

**UNIVERSIDADE DO EXTREMO SUL CATARINENSE - UNESC
CURSO DE CIÊNCIA DA COMPUTAÇÃO**

MARCELO MINATO REZIN

**SISTEMAS EMBARCADOS UTILIZANDO FPGA: ESTUDO DE CASO NO
CONTROLE DE TEMPERATURA NA ETAPA DE BRASSAGEM**

**CRICIÚMA
2020**

MARCELO MINATO REZIN

**SISTEMAS EMBARCADOS UTILIZANDO FPGA: ESTUDO DE CASO NO
CONTROLE DE TEMPERATURA NA ETAPA DE BRASSAGEM**

Trabalho de Conclusão de Curso, apresentado para obtenção do grau de Bacharel no curso de Ciência da Computação da Universidade do Extremo Sul Catarinense, UNESC.

Orientador: Prof. Me. Paulo João Martins
Coorientador: Prof. Me. Emerson Colonetti

CRICIÚMA

2020

MARCELO MINATO REZIN

SISTEMAS EMBARCADOS UTILIZANDO FPGA: ESTUDO DE CASO NO CONTROLE DE TEMPERATURA NA ETAPA DE BRASSAGEM

Trabalho de Conclusão de Curso aprovado pela Banca Examinadora para obtenção do Grau de Bacharel, no Curso de Ciência da Computação da Universidade do Extremo Sul Catarinense, UNESC, com Linha de Pesquisa em Sistemas Embarcados.

Criciúma, 05 de agosto de 2020.

BANCA EXAMINADORA

DocuSigned by:

Paulo João Martins

2C63A5088FBE4A5...

Prof. Paulo João Martins - Mestre - UNESC – Orientador

DocuSigned by:

Emerson Colonetti

10493B10B2E74CA...

Prof. Emerson Colonetti - Mestre - UNESC - Coorientador

DocuSigned by:

Ênio José Peruchi

0D7C4B8362DA461...

Prof. Ênio José Peruchi - Especialista – UNESC

DocuSigned by:

Gilberto Vieira da Silva

DA5D215462F64BA...

Prof. Gilberto Vieira da Silva - Especialista - UNESC

Aos meus pais.

AGRADECIMENTOS

Agradeço aos meus pais e ao meu irmão pelo apoio em todas as fases da jornada acadêmica, bem como colegas e amigos que estiveram presentes durante esse período e que contribuíram de alguma forma.

Também agradeço ao governo federal, que por meio da bolsa PROUNI viabilizou a conclusão de mais essa etapa da minha vida.

Meu agradecimento também ao meu orientador e professor Paulo João Martins, que aceitou o desafio e me auxiliou nesse processo. Agradeço também ao meu coorientador Emerson Colonetti, que com seu conhecimento e experiência contribuiu para a qualidade desta pesquisa.

“Se a aparência e a essência das coisas coincidissem, a ciência seria desnecessária.”

Karl Marx

RESUMO

Os sistemas embarcados são amplamente difundidos nos ambientes residenciais e industriais. Essa alta demanda torna necessário a busca por alternativas que sejam versáteis e flexíveis, como os circuitos FPGA. Essa tecnologia tem a principal característica de ser um circuito programável, o que permite um bom aproveitamento dos recursos, aliado ao alto desempenho. Dado o grande potencial de aplicação dos FPGAs, o controle de temperatura é um processo importante em vários segmentos, e que pode usufruir das suas capacidades. Na produção de cerveja, esse tipo de controle representa um importante papel, e influencia diretamente nas características da bebida, como na conversão de amido em açúcares fermentáveis. Com base nisso, este trabalho propôs a construção de um sistema embarcado baseado em FPGA para o controle de temperatura na etapa de brassagem no processo de produção de cerveja. Para tal foi utilizada uma placa FPGA de prototipação, juntamente com a linguagem de descrição VHDL para implementar o modo de controle proporcional derivativo, que a partir da aferição da temperatura por meio de um sensor, realizava o ajuste do sistema de aquecimento por controle de fase. Também foi desenvolvido uma aplicação desktop para ser utilizada com interface humano máquina, e que realizava a troca de informações com o protótipo por comunicação serial. Foram realizados três testes de produção de cerveja com o protótipo, em que foram obtidos os dados referentes a temperatura e a transformação do amido para que fosse possível avaliar seu desempenho. Com base nos resultados alcançados, se concluiu que os objetivos propostos foram atingidos.

Palavras-chave: Sistemas embarcados. FPGA. Controle de temperatura. Brassagem.

ABSTRACT

Embedded systems are widespread in residential and industrial environments. This high demand makes it necessary to search for alternatives that are versatile and flexible, such as FPGA circuits. This technology has the main characteristic of being a programmable circuit, which allows a good use of resources, combined with high performance. Given the great potential for application of FPGAs, temperature control is an important process in several segments, and that can take advantage of its capabilities. In beer production, this type of control plays an important role, and directly influences the characteristics of the drink, as in the conversion of starch into fermentable sugars. From that, this research proposed the build of an FPGA based embedded system for temperature control in the mashing step in the brewing process. For this purpose, a prototype FPGA board was used, together with the description language VHDL to implement the derivative proportional control mode, which from the measurement of temperature by means of a sensor, performed the adjustment of the heating system by phase control. A desktop application was also developed to be used as interface human machine, and that performed the exchange of information with the prototype by serial communication. Three beer production tests were made with the prototype, in which the temperature and starch transformation data were obtained so that it was possible to evaluate its performance. With base on the results achieved, were concluded that the proposed objectives were reached.

Keywords: Embedded systems. FPGA. Temperature control. Mashing.

LISTA DE ILUSTRAÇÕES

Figura 1 – Exemplo dos degraus de temperatura no processo de brassagem.	12
Figura 2 – Diagrama de blocos de um sistema de controle retroativo para o processo de brassagem.....	7
Figura 3 – Comparação entre a tensão da fonte CA e a tensão da entre à carga ao longo do tempo.....	11
Figura 4 – Formatos de onda para cada potência aplicada à carga e do disparo do TRIAC.....	12
Figura 5 – Estrutura simplificada de um circuito FPGA.	14
Figura 6 – Estrutura de um projeto VHDL.	15
Figura 7 – Projeto VHDL de uma porta and.	16
Figura 8 – Simulação dos estímulos de uma entidade VHDL.	17
Figura 9 – Transmissão do caractere A em UART.....	18
Figura 10 – Recursos incorporados à placa de prototipação DE10-Lite.	24
Figura 11 – Apresentação da panela.	25
Figura 12 – Circuito para detecção da passagem de zero.....	26
Figura 13 – Circuito para controle de fase.....	26
Figura 14 – Termopar tipo K e módulo MAX6675.	27
Figura 15 – Sonda para aferição da temperatura.....	28
Figura 16 – Código par utilização do Arduino como conversor.	28
Figura 17 – Controlador e circuitos do protótipo.....	29
Figura 18 – Representação do software do protótipo.....	30
Figura 19 – Apresentação do Intel Quartus Prime.	30
Figura 20 – Simulação de código VHDL pelo Modelsim.....	31
Figura 21 – Simulação da entidade responsável pela leitura da temperatura.	32
Figura 22 – Cálculo da porcentagem de potência a partir do cálculo do PID.	32
Figura 23 – Diagrama de caso de uso da IHM.	34
Figura 24 – Tela para edição da receita.....	34
Figura 25 – Código responsável pela persistência das receitas.....	35
Figura 26 – Tela de acompanhamento do processo.	36
Figura 27 – Sequência de bytes enviados pelo controlador à IHM.	36
Figura 28 – Rampas de temperatura da receita utilizada nos testes.....	37
Figura 29 – Temperatura de acordo com o tempo no primeiro teste.....	38

Figura 30 – Taxa de açúcar de acordo com o tempo no primeiro teste.	39
Figura 31 – Temperatura de acordo com o tempo no segundo teste.....	39
Figura 32 – Taxa de açúcar de acordo com o tempo no segundo teste.....	40
Figura 33 – Temperatura de acordo com o tempo no terceiro teste.....	40
Figura 34 – Taxa de açúcar de acordo com o tempo no terceiro teste.	41

LISTA DE TABELAS

Tabela 1 – Principais enzimas e suas temperaturas de ativação.....	11
Tabela 2 – Composição e faixa de operação dos tipos de termopares.....	10

LISTA DE ABREVIATURAS E SIGLAS

ASCII	American Standard Code for Information Interchange
ASIC	Application Specific Integrated Circuits
CA	Corrente Alternada
CI	Circuito Integrado
CPU	Central Processing Unit
DIAC	Diode for Alternating Current
FAN	Free Amino Nitrogen
FPGA	Field Programmable Gate Array
HDL	Hardware Description Language
IHM	Interface Humano-Máquina
IDE	Integrated Development Environment
LED	Ligth-Emitting Diode
LUT	LookUp Table
MOS	Metal Oxide Semiconductor
NTC	Negative Temperature Coeficient
PID	Proporcional Integral Derivativo
PTC	Positive Temperature Coeficient
RAM	Random Access Memory
RTD	Resistance Temperature Detector
RX	Receptor
SCR	Silicon Controlled Rectifier
SRAM	Static Random Access Memory
TRIAC	Triode for Alternating Current
TX	Transmissor
UART	Universal Asynchronous Receiver-Transmitter
USB	Universal Serial Bus
VHDL	VHSIC Hardware Description Language
VHSIC	Very-High-Speed Integrated Circuit

SUMÁRIO

1 INTRODUÇÃO	6
1.1 OBJETIVO GERAL.....	7
1.2 OBJETIVOS ESPECÍFICOS	7
1.3 JUSTIFICATIVA	8
1.4 ESTRUTURA DO TRABALHO	8
2 PROCESSO DE PRODUÇÃO DE CERVEJA	10
2.1 ETAPAS DA PRODUÇÃO.....	10
2.2 BRASSAGEM.....	11
3 CONTROLE DE TEMPERATURA	7
3.1 MODOS DE CONTROLE	7
3.2 MEDIÇÃO DE TEMPERATURA.....	8
3.2.1 Termorresistores	9
3.2.2 Termopares	9
3.3 SISTEMA DE AQUECIMENTO ELÉTRICO	10
3.4 CONTROLE DE POTÊNCIA EM CORRENTE ALTERNADA	11
4 SISTEMAS EMBARCADOS	13
4.1 FPGA.....	13
4.2 VHDL.....	15
4.2.1 Testes	17
4.3 COMUNICAÇÃO SERIAL	17
4.4 INTERFACE HUMANO-MÁQUINA	19
5 TRABALHOS CORRELATOS	20
5.1 CONTROLE DE TEMPERATURA POR MEIO DE COMPUTAÇÃO EMBARCADA	20
5.2 DESENVOLVIMENTO DE DISPOSITIVO DE CONTROLE AUTOMÁTICO PARA OS PROCESSOS DE MOSTURAÇÃO E FERVURA DA FABRICAÇÃO DE CERVEJA ARTESANAL	20
5.3 SISTEMA DE CONTROLE DE TEMPERATURA USANDO FPGA.....	21
5.4 INSTRUMENTAÇÃO E CONTROLE DO PROCESSO DE PRODUÇÃO DE UMA MICROCERVEJARIA.....	21
5.5 DESIGN AND IMPLEMENTATION OF DIGITAL PID CONTROLLER USING FPGA FOR PRECISION TEMPERATURE CONTROL.....	22

6 SISTEMAS EMBARCADOS UTILIZANDO FPGA: ESTUDO DE CASO NO CONTROLE DE TEMPERATURA NA ETAPA DE BRASSAGEM.....	23
6.1 METODOLOGIA.....	23
6.1.1 Definição e construção dos componentes	23
6.1.1.1 Escolha da placa FPGA	24
6.1.1.2 Escolha da panela	24
6.1.1.3 Controle de potência	25
6.1.1.4 Aferição da temperatura	27
6.1.1.5 Comunicação serial	28
6.1.1.6 Montagem do protótipo.....	28
6.1.2 Desenvolvimento do software.....	29
6.1.2.1 Descrição em VHDL	30
6.1.2.2 Programação da IHM	33
6.1.3 Testes com o protótipo.....	37
6.2 RESULTADOS	37
6.2.1 discussão dos resultados	41
7 CONCLUSÃO	43
REFERÊNCIAS.....	45

1 INTRODUÇÃO

Várias aplicações, desde as mais simples, como máquinas de café até as mais complexas, como dispositivos de segurança automotiva, demandam de sistemas computacionais que realizam funções dedicadas (NULL; LOBUR, 2011). Para suprir essa necessidade são utilizados os sistemas embarcados, que, diferentemente dos computadores de propósito geral atuam de forma mais específica, como parte de um dispositivo completo (LAMB, 2015).

Sistemas embarcados atuam muitas vezes em ambientes com poucos recursos, como hardware de baixo desempenho e limitações no consumo de energia. Outra dificuldade são o local onde estes sistemas estão inseridos que por vezes são hostis e, em alguns casos, possuem severas restrições de tempo real (NULL; LOBUR, 2011).

Uma alternativa para atender os requisitos de hardware dos sistemas embarcados são os circuitos integrados de aplicação específica, do inglês, *Application Specific Integrated Circuits* (ASIC). Esses circuitos são desenvolvidos especificamente para uma finalidade e possuem todos os recursos necessários, o que torna os custos de produção mais baixos, porém com a desvantagem de tempo de desenvolvimento e custo do projeto elevados (BRAGA, 2017).

Outra opção são microcontroladores, que possuem todo o hardware necessário para execução dos programas, como CPU e memória RAM, alocados em um único componente (BERTINI, 2010). Esse chip permite que as funções necessárias sejam programadas, porém, em alguns casos, esses componentes não possuem todos os recursos necessários demandando de circuitos externos, ou possuem funções a mais, o que causa desperdício (BRAGA, 2017).

Tendo em vista as desvantagens das alternativas apresentadas, uma solução cada vez mais aplicada é o arranjo de portas programáveis em campo, do inglês, *Field Programmable Gate Array* (FPGA). Este circuito é composto por um conjunto de blocos lógicos programáveis organizados em forma de matriz, o que atribui a eles a característica de ser um circuito programável, permitindo o desenvolvimento de qualquer circuito lógico, evitando a falta ou a sobra de recursos, com a vantagem da reconfiguração do arranjo quando necessário (ORDONEZ et al., 2003).

Para definir os circuitos lógicos de um FPGA utiliza-se uma linguagem de descrição de hardware, do inglês, Hardware Description Language (HDL), onde, a linguagem de projeto de hardware para circuitos integrados de velocidade muito alta, do inglês VHSIC *Hardware Description Language*, onde VHSIC é a sigla de *Very High Speed Integrated Circuits* (VHDL) é a padronizada para descrever os componentes digitais, sendo um padrão internacional, e possui uma grande capacidade de modelagem e um alto nível de abstração (ORDONEZ et al., 2003).

Dentre as várias aplicações possíveis dos sistemas embarcados, o controle de temperatura representa um ponto importante de vários processos, tanto industriais quanto residenciais. Na produção de cerveja é um controle fundamental para etapa de brassagem. Por meio do controle de temperatura o mestre cervejeiro tem a capacidade de regular a ativação de certas enzimas que são responsáveis por características fundamentais da bebida, como teor alcoólico e corpo (VENTURINI FILHO, 2018).

Dessa forma, este trabalho aqui proposto visa utilizar o FPGA em uma aplicação embarcada de controle de temperatura na etapa de brassagem no processo de produção de cerveja.

1.1 OBJETIVO GERAL

Desenvolver um protótipo de sistema embarcado utilizando FPGA para o controle de temperatura no processo de brassagem da produção de cerveja.

1.2 OBJETIVOS ESPECÍFICOS

Os objetivos específicos são:

- a) compreender o conceito de sistema embarcado;
- b) estudar o processo de brassagem na produção de cerveja;
- c) pesquisar meios de controle de temperatura;
- d) compreender o processo de desenvolvimento utilizando VHDL;
- e) desenvolver um controlador baseado em FPGA para o processo proposto.

1.3 JUSTIFICATIVA

Os sistemas embarcados atuam, segundo Almeida, Moraes e Seraphim (2017) em ambientes com restrições de recursos, tanto de hardware quando físicos. Isso torna necessário a utilização de sistemas computacionais que sejam eficientes e ao mesmo tempo tenham baixo custo de projeto e produção.

A utilização de FPGAs permite que vários circuitos lógicos diferentes sejam implementados no mesmo chip de acordo com a necessidade e, possibilita a sua alteração quando preciso sem a substituição do hardware ou desenvolvimento de um novo projeto. O consumo de energia segundo George e Rabaey (2012, tradução nossa) pode ser melhorado com a otimização dos circuitos desenvolvidos, aumentando a possibilidade de aplicações.

A flexibilidade proporcionada pelos chips FPGAs pode ser uma alternativa a alta demanda de embarcados que tende a aumentar com a popularização da Internet das coisas. Segundo Santos (2018) estimativas apontam que até 2020 haverá 50 bilhões de dispositivos conectados.

Já o controle de temperatura, é um recurso essencial de vários processos onde diferenças de cerca 1°C podem ser prejudiciais (IBRAHIM, 2002, tradução nossa). No processo de produção de cerveja esse tipo de controle na etapa de brassagem é um fator importante e influência diretamente na bebida final.

Ao utilizar FPGA em um sistema para controle de temperatura no processo de brassagem, é possível fazer uso das vantagens da arquitetura em prol da qualidade do produto. O paralelismo da execução das instruções, garante um controle estável, obedecendo aos requisitos do mestre cervejeiro, juntamente com a característica da implementação a nível de circuito, que provê o desempenho necessário para os cálculos e comparações dos métodos de controle.

1.4 ESTRUTURA DO TRABALHO

A estrutura do trabalho está organizada em sete capítulos. No primeiro é abordado a introdução sobre a pesquisa, juntamente com a apresentação do objetivo geral, dos objetivos específicos e da justificativa.

