



**Centro de Tecnologia
Curso de Engenharia de Computação**

Mairon Piber Klimeck

SISTEMA AUTOMATIZADO PARA AQUÁRIO

Trabalho de Conclusão de Curso

Santa Maria – RS, Brasil

2018



MAIRON PIBER KLIMECK

SISTEMA AUTOMATIZADO PARA AQUÁRIO

Trabalho de Conclusão de Curso (TCC) apresentado ao Curso de Graduação em Engenharia de Computação da Universidade Federal de Santa Maria (UFSM) como requisito parcial para obtenção do grau de Engenheiro de Computação.

Orientador: Prof. Dr. José Eduardo Baggio

Santa Maria – RS, Brasil
2018



CURSO DE ENGENHARIA DE COMPUTAÇÃO

A Banca Examinadora, abaixo assinada, aprova o Trabalho de Conclusão de Curso

SISTEMA AUTOMATIZADO PARA AQUÁRIO

Elaborado por

Mairon Piber Klimeck

Como requisito parcial para obtenção do grau de Engenheiro de Computação.

Banca Examinadora:

Prof. Dr. José Eduardo Baggio – UFSM

Prof. Dr. Fábio Ecke Bisogno – UFSM

Prof. Dr. Jorge Rodrigo Massing – UFSM

Santa Maria - RS, Brasil.

2018

DEDICATÓRIA

*Dedico este trabalho aos meus pais Josemar e Leonice,
e à minha namorada Luana.*

“O navio está em segurança no porto, mas não é para isso que servem os navios.”

(C.S Lewis)

RESUMO

SISTEMA AUTOMATIZADO PARA AQUÁRIO

Autor: Mairon Piber Klimeck

Orientador: Prof. Dr. José Eduardo Baggio

O presente trabalho apresenta um sistema automatizado para aquários denominado Aquadroit, projetado especificamente para controle de aquários residenciais, utilizando o conceito de Internet das Coisas (IoT). O objetivo principal do sistema é controlar automaticamente e monitorar alguns fatores que são cruciais para o bem-estar de um habitat aquático dentro de um aquário. Este trabalho foi planejado e desenvolvido, levando-se em consideração que o aquarismo é uma das paixões cada vez mais aclamadas pelo mundo. O sistema é composto basicamente por um microcontrolador conectado à rede wi-fi, e alguns sensores e atuadores. Com o sistema desenvolvido, é possível obter o controle remoto dos principais fatores que envolvem um aquário, assim como fazer o controle de maneira manual ou programada de alguns parâmetros do aquário, utilizando o aplicativo específico para smartphones/tablets, ou via browsers em computadores e afins.

Palavras-chave: Sistema para Aquário, Internet das Coisas, Microcontrolador, IoT.

ABSTRACT

AUTOMATED SYSTEM FOR AQUARIUMS

Author: Mairon Piber Klimeck

Advisor: Prof. Dr. José Eduardo Baggio

The present work presents an automated for aquariums system called Aquadroit, specifically designed for the control of residential aquariums, using the Internet of Things (IoT) concept. The main purpose of the system is to automatically control and monitor some factors that are crucial to the well-being of an aquatic habitat within an aquarium. This work was planned and developed, taking into account that aquarism is one of the most acclaimed passions in the world. The system consists basically of a microcontroller connected to the wi-fi network, and some sensors and actuators. With the system developed, it is possible to obtain remote control of the main factors that involve an aquarium, as well as manually or programmed control of some parameters of the aquarium, using the specific application for smartphones / tablets, or via browsers in computers and related.

Keywords: System for Aquarius, Internet of Things, Microcontroller, IoT.

