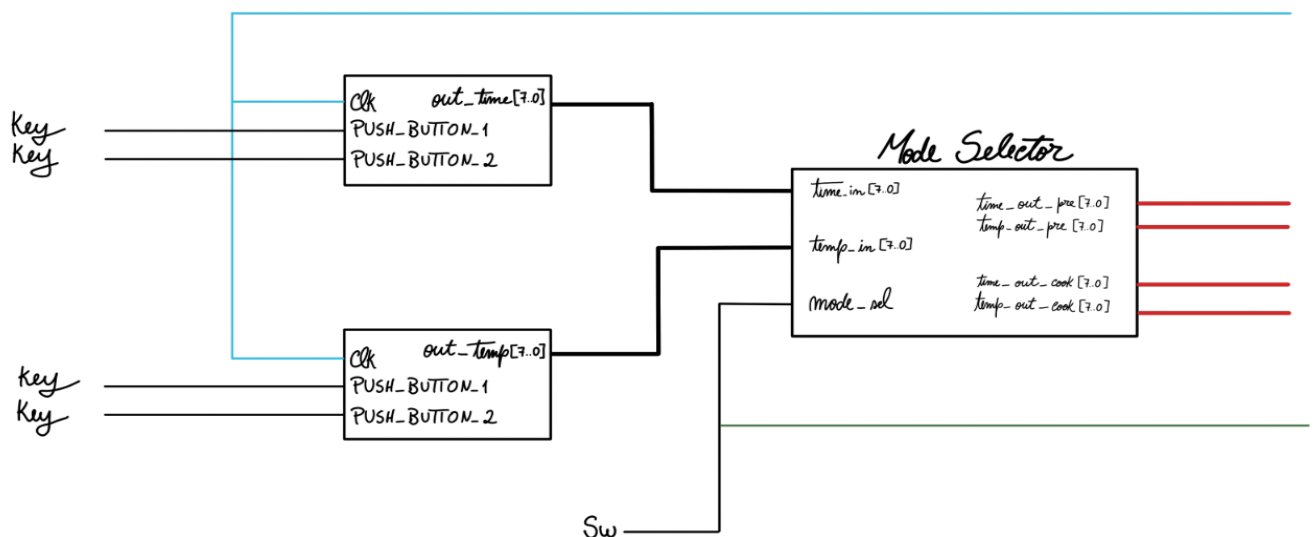


Figura 3.2: Diagrama de blocos do User

3.1 State Machine

A StateMachine é o componente principal da máquina e tem como entradas alguns inputs dados pelo utilizador através de SWs (on/off, run, open_oven), e o timer_done que é um sinal dado pelo Timer à Máquina de Estados que a informa que o tempo a ser contado já terminou.

Como saídas tem o state_out que informa o ProgramController em que estado estamos, o status que serve para ter LEDs na placa a indicar o estado atual, o food_in que indica os momentos em que o utilizador deve retirar a cuba. E ainda o starter_time que envia para o Timer a informação de que este deve começar a contar.

3.2 Timer

O Timer é responsável por receber um sinal que vem da Máquina de Estados (e a partir daí começar a contagem). Recebe ainda um enable que vem de um gerador de pulsos a 1 Hz para que a contagem tenha essa frequência e recebe o valor de tempo desejado a contar que vem do ProgramController (initial_val).

Dá como saídas o time_left que sai para os Displays mostrando o tempo ao utilizador a cada momento. E por fim o done que vai dar a informação à Máquina de Estados que a contagem acabou.

3.3 Program Controller

O Program Controller recebe o estado atual que sai da maquina de estados, recebe o programa escolhido pelo utilizador através de SWs na placa, e com essas duas informações consegue transmitir para a saída o tempo e temperatura desejados a cada momento.

