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 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].

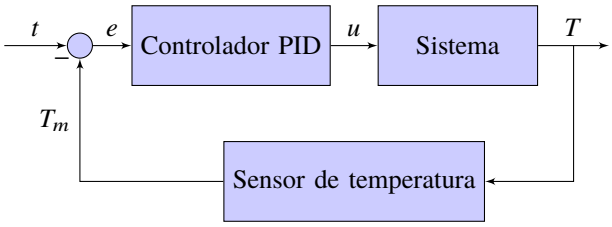

Figura 1. Controle PID a ser usado no sistema.Onde t é a temperatura interpretada pela raspberry pi, e o erro associado a diferença de temperatura entre o que o sensor mede e a solicitada pelo usuário, u a variável de controle no transdutor, T_m a temperatura lida pelo sensor e T é a temperatura da água.

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 logico 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

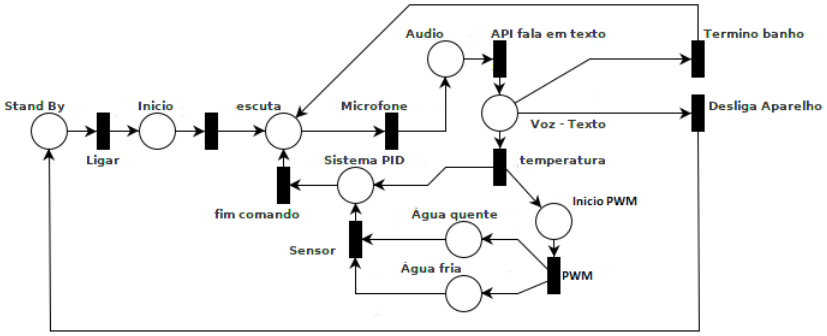

Figura 2. Rede de Petri demonstrando a sequencia de ações e os principais processos.

etc. mais 4 arquivos contendo a gramática, o dicionário, alguns comandos e uma amostra da gramática. Para configuração o primeiro passo é gerar o arquivo .voca, nele, grupos de palavras são associadas da seguinte forma:

%mnemônico
palavra1 fonemas
palavra2 fonemas
palavra3 fonemas

Cada mnemônico representa um conjunto de palavras parecidas, como, por exemplo, a palavra aquecer, que pode ter variações como: aquecimento, aquece e etc. Formados os mnemônicos no arquivo .voca é necessário introduzir sentenças, ou orações, que contenham uma sequência destes mnemônicos, todo grupo de palavras no arquivo .voca deve fazer parte de ao menos uma sentença. Este arquivo recebe a extensão de .grammar e a partir do conjunto destes dois elementos mais as ferramentas *mkdfa.pl* e *generate*, obtidas na instalação do Julius, gera-se, então, o dicionário para reconhecimento da fala. Após realizado este processo, foi necessário remover alguns caracteres para formatação do dicionário: 1 dígito numérico mais tabulação no início de cada linha e o conjunto de colchetes que encorpora a primeira *string* de cada linha. Para isso foi utilizado o *Sublime Text* com o auxílio de expressão regex para substituição, figura 3:

```
^\d{1,2}\t[[].
```

Figura 3. Expressão regex utilizada para substituir carcteres no arquivo .dict.

Logo após, foi feita a remoção do colchete remanescentes pela expresão da figura 4:

```
[[].
```

Figura 4. Expressão regex utilizada para substituir carcteres no arquivo .dict.

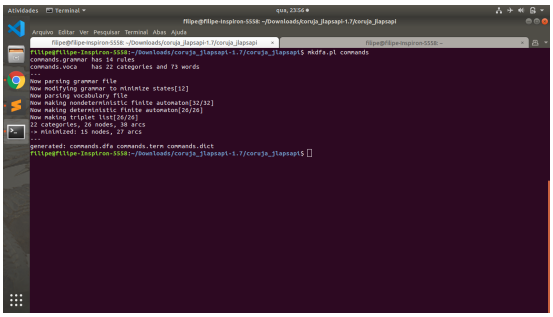

Figura 5. Figura acerca do dicionario gerado

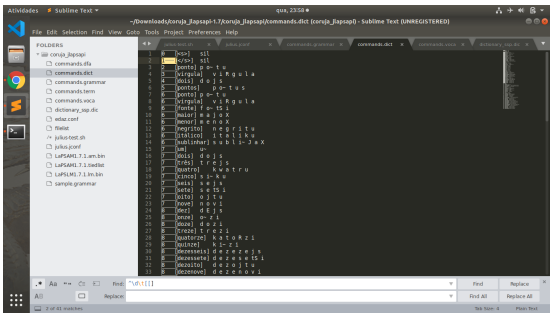

Figura 6. Figura acerca da expressão regex da figura 3.