Figura 1 - A primeira torradeira conectada à internet.	13
Figura 2 – Protocolo MQTT.	15
Figura 3 – Locais de execução do broker MQTT.	16
Figura 4 – Área de programação no Node-RED.	16
Figura 5 - NodeMCU ESP-12E.	18
Figura 6 - Arquitetura de um microprocessador básico.	18
Figura 7 - Módulos com ESP8266.	20
Figura 8 - Tamanho do Módulo ESP-12E.	20
Figura 9 - Pinagem e esquemático da ESP-12E.	21
Figura 10 - Sensor de PH-4502C.	23
Figura 11 - Sensor DS18B20.	24
Figura 12 - Diagrama de blocos do sensor DS18B20.	24
Figura 13 - Aquecedor Heater-100W.	25
Figura 14 – Formas de resfriamento.	26
Figura 15 – Cooler para Refrigeração.	27
Figura 16 - Filtro Externo.	28
Figura 17 - Compressor de Ar.	29
Figura 18 - Motor de Toca Fitas.	29
Figura 19 – Vista inferior do Alimentador.	30
Figura 20 - Vista superior do Alimentador.	31
Figura 21 - Esquemático de funcionamento de um Relé.	32
Figura 22 - Módulo Relé de 4 Canais.	33
Figura 23 - Módulo RTC DS3231.	34
Figura 24 - Fita LED 3528 SMD.	34
Figura 25 - Configurações do transistor.	36
Figura 26 - TIP120.	37
Figura 27 - Diagrama em blocos da fonte de alimentação chaveada.	37
Figura 28 - Fonte 12 volts.	38
Figura 29 - Sensor de Nível P45.	38
Figura 30 - Display LCD 16x2.	39
Figura 31 – Pinagem MC74HC595.	40
Figura 32 - Disposição dos pinos no Microcontrolador.	41
Figura 33 - Dispositivos ligados à ESP-12E.	42
Figura 34 - Código utilizado para mensurar o pH.	43
Figura 35 - Divisor resistivo.	44
Figura 36 - Equação Divisor Resistivo.	44
Figura 37 - Código utilizado para mensurar a temperatura.	45
Figura 38 - Código utilizado para atualizar o horário.	45
Figura 39 - Código verificar se a fonte está sendo ocupada.	46
Figura 40 - Código utilizado para mensurar a temperatura.	48
Figura 41 – Tela do broker responsável por ajustar valores de temperatura.	48
Figura 42 – Lógica para ligar/desligar o aquecedor.	49
Figura 43 - Lógica para ligar/desligar o resfriador.	49
Figura 44 – Tela do broker responsável por manipular valores da alimentação.	50
Figura 45 - Código utilizado realizar a alimentação automática.	51
Figura 46 - Código utilizado realizar a alimentação manual.	52
Figura 47 – Parte do código de interpretação do tópico comando.	53
Figura 48 – Tela do broker responsável por controlar a iluminação.	54
Figura 49 – Tela de programação do broker responsável por controlar a iluminação.	55
Figura 50 – Código responsável pela leitura do nível de água.	56
Figura 51 – Maneira para exportar um código.	56
Figura 52 – Maneira para importar um código.	57
Figura 53 – Menu de navegação disponibilizado do lado superior esquerdo.	58
Figura 54 – Tela de Início no broker.	58
Figura 55 – Tela Central de Controle no broker parte 1.	59

Figura 56 – Tela Central de Controle no broker parte 2.	60
Figura 57 – Tela Gráficos no broker parte 1.	61
Figura 58 – Tela Gráficos no broker parte 2.	62
Figura 59 – Aplicativo MQTT Dash.	63
Figura 60 – Captura de tela 1 MQTT Dash.	63
Figura 61 – Captura de tela 2 MQTT Dash.	64
Figura 62 – Função de rotina do microprocessador.	66
Figura 63 – Configuração do Telegram no Node-RED.	68
Figura 64 – Mensagem alerta no telegram.	69
Figura 65 – Interface MQTT Dash.	70
Figura 66 – Captura de tela MQTT Dash.	71
Figura 67 – Vista Superior Aquadroit.	72
Figura 68 – Vista traseira Aquadroit.	72
Figura 69 – Vista frontal Aquadroit.	73

