

Controlo de Radiador (temperaturas programáveis)

Laboratório de Sistemas Digitais

Bernardo Marujo, Diogo Carvalho
(107332) bernardomarujo@ua.pt, (113221) diogo.tav.carvalho@ua.pt

UNIVERSIDADE DE AVEIRO

Aveiro, maio 2023

Introdução

Este projeto visa desenvolver um controlador para um radiador utilizando o kit de desenvolvimento Altera DE2-115. O objetivo é criar um sistema realista que permita controlar a ativação do radiador de forma eficiente e programável. O sistema utiliza o princípio de controle ON/OFF, onde um único sensor de temperatura ambiente é constantemente comparado com um valor de referência para decidir quando ligar ou desligar o radiador.

Funcionalidades:

Controle ON/OFF: O sistema utiliza um elemento de aquecimento controlado em modo ON/OFF, com uma janela de histerese para evitar comutações excessivas. A temperatura ambiente medida é comparada com a temperatura de referência, e o radiador é ligado ou desligado com base nessa comparação.

Programação horária: O sistema permite a programação da temperatura de referência ao longo do dia.

Interface com o utilizador: O sistema possui botões para os comandos de programação, ajuste, incremento e decremento. O relógio também pode ser acertado pelo utilizador. Os valores de temperatura e comandos são exibidos em displays e LEDs, proporcionando uma interação intuitiva.

Unidade de controlo: É implementado numa unidade de controlo a execução e lógica de controlo e processamento dos sinais, enviando um sinal de ativação para o simulador de ambiente.

O projeto foi dividido em fases para garantir a implementação adequada de cada funcionalidade. O objetivo final foi construir um sistema eficiente e programável que possa controlar o radiador de forma precisa e intuitiva.

Manual de Utilização

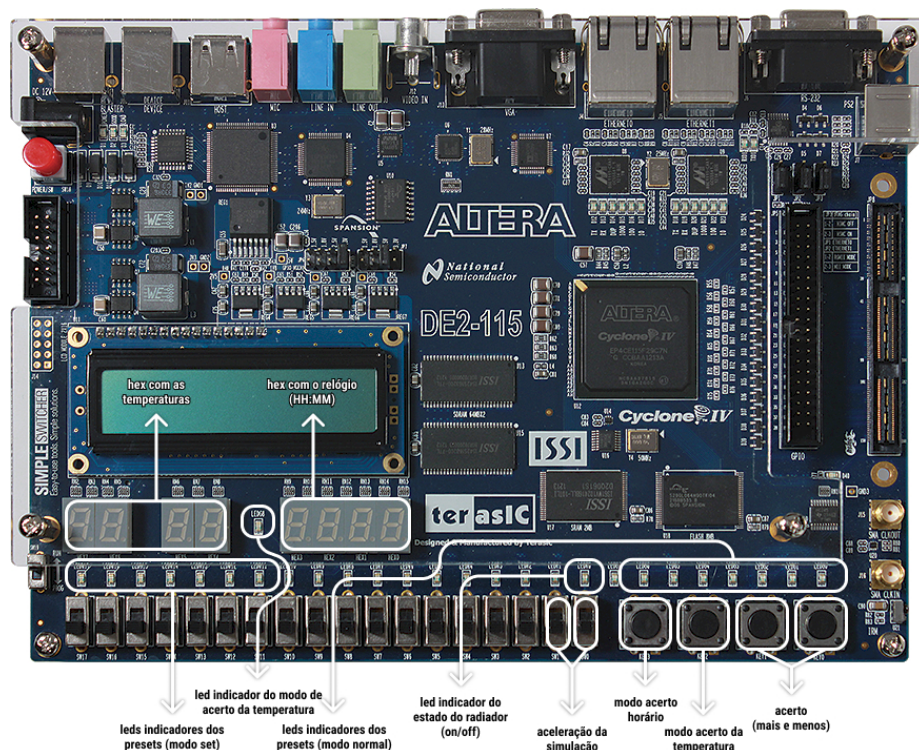


Figura 1: FPGA DE2-115 com as funções implementadas pelo projeto

Funcionalidades a realçar:

1. Os botões - e + (KEY0 e KEY1, respetivamente), funcionam de forma diferente, consoante o modo. No modo P (acerto das temperaturas), é recebido apenas um pulso por clique, enquanto que no modo S (acerto das horas), é recebido um pulso por clique durante o primeiro segundo, e são recebidos dez pulsos por segundo a partir do primeiro segundo, para facilitar o acerto. Foi implementado assim, depois de algumas considerações, tendo sido decidido deixar o modo P com apenas um pulso por segundo, sem toques longos, visto que os acertos são menos frequentes e mais pequenos.
2. Os switches são responsáveis pela aceleração da simulação. As diferentes combinações mudam a velocidade entre a tempo real, 60x, 600x, e 3600x.
3. Os hex indicadores da temperatura mostram a temperatura ambiente durante uso normal, mas mostram as temperaturas que estão a ser alteradas,

quando em modo de acerto, indicando o preset ativo pelos leds vermelhos, e o led verde 8.

4. O radiador liga e desliga automaticamente, com uma histerese fixa de cerca de 2.8°C (1.4 acima da referência, e 1.4 abaixo). O estado (on/off) é indicado pelo led vermelho 0.
5. Durante o acerto do relógio, o ecrã pisca para notificar o utilizador. Um primeiro toque muda para o acerto das horas, e um segundo toque muda para o acerto dos minutos.
6. O simulador ambiente, quando o radiador está ativo, sobe ao dobro da velocidade de quando está desligado, a descer, adicionando um elemento de detalhe e profundidade.

Arquitetura e Implementação

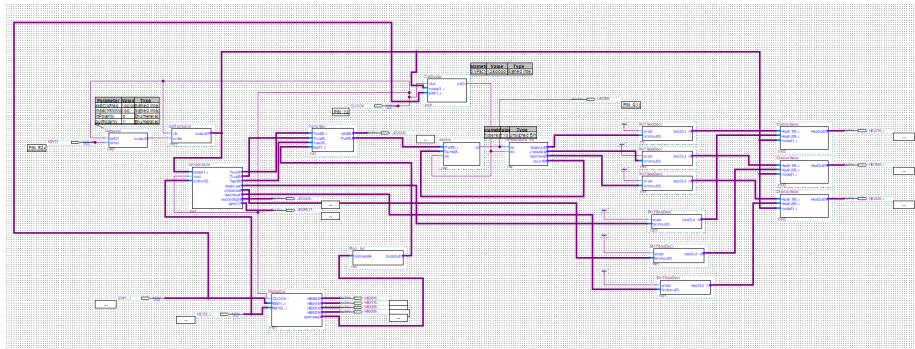


Figura 2: Diagrama de funcionamento do filtro de média móvel.

De uma forma simplista, o projeto é controlado por uma unidade central de controle (uma máquina de estados) que controla um gerador de endereços ligado diretamente à ROM, um registo, responsável por armazenar momentaneamente os dados obtidos da ROM e enviá-los para uma unidade aritmética capaz de efetuar os cálculos de filtragem do sinal recebido da ROM. Em seguida, este sinal devidamente filtrado é guardado na RAM.

Tanto os valores armazenados na ROM como os na RAM passam por módulos que os preparam para serem mostrados nos displays de 7 segmentos do kit Terasic DE2-115 como números inteiros com sinal em base 10.

SetClock

Componente responsável pelo modo atualmente ativo, ou seja, modo normal, modo do acerto das horas, e modo do acerto dos minutos. Gera apenas um output, que é depois interpretado noutro componente.

Counter4Bits

Componente responsável por contar as horas, para cima e para baixo, seja com o clock, num tempo determinado, seja quando é pedido pelo utilizador, através dos botões.

HourJoin

Componente responsável por juntar as unidades e dezenas, para que depois sirva como endereço para escolher o modo de temperatura programado para cada hora, guardado na rom.