De acordo co documentação oficial no *GitHub* [4], o jack de áudio serve apenas como *output*, não há hardware nativo para processar a entrada de áudio, por tanto, faz necessário a utilização de um microfone com conexão USB. Para testes foi descrito um shell script simples que faz a gravação do áudio em um computador, através do *framework* alsa, gerando um arquivo de saída que é armazenado na pasta temporária do computador */tmp/* e então enviado por ssh ao respectivo diretório, */tmp/*, onde a raspberry executa o reconhecimento de fala através do Julius. O arquivo de saída gerado com as possíveis sentenças é então enviado de volta ao computador para validação entre o que foi dito e o que foi reconhecido.

C. Sistema completo

A rede de Petri da figura 2 representa o processo de forma geral, com o começo quando o equipamento é ligado e entra

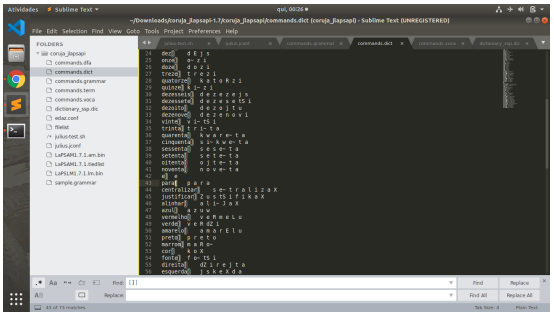

Figura 7. Figura acerca da expressão regex da figura 4.

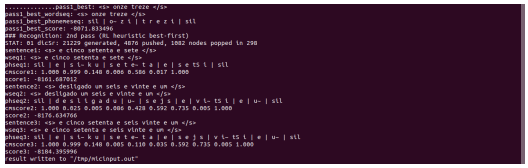

Figura 8. Caption

em modo de espera para um comando. Quando o áudio é processado e indica uma temperatura o sistema de controle PID é acionado com o parâmetro solicitado, quando o comando interpretado refere-se ao controle da resistência para controlar a temperatura e o equipamento entra em modo *standby*. A saída do sistema de reconhecimento está representada na figura 8.

D. Sensor de temperatura

São dispositivos que ajudam a identificar o nível de aquecimento de um determinado equipamento ou processo.

Em virtude dos infinitos tipos de aplicações existentes, é fácil perceber a necessidade de uma vasta gama de sensores, os quais, de maneira individual, cumprem requisitos particulares de cada situação.

Em geral, sensores de temperaturas são analógicos, mas o sensor DS18B20 é capaz de ler a temperatura, interpreta-la e enviar a informação do valor de temperatura em graus Celsius para o microcontrolador usando um barramento de apenar um fio.

O sensor de temperatura DS18B20 pode medir temperaturas entre -55 ºC e 125 ºC com uma precisão de cerca de 0,5 ºC na faixa de -10 ºC e +85 ºC

III. Resultados

Os primeiros testes consistiram em um conjunto de palavras que se referem a aquecer, esfriar, ligar ou desligar o equipamento e conjunções. A imagem 8 apresenta a saída obtida para a frase "Aquece a 30 gruas".

Referências

[1] "Como funciona um chuveiro elétrico?" Mundo da Elétrica. [Online]. Available: www.mundodaeletrica.com.br/como-funciona-um-chuveiro-eletrico/

[2] "Como funciona o aquecedor solar de Água soletrol," Soletrol Indústria e Comércio LTDA. [Online]. Available: www.soletrol.com.br/extras/como-funciona-o-aquecedor-solar-soletrol/

[3] N. S. Nise, *Engenharia de sistemas de controle*. Rio de Janeiro: LTC — Livros Técnicos e Científicos, 2013.

[4] "Raspi git," raspberrypi.org. [Online]. Available: https://github.com/raspberrypi/documentation/tree/master/setup