No segundo capítulo é descrito sobre os processos de produção da cerveja, sendo abordado em maiores detalhes a etapa de brassagem, um dos focos do trabalho.

No terceiro capítulo é apresentado o controle de temperatura, onde é apontado os métodos de controle e apresentado uma visão sobre sensores de temperatura, sistemas de aquecimento elétrico e controle de potência em corrente alternada.

O quarto capítulo aborda uma contextualização sobre sistemas embarcados, juntamente com uma visão mais detalhada sobre FPGA e VHDL, também sendo abordados elementos da comunicação serial e de interface humano-máquina.

No quinto capítulo são apresentados trabalhos relacionados a este, que abordam alguns objetivos que este projeto de pesquisa também contém.

O sexto capítulo relata cada etapa do desenvolvimento do protótipo e dos testes realizados, juntamente com os resultados obtidos e a comparação com os trabalhos correlatos.

Por fim, o sétimo capítulo é composto pela conclusão do projeto e apresentação de sugestões de trabalhos futuros.

2 PROCESSO DE PRODUÇÃO DE CERVEJA

A cerveja é uma antiga bebida alcoólica com altos números de consumo atualmente. Sendo composta principalmente por cereais maltados ou não, água, lúpulo e levedura, possui várias etapas em seu processo de produção, que garantem seu aroma e sabor característico (VENTURINI FILHO, 2018).

2.1 ETAPAS DA PRODUÇÃO

A produção de cerveja é composta de vários processos que devem ser seguidos para que se obtenha o produto desejado. A primeira etapa é chamada de moagem, nela os cereais, que podem ser maltados ou não, composto por cevada pura ou misturada com outros cereais, tem sua casca quebrada, com o objetivo de expor seus amidos, para que se dê início à fabricação da bebida (NACHEL; ETTLINGER; BRAGA, 2018).

O próximo passo é a etapa de brassagem. Nesta etapa os grãos moídos anteriormente são infundidos em água para converter os amidos do malte em açúcares naturais, formando-se o mosto, que posteriormente será fermentado (NACHEL; ETTLINGER; BRAGA, 2018). Esse processo necessita de um rigoroso controle de temperatura e tempo, sendo o foco deste trabalho, e, será abordado mais profundamente.

Após a brassagem inicia-se o processo de fervura dos ingredientes para realizar a esterilização do mosto. Nesta etapa também é adicionado lúpulo à mistura para controlar o aroma e o amargor da bebida final (GURGEL; CUNHA, 2019).

Ao fim da fervura, é necessário resfriar o mosto para criar um ambiente propício as leveduras. Estes fungos são responsáveis pela fase de fermentação, onde o mosto é colocado em um recipiente que pode ser vedado ou não, para que elas transformem os açúcares em etanol e gás carbônico (NACHEL; ETTLINGER; BRAGA, 2018).

Com o término da fermentação, se inicia o processo de maturação, nele ocorre o envelhecimento da bebida, sendo que o tempo pode variar de acordo com o estilo e receita. Ao fim da maturação, pode-se embalar a cerveja para que seja distribuída (NACHEL; ETTLINGER; BRAGA, 2018).

2.2 BRASSAGEM

O processo de brassagem, também conhecido como mosturação, segundo Müller (2002), tem o objetivo de transformar o amido extraído do malte em açúcares por meio da quebra de suas moléculas pela adição de água (hidrólise) e ação de enzimas, para que posteriormente sejam fermentados.

Existem várias enzimas que podem ser utilizadas no processo, e, suas respectivas ações podem ser favorecidas pela faixa de temperatura utilizada e níveis de pH. O ajuste de temperatura e consequente ativação de enzima permite ao cervejeiro, de acordo com a receita ou o seu gosto, destacar características da bebida, como uma maior quantidade de açúcares que podem ser fermentados, tornando-a mais alcoólica, ou maior quantidade de substâncias com mais peso molecular, tornando-a mais encorpada (VENTURINI FILHO, 2018).

A Tabela 1 apresenta as principais enzimas presentes no processo, suas faixas de temperatura ideais e funções:

Tabela 1 – Principais enzimas e suas temperaturas de ativação.

Enzima	Melhor faixa de temperatura	Função
Beta Glucanase	35° - 45° C	Melhor quebra da goma.
Peptidase	45° - 55° C	Produz amino livre de nitrogênio (FAN).
Protease	45° - 55° C	Quebra proteínas grandes da cevada.
Beta Amilase	55° - 65° C	Produz maltose.
Alfa Amilase	67° - 72° C	Produz vários tipos de açúcar, incluindo maltose.

Fonte: Palmer (2006).

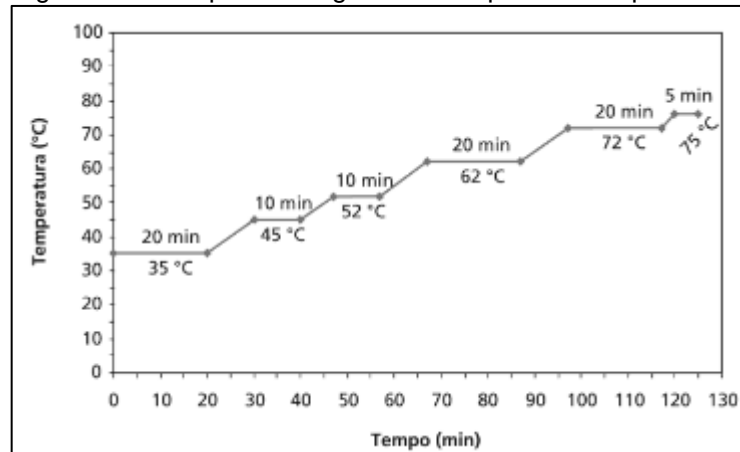
De acordo com as faixas para cada enzima, o mosto é submetido a diferentes rampas de temperatura para extrair os açúcares e nutrientes necessários para o processo de fermentação, como exibido na Figura 1. A primeira enzima a ser ativada é a beta glucanase, que auxilia na quebra da goma da cevada não maltada, para a liberação de outras enzimas (PALMER, 2006).

Em seguida a temperatura do mosto é elevada até a faixa de 45° à 55° C para solubilizar as várias proteínas e enzimas da cevada que não são solúveis naturalmente. Nessa faixa entra em a ação as enzimas peptidase, que fornece à

mistura os nutrientes que serão consumidos pelas leveduras e a protease, que quebra grandes proteínas (PALMER, 2006).

A partir desse ponto, a temperatura do mosto é elevada até valores entre 55° e 65° C para que a enzima beta amilase entre em ação. Essa enzima produz apenas açúcares que podem ser fermentados, e de acordo com o tempo de ação é possível controlar o teor alcoólico da bebida final. A última enzima a ser ativada é a alfa amilase, na faixa que varia entre 67° e 72°C. A ação alfa amilase produz açúcares que podem ou não ser fermentados, contribuindo para o corpo da cerveja (PALMER, 2006). Após a conclusão da ação da alfa amilase, o mosto é aquecido a temperaturas por volta dos 76°C a fim de desativar a ação das enzimas (VENTURINI FILHO, 2018).

Figura 1 – Exemplo dos degraus de temperatura no processo de brassagem.



Fonte: Venturini Filho (2018).

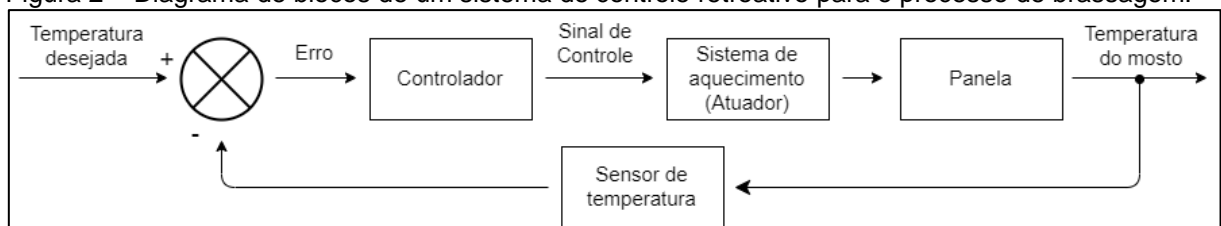
Após o término da brassagem, o mosto já possui os nutrientes suficientes para alimentar as leveduras e pode seguir para as demais fases do processo de produção.

Essa etapa da fabricação tem um grande peso nas características do produto, o que torna de suma importância o controle da temperatura de modo contínuo. Com será visto a seguir, a partir de um sistema de controle retroativo é possível realizar tal ajuste de acordo com o as necessidades do mestre cervejeiro e sem que haja necessidade de interferência humana.

3 CONTROLE DE TEMPERATURA

O controle de temperatura é um processo essencial em diversas áreas, e representa um papel importante na produção da cerveja. Esse tipo de controle é chamado de retroativo (Figura 2), em que, a temperatura medida por meio de um sensor é enviada a um controlador, que compara o valor obtido com um parâmetro de referência, e de acordo com o resultado, age para influenciar a variável por meio de um atuador (FRANKLIN; POWELL; EMAMI-NAEINI, 2013).

Figura 2 – Diagrama de blocos de um sistema de controle retroativo para o processo de brassagem.



Fonte: Do autor.

3.1 MODOS DE CONTROLE

Um controlador pode agir sobre a temperatura a partir de dois modos de operação básicos: A ação de *liga/desliga* e a ação contínua. Esses modos determinam o sinal de controle para garantir que a variável medida se mantenha dentro dos limites estabelecidos (DUNN, 2013).

A ação *liga/desliga* é a maneira mais simples de controle. Nesse método o sinal de saída pode assumir *ligado* ou *desligado*. A geração desse sinal se dá a partir da comparação da variável monitorada com uma referência previamente estabelecida, e, em um sistema de aquecimento, caso a temperatura esteja acima, o atuador é desligado, caso esteja abaixo, é ligado. Uma desvantagem desse método é o atraso que pode ser gerado entre o acionamento e a estabilização da temperatura no nível desejado, causando oscilações na mesma (DUNN, 2013).

Na ação de controle contínuo, a partir da variável medida se determina um erro, que é utilizado pelo controlador para produzir um sinal sobre o atuador a fim de diminuir esse erro. Essa ação é realizada de forma contínua e pode ter três formas de operação: proporcional, integral e derivativa, sendo que os controladores podem utilizar uma dessas funções, a combinação de duas ou das três de acordo com a necessidade (DUNN, 2013).

No controle por ação proporcional, o sinal de um sensor é comparado ao ponto de referência, a diferença entre esses dois valores gera um sinal de erro que é amplificado e realimentado, sendo transformado em um sinal de correção proporcional a amplitude do erro. Devido a proporcionalidade do sistema, situações em que o sinal de correção é muito grande ou muito pequeno podem provocar oscilações ou atrasos na estabilização da variável controlada (DUNN, 2013).

Quando se utiliza apenas o controlador proporcional, a variação medida não é totalmente corrigida em longos períodos. Para realizar esse ajuste, pode-se adicionar a ação integral, que fornece correções para cargas de longa duração, reduzindo o erro a valores próximo a zero. Nessa forma de controle, o sinal de correção é proporcional a magnitude e a duração do erro, ou seja, correspondente a área de mudança da variável (DUNN, 2013).

A ação puramente proporcional também pode ter um tempo de resposta alto. Para tentar resolver isso, pode-se incluir a ação derivativa. O sinal derivativo aplica uma correção apenas à taxa de variação, antecipando o erro, e, ao combinar esse sinal ao sinal proporcional, o sinal de controle é acelerado, diminuindo o tempo necessário para a estabilização da variável controlada (DUNN, 2013).

Todos os meios de controle contínuo podem ser combinados, gerando a ação PID, a forma de controle mais frequentemente utilizada em controle de processos (DUNN, 2013). Esse modelo de controle permite solucionar problemas dos regimes integrais e derivativos, acelerando o tempo de resposta e reduzindo o erro a longo prazo (PENEDO, 2014).

Uma das principais informações que o controlador necessita, independente da ação implementada é a temperatura da mistura dentro da panela. Para obter essa variável, são utilizados sensores que transformam o valor da temperatura em sinais que podem ser interpretados pelo meio de controle.

3.2 MEDIÇÃO DE TEMPERATURA

A temperatura é uma medida relacionada a vibração dos átomos de um corpo e sua aferição pode ser realizada de diferentes formas. Em sistemas computacionais, são utilizados sensores que transformam essa grandeza física em sinais elétricos para que possam ser compreendidos pelo controlador.

3.2.1 Termorresistores

Um sensor comumente utilizado é o termorresistor. Seu funcionamento se baseia na variação da resistência de um condutor, geralmente uma liga metálica, de acordo com a temperatura (LIRA, 2016). Também existem os detectores de temperatura de resistência, do inglês *Resistance Temperature Detector* (RTD), com funcionamento semelhante, porém, utilizam material puro em sua composição, na maioria das vezes platina (LAMB, 2015).

Os termorresistores são divididos em dois grupos: o controle de temperatura positivo, do inglês *Positive Temperature Coefficient* (PTC) e o controle de temperatura negativo, do inglês *Negative Temperature Coefficient* (NTC). No PTC a resistência do componente aumenta de acordo com a temperatura. Já no NTC, mais comumente utilizado, a resistência diminui com o aumento de temperatura (FRADEN, 2010).

Este tipo de sensor apresenta uma ampla faixa de operação em conjunto com uma grande estabilidade e baixo erro, porém a relação entre resistência e temperatura não é linear, o que torna necessário a utilização de equações matemáticas juntamente com parâmetros fornecidos pelo fabricante para que seja possível a obtenção da temperatura aferida (LIRA, 2016).

3.2.2 Termopares

Uma alternativa aos termorresistores, e a mais utilizada, é o sensor termopar (LAMB, 2015). Este tipo de sensor é formado pela junção de dois metais distintos e seu princípio de funcionamento se baseia no efeito termoelétrico, em que diferentes temperaturas, produzem diferentes tensões de forma previsível, permitindo a aferição (FIALHO, 2010).

Existem disponíveis no mercado vários tipos de termopares e sua classificação se dá por letras de acordo com os metais utilizados em sua composição, sendo que cada um deles opera em uma faixa de temperatura diferente, como apresenta a Tabela 2 (DUNN, 2013).

Os termopares são precisos e possuem uma relação linear entre a temperatura e a tensão produzida, porém, o valor fica na casa dos milivolts, o que torna necessário à amplificação para que seja possível a interpretação pelo

controlador. Outro fator é que a função de relação é obtida de forma padronizada pelos fabricantes a uma temperatura de 0°C, no entanto, a temperatura do ambiente em aplicações práticas normalmente é diferente disso. Para considerar essa variação se utiliza a técnica de compensação de junta fria, no qual é adicionada à tensão fornecida pelo sensor a milivoltagem correspondente a temperatura do ambiente em que se encontra os terminais de leitura do componente (FIALHO, 2010).

Tabela 2 – Composição e faixa de operação dos tipos de termopares.

Tipo	Composição	Faixa aproximada (°C)
T	Cobre-Constantan	De -140 a 400
E	Cromel-Constantan	De -180 a 1000
J	Ferro-Constantan	De 30 a 900
K	Cromel-Alumel	De 30 a 1400
N	Nicrosil-Nisil	De 30 a 1400
S	Platina (10% ródio) - Platina	De 30 a 1700
R	Platina (13% ródio) -Platina	De 30 a 1700

Fonte: Dunn (2013).

A partir da obtenção da temperatura atual e pelo meio de controle implementado, o controlador tem a capacidade de atuar sobre a temperatura do mosto dentro da panela. Para realizar essa tarefa, se utiliza de um sistema de aquecimento.

3.3 SISTEMA DE AQUECIMENTO ELÉTRICO

Um meio frequentemente utilizado para o aquecimento por eletricidade é a resistência de fio. Geralmente é composta por uma liga de níquel-cromo (nicromo), e possui uma alta resistência. Essa característica faz com que ao aplicar uma tensão no elemento de aquecimento, parte da energia elétrica é convertida em energia térmica, produzindo calor (PETRUZELLA, 2014).

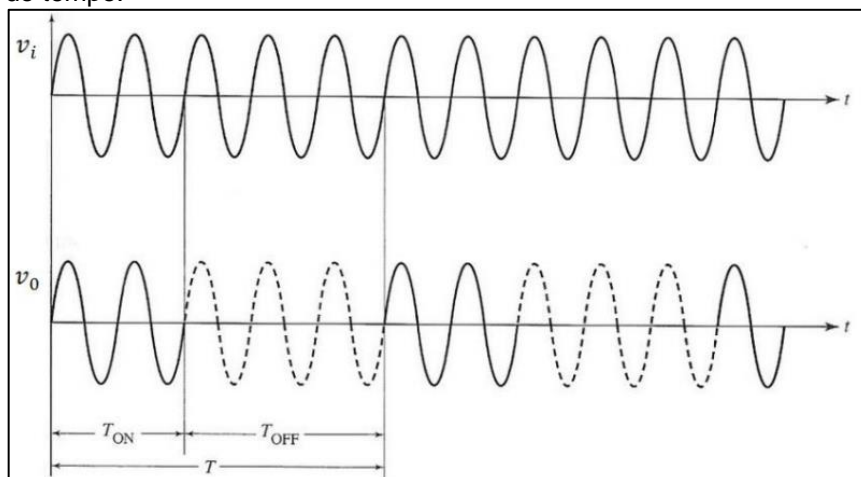
Por utilizar a eletricidade para produzir calor, o controlador tem a capacidade de regular o fornecimento de energia do elemento resistivo e assim adequar o a temperatura de acordo com o meio de controle.

3.4 CONTROLE DE POTÊNCIA EM CORRENTE ALTERNADA

Para atuar sobre a temperatura do mosto é necessário efetuar o controle da potência fornecida ao sistema de aquecimento. Um meio de se fazer isso é transformando uma fonte de tensão de Corrente Alternada (CA) fixa em uma fonte de tensão CA que pode ser variável. Para tal, é possível utilizar duas técnicas: a de *liga-desliga* e a de controle de fase (AHMED, 2000).