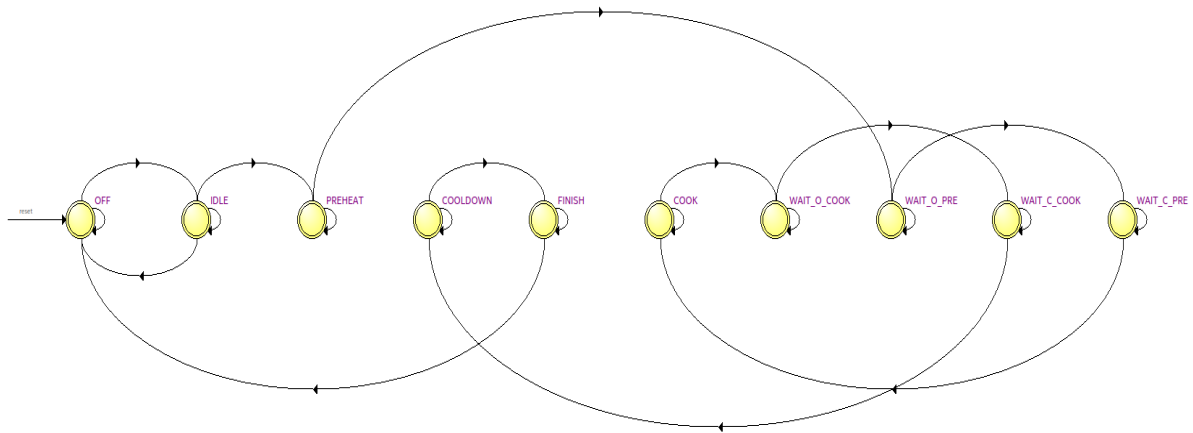


Figura 3.3: Esquema da Máquina de Estados

	Source State	Destination State	Condition
1	COOK	COOK	(!prev_timer_done).(!timer_done) + (prev_timer_done)
2	COOK	WAIT_O_COOK	(!prev_timer_done).(timer_done)
3	COOLDOWN	FINISH	(!prev_timer_done).(timer_done)
4	COOLDOWN	COOLDOWN	(!prev_timer_done).(!timer_done) + (prev_timer_done)
5	FINISH	OFF	(!on_off)
6	FINISH	FINISH	(on_off)
7	IDLE	OFF	(!on_off)
8	IDLE	IDLE	(!run).(on_off)
9	IDLE	PREHEAT	(run).(on_off)
10	OFF	OFF	(!on_off)
11	OFF	IDLE	(on_off)
12	PREHEAT	PREHEAT	(!prev_timer_done).(!timer_done) + (prev_timer_done)
13	PREHEAT	WAIT_O_PRE	(!prev_timer_done).(timer_done)
14	WAIT_C_COOK	WAIT_C_COOK	(!run) + (run).(open_oven)
15	WAIT_C_COOK	COOLDOWN	(run).(lopen_oven)
16	WAIT_C_PRE	COOK	(run).(lopen_oven)
17	WAIT_C_PRE	WAIT_C_PRE	(!run) + (run).(open_oven)
18	WAIT_O_COOK	WAIT_C_COOK	(open_oven)
19	WAIT_O_COOK	WAIT_O_COOK	(lopen_oven)
20	WAIT_O_PRE	WAIT_O_PRE	(lopen_oven)
21	WAIT_O_PRE	WAIT_C_PRE	(open_oven)

Figura 3.4: Sidas da Máquina de Estados

Capítulo 4

Validações

No decorrer do nosso projeto, enfrentamos várias adversidades relacionadas à simulação e validação. Portanto, a principal forma de verificação foi prática, realizada diretamente na placa, já que o trabalho lida principalmente com intervalos de tempo de segundos, o que é difícil de se trabalhar tanto no simulador quanto na testbench.

Capítulo 5

Conclusões e Contribuições

5.1 Conclusões

A AirFryer foi implementada com sucesso, permitindo um controle eficiente do sistema. O projeto atendeu aos objetivos definidos. Durante o desenvolvimento, enfrentamos diversas adversidades, especialmente relacionadas à simulação e validação.

No geral, cumprimos todos os objetivos definidos e ainda implementamos novas funcionalidades. Autoavaliamos nosso trabalho com 18 valores, reconhecendo os desafios enfrentados, mas estando satisfeitos com o resultado final.

5.2 Contribuições dos autores

Neste projeto, a percentagem de participação de cada um dos elementos é: Dinis Batista 5% e Francisco Ribeiro 95%