

CHUVEIRO ELETRÔNICO

Filipe de Souza Freitas
Graduação em Engenharia Eletrônica
Universidade de Brasília
Gama, DF, Brasil
14/0020161
filipe.desouzafreitas@gmail.com

Nauam Victor Reis de Oliveira
Graduação em Engenharia Eletrônica
Universidade de Brasília
Gama, DF, Brasil
16/0140056
nauamvictor@outlook.com

Resumo—O projeto compreende a construção de um sistema embarcado que controle a temperatura da água de um chuveiro, com acionamento feito por voz ao conjunto de opções pre-estabelecidas. Utilizando uma API de conversão de fala para texto pode-se definir a temperatura da água.

Index Terms—Controle, temperatura, embarcado, voz, chuveiro.

I. INTRODUÇÃO

Atualmente existe duas formas principais de aquecimento de água para chuveiros por meio da resistência elétrica e outra por energia solar.

O chuveiro elétrico é constituído por uma resistência. O resistor é uma peça metálica que tem a capacidade de chegar a altas temperaturas sem se danificar, assim a água que passa por ele é aquecida [1].

Por outro lado, existe o aquecedor solar que é um sistema de placas coletoras responsáveis pela absorção da radiação solar. O calor do sol, captado pelas placas do aquecedor solar, é transferido para a água que circula no interior de suas tubulações de metal. O reservatório térmico é um recipiente para armazenamento da água aquecida. São cilindros de metais isolados termicamente, dessa forma, a água é conservada aquecida [2].

Os chuveiros em geral tem problemas para o ajuste da temperatura da água que requer um certo tempo e consumo de água.

II. METODOLOGIA

O projeto contém 2 subsistemas, um sistema de controle de temperatura, um sistema de conversão de fala para texto. Através da conversão da fala em texto é gerado um comando que pode ser interpretado de duas formas: ligar ou desligar o sistema e informar a temperatura desejada.

A. Sistema de controle de temperatura

O sistema de controle de temperatura adotado será de malha fechada do tipo PID, controle proporcional, integral e derivativo como mostra a figura 1. O controle integral melhora a resposta em regime permanente ao acrescentar um polo enquanto que o controle derivativo melhora a resposta em regime transitório ao acrescentar um zero, já o controlador

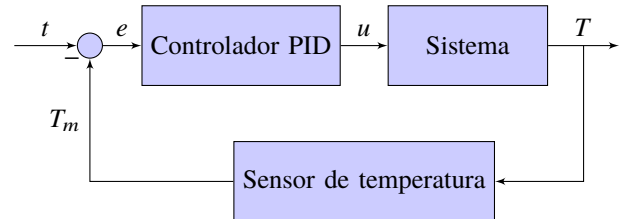


Figura 1. Controle PID a ser usado no sistema. Onde t é a temperatura interpretada pela raspberry pi, e o erro associado a diferença de temperatura entre o que o sensor mede e a solicitada pelo usuário, u a variável de controle no transdutor, T_m a temperatura lida pelo sensor e T é a temperatura da água.

proporcional é responsável por alimentar o erro, diferença entre o valor de entrada e o valor lido pelo sensor de temperatura, adiante por um fator de escala [3].

Será utilizado um relé para o acionamento da resistência do chuveiro. O PWM terá a função de ligar e desligar o relé em uma frequência muito baixa, já que se a transição de níveis forem muito rápidas, acaba que não há tempo para a resistência esquentar. O tempo de cada nível lógico será controlado pelo PID.

B. API de conversão de fala em texto

O "Julius" é um software decodificador de reconhecimento contínuo de fala de alto desempenho e tamanho reduzido para pesquisadores e desenvolvedores relacionados à fala. Com base na palavra N-gram e no HMM dependente do contexto, ele pode executar decodificação em tempo real em vários computadores e dispositivos, do microcomputador ao servidor em nuvem. O algoritmo é baseado na pesquisa em treliça em árvore de 2 passagens, que incorpora totalmente as principais técnicas de decodificação, como o léxico organizado em árvore, a aproximação do contexto de um par de palavras, pode classificar, fatorar N-grama, contexto de palavras cruzadas manipulação de dependência, busca por feixe envelopado, poda gaussiana, seleção gaussiana etc.. A validação e verificação da API Julius foi feita a partir do uso do projeto coruja produzido e mantido pelo grupo de pesquisa FalaBrasil da Universidade Federal do Paraná (Ufpa), são 9 arquivos, o modelo acústico LAPSAM 1.7, mantido pelo mesmo grupo de pesquisa, contendo 3 binários, 1 arquivo de configuração do Julius, contendo taxa de amostragem, modo de saída, entrada e