A técnica de *liga-desliga* consiste em ligar ou desligar o fornecimento de tensão durante vários ciclos, resultando no controle da potência média da carga de acordo com o tempo, como apresentado na Figura 3 (AHMED, 2000). Uma forma eficiente de se realizar esse controle é utilizando interruptores eletrônicos de onda completa como o Tríodo de Corrente Alternada, do inglês *Triode for Alternating Current* (TRIAC). Este componente é formado por dois Retificadores Controlados de Silício, do inglês *Silicon Controlled Rectifier* (SCR), que é interruptor eletrônico unidirecional formado pela junção de dois transistores, posicionados em antiparalelo. O TRIAC possui dois terminais principais, T1 e T2 e um terminal de disparo denominado gatilho. Sempre que houver presença de corrente elétrica entre o gatilho e o T1, o TRIAC passa a conduzir corrente entre o T1 e o T2. Após o disparo, a condução é mantida de forma independente ao gatilho, e só será interrompida quando a corrente conduzida caia abaixo do valor de corte. (SCHULER, 2013).

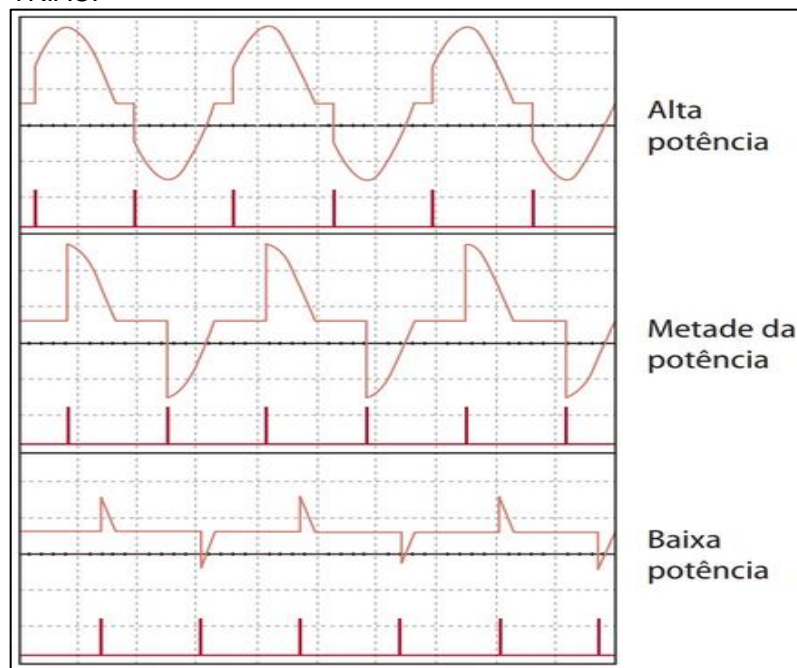
Figura 3 – Comparação entre a tensão da fonte CA e a tensão da entre à carga ao longo do tempo.



Fonte: Ahmed (2000).

Já na técnica de controle de fase, se faz o ajuste da tensão controlando em que ponto do semiciclo da onda senoidal será feita a ligação entre a carga e a fonte CA (AHMED, 2000). Por meio disso é possível obter um controle mais suave e a possibilidade de inferir determinadas porcentagens de potência sobre a carga. Para tal, se utiliza um TRIAC, que deve ser disparado após o início de cada semiciclo. De acordo com o atraso no disparo, se ajusta o quanto da onda será fornecida a carga, conseqüentemente, ajustando a potência. Quando o semiciclo termina, a tensão se aproxima de zero e o TRIAC interrompe a corrente, aguardando um novo disparo (SCHULER, 2013). A Figura 4 ilustra o formato de onda um controle de fase para potência alta, média e alta, juntamente com a onde de disparo do TRIAC.

Figura 4 – Formatos de onda para cada potência aplicada à carga e do disparo do TRIAC.



Fonte: Schuler (2013).

A partir do meio de controle pode-se influenciar na temperatura da panela controlando o sistema de aquecimento com base na temperatura obtida. Esse gerenciamento é realizado pelo controlador. Ao se fazer uso de meios computacionais, esse papel pode ser exercido pelos sistemas embarcados, dispositivos dedicados a tarefas específicas, como será abordado no próximo capítulo.

4 SISTEMAS EMBARCADOS

No gerenciamento da temperatura por meios digitais é necessário a presença de um controlador. Para essa função pode-se utilizar um sistema embarcado. Este tipo de dispositivo é um sistema de computação dedicado que realiza funções específicas, diferentemente de computadores de uso geral, que realizam diversas tarefas diferentes (LAMB, 2015).

Além do controle de temperatura, os sistemas embarcados estão presentes em muitas outras aplicações, que partem de simples máquinas de café, até complexos sistemas de controles de segurança em aeronaves e carros. Esta vasta gama de aplicação faz com que em muitos casos estes sistemas operem com número limitado de recursos computacionais, com requisitos de baixo consumo de energia e restrições de tempo real, exigindo um conjunto de hardware e software robusto e confiável (NULL; LOBUR, 2011).

Os sistemas embarcados podem ser implementados de diversas maneiras, como a partir da utilização de microcontroladores, ou baseados em circuitos ASICs, porém a flexibilidade proporcionada pelos FPGAs garante ao sistema desempenho juntamente a otimização dos recursos.

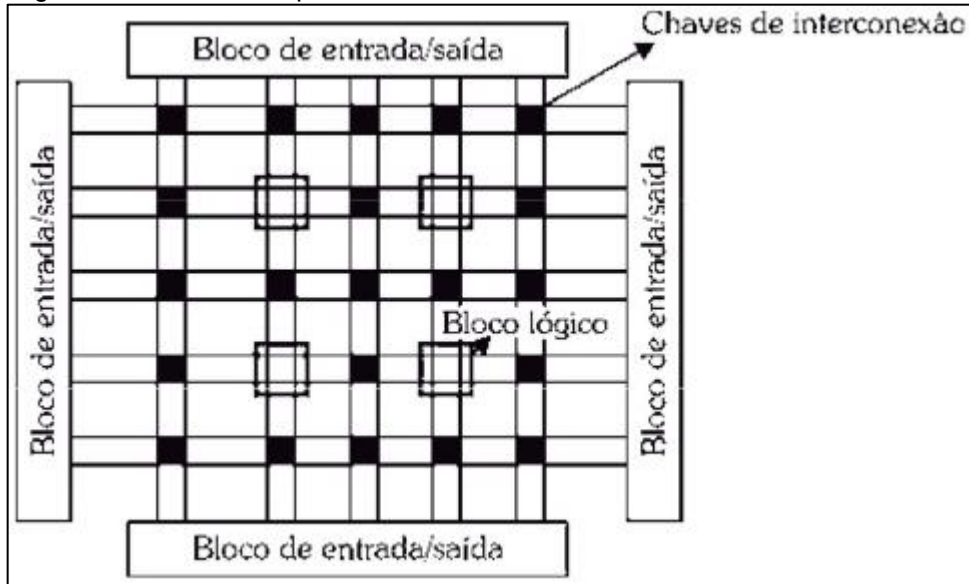
4.1 FPGA

Os FPGAs são circuitos formados por células lógicas ou blocos lógicos configuráveis organizados em forma de matriz. Cada uma dessas células tem a capacidade de implementar funções lógicas e de se comunicar com outras células. De acordo com cada projeto os blocos são conectados pela matriz, interligando as funções lógicas, permitindo o fornecimento e recebimento informações pelos blocos de entrada e saída (COSTA, 2014). A Figura 3 apresenta de forma simplificada a estrutura de um circuito FPGA.

A maneira mais utilizada pelas fabricantes para implementação das funções lógicas é por bloco de memória, do inglês *LookUp Table* (LUT). Nesse método, não é feito de fato uma operação lógica, mas sim, uma consulta, onde, cada endereço da memória corresponde ao resultado da função lógica para aquele endereço. Os blocos LUTs são voláteis, o que resulta na perda das informações sempre que a fonte de energia é cortada, tornado necessário a utilização de pequenas

memórias não voláteis para armazenar e posteriormente carregar as funções sempre que o circuito for energizado (COSTA, 2014).

Figura 5 – Estrutura simplificada de um circuito FPGA.



Fonte: Costa (2014).

Os blocos lógicos podem possuir diferenças na capacidade de implementar as funções lógicas. Essas diferenças são classificadas por granularidade, que é dividida em 3 categorias: fina, grossa e média (SKLIAROVA; FERRARI, 2003).

Na granularidade fina, as células lógicas implementam funções simples e com poucas entradas, o que resulta em um alto nível de aproveitamento dos recursos disponíveis, porém gera necessidade de mais interligações e chaves de roteamento, tornando o circuito e a configuração mais difíceis. Quando a granularidade é grossa, os dispositivos possuem muitas entradas e são voltados para funções complexas, executando operações de forma mais rápida, com a desvantagem de um mau aproveitamento dos blocos quando são implementadas funções simples. Para minimizar as desvantagens das granularidades extremas as fabricantes adotam na maioria dos casos a granularidade média nos circuitos, equilibrando as capacidades de implementação com o aproveitamento dos recursos (SKLIAROVA; FERRARI, 2003).

As células lógicas de um FPGA necessitam ser interconectadas de uma maneira que seja possível a seleção dos blocos necessários para o arranjo do projeto e ao mesmo tempo permitam uma grande quantidade de portas lógicas no circuito. Para isso existem três tecnologias de chaves de roteamento comumente utilizadas:

por memória de acesso randômico estático, do inglês Static Random Access Memory (SRAM), antifuse e gate flutuante. Na SRAM, uma memória desse tipo é utilizada para controlar um multiplexador, o que ocupa muito espaço no circuito, porém tem uma reprogramação rápida. Na antifuse, utiliza-se um dispositivo com dois terminais que apresenta alta impedância quando não programado, e baixa impedância quando aplicado uma tensão. Na gate flutuante, as chaves são compostas por transistores *Metal Oxide Semiconductor* (MOS), tendo a vantagem da retenção de dados e a capacidade de programação (COSTA, 2014).

Para montar um arranjo de portas lógicas em um circuito FPGA se utiliza uma linguagem de descrição de hardware, que permite modelar os circuitos de forma que possam ser carregados na placa. Neste trabalho a HDL utilizada será a linguagem VHDL.

4.2 VHDL

A linguagem VHDL é uma HDL que facilita a construção de projetos complexos que partem do nível hierárquico mais alto para o mais baixo, seguindo um padrão de design *top down*. Tirando alguns trechos específicos de código, seus comandos são sempre executados de forma paralela (D'AMORE, 2012). A Figura 4 apresenta as etapas que são aplicadas a um projeto VHDL para que o código seja implementado em um circuito FPGA.

Figura 6 – Estrutura de um projeto VHDL.



Fonte: D'amore (2012).

O resultado das etapas é um arquivo com a rede de ligações das células lógicas para o circuito especificado, que será utilizado pela ferramenta da fabricante para configurar o roteamento e os blocos do FPGA (D'AMORE, 2012).

Um projeto VHDL é composto por duas partes: bibliotecas e entidades. Nas bibliotecas são importadas, quando necessário, funcionalidades prontas que podem ser do próprio projetista ou de outros desenvolvedores. As entidades são a representação do projeto em si. Nela são declaradas a interface entre e o mundo exterior e a entidade, e a arquitetura, que é onde é feita a descrição do circuito desejado, juntamente com ligação entre os componentes (D'AMORE, 2012). A Figura 5 ilustra um projeto de uma porta *and* com dois sinais de entrada (a e b) e um sinal de saída (z).

Figura 7 – Projeto VHDL de uma porta and.

```
-- Entidade
entity and_gate is

    -- Interface com o meio externo
    port(
        a, b    : in bit;
        z      : out bit
    );
end and_gate;

--Arquitetura
architecture main of and_gate is
begin

    -- Implementação da lógica
    z <= a and b;

end main;
```

Fonte: DO autor.

Em VHDL também é possível construir várias entidades que realizam pequenas funções, para posteriormente reaproveitá-las em outra entidade, compondo um projeto de forma semelhante ao mundo real, onde vários componentes eletrônicos formam o circuito final (CAPPELATTI; TOROK, 2010).

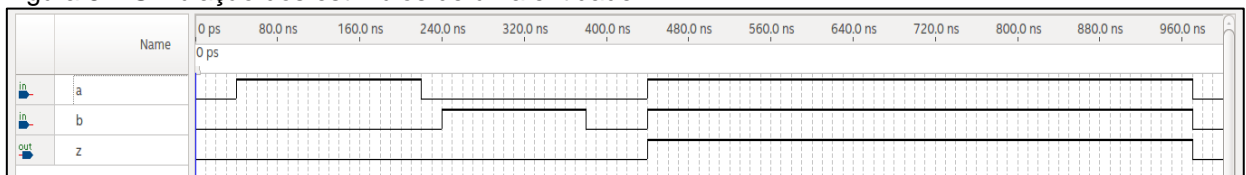
Ao descrever uma entidade é crucial garantir o seu funcionamento para que ela possa ser utilizada por outros componentes da arquitetura. Para tal, pode-se o recurso de testes, que permite exercitar a descrição na fase de desenvolvimento.

4.2.1 Testes

Uma etapa essencial da descrição de hardware é a de testes. A partir dela é possível verificar a operação do sistema e detectar problemas de implementação. O processo consiste em gerar estímulos nas entradas da arquitetura e averiguar se as saídas correspondem ao esperado (D'AMORE, 2012).

Em VHDL, para realizar esse processo pode-se criar uma entidade exclusivamente de testes, e com isso estabelecer os estímulos que a entidade testada irá receber. A linguagem também permite que nesse tipo de entidade seja possível informar parâmetros de tempo para facilitar o processo. Após o término da descrição do teste é possível utilizá-lo em um software de simulação, que possibilita analisar os sinais de forma visual (D'AMORE, 2012). A Figura 8 ilustra a visualização de um teste em um software de simulação

Figura 8 – Simulação dos estímulos de uma entidade VHDL.



Fonte: Do autor.

Muitas vezes apenas o sistema embarcado em si não consegue atender a demanda proposta, sendo necessária a comunicação do mesmo com outros dispositivos e periféricos. Para tal tarefa, pode-se utilizar um meio de transmissão de dados como a comunicação serial.

4.3 COMUNICAÇÃO SERIAL

Uma maneira de transmissão de dados digitais é a comunicação serial. Nessa modalidade é enviado um bit de cada vez de forma sequencial por meio de um único cabo. Cada palavra a ser transmitida inicia com um bit padrão, denominado bit de início, em seguida são enviados os bits de dados e por último é transmitido um bit de interrupção (PETRUZELLA, 2013).

Um protocolo serial comumente utilizado é o Receptor-Transmissor Assíncrono Universal, do inglês *Universal Asynchronous Receiver-Transmitter*

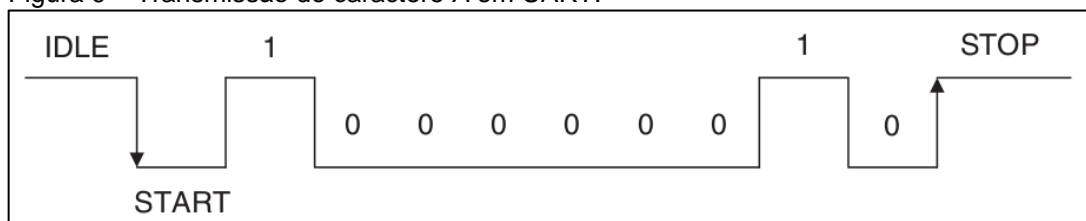
(UART). Nele são utilizados um cabo para transmissão de dados, denominado TX e outro cabo para recepção dos dados, o RX. Essa configuração permite enviar dados em qualquer direção de forma simultânea (IBRAHIM, 2008).

Por ser uma comunicação assíncrona, o controle de tempo é crucial para a transmissão ocorrer de forma correta, necessitando que o transmissor e o receptor tenham a mesma taxa de transferência de bits por segundo, essa configuração é denominada *baud rate*. Valores comuns desse parâmetro são de 4800, 9600, 19200 e 38400 bits por segundo. A partir dessa taxa, ao iniciar uma transmissão o receptor sabe o intervalo de tempo em que deve verificar o estado lógico presente no cabo, da mesma forma que o transmissor sabe quanto tempo ele deve deixar um determinado bit disponível para leitura no barramento (IBRAHIM, 2008).

Outro fator importante é que em UART a transmissão se dá a partir do bit menos significativo (bit mais à direita) até o bit mais significativo (bit mais à esquerda). Também é possível enviar um bit extra para verificar a integridade dos dados recebidos, esse bit é denominado bit de paridade (IBRAHIM, 2008).

A Figura 9 ilustra o envio caractere A, definido no Código Padrão Americano para o Intercâmbio de Informação, do inglês *American Standard Code for Information Interchange* (ASCII), que possui o valor 01000001 em binário, utilizando UART. É possível visualizar que o receptor fica em estado ocioso (*idle*) até receber um bit conhecido para o início da transmissão (*start*), após isso, recebe os bits de dados e interrompe a transmissão ao receber o bit de parada (*stop*), voltando para o estado ocioso.

Figura 9 – Transmissão do caractere A em UART.



Fonte: Ibrahim (2008).

Além da comunicação com outros dispositivos, os sistemas embarcados necessitam de um meio para a interação com os seres humanos. Para tal finalidade existem dispositivos que realizam esse contato.

4.4 INTERFACE HUMANO-MÁQUINA

Em várias situações é necessário que haja a comunicação entre os seres humanos e o sistema embarcado. Para exercer essa função se utiliza um meio de contato entre essas duas partes, denominado Interface Humano-Máquina (IHM). A partir da IHM o usuário tem os recursos fundamentais para informar os parâmetros ao sistema e para obter as informações fornecidas por ele (ROCHA, 2003).