HexEnable

Componente responsável pelo piscar dos displays, quando estão no modo de acerto. Isto foi implementado para que o utilizador saiba, facilmente, se está no modo normal, se está a acertar as horas, ou se está a acertar os minutos.

SetTemperature

Componente responsável pelo modo atualmente ativo, ou seja, modo normal, modo do acerto do TSol, modo acerto do TLua, e modo do acerto do TGel. Gera apenas um output, que é depois interpretado noutro componente, tal como no SetClock.

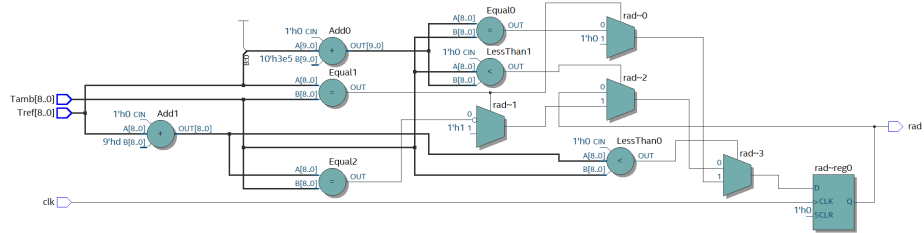


Figura 5: RTL viewer do Controlo ON/OFF.

DisplaySelector

Componente responsável por escolher quais temperaturas são mostradas nos hex (7 a 4). Ou seja, escolhe se mostra a temperatura ambiente, ou as demais temperaturas, durante o acerto.

TemperatureCounter

Componente responsável por simular a temperatura ambiente. Aumenta mais rapidamente do que diminui, pelas razões descritas acima.

TemperatureKey

Componente responsável pelos ajustes da temperatura.

TempSelect

Componente responsável pela escolha da temperatura.

RomTemp

Componente responsável pelo armazenamento do programa horário.

Validações

Nesta secção estão exemplos de como as simulações dos componentes foram feitas. São mostradas nas imagens seguintes, as testbenches da máquina de

estado usada no botão do relógio que é usado para definir se estamos a alterar as horas ou os minutos e da ROM, onde são armazenadas as temperaturas de referência a serem usadas nas várias horas do dia.

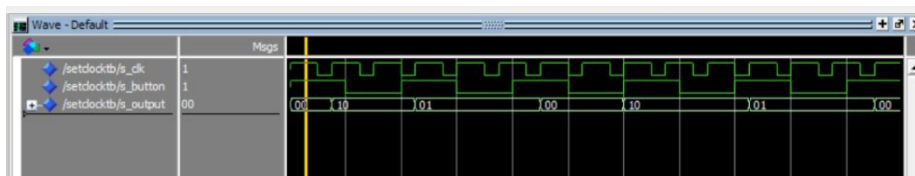


Figura 6: Simulação da máquina de estados do botão do Relógio.

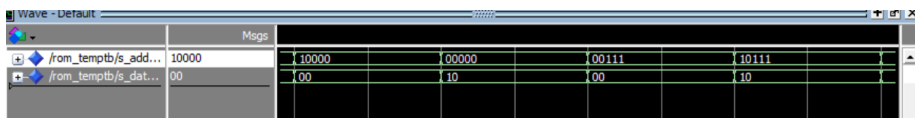


Figura 7: Simulação da ROM.

Conclusão

De acordo com o esperado, todas as funcionalidades solicitadas no enunciado foram implementadas com sucesso no projeto. Todas as dificuldades enfrentadas durante o planeamento e desenvolvimento foram superadas graças ao empenho e comprometimento dos responsáveis. Dado o esforço, dedicação e tempo que foi necessário gastar para realizar o projeto, e tendo em consideração o facto de ter sido necessário negligenciar outras cadeiras para conseguir concluir tudo a tempo, auto-avaliámos este trabalho com 20.

Contribuições dos autores

Ambos os autores contribuíram ativamente e dedicaram-se totalmente à realização deste projeto. Desta forma, a participação neste trabalho é distribuída de forma equilibrada, com cerca de 50% para cada membro do grupo.