SUMÁRIO

1	INTRODUÇÃO	12
2	REVISÃO LITERÁRIA	13
2.1	INTERNET DAS COISAS(IoT).....	13
2.2	PROTOCOLO MQTT.....	14
2.3	NODE-RED.....	16
2.4	MICROCONTROLADOR.....	17
2.5	NODEMCU ESP8266 ESP-12E	19
2.6	SENSOR DE pH	22
2.7	SENSOR DE TEMPERATURA	23
2.8	AQUECEDOR	25
2.9	RESFRIAMENTO	26
2.10	FILTRO MECÂNICO EXTERNO	28
2.11	COMPRESSOR DE AR.....	28
2.12	ALIMENTADOR AUTOMÁTICO	29
2.13	MÓDULO RELÉ	32
2.14	MÓDULO RTC DS3231.....	33
2.15	FITA LED 3528 SMD.....	34
2.16	TRANSISTOR	35
	2.16.1 Transistor NPN	36
2.17	FONTE DE ALIMENTAÇÃO.....	37
2.18	NÍVEL DE ÁGUA	38
2.19	DISPLAY LCD	39
2.20	REGISTRADOR DE DESLOCAMENTO MC74HC595	40
3	DESENVOLVIMENTO	41
3.1	DISPOSITIVOS LIGADOS À ESP-12E	41
3.2	CALIBRAGEM DO SENSOR DE pH	43
3.3	INSTALAÇÃO DO DISPLAY E MÓDULO RTC	44
3.4	FONTE 12V	46
3.5	MÓDULO RELÉ	47
3.6	CONFIGURAÇÃO DO SENSOR DE TEMPERATURA	48
3.7	ALIMENTADOR AUTOMÁTICO	50
3.8	ILUMINAÇÃO	53
3.9	NÍVEL DE ÁGUA	55

3.10	Node-RED e BROKER MQTT.....	56
3.11	APLICATIVO MQTT DASH.....	63
4	RESULTADOS.....	66
4.1	COMUNICAÇÃO ENTRE A ESP-12E E BROKER MQTT.....	66
4.2	COMUNICAÇÃO DO BROKER COM O TELEGRAM	68
4.3	INTERFACE DO APLICATIVO MQTT DASH	70
4.4	PROJETO FISICO AQUADROIT.....	71
5	CONCLUSÃO	74
	REFERÊNCIAS.....	75

1 INTRODUÇÃO

Estamos vivendo uma época em que presenciamos uma constante evolução da tecnologia, e nessa evolução o conceito de internet das coisas (IoT) está cada vez mais presente nos dispositivos atuais. Este conceito se faz presente desde equipamentos eletrodomésticos até mesmo em roupas conectadas à internet, a computadores e a outros dispositivos móveis como smartphones.

IoT se conceitua em uma rede que interconecta equipamentos pela Internet. É o conceito de integrar vários tipos de dispositivos eletrônicos, e os mais diversos sensores, com tudo conectado podemos realizar a comunicação e troca de informação. A partir disso, obtemos uma rede capaz de possuir funções de reconhecimento inteligente, localização, rastreamento e gerenciamento destes dispositivos (AGRAWAL; VIEIRA, 2013).

Atualmente, a maioria das pessoas possuem dentro de suas residências algum animal de estimação. A grande maioria desses animais, ainda são cães e gatos. Entretanto, uma nova espécie de animal doméstico vem ganhando espaço dentro das residências, os peixes. A presença desses seres no ambiente familiar tem se tornado cada vez mais frequente e, conseqüentemente, a preocupação com o bem-estar dos mesmos também.

Tendo em vista essa crescente demanda na criação de peixes, tecnologias têm sido criadas para que a criação domiciliar seja vantajosa para o dono e para o animal. Visando auxiliar na maior sobrevivência dos peixes de aquário, com maior autonomia e dinamização, foi desenvolvido um sistema de automação para um aquário, denominado Aquadroit. O nome é o resultado da fusão de aquário, Android e IoT.

Esse dispositivo, se propõe a utilizar o conceito de Internet das Coisas para desenvolver um sistema que permita ao usuário determinar a temperatura, a iluminação, os horários e a quantidade de ração que será dada aos peixes, além de realizar o monitoramento do pH e nível de água, criando um alerta sempre que esses entrarem em um nível crítico preestabelecido, viabilizando assim maior flexibilidade para o dono em monitorar e controlar tudo em seu smartphone quando ele não estiver em sua residência para fazê-lo.

Para obter os resultados esperados serão estudados os microcontroladores, módulos de conexão com a rede e com o usuário, os sensores e atuadores que podem ser usados em conjunto com esses microcontroladores, e também será estudado os fatores ideais para se obter uma boa qualidade de vida no habitat aquático.