Para construir a IHM podem ser utilizadas telas que apresentam as informações bem como luzes que indicam algum status do sistema. Para receber os dados fornecidos pelos os usuários, teclados e botões são capazes de atender a demanda.

Durante a pesquisa bibliográfica, foram encontrados alguns trabalhos que abordam temas semelhantes ao deste projeto, e que serão discutidos a seguir.

5 TRABALHOS CORRELATOS

Neste capítulo serão apresentados alguns trabalhos com o tema relacionado com o abordado neste documento.

5.1 CONTROLE DE TEMPERATURA POR MEIO DE COMPUTAÇÃO EMBARCADA

Publicado em 2008 pelo acadêmico André Feltrin Baldessar como trabalho de conclusão de curso pela Universidade do Extremo Sul Catarinense, aborda os conceitos de computação embarcada para a elaboração de um protótipo constituído de hardware e software focado em controle de temperatura na avicultura.

Para tal, foi utilizado um microcontrolador PIC 16F877A como controlador, que usa o *Assembly* como linguagem de programação. Na aquisição da temperatura o sensor LM35, que varia a tensão de acordo com a temperatura, foi o escolhido. O modo de controle implementado foi o proporcional, que para influenciar na temperatura usa modulação de largura de pulso. A IHM foi implementada em um computador, e a comunicação com o controlador acontece de forma serial, utilizando o protocolo RS-232-C (BALDESSAR, 2008).

Os testes foram baseados em simulações com cargas de corrente continua para atuar no aquecimento e no resfriamento do meio (BALDESSAR, 2008).

O autor concluiu que todos os objetivos da pesquisa foram alcançados, tendo como resultado o hardware e software propostos (BALDESSAR, 2008).

5.2 DESENVOLVIMENTO DE DISPOSITIVO DE CONTROLE AUTOMÁTICO PARA OS PROCESSOS DE MOSTURAÇÃO E FERVURA DA FABRICAÇÃO DE CERVEJA ARTESANAL

Um trabalho de conclusão de curso publicado em 2015 pelos autores Jean Carlos Borges e Thiago Fröhlich pelo Instituto Federal De Educação, Ciência E Tecnologia De Santa Catarina Campus Joinville, descreve um projeto de automação para os processos de mosturação e fervura na produção de cerveja artesanal.

Foi construído o protótipo utilizando um Arduino como controlador, com a ação de controle PID implementada, atuando sobre o sistema de aquecimento por meio do controle de *liga-desliga*. Para aferição da temperatura foi usado o sensor

DS18B20, que utiliza o protocolo *1-wire* para fornecer os dados. A IHM se deu por uma tela sensível ao toque acoplada ao Arduino. A panela usada foi uma de alumínio, com algumas adaptações para utilização na produção de cerveja (BORGES; FRÖHLICH, 2012).

O teste do protótipo foi realizado a partir da produção de uma receita de cerveja *Belgian Blond Ale*, e o autor conclui que todos os objetivos do projeto foram concluídos com êxito (BORGES; FRÖHLICH, 2012).

5.3 SISTEMA DE CONTROLE DE TEMPERATURA USANDO FPGA

Publicado pelos acadêmicos Alexssandro Braz de Souza e Deise Martinello em 2013 como trabalho de conclusão de curso pela Universidade Tecnológica Federal do Paraná, com o objetivo de implementar um sistema de controle automático de temperatura baseado em FPGA.

O protótipo tem como controlador uma placa FPGA, e utilizou VHDL com HDL, implementando a ação de controle *liga-desliga*. Para aferição da temperatura o sensor utilizado foi o LM35. Como atuador para o aquecimento foram utilizadas lâmpadas incandescentes e para o resfriamento, ventoinhas (SOUZA; MATINELLO, 2013).

Para os testes os autores desenvolveram uma maquete e estipularam uma temperatura alvo. A partir desse valor, o protótipo atuava no aquecimento ou resfriamento do ambiente (SOUZA; MATINELLO, 2013).

Os autores concluíram em sua pesquisa que o objetivo de realizar o controle de temperatura baseado em FPGA foi atingido (SOUZA; MATINELLO, 2013).

5.4 INSTRUMENTAÇÃO E CONTROLE DO PROCESSO DE PRODUÇÃO DE UMA MICROCERVEJARIA

No projeto de graduação submetido na Universidade Federal Do Rio De Janeiro pelo Lucas Rodrigues de Moraes Tostes em 2015, o objetivo era criar a instrumentação e controle para produção de cerveja em uma microcervejaria.

No protótipo desenvolvido se teve como base uma planta industrial e foi utilizado um Arduino uno como controlador. O sistema de aquecimento foi baseado na condução do mosto por uma serpentina imersa em água quente, sendo conduzido por bombas e válvulas. A aferição de temperatura se deu pelo sensor DS18B20. Para

a IHM, foi utilizado um software em um computador para sincronizar as informações com o controlador (TOSTES, 2015).

Os testes se basearam em três simulações com faixas de temperaturas semelhantes à do processo de produção. Baseado nesses testes o autor conclui que os objetivos da pesquisa foram atingidos com sucesso (TOSTES, 2015).

5.5 DESIGN AND IMPLEMENTATION OF DIGITAL PID CONTROLLER USING FPGA FOR PRECISION TEMPERATURE CONTROL

Um artigo publicado em 2014 na IEEE Power India International Conference, pelos autores Prachi Rusia e Sandeep Bhongade propõe um sistema de controle de temperatura que utiliza a ação de controle PID baseado em FPGA.

Para o projeto foram utilizados o controle PID implementado em um circuito FPGA, com o controle da carga por modulação de largura de pulso. Para leitura da temperatura se usou um conversor analógico digital convertendo o sinal fornecido pelo sensor (RUSIA; BHONGADE, 2014).

Os testes foram baseados em simulações virtuais e utilizando um sistema de aquecimento de 591mW. A partir disso os autores concluíram que os objetivos foram atingidos, obtendo um sistema estável com tempo de resposta rápido (RUSIA; BHONGADE, 2014).

6 SISTEMAS EMBARCADOS UTILIZANDO FPGA: ESTUDO DE CASO NO CONTROLE DE TEMPERATURA NA ETAPA DE BRASSAGEM

O trabalho consiste na criação de um protótipo de sistema embarcado utilizando um circuito FPGA como controlador de temperatura na etapa de brassagem no processo de produção de cerveja.

A partir da aquisição da temperatura do mosto por meio de um sensor, o controlador irá aplicar a ação de controle implementada, e com base no resultado obtido, ajustar o sistema de aquecimento para que a temperatura da mistura se adeque aos parâmetros do mestre cervejeiro.

O meio de interação entre o usuário e o sistema embarcado será por meio de uma aplicação desktop, que utilizando a comunicação UART será capaz de enviar os parâmetros necessários e receber os dados sobre o status atual do processo.

6.1 METODOLOGIA

A pesquisa é de abordagem qualitativa, com natureza aplicada, tendo isso em vista, para iniciar o trabalho, foi feito o levantamento bibliográfico a fim de buscar o aprendizado do conteúdo necessário para o desenvolvimento do projeto. Para tal, foi utilizada a biblioteca da UNESC e as bases da IEEE Xplore, Google Acadêmico, periódicos da CAPES entre outras. A partir do conhecimento adquirido se definiu a placa FPGA a ser utilizada bem como os outros componentes necessários para a construção do protótipo.

Na etapa seguinte foi realizada a descrição do hardware que seria utilizado na placa, a montagem dos circuitos necessários para o controle da temperatura e desenvolvimento da IHM.

A etapa final teve como base a realização dos testes com o protótipo para a obtenção dos dados necessários para averiguar os resultados obtidos.

6.1.1 Definição e construção dos componentes

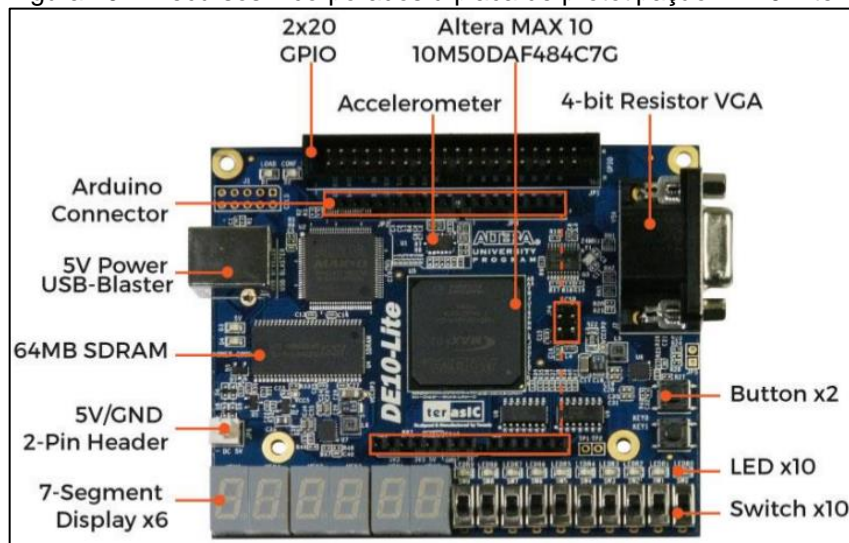
Para o desenvolvimento do protótipo foi necessário a escolha dos equipamentos e periféricos que seriam utilizados, juntamente com a construção de alguns circuitos.

6.1.1.1 Escolha da placa FPGA

A fim de se definir o circuito FPGA que seria utilizado, foi realizada uma pesquisa dos modelos disponíveis e seus respectivos valores. Após isso, determinou-se que seria utilizado uma placa FPGA de prototipação, devido a facilidade de configuração e utilização de dispositivos desse gênero. Também foi considerado a quantidade de elementos lógicos disponíveis e a capacidade de integração com periféricos, bem como a quantidade de recursos integrados.

A partir desses requisitos foi escolhida a placa de prototipação DE10-Lite (Figura 10), fornecida pela fabricante Terasic. A mesma conta com um circuito FPGA Altera Max 10, desenvolvido pela Intel, com 50 mil elementos lógicos, 40 entradas/saídas de propósito geral, LEDs, botões, displays de 7 segmentos, dentre outros recursos incorporados à placa.

Figura 10 – Recursos incorporados à placa de prototipação DE10-Lite.



Fonte: Terasic (2020).

6.1.1.2 Escolha da panela

Para a produção da cerveja foi definida uma panela industrial em inox de parede dupla com capacidade máxima de 120 litros (Figura 11). O equipamento também possui uma resistência elétrica do tipo flange com 6kW de potência, sendo que o sistema de aquecimento não entra em contato com o conteúdo, e o calor produzido é conduzido por um fluido térmico.

Figura 11 – Apresentação da panela.



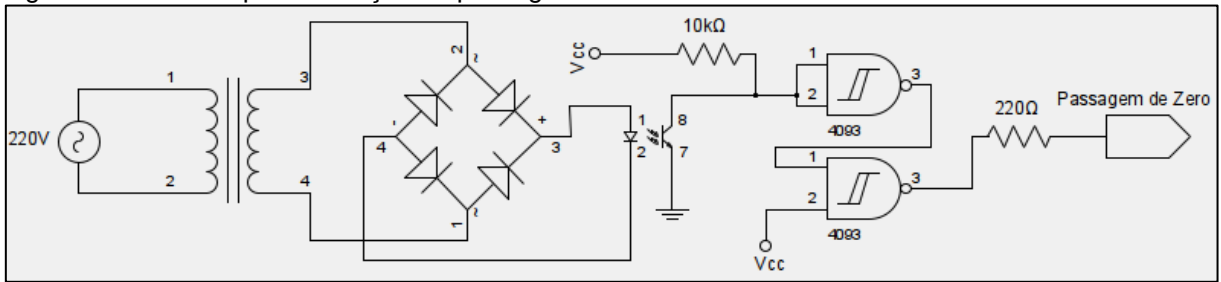
Fonte: Do autor.

6.1.1.3 Controle de potência

Para controlar a potência do sistema de aquecimento, foi desenvolvido um circuito para detecção da passagem por zero da senoide da rede elétrica e outro para controlar o fornecimento de tensão à resistência pela técnica de controle de fase.

Em relação a detecção de zero, se utilizou um transformador para diminuir a tensão da rede elétrica de 220V para 3V e se fez a retificação para obter um pulso de corrente contínua na duração de meio ciclo. Esse pulso foi ligado a entrada de um acoplador óptico para isolar o circuito e mantê-lo seguro para o controlador. Na saída desse componente, foi ligado um resistor de *pull-up*, que garante o nível lógico alto sempre que o optoacoplador não está conduzindo. O circuito descrito fornece em sua saída um pulso sempre que a senoide passa pelo zero, porém, nos testes realizados, o sinal fornecido não era uma onda quadrada ideal que poderia ser lida pelo controlador. Para resolver isso foi adicionado ao circuito o CI 4093, que possui quatro portas nand Schmitt Triggers. Dessas, foram utilizadas 2 portas para montar um formador de sinais quadrados a partir da onda obtida do optoacoplador. Após essa alteração o circuito passou a oferecer uma onda em formato adequado que poderia ser utilizada pelo controlador. A Figura 12 apresenta o circuito final responsável por fornecer o sinal de passagem de zero.

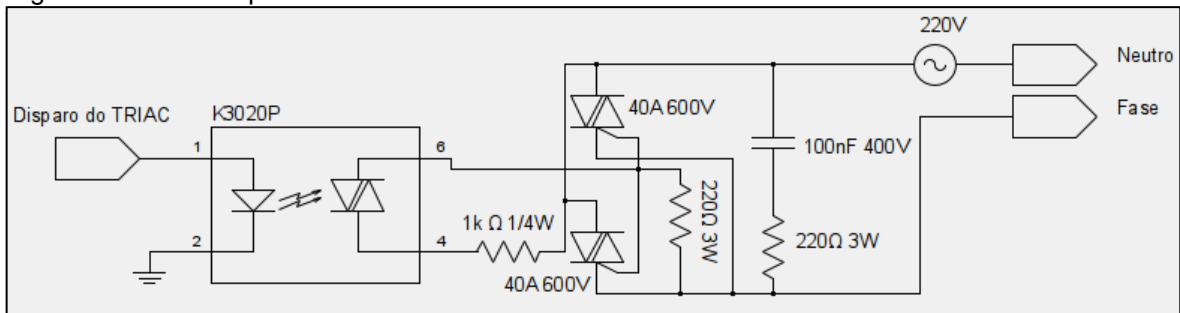
Figura 12 – Circuito para detecção da passagem de zero.



Fonte: Do autor.

Já no controle de fase, foi desenvolvido um circuito que aciona o TRIAC a partir de um sinal fornecido pelo controlador. Para efetuar o acionamento, foi utilizado um acoplador óptico com Diodo para Corrente Alternada, do inglês *Diode for Alternating Current* (DIAC), que isola o controlador, o mantendo seguro, conectado ao terminal de disparo do TRIAC. Os outros terminais foram conectados à rede elétrica e ao sistema de aquecimento da panela. Também foi adicionado em paralelo um circuito *snubber*, que utilizar um resistor e um capacitor em série para proteger o TRIAC de variações abruptas de tensão que acontecem ao chavear a carga. Nos testes realizados utilizando um TRIAC para 40A, ele apresentava comportamento errado quando a corrente drenada pela da resistência chegava na casa dos 20A. Após pesquisas, descobriu-se que o problema se dava pelo calor. Em primeira tentativa, foi testado um dissipador maior acoplado ao TRIAC, porém sem sucesso. Em seguida foi aumentado o tamanho do dissipador e adicionada uma ventoinha para forçar a passagem do ar, que também não surtiu efeito. A solução final para o problema foi ligar dois TRIACs em paralelo, para dividir a corrente conduzida, juntamente com um dissipador de calor e uma ventoinha. A Figura 13 ilustra o circuito final para controle de fase da resistência a partir do sinal do controlador.

Figura 13 – Circuito para controle de fase.

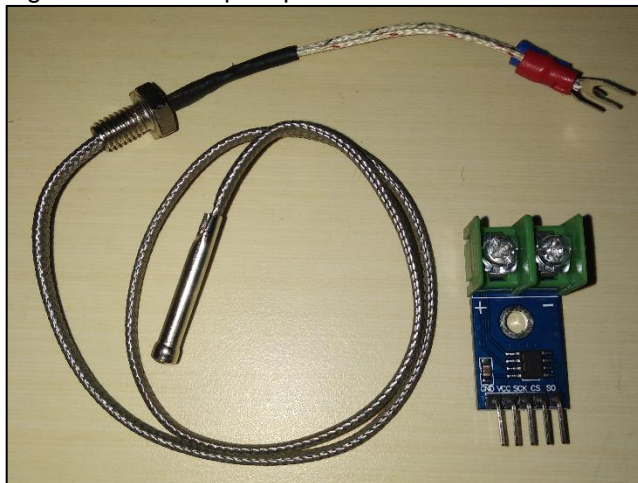


Fonte: Do autor.

6.1.1.4 Aferição da temperatura

Para aferição da temperatura optou-se pelo sensor do tipo termopar devido a precisão e baixo custo. A categoria escolhida foi a K, para que fosse possível a utilização em conjunto com o módulo MAX6675. Esse módulo realiza a compensação de junta fria e amplifica a tensão termopar, fornecendo uma faixa de leitura 0°C até +700°C, com uma resolução de 0,25°C. Para obter a temperatura aferida, o módulo possui uma interface de comunicação serial SPI, por onde é possível receber os 12 bits que representam a temperatura. A Figura 14 ilustra o termopar juntamente com o módulo de leitura.

Figura 14 – Termopar tipo K e módulo MAX6675.

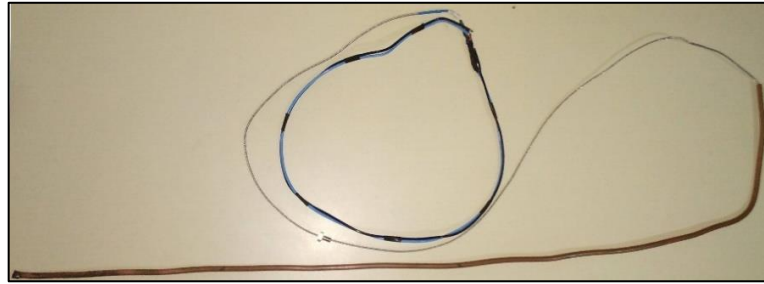


Fonte: Do autor.

