**CENTRO PAULA SOUZA**

**RUBENS DE FARIA E SOUZA**

**Eletrônica**

**Antonio Zanini de Farias**

**Caio César Corrá**

**Gustavo Teixeira Morais**

**Hebert Prestes Lima**

**CONTROLE DE ALIMENTAÇÃO ANIMAL**

**SOROCABA**

**2020**

**Antonio Zanini de Farias**

**Caio César Corrá**

**Gustavo Teixeira Morais**

**Hebert Prestes Lima**

**CONTROLE DE ALIMENTAÇÃO ANIMAL**

Trabalho de Conclusão de Curso apresentado ao Curso Técnico em Eletrônica da Etec Rubens de Faria e Souza, orientado pelo Prof. Magoga, como requisito parcial para obtenção do título de técnico em Eletrônica.

**SOROCABA**

**2020**

BANCA DE AVALIAÇÃO

Examinador 1: Nome: Assinatura:

Titulação:

Examinador 2: Nome: Assinatura:

Titulação:

Examinador 3: Nome: Assinatura:

Titulação:

Data: Menção:

RESUMO

O projeto consiste no desenvolvimento de um dispositivo capaz de automatizar a alimentação animal controlando a frequência de abastecimento, que é configurada no mesmo.

Utilizando uma placa de prototipagem da plataforma Arduino para controlar as funções do equipamento.

Para o transporte da ração pela passagem entre o container e o pote de ração foi empregado um motor.

ABSTRACT

This project consist in the development about a device capable of automate an animal feeding by controlling the supply frequency.

Resorting to an Arduino single-board microcontroller to control the equipment features.

A DC motor has the function of transporting the animal food on container down to the food pot.

SUMÁRIO

[1 INTRODUÇÃO 7](#_Toc52829041)

[2 DESENVOLVIMENTO 8](#_Toc52829042)

[2.1 Revisão Bibliográfica 8](#_Toc52829043)

[2.1.1 IOT 8](#_Toc52829044)

[2.1.2 WI-FI 8](#_Toc52829045)

[2.1.3 MQTT 8](#_Toc52829046)

[2.1.3.1 Características 9](#_Toc52829047)

[2.1.3.2 Funcionamento do MQTT 9](#_Toc52829048)

[2.1.4 NodeRED 10](#_Toc52829049)

[2.1.5 I2C 10](#_Toc52829050)

[2.2 Metodologia 10](#_Toc52829051)

[2.3 Projeto 10](#_Toc52829052)

[2.3.1 Hardware 11](#_Toc52829053)

[2.3.1.1 Abastecimento 11](#_Toc52829054)

[2.3.1.2 Sensores e Atuadores Secundários 11](#_Toc52829055)

[2.3.1.2.1 Sensores Abastecimento 11](#_Toc52829056)

[2.3.1.2.1.1 Características: 12](#_Toc52829057)

[2.3.1.2.2 Display informativo 12](#_Toc52829058)

[2.3.1.2.2.1 Características: 12](#_Toc52829059)

[2.3.1.2.3 Módulo I2C 12](#_Toc52829060)

[2.3.1.2.3.1 Características: 13](#_Toc52829061)

[2.3.1.2.4 Buzzer Sinalizador 13](#_Toc52829062)

[2.3.1.2.4.1 Características: 13](#_Toc52829063)

[2.3.2 Software 13](#_Toc52829064)

[2.3.2.1 Linguagem C (Arduino) 14](#_Toc52829065)

[2.3.2.1.1 WI-FI 14](#_Toc52829066)

[2.3.2.1.2 MQTT 14](#_Toc52829067)

[2.3.2.2 NodeRed 15](#_Toc52829068)

[3 CONCLUSÃO 16](#_Toc52829069)

[REFERÊNCIAS 17](#_Toc52829070)

[ANEXO 1 – Programa em linguagem C (Arduino) 18](#_Toc52829071)

1. INTRODUÇÃO

Este trabalho visa experimentar e solidificar conceitos de eletrônica aplicada a automatização de tarefas, circuitos eletrônicos são a base de todos os dispositivos de automação modernos e ao projetar uma nova solução se torna natural o estudo deles.

Este projeto desenvolve o uso de uma placa de prototipagem, que manipula os sinais elétricos de entradas para acionar um motor que abastecerá, de forma programável, o recipiente de um animal.

Uma placa microcontroladora baseada em arduino será responsável por controlar a alimentação, acionando o motor que impulsiona a ração no reservatório em direção à saída alimento. O intervalo de tempo em que este processo ocorre é definido pelas configurações inseridas no painel de entrada e armazenadas na memória do arduino, e utilizará um módulo RTC para o controle de horário. Quando o reservatório estiver em desabastecido, um LED de alerta será ativado, e uma mensagem será exibida no display do painel. O dispositivo também contará com uma interface wireless para a entrada de configurações, sendo disponíveis através da conexão do dispositivo a uma rede e a utilização de um navegador para a exibição e edição de informações

1. DESENVOLVIMENTO

t.