2 REVISÃO LITERÁRIA

2.1 INTERNET DAS COISAS(IoT)

Internet das coisas é o conceito tecnológico em que todos os objetos da vida cotidiana estariam conectados à internet, agindo de modo inteligente e sensorial. Consiste na ideia da fusão do “mundo real” com o “mundo digital”, fazendo com que o indivíduo possa estar em constante comunicação e interação, seja com outras pessoas ou objetos (LIMA, 2017).

Teixeira apresenta uma definição mais técnica sobre a Internet das Coisas, ele dá destaque para os protocolos de comunicação para que seja possível a comunicação entre uma rede de dados, para (TEIXEIRA, 2014, p. 589) as coisas “serão capazes de interagir e comunicar entre elas mesmas, trocar informações coletadas do ambiente, reagindo autonomamente aos eventos do mundo físico real, bem como influenciar esse contexto sem intervenção direta do ser humano”.

O primeiro dispositivo a ser conectado a um computador foi criado por John Romkey em 1990. Ele criou uma torradeira que poderia ser controlada remotamente e apresentou na INTEROP '89 Conference. Dan Lynch, presidente da INTEROP na época, prometeu para John Romkey, que se a torradeira fosse conectada à internet, seu aparelho seria colocado em exposição durante a conferência, mas para isso a torradeira teria que ser ligada e desligada pela internet. John conectou a torradeira a um computador com rede TCP/IP, e foi um sucesso, porém o pão ainda tinha que ser inserido manualmente. Após um ano, e com a ajuda de um pequeno guindaste robótico controlado pela internet, o pão então foi também colocado na torradeira de maneira remota, automatizando assim o sistema por completo. Na Figura 1 temos a ilustração da torradeira de John Romkey.

Figura 1 - A primeira torradeira conectada à internet.



Fonte: <https://www.linkedin.com/pulse/hist%C3%B3ria-da-internet-das-coisas-ou-things-iot-m%C3%B4nica-mancini>

Para o equipamento se adequar no conceito da IoT, é indispensável uma conexão à rede. Essa conexão pode ser cabeada ou sem fio, sendo essa última mais desejada por permitir maior liberdade no uso dos equipamentos.

Segundo (TANENBAUM, 2003) as redes sem fio surgiram em conjunto com os notebooks, as pessoas sonhavam que ao chegar em seus escritórios, suas máquinas conectassem na internet de maneira automática. Para o funcionamento da rede sem fio seria necessário ser definido um padrão para estabelecer esta comunicação, e foi assim que o IEEE criou o padrão 802.11.

Para (MARTINS; COSTA; QUEIROZ, 2013) o padrão IEEE 802.11n, publicado em 2009, foi projetado para garantir um aumento de até seis vezes na quantidade limite de banda dos padrões 802.11g e 802.11a. Além disso, esse padrão foi planejado com o intuito de ser retro compatível com os padrões anteriores, tendo adicionado canais de 40 MHz à camada física e podendo operar nas bandas de 2,4GHz (802.11b/g) e 5GHz (802.11a). Um dos principais avanços veio com a introdução da tecnologia MIMO (Multiple-Input Multiple-Output), uma técnica de processamento de sinais para transmitir múltiplos fluxos de dados através de várias antenas.

Para possibilitar a troca de informações entre o usuário e o controlador é necessário um protocolo de comunicação. Para isso, será usado o MQTT e seu funcionamento será explicado a seguir.

2.2 PROTOCOLO MQTT

O MQTT é um protocolo de ligação que especifica como os bytes de dados são organizados e transmitidos pela rede TCP/IP. Mas por motivos práticos, os desenvolvedores não precisam entender o protocolo de ligação. Basta saber que cada mensagem tem uma carga útil de comando e dados. O comando define o tipo de mensagem (por exemplo, uma mensagem CONNECT ou uma mensagem SUBSCRIBE). Todas as bibliotecas e ferramentas do MQTT oferecem maneiras simples de manipular essas mensagens diretamente e podem preencher automaticamente alguns campos necessários, como os IDs da mensagem e do cliente (www.ibm.com).