Também foi executado testes com o sensor e o módulo a fim de averiguar a sua precisão. Para tal, foi comparado as leituras obtidas em testes com líquido frio e quente com as leituras de um termômetro de mercúrio. Nesses testes não foi constada nenhuma diferença que influenciasse no controle do processo.

Para que fosse possível inserir o sensor na panela, foi necessário colocá-lo dentro de um cano de cobre (Figura 15). A sonda desenvolvida foi capaz de ler a temperatura sem prejudicar o funcionamento do sensor. Outro detalhe foi uma emenda no fio do termopar para que ele que chegasse até o módulo. Como a emenda estava na temperatura ambiente, a utilização do cabo flexível de cobre não interferiu na leitura.

Figura 15 – Sonda para aferição da temperatura.



Fonte: Do autor.

6.1.1.5 Comunicação serial

No decorrer do projeto se constatou a necessidade de um conversor USB para comunicação serial UART, tornando possível a troca de informações entre o controlador e a IHM. Para isso poderia ser utilizado um hardware específico, porém se optou por usar um Arduino Mega 2560, já que este possui o conversor, e o acadêmico tinha um disponível. Foi conectado o transmissor e o receptor do controlador ao Arduino, juntamente com uma programação para que tudo o que viesse do controlador fosse transmitido ao USB e tudo o que viesse do USB fosse transmitido ao controlador, como apresentado na Figura 16.

Figura 16 – Código par utilização do Arduino como conversor.

```
void loop(){  
    if(Serial1.available() > 0){  
        Serial.write(Serial1.read()); //Controlador > IHM  
    }  
    if(Serial.available() > 0){  
        Serial1.write(Serial.read()); //IHM > Controlador  
    }  
}
```

Fonte: Do autor.

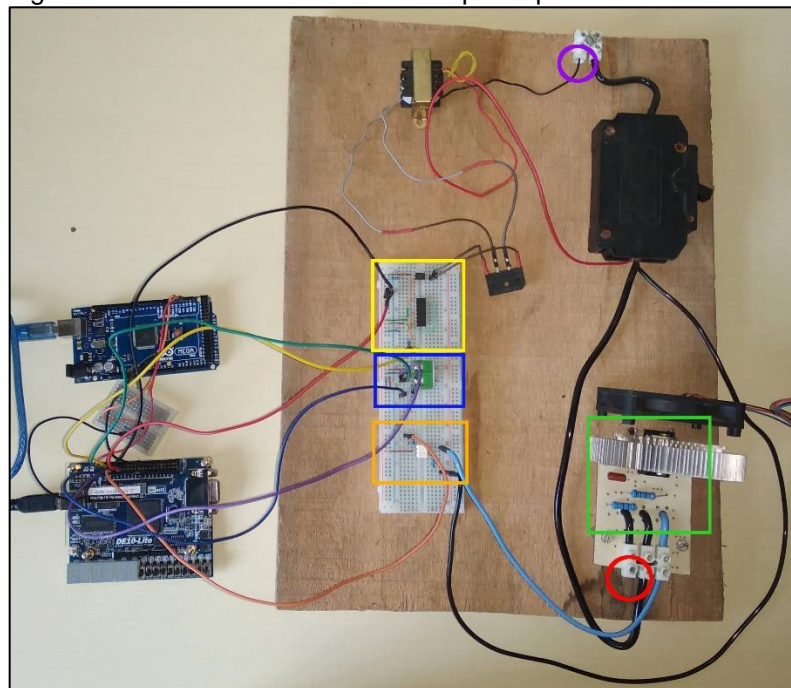
6.1.1.6 Montagem do protótipo

Ao término da definição dos itens necessários, foi feita a compra dos materiais para realizar a montagem do protótipo. Os itens foram montados sobre uma bancada para facilitar o transporte. O circuito de potência foi montado em uma placa separada, evitando possíveis danos aos outros componentes devido a corrente conduzida. Os demais circuitos foram montados em uma protoboard. Também foi

adicionado para proteger o esquema um disjuntor. A alimentação das placas se deu por USB e dos componentes do circuito pelos pinos de energia da placa DE10-Lite.

A Figura 17 apresenta o controlador e os circuitos do protótipo. A seleção em amarelo indica o circuito de detecção da passagem de zero, a em azul o módulo de leitura de temperatura, em laranja o disparo do TRIAC e em verde o controle de potência. O círculo em vermelho indica onde deve ser conectado a fase do sistema de aquecimento e o roxo o neutro.

Figura 17 – Controlador e circuitos do protótipo.

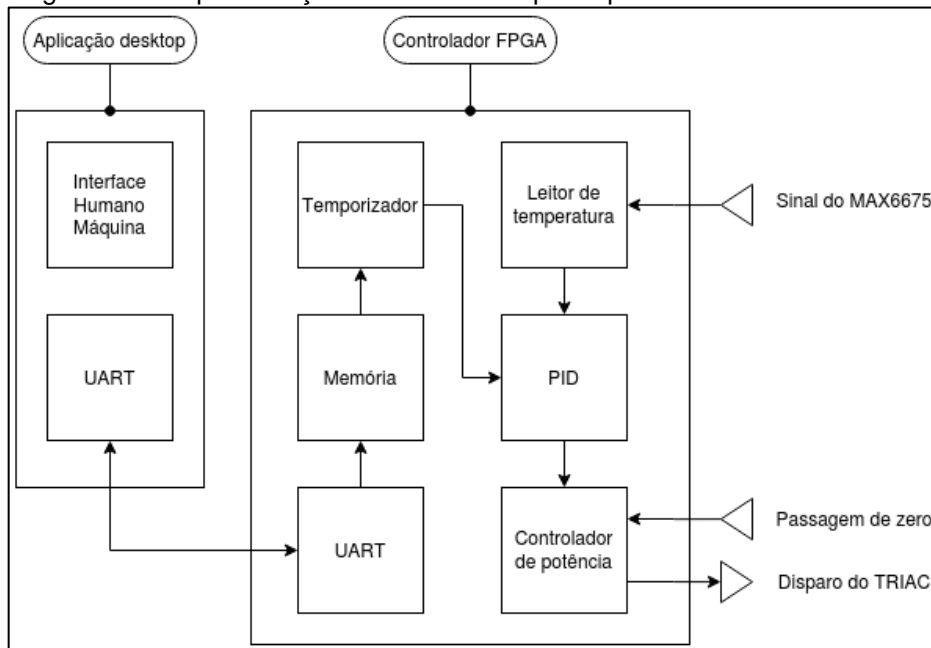


Fonte: Do autor.

6.1.2 Desenvolvimento do software

O desenvolvimento do software foi dividido entre a descrição do hardware do controlador e da programação da IHM. A Figura 18 ilustra os componentes de software do protótipo e o relacionamento entre eles.

Figura 18 – Representação do software do protótipo.

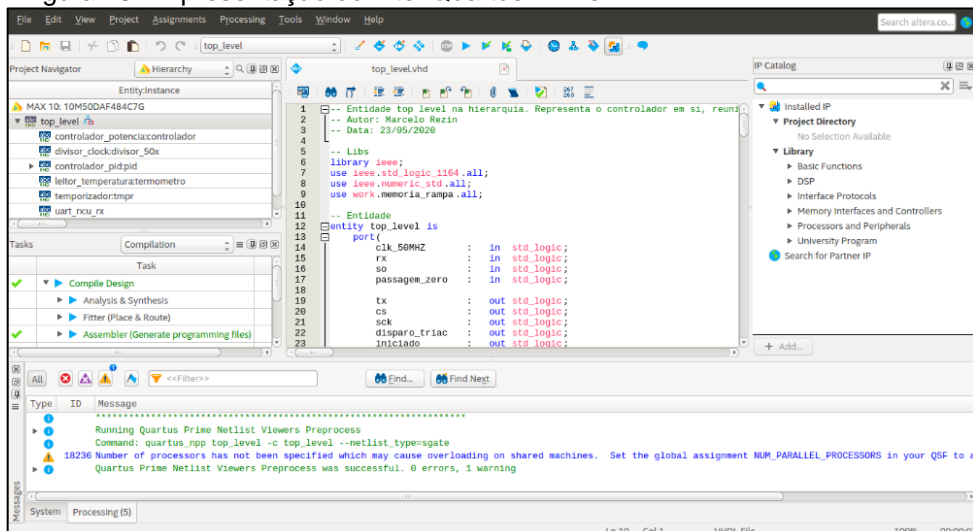


Fonte: Do autor.

6.1.2.1 Descrição em VHDL

O início do desenvolvimento em VHDL se deu a partir da definição das ferramentas que seriam usadas. Com base na placa FPGA escolhida, a IDE utilizada foi a Intel Quartus Prime (Figura 19), software oficial de desenvolvimento para o dispositivo selecionado, na edição Lite, que é gratuita, versão 18.1. O Quartus Prime permite escrever o código, realizar simulações e efetuar a compilação e gravação do código em VHDL na placa.

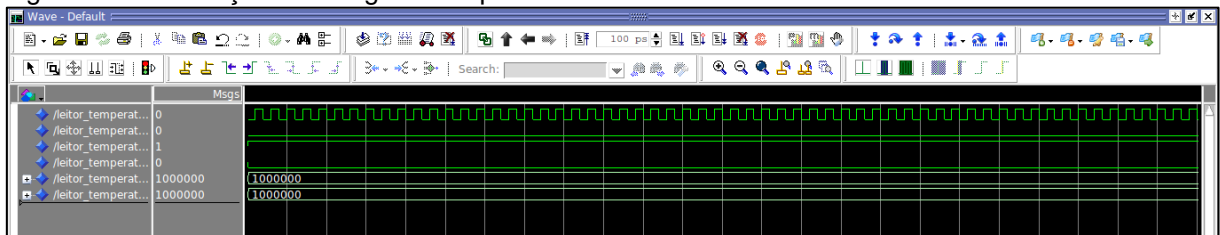
Figura 19 – Apresentação do Intel Quartus Prime.



Fonte: Do autor.

Para realização dos testes do código em VHDL, foi selecionado o ambiente de simulação o Modelsim (Figura 20), que possui uma edição gratuita para estudantes, na versão 10.5. O Modelsim oferece recursos avançados de simulação e capacidade de integração com o Quartus.

Figura 20 – Simulação de código VHDL pelo Modelsim.



Fonte: Do autor.

Após a escolha das ferramentas, se deu continuidade ao projeto descrevendo o hardware. A arquitetura foi dividida em várias entidades que realizam funções específicas para posteriormente serem unidas em uma arquitetura que representa o controlador em si. Esse modelo de desenvolvimento permitiu realizar os testes de forma isolada, tornando o processo mais assertivo.

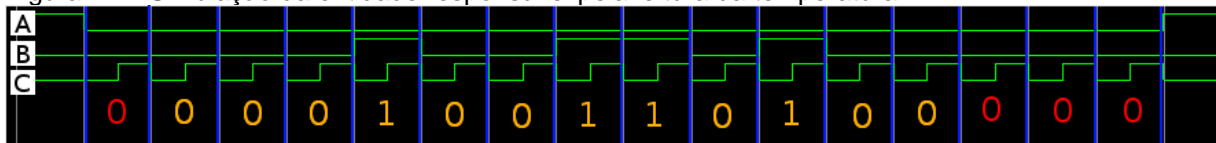
Inicialmente foi descrito uma entidade utilitária para dividir o *clock* da placa de 50MHz para 1MHz. Esse *clock* dividido foi passado à todas entidades da arquitetura, o que tornou alguns cálculos mais simples e removeu alguns sinais indesejados que alteravam o comportamento do controlador, principalmente na detecção da passagem por zero.

Em seguida foi descrito a entidade responsável pelo controle de potência. Sua arquitetura se baseia no recebimento do sinal da passagem de zero e de uma porcentagem de potência que deve ser aplicada ao sistema de aquecimento. De acordo com o percentual indicado a entidade aguarda um determinado tempo a partir da passagem de zero para enviar o sinal de disparo do TRIAC, efetuando o controle de fase.

Após isso foi descrita a arquitetura que recupera a temperatura do mosto a partir do módulo MAX6675. Para isso foi necessário implementar a comunicação serial SPI para efetuar a recepção dos bits. Dos 16 bits recebidos, o bit 15 ao 4 representavam a temperatura, os outros eram parâmetros que não foram utilizados. Outro ponto é que o módulo envia os valores representados por ponto fixo, sendo que os 2 últimos bits correspondiam a parte decimal da temperatura aferida.

A Figura 21 ilustra a simulação da leitura de temperatura por SPI. O bit mais significativo é recebido primeiro e o menos significativo por último. O sinal A é enviado ao módulo e indica o início da leitura, o B são os bits recebidos e o C é o *clock* fornecido ao módulo. A linha azul indica a divisão para leitura de cada bit presente no sinal B, os bits descartados estão na cor vermelha e os bits que representam a temperatura na cor amarela.

Figura 21 – Simulação da entidade responsável pela leitura da temperatura.



Fonte: Do autor.

A próxima entidade construída foi a que realiza o controle PID, tendo como base a temperatura alvo e a temperatura atual, fornecida pela arquitetura de leitura. Com base nesses dois parâmetros, são calculados o erro, o ganho proporcional, o ganho integral e o ganho derivativo. Para realização dos cálculos foi adotado a representação por ponto fixo, utilizando números com sinal de 18 bits, sendo que 3 bits eram a parte decimal dos valores. Também foi definido a base de tempo em 1s, o que simplifica os cálculos sem prejudicar o controle, dado tipo de equipamento. Tendo o valor do PID, a entidade converte para uma porcentagem válida para que a arquitetura de controle de potência possa regular o sistema de aquecimento e inferir na temperatura. Essa entidade também é responsável por informar as outras quando o erro era superior a 3°C, o que indicava que a panela estava em aquecimento. A Figura 22 apresenta o trecho do código responsável por calcular a porcentagem de potência do sistema de aquecimento a partir do PID.

Figura 22 – Cálculo da porcentagem de potência a partir do cálculo do PID.

```
begin
  if rising_edge(clk) then
    temperatura_atual_signed := signed(shift_left(resize(unsigned(temperatura_atual), temperatura_atual_signed'length), 1));
    erro := signed(shift_left(unsigned(set_point), 1)) - temperatura_atual_signed;
    p := shift_right(erro * kp, 3);
    i := i + shift_right(erro * ki, 3);
    d := shift_right(ultima_temperatura - temperatura_atual_signed) * kd, 3);
    pid := shift_right(p + i + d, 3);
    ultima_temperatura <= temperatura_atual_signed;
    if pid > 9 then
      percentual_potencia <= 100;
    elsif pid < 0 then
      percentual_potencia <= 0;
    else
      percentual_potencia <= to_integer(pid) * 10;
    end if;
    if erro > 24 then --24 = 3 no ponto fixo - Aguarda a panela aquecer
      paralizar_contagem_tmp <= '1';
    else
      paralizar_contagem_tmp <= '0';
    end if;
  end if;
end process;
```

Fonte: Do autor.

Nos testes realizados a fim de ajustar o controle, foi constatado que o ganho integral acelerava de mais o aumento da temperatura, fazendo com que o mosto ficasse mais quente que o ideal, por esse motivo ele foi desativado, restando o controle P, e o controle D, multiplicado por 0,125, o que levou a um aumento mais suave da temperatura, sem ultrapassar em muitos graus o limite estipulado.

Na sequência do desenvolvimento, foi criada uma entidade de memória para armazenar os parâmetros de duração e temperatura das rampas informados pelo usuário. Para atender as demandas das receitas de cerveja, a arquitetura foi projetada para armazenar até 10 rampas, cada uma podendo durar no máximo 2 horas e 15 minutos com temperaturas de até 120 graus.

Para executar as rampas armazenadas, foi desenvolvida uma entidade responsável pela temporização. Com base no tempo decorrido a partir do início da brassagem, essa arquitetura contabiliza o tempo e informa ao módulo PID a temperatura alvo da rampa. Quando a rampa chegava ao final, ou era enviado um sinal as outras entidades informando que o processo tinha terminado, ou se seguia para a próxima rampa. O temporizador também pausava a contagem de tempo caso a panela estivesse em fase de aquecimento, para que a receita fosse seguida corretamente.

Para que fosse possível receber os parâmetros provenientes da IHM, foram descritas duas entidades para comunicação UART, responsáveis pela transmissão, e recepção dos dados, projetadas para operar a uma taxa de bits 9600 por segundo, sem bit de paridade.