* 1. Revisão Bibliográfica

t.

* + 1. IOT

t.

* + 1. WI-FI

Wi-Fi é uma tecnologia de comunicação de rede sem fio baseada em transmissões de radiofrequência na faixa entre 2.4Ghz e 5Ghz.

Os padrões da família IEEE 802.11 definem os protocolos que envolvem a conexão entre computadores, dispositivos móveis e outros equipamentos. Cada padrão é estabelecido e ratificado, sucedendo o padrão vigente e estabelecendo uma versão com mais funcionalidades.

* + 1. MQTT

Este protocolo foi criado pela empresa IBM, no final dos anos 90, como o objetivo vincular suas estruturas de sensores em oleodutos prolíferos à rede de satélites. Como o desenvolvimento do IoT, o protocolo MQTT se tornou o protocolo mais utilizado na comunicação entre pequenos dispositivos.

* + - 1. Características

O MQTT possui a seguintes características que o qualificam como uma excelente opção de comunicação entre sistemas embarcados:

* Protocolo leve: Este protocolo foi desenvolvido para suportar bom desempenho hardware com capacidades limitadas, larguras de banda pequenas e transmissões de alta latência;
* Flexibilidade: O modo de transmissão e a organização de tópicos confere uma alta maleabilidade sobre qual a informação será trafegada o como ela o será;
* Suporte à Segurança: Apesar de leve e flexível, o MQTT ainda é capaz de fornecer funcionalidades de segurança como autenticação, restrição a conteúdos e suporte a conexões seguras (SSL/TSL).
  + - 1. Funcionamento do MQTT

O MQTT funcionará através da rede sem fio, sendo executado em cima do protocolo TCP/IP. Em sua organização, este protocolo possui duas entidades (broker e cliente), dois papéis (assinante e publicador) e uma estrutura de organização de mensagens (tópicos).

O **broker**, é um centralizador e redistribuidor de mensagens, mantendo registro dos destinatários das mensagens e redirecionando as mensagens tão logo elas são recebidas. Apenas um dispositivo pode ser o broker de uma conexão MQTT.

O **cliente** é qualquer dispositivo que mande ou receba mensagens através do broker. Quando está enviando, o cliente está desempenhando o papel de **publicador**, onde ele enviar uma mensagem ao broker para ser aplicada em um determinado tópico de mensagens. Ao receber mensagens, o cliente está no papel do **assinante**, onde ele define, geralmente no momento do estabelecimento de sua conexão MQTT, os tópicos de mensagens aos quais as publicações devem ser redirecionadas a ele. Múltiplos dispositivos podem ser clientes, publicadores e assinantes em uma conexão MQTT, além disso, o cliente pode desempenhar os papeis de Publisher e Subscriber simultaneamente.

Os **tópicos** são uma estrutura de armazenamento de mensagens primariamente temporário, para reter a mensagem apenas durante o período entre o broker receber a mensagem do publicador e o momento em que terminar a entrega para o último assinante de seu tópico.

* + 1. NodeRED

t.

* + 1. I2C

t.

* 1. Metodologia

t.

* 1. Projeto

t.

* + 1. Hardware

t.

* + - 1. Abastecimento

Para transportar a ração do armazenador para a tigela do pet foi empregado uma rosca helicoidal que, ao rotacionar em seu eixo, movimenta a ração até a saída, estrategicamente posicionada.

O giro da rosca helicoidal é proporcionado por um motor DC 12 V, corrente de até 0,07 A, com caixa de redução 1:10, 200 RPM e torque de 2 kgf.cm. O motor entra em ação quando o Arduino, no momento certo, envia um sinal de 5 V, que energiza a bobina de um módulo relé de 1 Canal 5 V, com capacidade de 30 V DC e 10A ou 250V AC e 10A, chaveando o contato NA com 12 V <>

t.

* + - 1. Sensores e Atuadores Secundários
         1. Sensores Abastecimento

Utilizaremos dois Sensores Capacitivos de Proximidade que serão responsáveis por nos informar a quantidade de mantimentos contidos no reservatório no qual se localiza o alimento do animal desejado. Um dos sensores ficará no meio do reservatório e o outro próximo ao fundo.

Características:

Sensor Capacitivo de Proximidade do tipo PNP é um dispositivo eletrônico capaz de detectar a aproximação / presença de objetos metálicos (ferro, aço, alumínio e outros) e objetos não metálicos.

* + - * 1. Display informativo

Um Display LCD 16x2 será o responsável por nos informar as situações presentes no reservatório referente ao estado dos sensores.

Características:

Display LCD 16×2 Backlight Verde possui 2 linhas com 16 colunas e controlador HD44780.

* + - * 1. Módulo I2C

Um Circuito Integrado responsável pela comunicação entre a ESP32 e o display de LCD, pois o ESP32 possui um número limitado de portas e não há a necessidade de usar todas.