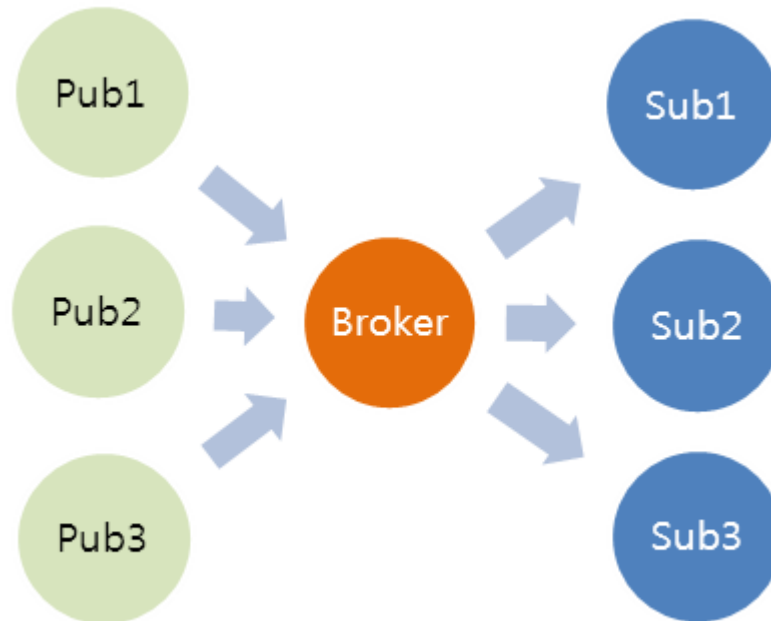
A comunicação MQTT funciona basicamente com 3 divisões:

- Publishers;
- Subscribers;

- Broker.

A figura 2 abaixo ajuda no entendimento dessa divisão:

Figura 2 – Protocolo MQTT.



Fonte: <https://www.filipeflop.com/blog/control-monitoring-iot-nodemcu-e-mqtt/>

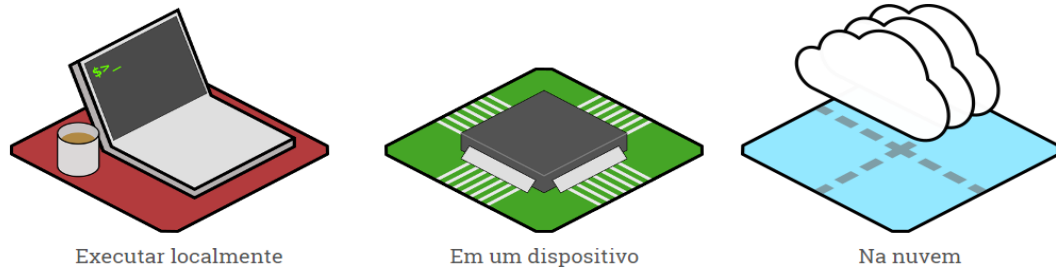
Publishers é quem irá publicar informações no broker, no nosso caso será tanto a ESP-12E quanto os dispositivos conectados, a ESP-12E irá ler os sensores conectados e disponibilizar essas informações no broker, para cada publicação realizada deve-se saber o tópico a ser publicado, para que o broker possa entender o que aquele dado significa e processá-lo devidamente, já os dispositivos conectados iram publicar no broker comandos para acionar a iluminação na ESP-12E por exemplo, que posteriormente irá receber a informação a partir da subscrição no tópico.

Subscribers é quem irá receber informações através de uma subscrição em um ou vários tópicos contidos no broker, pode ser um aplicativo, um computador, outro microprocessador, enfim qualquer dispositivo que seja capaz de interagir com o broker e receber mensagens.

E por fim, existem os Brokers MQTT, que são servidores que recebem todas as mensagens dos clientes e, em seguida, roteiam essas mensagens para os clientes inscritos nos

tópicos, existem em três locais para a execução do broker as quais estão ilustradas na figura 3, quando utilizado na nuvem possibilita acessá-lo de qualquer lugar do planeta através da conexão com a internet.

Figura 3 – Locais de execução do broker MQTT.