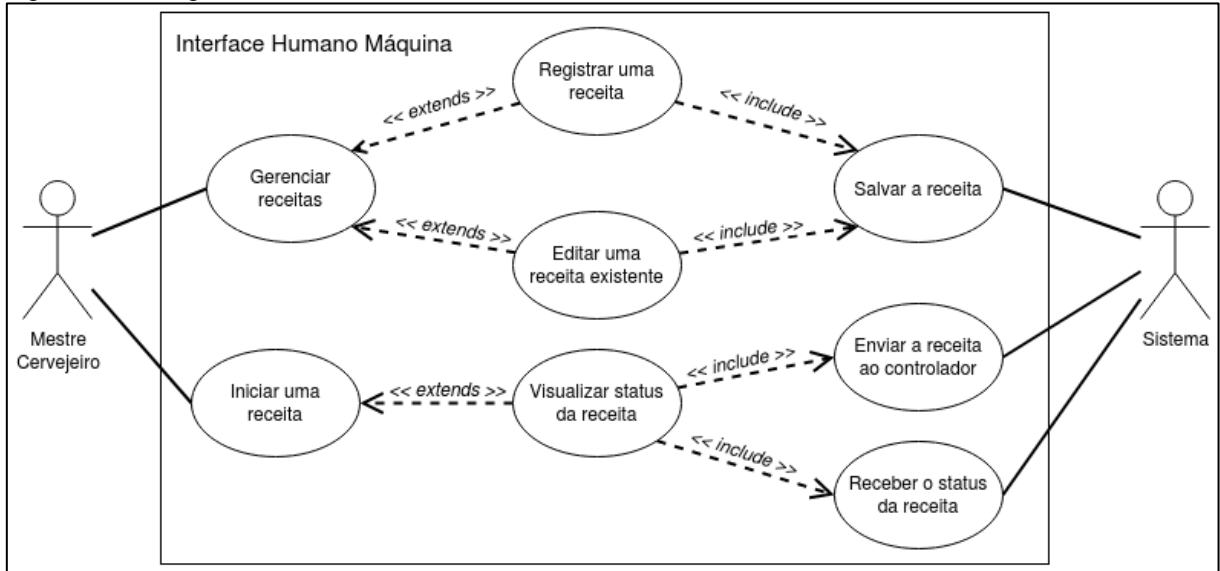
Ao final, foi descrito uma entidade que comporta e interliga todas as outras, e que representa o controlador final. Ela também é responsável por realizar a sincronização com a IHM, recebendo os parâmetros da receita e enviando informações do status do processo. Essa arquitetura foi compilada e gravada na placa FPGA, ocupando 1082 (2%) elementos lógicos, 515 registradores e 11 pinos de propósito geral.

6.1.2.2 Programação da IHM

Para o desenvolvimento da IHM, a IDE escolhida foi o Eclipse que é gratuita, na versão 4.12. O software projetado foi uma aplicação desktop na linguagem Java, versão 8, e permite ao usuário registrar receitas de cerveja e acompanhar o

processo em tempo real. A Figura 23 ilustra o diagrama de caso de uso com as principais funcionalidades da IHM, juntamente com suas interações com as partes internas do sistema.

Figura 23 – Diagrama de caso de uso da IHM.



Fonte: Do autor.

A primeira funcionalidade desenvolvida foi a de criação e edição de receitas, em que o usuário informa o nome e as rampas e em seguida salva os dados. A Figura 24 ilustra a tela no momento da edição de uma receita.

Figura 24 – Tela para edição da receita.

Alterar os dados da receita:			
Nome:	<input type="text" value="Golden"/>		
Rampas			
Temperatura 1 (°C):	<input type="text" value="54"/>	Tempo 1 (Minutos):	<input type="text" value="2"/>
Temperatura 2 (°C):	<input type="text" value="50"/>	Tempo 2 (Minutos):	<input type="text" value="20"/>
Temperatura 3 (°C):	<input type="text" value="67"/>	Tempo 3 (Minutos):	<input type="text" value="80"/>
Temperatura 4 (°C):	<input type="text" value="75"/>	Tempo 4 (Minutos):	<input type="text" value="10"/>
Temperatura 5 (°C):	<input type="text" value="85"/>	Tempo 5 (Minutos):	<input type="text" value="10"/>
Temperatura 6 (°C):	<input type="text"/>	Tempo 6 (Minutos):	<input type="text"/>
Temperatura 7 (°C):	<input type="text"/>	Tempo 7 (Minutos):	<input type="text"/>
Temperatura 8 (°C):	<input type="text"/>	Tempo 8 (Minutos):	<input type="text"/>
Temperatura 9 (°C):	<input type="text"/>	Tempo 9 (Minutos):	<input type="text"/>
Temperatura 10 (°C):	<input type="text"/>	Tempo 10 (Minutos):	<input type="text"/>
<input type="button" value="Salvar"/>		<input type="button" value="Excluir"/>	
<input type="button" value="Voltar"/>			

Fonte: Do autor.

Para a persistência das informações foi utilizada a técnica de serialização de objetos da linguagem, em que é possível representar o objeto em bytes que podem

ser escritos em formato de arquivo. Para recuperação das receitas salvas foi implementado um processo de desserialização, que transforma o arquivo armazenado de volta em objetos da linguagem. Essa técnica foi adotada devido a pequena quantidade de informações que são armazenadas e a baixa complexidade para realizar o processo. A Figura 25 apresenta os métodos responsáveis por persistir as receitas (*save*) e por carregar as receitas salvas (*load*).

Figura 25 – Código responsável pela persistência das receitas.

```
private static final String ARQUIVO = "receitas.dat";
public static void save() {
    try {
        File arquivo = new File(ARQUIVO);
        arquivo.delete();
        arquivo.createNewFile();

        ObjectOutputStream outReceitas = new ObjectOutputStream(new FileOutputStream(arquivo));
        outReceitas.writeObject(ReceitaWrapper.getInstance().getReceitas());
        outReceitas.close();
    } catch (Exception e) {
        JOptionPane.showMessageDialog(null, "Não foi possível salvar as receitas!", "Erro:", JOptionPane.ERROR_MESSAGE);
    }
}

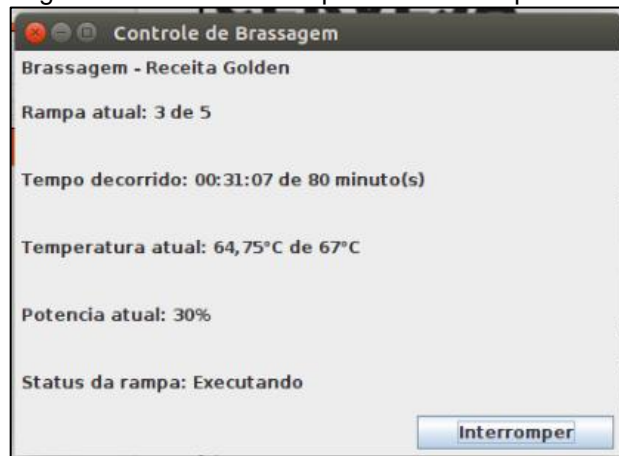
@SuppressWarnings("unchecked")
public static void load(){
    try {
        File arquivo = new File(ARQUIVO);

        if (arquivo.exists()) {
            ObjectInputStream inReceitas = new ObjectInputStream(new FileInputStream(arquivo));
            ReceitaWrapper.getInstance().setReceitas((ArrayList<Receita>)inReceitas.readObject());
            inReceitas.close();
        } else {
            ReceitaWrapper.getInstance().setReceitas(new ArrayList<Receita>());
        }
    } catch (Exception e) {
        JOptionPane.showMessageDialog(null, "Não foi possível carregar receitas!", "Erro:", JOptionPane.ERROR_MESSAGE);
    }
}
```

Fonte: Do autor.

Na sequência do desenvolvimento foi implementada a funcionalidade para execução da receita. Nessa parte, o software envia os dados da receita para o controlador e sinaliza a ele que o processo deve ser iniciado. A partir desse momento o sistema também fica recebendo as informações do status da execução fornecidas pelo controlador e as exibe para que o usuário possa acompanhar. Na tela são mostradas a rampa executada, tempo, temperatura atual a potência aplicada ao sistema de aquecimento e se o temporizador está executando ou aguardando o aquecimento, como ilustra a Figura 26.

Figura 26 – Tela de acompanhamento do processo.



Fonte: Do autor.

A fim de realizar a sincronização entre o controlador e a IHM, foi desenvolvido um protocolo básico para envio das informações via UART. Para comunicação serial, foi utilizada a biblioteca rtxcommon¹, que é gratuita e de uso simples.

Primeiramente o software envia um byte conhecido ao controlador solicitando a sincronização, caso o controlador retorne uma confirmação, ela é iniciada. Nessa sincronização são enviadas as rampas definidas pelo usuário, separadas em 2 bytes cada uma, sendo o primeiro a temperatura alvo e o segundo o tempo. Ao término, é enviado um byte que informa o controlador que a brassagem deve ser iniciada. A partir desse ponto o controlador envia a IHM uma sequência de 8 bytes a cada 200 milissegundos contendo as informações do status do processo, com apresentado na Figura 27. O software também pode enviar um byte ao controlador sinalizando que o processo deve ser interrompido ou o controlador pode avisar a IHM que a brassagem terminou. Nas duas situações os dispositivos limpam os dados referente a receita executada e ficam aguardando o início de um novo processo.

Figura 27 – Sequência de bytes enviados pelo controlador à IHM.

Sinaliza o início da atualização	Rampa atual	Bits mais significativos do tempo decorrido	Bits menos significativos do tempo decorrido	Bits mais significativos da temperatura atual	Bits menos significativos da temperatura atual	Potência atual	Status do temporizador
1 byte	1 byte	1 byte	1 byte	1 byte	1 byte	1 byte	1 byte

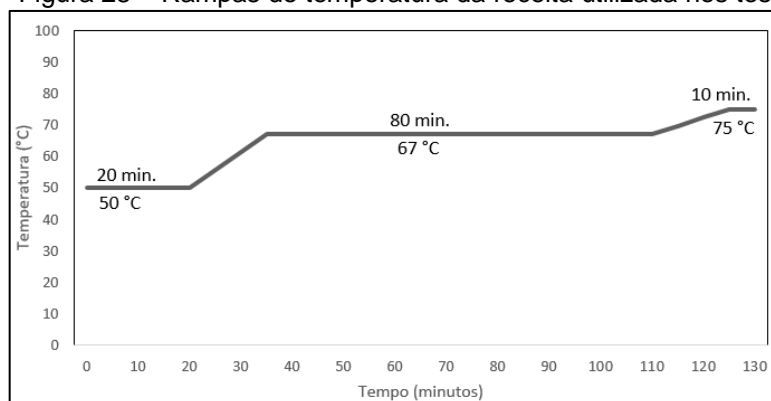
Fonte: Do autor.

¹ A biblioteca se encontra disponível no seguinte endereço: http://jlog.org/v4/linux/x86_64-unknown-linux-gnu/RXTXcomm.jar

6.1.3 Testes com o protótipo

Após o término do desenvolvimento e da montagem dos componentes do protótipo, deu-se início aos testes. Para essa etapa foi selecionada a cerveja *belgian golden strong* com as seguintes fases no processo de brassagem: primeiro a água dentro da panela deve ser aquecida a 54°C. Assim que essa temperatura for atingida, deve-se colocar o malte na água, e manter a temperatura em 50°C por 20 minutos. Ao término desse período a temperatura deve ser elevada a 67°C durante 80 minutos. Por fim, o malte deve ser aquecido até 75°C por 10 minutos. A Figura 28 ilustra as rampas de temperatura da receita.

Figura 28 – Rampas de temperatura da receita utilizada nos testes.



Fonte: Do autor.

A partir da receita, foram salvas no protótipo as rampas de temperatura e se iniciou a brassagem. O processo foi realizado em três receitas de 20 litros. Para aferir a temperatura de acordo com o tempo e a conversão do amido, foram registrados para cada teste, a temperatura, obtida pela IHM do protótipo e a taxa de açúcar do mosto, obtida por meio de um refratômetro, em uma frequência de cinco em cinco minutos.

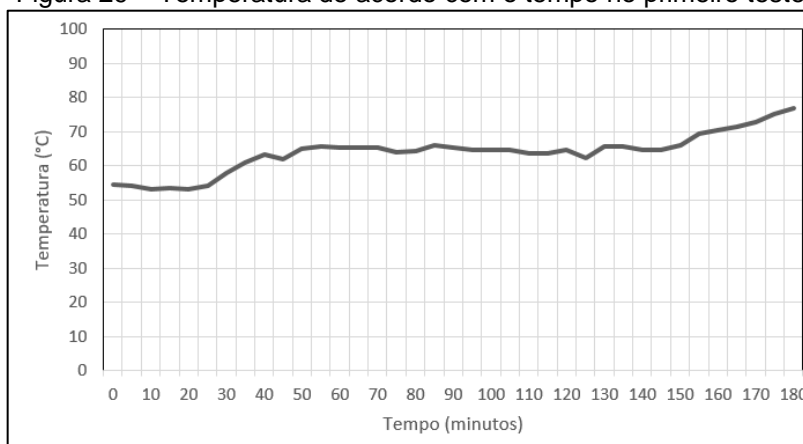
6.2 RESULTADOS

Tendo como base os dados obtidos nos testes citados anteriormente, é possível avaliar a capacidade do protótipo em realizar o processo. A seguir serão apresentadas as figuras referentes aos gráficos dos dados coletados, dentre essas, a

29, 31 e 33 representam a temperatura (°C) de acordo com o tempo, e a 30,32 e 34 mostram a taxa de açúcar (°Bx) ao longo do tempo.

Na Figura 29 é possível observar as rampas de temperatura geradas pelo protótipo no primeiro teste. Na primeira rampa (de 50°C), dada a inércia térmica da panela utilizada, é possível visualizar que a temperatura não diminuiu até o valor ideal a tempo, gerando uma média de 53,71°C com desvio padrão de 0,64°C, porém, como visto no capítulo 2.2, a faixa de temperatura dessa fase é de 45°C à 55°C, o que resulta em um valor dentro do estipulado. Passados os 20 minutos, é possível visualizar que o protótipo acionou o sistema de aquecimento de maneira correta para chegar até a segunda rampa (de 67°C). Nessa fase o controlador fez com que a média da temperatura ficasse em 64,75°, com desvio padrão de 0,90°C. A média ficou 2,25°C abaixo do apontado e fora da faixa de atuação da enzima estipulada (66°C à 72°C), fazendo com que outra enzima atuasse, o que pode alterar as características previstas para a receita. Completado o tempo dessa rampa, novamente o controlador aqueceu o mosto, entrando na última fase. Na rampa final (de 75°C), foi obtida uma média de 74,83°C, com desvio padrão de 2°C, valor próximo do estipulado, sem ultrapassar a temperatura limite (76°C), evitando a extração de compostos indesejáveis do malte. Ao término dessa rampa o controlador indicou o fim e processo.

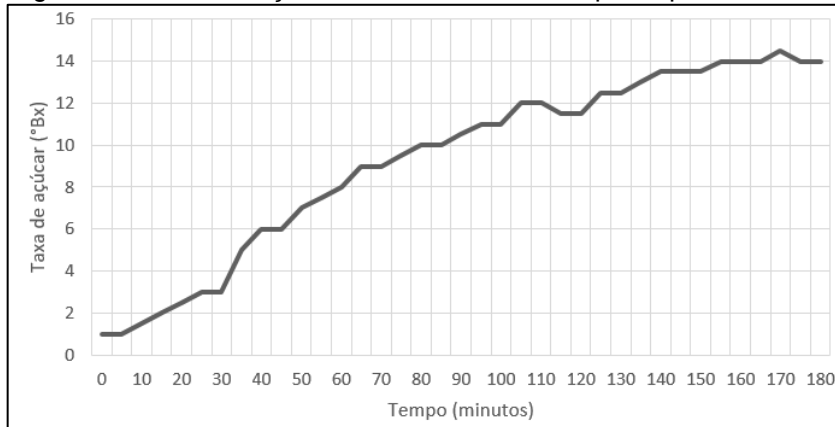
Figura 29 – Temperatura de acordo com o tempo no primeiro teste.



Fonte: Do autor.

Em relação a conversão do amido, a Figura 30 apresenta a evolução da quantidade de açúcares presentes no mosto. Inicialmente a mistura contava com 1°Bx e no fim do teste o valor foi de 14°Bx. O valor estimado para a receita é de 13,6°Bx, apontando que o objetivo da brassagem foi concluído.

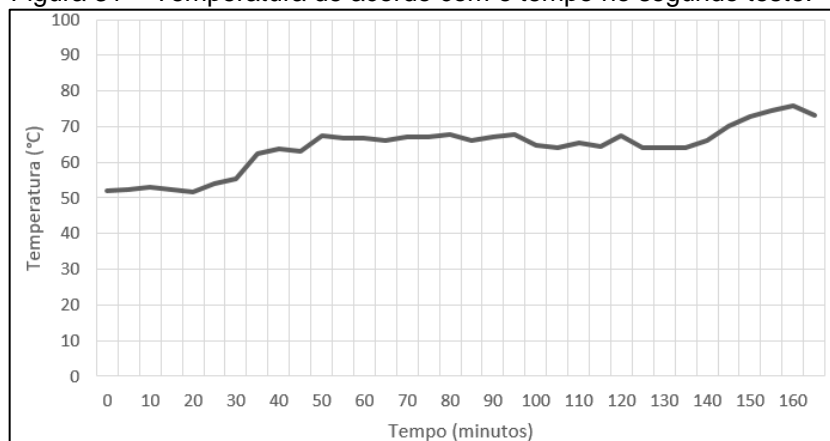
Figura 30 – Taxa de açúcar de acordo com o tempo no primeiro teste.



Fonte: Do autor.

No segundo teste, o protótipo gerou resultados semelhantes, como ilustrado na Figura 31. Para a primeira rampa, foi obtida uma média de 52,3°C, com desvio padrão de 0,48°C, novamente acima do estipulado, mas dentro da faixa aceitável. Ao término do tempo da primeira fase, o controlador elevou a temperatura para o valor definido na segunda fase. Essa rampa gerou uma média de 66,08°C com o desvio padrão de 1,34°C, valor próximo do estipulado, e dentro da faixa de atuação da enzima prevista. Por fim, o controlador prosseguiu para a última etapa, que ao término gerou uma média de 75,5°C, com o desvio padrão de 1,25°C. Nessa fase a média ficou 0,5°C acima do estipulado, porém o limite de 76°C não foi ultrapassado.

Figura 31 – Temperatura de acordo com o tempo no segundo teste.