Características:

O CI PCF8574 é um expansor de portas I2C de 8 bits que permite o controle de até 8 portas utilizando apenas 2 pinos do microcontrolador. Funciona com tensões entre 2.5 e 6V, e os pinos A0, A1 e A2 definem o endereço utilizado para comunicação I2C. As 8 portas podem ser configuradas como entrada ou saída e os pinos da interface I2C (SDA e SCL) necessitam de resistores pull-up.

* + - * 1. Buzzer Sinalizador

Um Buzzer Passivo será acionado quando o reservatório estiver quase vazio ou seja, quando ambos os sensores estiverem enviando sinal baixo para o ESP32. Irá emitir sons sincronizados em forma de uma música.

Características:

O buzzer 5V passivo permite que você tenha mais controle sobre a melodia, já que as formas de onda dos sons são enviadas diretamente para o pino de I/O, resultando em um som mais limpo.

* + 1. Software

t.

* + - 1. Linguagem C (Arduino)
         1. WI-FI

Para a utilização de WI-FI no ESP32 utilizamos a biblioteca WiFi.h, ela traz funcionalidades que permitem o manuseio simplificado de uma conexão de rede sem fio.

Na implementação se faz necessária a definição do identificador da rede sem fio (SSID) e da sua respectiva senha de acesso, tendo estas informações um comando de inicialização da conexão WI-FI deve ser executado até a conexão seja estabelecida.

Uma vez tendo a conexão estabelecida, o ESP32 terá que verificar periodicamente o estado da conexão para se informar se a conexão ainda está ativa ou foi perdida e, nesse caso, reiniciar o processo de reconexão.

* + - * 1. MQTT

Como o protocolo principal de comunicação entre os clientes WI-FI, o projeto utilizará o protocolo MQTT.

Utilizamos um broker Eclipse Mosquitto com configurações iniciais instalado em rede local, como um projeto conceitual esta configuração será efetiva para os propósitos desejados. Entretanto, o broker poderia ser utilizado em um serviço hospedado em algum servidor web, adicionando a possibilidade de utilizar as funcionalidades de o projeto de forma remota através da internet.

Na utilização do MQTT pelo ESP32 precisamos da conexão WI-FI, previamente definida, assim como uma biblioteca para fornecer-nos o conjunto de comandos para sua manipulação, nesse caso, utilizaremos a biblioteca PubSubClient.h.

Para estabelecer a conexão MQTT, é necessário o endereço IP (ou web) do computador do broker e a porta configurada para o MQTT. Após realizada a conexão, deve-se configurar nesta o método de comunicação (callback), assim o ESP32 poderá receber as mensagens publicadas nos tópicos aos quais assinar.

Assim como uma conexão WI-FI, os comandos da conexão MQTT devem ser iterados até a conexão seja efetivamente estabelecida e deve ser monitorada periodicamente para tratar possíveis falhas de conexão.

Sempre após realizar a conexão (ou reconexão) o ESP32 deverá assinar os tópicos aos quais deseja receber as mensagens do broker, esses tópicos seriam: “alimentador/horariosAlimentacao” e “alimentador/tempoAlimentacao”.

Nos casos de publicação, o ESP32 apenas terá de enviar para o broker uma mensagem contendo o tópico no qual ele deseja publicar e o conteúdo publicado.

* + - 1. NodeRed

t.

1. CONCLUSÃO

t.

REFERÊNCIAS

BERTOLETI, Pedro. ESP32 e MQTT DASH: controle e monitoramento através de um dashboard MQTT para Android. **Filipeflop**, 2019. Disponível em: https://www.filipeflop.com/blog/esp32-e-mqtt-dashboard-android/. Acesso em: 05 out. 2020.

ESP32 Series Datasheet. **Espressif**, 2020. Disponível em: https://www.espressif.com/sites/default/files/documentation/esp32\_datasheet\_en.pdf. Acesso em: 05 out. 2020.

LIMA, Izabelle. Aprenda a Reproduzir Músicas com o Módulo Buzzer. **Autocorerobotica**, 2018. Disponível em: https://autocorerobotica.blog.br/aprenda-a-reproduzir-musicas-com-o-modulo-buzzer-2/. Acesso em: 05 out. 2020.

O'LEARY Nick. ARDUINO Client for MQTT. **Knolleary**, 2020. Disponível em: https://pubsubclient.knolleary.net. Acesso em: 05 out. 2020.

YUAN, Michael, Conhecendo o MQTT. **IBM**, 2017. Disponível em: https://developer.ibm.com/br/articles/iot-mqtt-why-good-for-iot/. Acesso em: 05 out. 2020.

WIFI library. **Arduino**, 2019. Disponível em: https://www.arduino.cc/en/Reference/WiFi. Acesso em: 05 out. 2020.

ANEXO 1 – Programa em linguagem C (Arduino)