Fonte: <https://nodered.org/#get-started>

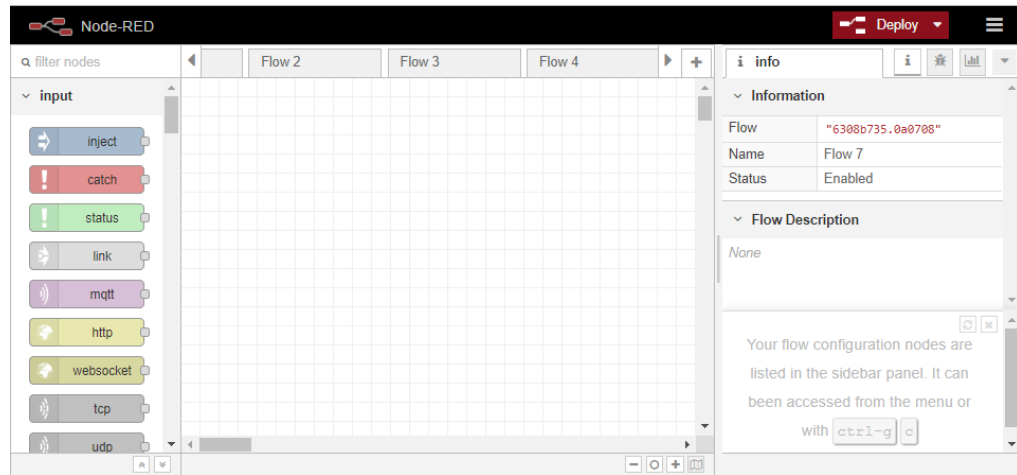
Para o processamento de informações em nosso broker iremos usar o Node-RED, que é uma programação baseada em nós, de simples entendimento e de grande agilidade no processamento de dados.

2.3 NODE-RED

O Node-RED fornece um editor de fluxo baseado em navegador que facilita a conexão de dispositivos, APIs e serviços on-line usando o amplo intervalo de nós na paleta. Os fluxos podem então ser implantados no tempo de execução do Node.js com um único clique (www.nodered.org).

A figura 4 mostra como é a área de programação no Node_RED.

Figura 4 – Área de programação no Node-RED.



Fonte: Arquivo pessoal.

A programação no Node-RED é baseada em nós interligados, gerando assim fluxos. No Node-RED já existe o protocolo para comunicação MQTT pronto, basta instalar o módulo correspondente, depois é só o arrastar para um fluxo desejado e configurar com o endereço e credencias do seu broker, além do tópico que deseja publicar ou se inscrever, tornando assim a programação simples, rápida e robusta.

O Node-RED apenas recebe o dado e o tópico relacionado a esse dado, e a partir desse tópico ele sabe em qual fluxo deve colocar o dado e assim fazer o processamento do mesmo. O fato dele receber apenas o dado o torna um protocolo leve em comparação aos demais.

Já falamos sobre a Internet das coisas e sobre o protocolo de comunicação a ser usado, vamos falar agora sobre o microprocessador a ser usado.

2.4 MICROCONTROLADOR

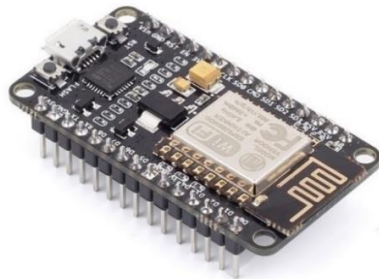
O microcontrolador é um dispositivo semiconductor em forma de circuito integrado, que integra as partes básicas de um microcomputador microprocessador, memórias não-voláteis e voláteis e portas de entrada e saída. Geralmente, é limitado em termos de quantidade de memória, principalmente no que diz respeito à memória de dados, é utilizada em aplicações específicas, ou seja, naquelas que não necessitam armazenar grandes quantidades de dados, como automação residencial, automação predial, automação industrial e automação embarcada. (GIMENEZ, 2005).

O microcontrolador é um circuito único integrado e programável, contemplando um CPU, memória de dados e programa, sistema de clock, portas, além de outros periféricos como conversores A/D entre outros. Pode ser usado para controlar uma grande quantidade de aparelhos, coordenando suas funções e ações. (JUCA, 2010)

Na Figura 5 é mostrado o microcontrolador utilizado no projeto, a NodeMCU ESP8266

ESP-12E.

Figura 5 - NodeMCU ESP-12E.