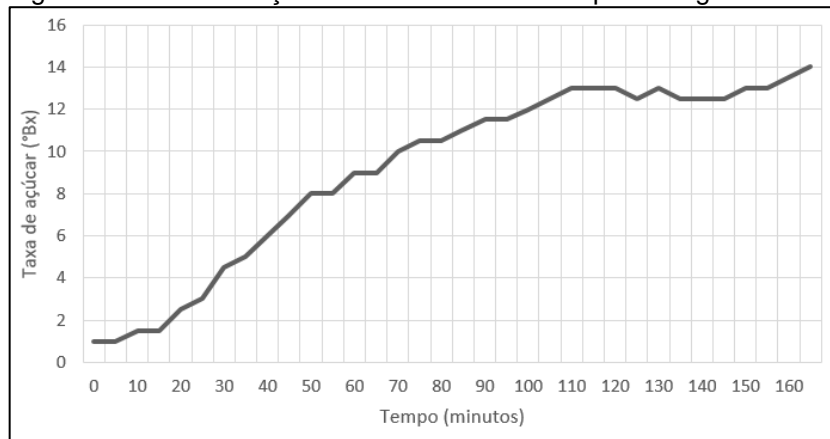


Fonte: Do autor.

A conversão de amido no segundo teste ocorreu como o previsto, sendo apresentado na Figura 32. Inicialmente o mosto continha uma taxa de açúcar de 1°Bx

e ao final do processo a taxa era de 14°Bx, 0,4°Bx acima do estipulado, indicando que o resultado para o processo também foi próximo do desejado.

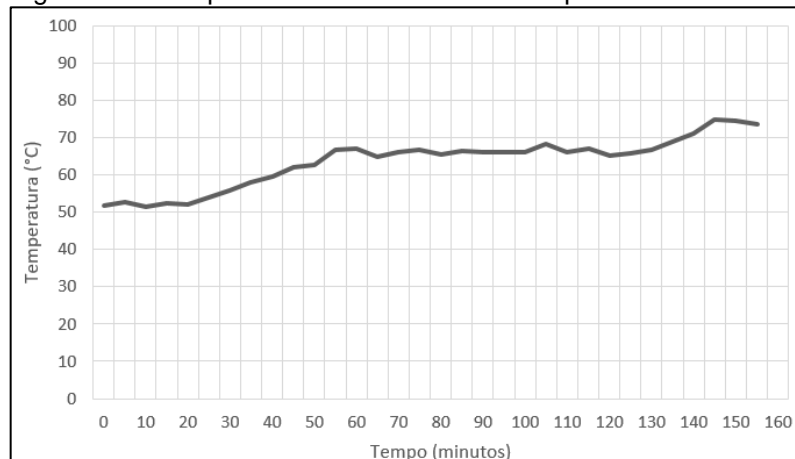
Figura 32 – Taxa de açúcar de acordo com o tempo no segundo teste.



Fonte: Do autor.

No último teste o controlador teve um desempenho similar, como indica a Figura 33. Na primeira rampa se obteve uma média de 52,1°C, com um desvio padrão de 0,52°C, comportamento observado também nos outros testes e dentro dos limites aceitáveis. Após o término do tempo, o controlador acionou o aquecimento e iniciou a segunda fase, que ao fim, obteve uma média de 66,33°C, com um desvio padrão de 0,81°C, valor dentro da faixa de atuação da enzima prevista. Dado o tempo configurado, o controlador elevou a temperatura do mosto nos parâmetros da terceira rampa. Ao término dessa, a média de temperatura foi de 74,33°C, com o desvio padrão de 0,52°C, valor próximo ao previsto, não ultrapassando o limite de atuação das enzimas.

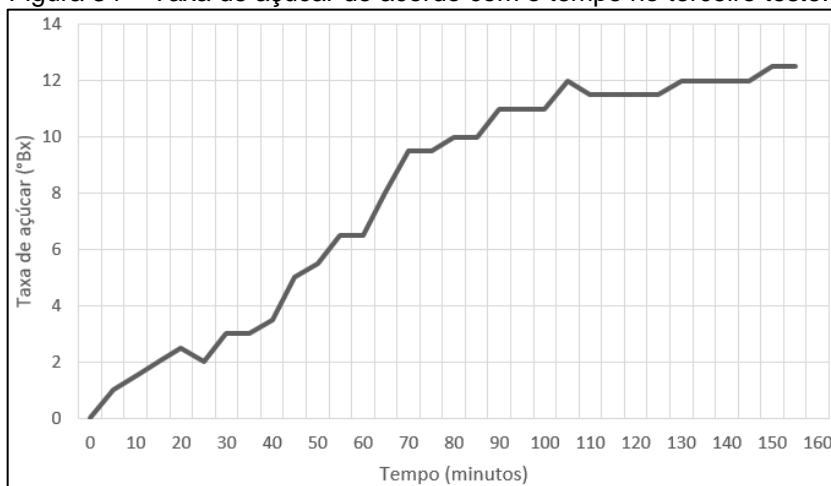
Figura 33 – Temperatura de acordo com o tempo no terceiro teste.



Fonte: Do autor.

Como é possível visualizar na Figura 34, o objetivo da brassagem também foi alcançado no último teste, porém, com uma diferença de valores em relação aos outros. O processo iniciou com 1°Bx, e ao término a taxa de açúcar foi de 12,5°Bx, valor abaixo de previsto, o que pode ocasionar na redução do teor alcoólico da bebida.

Figura 34 – Taxa de açúcar de acordo com o tempo no terceiro teste.



Fonte: Do autor.

É possível observar também diferenças de tempo total para concluir a brassagem entre os testes, sendo 180 minutos no primeiro, 165 minutos no segundo e 160 minutos no terceiro. Isso se deu devido a temperatura ambiente dos testes, que influencia na transição de uma rampa para outra.

Os resultados obtidos indicam que o controlador foi capaz de realizar a brassagem nos três testes realizados na receita utilizada.

6.2.1 Discussão dos Resultados

Em relação ao controle de temperatura, foi possível observar a estabilidade das médias obtidas, porém em algumas situações o valor pode ficar abaixo do estipulado. Também se constatou que a ação de controle implementada atuou de forma correta, evitando grandes oscilações de temperatura, mesmo com a inércia térmica do equipamento.

Outro ponto observado foi a capacidade de controle sem a necessidade de supervisão humana constante, quando comparado ao processo manual. O protótipo foi capaz de alterar a temperatura alvo da receita, dado o tempo estipulado e de aguardar a transição de uma rampa para outra. Além disso, em relação a fabricação

de modo convencional, as informações de tempo restante fornecidas pela IHM também auxiliam o mestre cervejeiro no processo.

A pesquisa apresentada por Baldessar (2008) realiza o controle de temperatura em outra área, e além do aquecimento é realizado também o resfriamento, com apenas a ação de controle proporcional, diferentemente dessa pesquisa, que focou no aquecimento e utilizou o controle proporcional e derivativo.

Na pesquisa de Borges e Fröhlich (2012), foram realizados o controle na etapa de brassagem e de fervura no processo de produção de cerveja. Dados os testes realizados por eles, o controle de temperatura teve desempenho superior ao deste trabalho, possuindo oscilações em níveis menores.

Em sua pesquisa, os autores Souza e Matinello (2013) realizam o controle de temperatura utilizando como controlador um circuito FPGA. Apesar da área de atuação ser diferente, a partir dos testes realizados por eles, é possível observar uma grande oscilação da temperatura controlada, apontando que o meio de controle implementado nesta pesquisa teve desempenho superior.

No trabalho apresentado por Tostes (2015), foram apresentados vários testes de controle para a etapa de brassagem, sendo que os resultados obtidos foram semelhantes ao deste trabalho em relação a margens de diferença de temperatura.

A pesquisa apresentada por Rusia e Bhongade (2014) realiza o controle de temperatura em baixos níveis, por meio de uma carga de potência inferior de corrente contínua e sem uma aplicação específica, não tornando possível a comparação com este trabalho.

7 CONCLUSÃO

Os sistemas embarcados estão difundidos por toda a parte, desde os ambientes domésticos até os industriais. Dentre as diversas aplicações possíveis, desses sistemas, o controle de temperatura exerce um papel importante em vários processos.

A fim de atender parte dessa demanda, o objetivo principal desse trabalho foi desenvolver um sistema embarcado para o controle de temperatura no processo de produção de cerveja baseado em FPGA. Para tal, foram implementadas a ação de controle e demais componentes do controlador utilizando VHDL, juntamente os circuitos necessários e a interface humano-máquina.

As dificuldades encontradas foram em relação à detecção da passagem por zero, em que o controlador interpretava o sinal de forma errada, o que impedia o controle de fase. Outro ponto foi o controle de potência, que apresentava o comportamento errado na condução de correntes altas.

Para o desenvolvimento utilizando FPGA foi necessário o estudo da linguagem VHDL, visto que os seus conceitos são diferentes da programação tradicional utilizados em outros embarcados.

Apesar dessas dificuldades, os resultados obtidos foram considerados satisfatórios para que o objetivo dessa pesquisa fosse atendido, tanto o geral quanto os específicos, sendo que o protótipo desenvolvido foi capaz de efetuar o controle de temperatura na etapa de brassagem.

Em todos os testes realizados, o protótipo influenciou a temperatura de acordo com os parâmetros definidos a fim de cumprir o papel da etapa de brassagem, que a conversão do amido do malte em açúcares fermentáveis. Exceto por rampa específica em que o valor obtido ultrapassou a margem aceitável, o controlador obteve médias satisfatórias, o que aponta seu bom resultado.

Tendo como base o conhecimento adquirido pelos estudos e pelo desenvolvimento do protótipo, pode-se observar as seguintes possibilidades de trabalhos futuros:

- a) estender o controle de temperatura também para o processo de fervura, com avisos sobre quando deve ser adicionado os lúpulos na mistura e pausas de tempo para as demais tarefas da etapa;

- b) realizar a coleta e avaliação dos dados adquiridos pelo controlador, dando a possibilidade da construção de gráficos e outros meios de visualizar informações que podem ser úteis ao mestre cervejeiro;
- c) implementar a IHM utilizando um aplicativo *mobile* e conexão sem fio para sincronização e obtenção dos dados.

REFERÊNCIAS

AHMED, A. **Eletronica De Potencia**. São Paulo: Pearson Prentice Hall, 2000.

ALMEIDA, R. M. A. DE; MORAES, C. H. V. DE; SERAPHIM, T. DE F. P. **Programação de Sistemas Embarcados: Desenvolvendo Software para Microcontroladores em Linguagem C**. 1. ed. [s.l.] Elsevier Brasil, 2017.

BALDESSAR, A. F. **Controle De Temperatura Por Meio De Computação Embarcada**. [s.l.] Universidade do Extremo Sul Catarinense, 2008.

BERTINI, L. **Microcontroladores Pic 16 F628/16 F628 A Em Assembly**. [s.l.] Clube do Autores', 2010.

BORGES, J. C.; FRÖHLICH, T. **Desenvolvimento De Dispositivo De Controle Automático Para Os Processos De Mosturação E Fervura Da Fabricação De Cerveja Artesanal**. [s.l.] Instituto Federal De Educação, Ciência E Tecnologia De Santa Catarina Campus Joinville, 2012.

BRAGA, N. C. **Curso de Eletrônica - Volume 4 - Eletrônica Digital - 2**. 1ª ed. São Paulo: Editora NCB, 2017.

CAPPELATTI, E.; TOROK, D. L. **Praticando VHDL**. Novo Haburgo: Editora Feevale, 2010.

COSTA, C. DA. **Projetos De Circuitos Digitais Com FPGA**. 3. ed. São Paulo: Erica, 2014.

D'AMORE, R. **VHDL: Descrição E Síntese De Circuitos Digitais**. 2. ed. Rio de Janeiro: Grupo Gen - LTC, 2012.

DUNN, W. C. **Fundamentos de Instrumentação Industrial e Controle de Processos**. Porto Alegre: Bookman Editora, 2013.

FIALHO, A. B. **Instrumentação Industrial: Conceitos, aplicações e análises**. 7ª ed. São Paulo: Saraiva Educação S.A., 2010.

FRADEN, J. **Handbook of Modern Sensors: Physics, Designs, and Applications**. 4ª ed. New York: [s.n.].

FRANKLIN, G. F.; POWELL, J. D.; EMAMI-NAEINI, A. **Sistemas de Controle para Engenharia - 6ed**. Porto Alegre: Bookman Editora, 2013.

GEORGE, V.; RABAEY, J. M. **Low-Energy FPGAs — Architecture and Design**. New York: Springer US, 2012.

GURGEL, M.; CUNHA, J. M. F. **Cerveja com design**. 1. ed. São Paulo: Editora Senac São Paulo, 2019.

IBRAHIM, D. **Microcontroller-Based Temperature Monitoring and Control**. Woburn: Newnes, 2002.

IBRAHIM, D. **Advanced PIC Microcontroller Projects in C: From USB to RTOS with the PIC 18F Series**. Burlington: Elsevier, 2008.

LAMB, F. **Automação Industrial na Prática - Série Tekne**. New York: AMGH Editora, 2015.

LIRA, F. A. DE. **Metrologia na indústria**. São Paulo: Saraiva Educação S.A., 2016.

MÜLLER, A. **Cerveja!** [s.l.] Editora da ULBRA, 2002.

NACHEL, M.; ETTLINGER, S.; BRAGA, S. V. **Cerveja Para Leigos**. Rio de Janeiro: Alta Books, 2018.

NULL, L.; LOBUR, J. **Princípios Básicos de Arquitetura e Organização de Computadores**. Porto Alegre: Bookman, 2011.

ORDONEZ, E. D. M. et al. **Projeto, desempenho e aplicações de sistemas digitais em circuitos programáveis (FPGAs)**. Pompéia: Bless, 2003.

PALMER, J. J. **How to Brew: Everything you need to know to brew beer right the first time**. Boulder: Brewers Publications, 2006.

PENEDO, S. R. M. **Sistemas de Controle: Matemática Aplicada a Projetos**. São Paulo: Saraiva Educação S.A., 2014.

PETRUZELLA, F. D. **Controladores Lógicos Programáveis**. 4. ed. Porto Alegre: AMGH Editora, 2013.

PETRUZELLA, F. D. **Eletrotécnica I: Série Tekne**. Porto Alegre: Bookman Editora, 2014.

ROCHA, H. V. DA. **Design e avaliação de interfaces humano-computador**. Campinas: Unicamp, 2003.

RUSIA, P.; BHONGADE, S. **Design And Implementation of Digital PID Controller Using FPGA For Precision Temperature Control**. Proceedings of 6th IEEE Power India International Conference, PIICON 2014, 2014.

SANTOS, S. **Introdução à IoT: Desvendando a Internet das Coisas**. [s.l.] CLUBE DE AUTORES, 2018.

SCHULER, C. **Eletrônica II - 7: Série Tekne**. 7ª ed. Porto Alegre: AMGH Editora, 2013.

SKLIAROVA, I.; FERRARI, A. B. **Introdução à computação reconfigurável**. Revista Do DETUA, set. 2003.

SOUZA, A. B. DE; MATINELLO, D. **Sistema de Controle de Temperatura Usando FPGA**. [s.l.] Universidade Tecnológica Federal do Paraná, 2013.

TERASIC. **DE10-Lite User Manual**. Disponível em: <https://www.terasic.com.tw/cgi-bin/page/archive_download.pl?Language=English&No=1021&FID=a13a2782811152b477e60203d34b1baa>. Acesso em: 23 jun. 2020.

TOSTES, L. R. DE M. **Instrumentação e Controle do Processo de Produção de uma Microcervejaria**. [s.l.] UNIVERSIDADE FEDERAL DO RIO DE JANEIRO, 2015.

VENTURINI FILHO, W. G. **Bebidas Alcoolicas Ciencia E Tecnologia**. São Paulo: BLUCHER, 2018.

APÊNDICE A – ARTIGO

SISTEMAS EMBARCADOS UTILIZANDO FPGA: ESTUDO DE CASO NO CONTROLE DE TEMPERATURA NA ETAPA DE BRASSAGEM

Marcelo Minato Rezin¹, Paulo João Martins¹, Emerson Colonetti²

¹Ciência da Computação – Universidade do Extremo Sul Catarinense (UNESC)

Criciúma – SC – Brasil

²Engenharia Química – Universidade do Extremo Sul Catarinense (UNESC)

Criciúma – SC – Brasil

marcelo.rezin@gmail.com, pjm@unesc.net, emersonc@unesc.net

Abstract. *The wide diffusion of embedded systems makes it necessary to search by alternatives that are versatile and flexible, such as FPGA circuits. The temperature control in the mash stage in brewing can enjoy the advantages of this type of circuit. Based on this, this research proposed the building of an FPGA based embedded system for such task, using a development board to implement a controller prototype that influenced the temperature of according to the parameters informed.*

Resumo. *A ampla difusão dos sistemas embarcados torna necessário a busca por alternativas que sejam versáteis e flexíveis, como os circuitos FPGA. O controle de temperatura na etapa de brassagem na produção de cerveja pode usufruir das vantagens desse tipo de circuito. Com base nisso, este trabalho propôs a construção de um sistema embarcado baseado em FPGA para tal tarefa, utilizando uma placa de desenvolvimento para implementar um protótipo de controlador que influenciava na temperatura de um recipiente de acordo com os parâmetros informados.*

1. Introdução

Os sistemas embarcados são dispositivos que atendem à demanda de sistemas computacionais que realizam funções dedicadas, tanto em ambientes residenciais quanto industriais (NULL; LOBUR, 2011).

As alternativas geralmente adotadas envolvem soluções baseadas em circuitos integrados de aplicação específica, do inglês, *Application Specific Integrated Circuits* (ASIC), que possuem baixos custos de produção, porém com a desvantagem de tempo de desenvolvimento e custo do projeto elevados (BRAGA, 2017), ou baseadas em microcontroladores, em que as funções necessárias são programadas, porém, em alguns casos, esses componentes não possuem todos os recursos necessários demandando de circuitos externos, ou possuem funções a mais, o que causa desperdício (BRAGA, 2017).

Dada as desvantagens das alternativas já citadas, é necessário que se explore alternativas versáteis e flexíveis, como o arranjo de portas programáveis em campo, do inglês, *Field Programmable Gate Array* (FPGA). Esses dispositivos tem como principal característica a

capacidade de ser um circuito reprogramável (ORDONEZ et al., 2003), o que permite sua utilização em diversas áreas.

Uma possível aplicação de sistemas embarcados é no controle de temperatura, que representa um ponto importante de vários processos, como na etapa de brassagem na produção de cerveja. Por meio deste controle o mestre cervejeiro tem a capacidade de regular a ativação de certas enzimas que são responsáveis por características fundamentais da bebida, como teor alcoólico e corpo (VENTURINI FILHO, 2018).

Dada a necessidade do controle de temperatura na etapa de brassagem, foi construído um protótipo de sistema embarcado baseado em FPGA que realizasse tal tarefa. O controlador desenvolvido utiliza como interface humano-máquina (IHM) uma aplicação desktop e possui a capacidade de influenciar o sistema de aquecimento do recipiente de acordo com os parâmetros informados.

2. Brassagem na Produção de Cerveja

A cerveja é uma antiga bebida alcoólica com altos números de consumo atualmente. Sendo composta principalmente por cereais maltados ou não, água, lúpulo e levedura, possui várias etapas em seu processo de produção, que garantem seu aroma e sabor característico (VENTURINI FILHO, 2018).

Dentre as etapas para sua produção, a fase de brassagem representa um importante papel, pois define características como teor alcoólico e corpo da bebida. Nessa fase o malte moído é infundido em água (formando o mosto), com o principal objetivo de transformar o amido presente nos grãos em açúcares por meio da quebra de suas moléculas pela adição de água (hidrólise) e ação de enzimas, para que posteriormente sejam fermentados (MÜLLER, 2002).

Existem várias enzimas que podem ser utilizadas no processo, e, suas respectivas ações podem ser favorecidas pela faixa de temperatura utilizada e níveis de pH. O ajuste de temperatura e consequente ativação de enzima permite ao cervejeiro, de acordo com a receita ou o seu gosto, destacar características da bebida, como uma maior quantidade de açúcares que podem ser fermentados, tornando-a mais alcoólica, ou maior quantidade de substâncias com mais peso molecular, tornando-a mais encorpada (VENTURINI FILHO, 2018).

A partir do conhecimento da temperatura de ativação das enzimas, é possível definir graus de temperatura em que o mosto será submetido, a fim de favorecer as características necessárias.

3. Sistemas Embarcados

Para gerenciar a temperatura de acordo com os parâmetros definidos pelo mestre cervejeiro, é necessário a presença de um controlador. Para essa função pode-se utilizar um sistema embarcado. Este tipo de dispositivo é um sistema de computação dedicado que realiza funções específicas, diferentemente de computadores de uso geral, que realizam diversas tarefas diferentes (LAMB, 2015).

Os sistemas embarcados podem ser implementados de diversas maneiras, como a partir da utilização de microcontroladores, ou baseados em circuitos ASICs, porém a flexibilidade proporcionada pelos FPGAs garante ao sistema desempenho juntamente a otimização dos recursos.

3.1 FPGA

Os FPGAs são circuitos formados por células lógicas ou blocos lógicos configuráveis organizados em forma de matriz. Cada uma dessas células tem a capacidade de implementar funções lógicas e de se comunicar com outras células. De acordo com cada projeto os blocos são conectados pela matriz, interligando as funções lógicas, permitindo o fornecimento e recebimento informações pelos blocos de entrada e saída (COSTA, 2014).

Para montar um arranjo de portas lógicas em um circuito FPGA se utiliza uma linguagem de descrição de hardware, do inglês, *Hardware Description Language* (HDL), que permite modelar os circuitos de forma que possam ser carregados na placa. Neste trabalho a HDL utilizada foi a linguagem de projeto de hardware para circuitos integrados de velocidade muito alta, do inglês VHSIC *Hardware Description Language*, onde VHSIC é a sigla de *Very High Speed Integrated Circuits* (VHDL).

4. Desenvolvimento do Protótipo

Esta pesquisa teve como objetivo criação de um protótipo de sistema embarcado utilizando um circuito FPGA como controlador de temperatura na etapa de brassagem no processo de produção de cerveja.

Para tal foi realizado o levantamento bibliográfico e a partir do conhecimento adquirido se definiu a placa FPGA a ser utilizada bem como os outros componentes necessários para a construção do protótipo.

Na etapa seguinte foi realizada a descrição do hardware que seria utilizado na placa, a montagem dos circuitos necessários para o controle da temperatura e desenvolvimento da IHM.

A etapa final teve como base a realização dos testes com o protótipo para a obtenção dos dados necessários para averiguar os resultados obtidos.

4.1. Definição e Construção dos componentes

Para o desenvolvimento do protótipo foi necessário a escolha dos equipamentos e periféricos que seriam utilizados, juntamente com a construção de alguns circuitos.

A placa escolhida foi a plataforma de prototipação DE10-Lite, fornecida pela fabricante Terasic. A mesma conta com um circuito FPGA Altera Max 10, desenvolvido pela Intel, com 50 mil elementos lógicos e 40 entradas/saídas de propósito geral.

Como recipiente se definiu uma panela industrial em inox (Figura 1), de parede dupla com capacidade máxima de 120 litros. O equipamento também possui uma resistência elétrica do tipo flange com 6kW de potência, sendo que o sistema de aquecimento não entra em contato com o conteúdo, e o calor produzido é conduzido por um fluido térmico.



Figura 1. Apresentação da panela.

Para controlar a potência do sistema de aquecimento, foi desenvolvido um circuito para informar ao controlador a passagem por zero da senoide da rede elétrica, e outro para controlar o fornecimento de tensão à resistência pela técnica de controle de fase a partir de um sinal de disparo fornecido pelo protótipo.

Em relação a aferição de temperatura, se utilizou um sensor termopar do tipo K juntamente com o módulo MAX6675, que possui uma interface de comunicação serial SPI.

A Figura 2 apresenta o controlador e os circuitos do protótipo. A seleção em amarelo indica o circuito de detecção da passagem de zero, a em azul o módulo de leitura de temperatura, em laranja o disparo do TRIAC e em verde o controle de potência. O círculo em vermelho indica onde deve ser conectado a fase do sistema de aquecimento e o roxo o neutro.

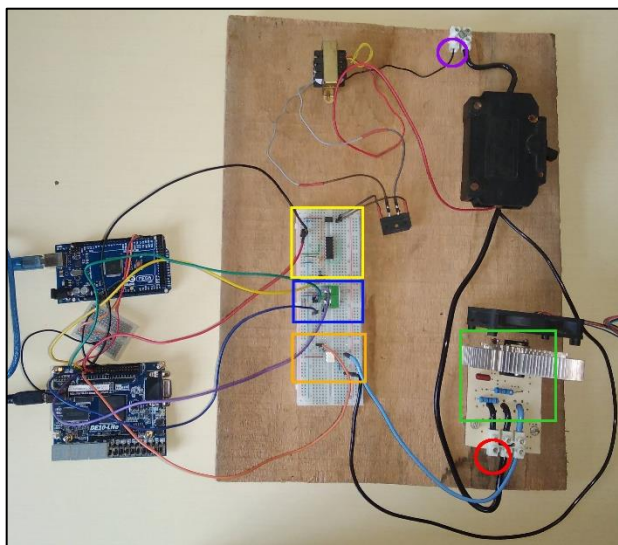


Figura 2. Controlador e circuitos do protótipo.

4.2 Desenvolvimento do Software

O desenvolvimento do software foi dividido entre a descrição do hardware do controlador, divididas em várias entidades que realizam funções específicas, e da programação da IHM, uma aplicação desktop responsável pela interação com o usuário.

Na descrição do hardware se criou as entidades responsáveis pelo controle de potência, pela comunicação com o módulo de temperatura, por realizar o controle PID, para armazenamento e execução das rampas e pela comunicação com a IHM. Cada uma dessas entidades foi testada de forma isolada e ao fim foram agrupadas para formarem o controlador em si.

A aplicação desktop para IHM foi desenvolvida utilizando a linguagem Java e teve como propósito permitir que o usuário registrasse os parâmetros das receitas e acompanhasse o processo de produção em tempo real. A comunicação e sincronização dos dados com o controlador se deu por meio de comunicação serial.

A Figura 3 apresenta a tela de acompanhamento do processo da IHM. Nela são mostradas a rampa executada, tempo, temperatura atual, a potência aplicada ao sistema de aquecimento e se o temporizador está executando ou aguardando o aquecimento.

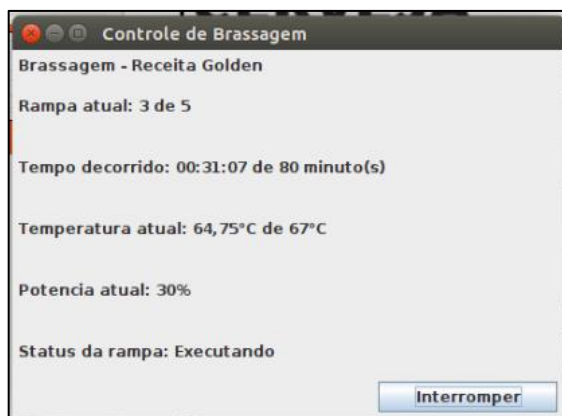


Figura 3. Tela de acompanhamento do processo.

4.3 Testes com o Protótipo

Para realização dos testes com o protótipo foi selecionada a cerveja *belgian golden strong*, com as seguintes fases no processo de brassagem: primeiro a água dentro da panela deve ser aquecida a 54°C. Assim que essa temperatura for atingida, deve-se colocar o malte na água, e manter a temperatura em 50°C por 20 minutos. Ao término desse período a temperatura deve ser elevada a 67°C durante 80 minutos. Por fim, o malte deve ser aquecido até 75°C por 10 minutos. A Figura 4 ilustra as rampas de temperatura da receita.

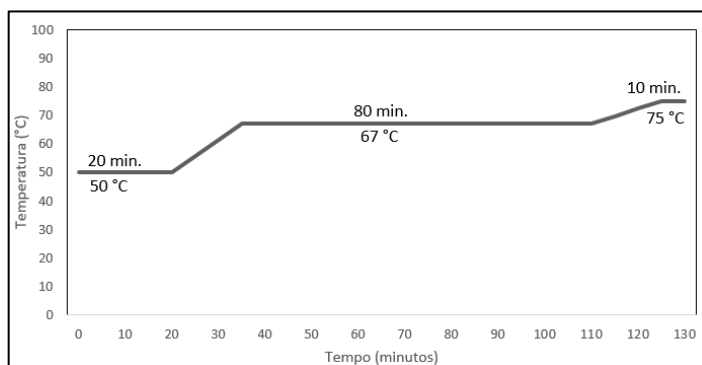


Figura 4. Rampas de temperatura da receita utilizada nos testes.

A partir da receita, foram salvas no protótipo as rampas de temperatura e se iniciou a brassagem. O processo foi realizado em três receitas de 20 litros. Para aferir a temperatura de acordo com o tempo e a conversão do amido, foram registrados para cada teste, a temperatura, obtida pela IHM do protótipo e a taxa de açúcar do mosto, obtida por meio de um refratômetro, em uma frequência de cinco em cinco minutos.

5. Resultados e Discussões

Tendo como base os dados obtidos nos testes citados anteriormente, é possível avaliar a capacidade do protótipo em realizar o processo.

No primeiro teste, o controlador foi capaz de influenciar na temperatura do mosto de acordo com os tempos definidos na receita. Foram observadas variações causadas pela inercia térmica do recipiente e na ação de controle do protótipo. A maior diferença observada foi na segunda rampa (67°C) em que o controlador obteve uma média de 64,75° com um desvio padrão 0,90°C. A média ficou 2,25°C abaixo do apontado e fora da faixa de atuação da enzima estipulada.

Em relação a conversão do amido desse primeiro teste, inicialmente a mistura contava com 1°Bx e no fim do teste o valor foi de 14°Bx. O valor estimado para a receita é de 13,6°Bx, apontando que o objetivo da brassagem foi concluído.

No segundo teste, o protótipo gerou resultados semelhantes, sendo capaz de ajustar a temperatura de acordo com o tempo, porém as médias obtidas ficaram dentro das faixas de atuação das enzimas estipuladas, como a Figura 5 ilustra.

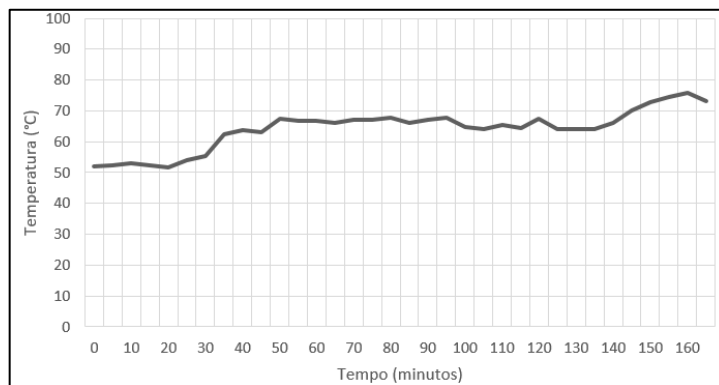


Figura 5. Temperatura de acordo com o tempo no segundo teste.

A conversão de açúcar no segundo teste também ocorreu de forma satisfatória, sendo apresentado na Figura 6. Inicialmente o mosto continha uma taxa de açúcar de 1°Bx e ao final do processo a taxa era de 14°Bx, 0,4°Bx acima do estipulado, indicando que o resultado para o processo também foi próximo do desejado.

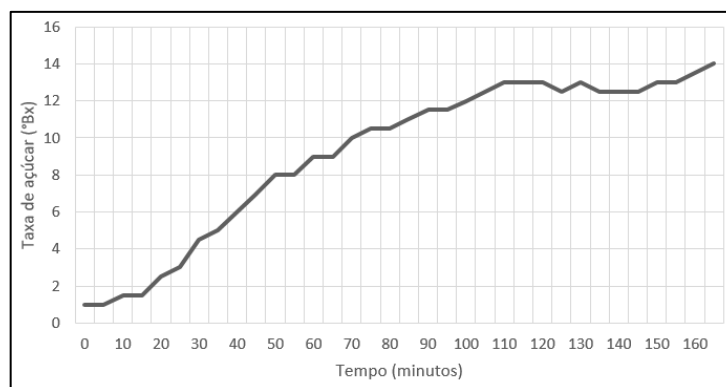


Figura 6. Taxa de açúcar de acordo com o tempo no segundo teste.

Para o último teste o controlador teve um desempenho similar, realizando o controle de temperatura dentro do estipulado nos parâmetros e sem ultrapassar a faixa de atuação das enzimas.

Quanto a conversão de açúcar, o objetivo da brassagem também foi alcançado no último teste, porém, com uma diferença de valores em relação aos outros. O processo iniciou com 1°Bx, e ao término a taxa de açúcar foi de 12,5°Bx, valor abaixo de previsto, o que pode ocasionar na redução do teor alcoólico da bebida.

Os resultados obtidos indicam que o controlador foi capaz de realizar a brassagem nos três testes realizados na receita utilizada, sendo possível observar a estabilidade das médias obtidas, porém em algumas situações o valor pode ficar abaixo do estipulado. Também se constatou que a ação de controle implementada atuou de forma correta, evitando grandes oscilações de temperatura, mesmo com a inércia térmica do equipamento.

Outro ponto observado foi a capacidade de controle sem a necessidade de supervisão humana constante, quando comparado ao processo manual. O protótipo foi capaz de alterar a temperatura alvo da receita, dado o tempo estipulado e de aguardar a transição de uma rampa para outra. Além disso, em relação a fabricação de modo convencional, as informações de tempo restante fornecidas pela IHM também auxiliam o mestre cervejeiro no processo.

6. Conclusão

Os sistemas embarcados estão difundidos por toda a parte, desde os ambientes domésticos até os industriais. Dentre as diversas aplicações possíveis desses sistemas, o controle de temperatura exerce um papel importante em vários processos.

A fim de atender parte dessa demanda, o objetivo principal desse trabalho foi desenvolver um sistema embarcado para o controle de temperatura no processo de produção de cerveja baseado em FPGA. Para tal, foram implementadas a ação de controle e demais componentes do controlador utilizando VHDL, juntamente os circuitos necessários e a interface humano-máquina.

As dificuldades encontradas foram em relação à detecção da passagem por zero, em que o controlador interpretava o sinal de forma errada, o que impedia o controle de fase. Outro ponto foi o controle de potência, que apresentava o comportamento errado na condução de correntes altas.

Apesar dessas dificuldades, os resultados obtidos foram considerados satisfatórios para que o objetivo dessa pesquisa fosse atendido, sendo que o protótipo desenvolvido foi capaz de efetuar o controle de temperatura na etapa de brassagem.

Em todos os testes realizados, o protótipo influenciou a temperatura de acordo com os parâmetros definidos a fim de cumprir o papel da etapa de brassagem, que é a conversão do amido do malte em açúcares fermentáveis. Exceto por rampa específica em que o valor obtido ultrapassou a margem aceitável, o controlador obteve médias satisfatórias, o que aponta seu bom resultado.

7. Referências

BRAGA, N. C. **Curso de Eletrônica - Volume 4 - Eletrônica Digital - 2.** 1ª ed. São Paulo: Editora NCB, 2017.

COSTA, C. DA. **Projetos De Circuitos Digitais Com FPGA.** 3. ed. São Paulo: Erica, 2014.

LAMB, F. **Automação Industrial na Prática - Série Tekne.** New York: AMGH Editora, 2015.

MÜLLER, A. **Cerveja!** [s.l.] Editora da ULBRA, 2002.

NULL, L.; LOBUR, J. **Princípios Básicos de Arquitetura e Organização de Computadores.** Porto Alegre: Bookman, 2011.

ORDONEZ, E. D. M. et al. **Projeto, desempenho e aplicações de sistemas digitais em circuitos programáveis (FPGAs).** Pompéia: Bless, 2003.

VENTURINI FILHO, W. G. **Bebidas Alcoolicas Ciencia E Tecnologia.** São Paulo: BLUCHER, 2018.