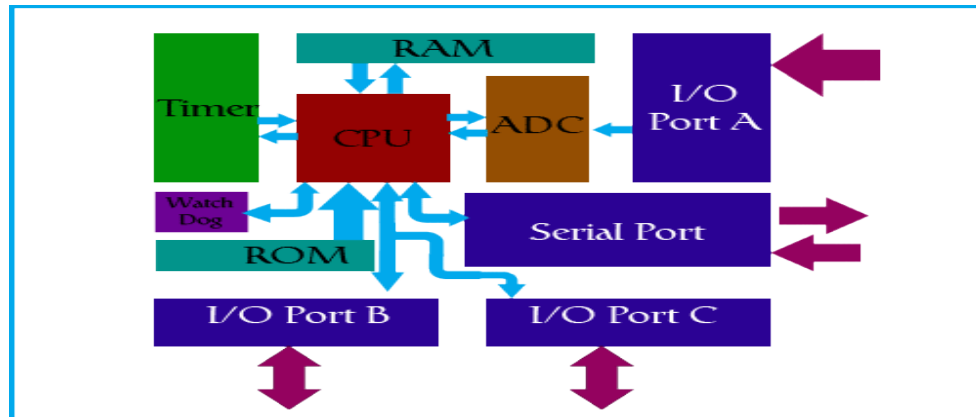
Fonte: <https://www.filipeflop.com/blog/qual-modulo-esp8266-comprar/>

Normalmente, microcontroladores são “embarcados” em algum outro aparelho, executando o controle de ações e respostas. O programa é registrado na memória ROM, fazendo com que o programa não mude depois de gravado na memória, dedicando o microcontrolador à uma tarefa única, rodando o programa específico. Possui um dispositivo de entrada dedicado e normalmente um display de LCD ou LED para saída, facilitando a comunicação com o usuário e a sua aplicação no produto. (PEREIRA, 2003)

Além de receber entradas do usuário, o microprocessador também recebe informações do próprio produto no qual está embarcado, e controla o aparelho mandando sinais para diversos componentes do mesmo. É também, de certa forma, resistente, podendo ser aplicado em condições mais extremas onde computadores normais não conseguiriam. (GROOVER, 2001)

Na Figura 6 é ilustrada a arquitetura que compõe o funcionamento de um microcontrolador básico.

Figura 6 - Arquitetura de um microprocessador básico.



Fonte: <http://www.marteltecnologia.com.br/microcontrolador/>

Um bom exemplo do funcionamento do microcontrolador seria observando um micro-ondas. Além de receber informações das teclas e exibi-las no display, controla o relé que liga ou desliga o funcionamento do aparelho. Pode-se observar a sua resiliência quando aplicado ao controle de motores de carros onde, dependendo da localização e clima, podem funcionar a temperaturas de -34°C até 49°C . Se for considerado o calor gerado pelo próprio motor, as temperaturas podem subir até 80°C , sem que o controlador pare de funcionar ou perca suas capacidades. (GROOVER, 2001)

Como dito anteriormente, nesse projeto usaremos a ESP-12E como nosso microcontrolador.

2.5 NODEMCU ESP8266 ESP-12E

O ESP8266 possui instruções de 32 bits e um núcleo microprocessador Tensilica L106 funcionando com um clock de 80 MHz podendo chegar a até 160 MHz, sendo que o protocolo Wi-Fi usa cerca de 20% desse processamento e o restante fica disponível para as aplicações do usuário. O ESP conta com uma memória interna de cerca de 50 kB para os dados de programas e 4 MB disponíveis para o programa principal, essa parte da memória pode ser acessada em funcionamento para programação pela rede Wi-Fi, conhecido como OTA (Over the Air), e além disso tem mais 512 kB para o programa principal que podem ser acessados apenas por programação via cabo USB/Serial (ESPRESSIF, 2018).

A sua tensão de operação é de 3,3V com um consumo de corrente que pode chegar em até 200 mA quando está em modo de transmissão por Wi-Fi. Em relação às suas interfaces de entrada e saída, o ESP8266 tem 17 interfaces GPIO (*General Purpose Input/Output*) sendo quatro delas são saídas PWM (*Pulse Width Modulation*) e uma entrada analógica com 10 bits de precisão. Como interface de comunicação, tem as seriais síncronas, SPI, I2C, I2S e as assíncronas UART, que podem ser usadas para interligar uma interface RS-232. Além disso conta com a interface Wi-Fi que pode atuar no modo AP, cliente-servidor, *ad-hoc* ou Wi-Fi *Direct* (ESPRESSIF, 2018). Segundo (CURVELLO, A., 2015) existe uma ampla variedade de

módulos com ESP8266, que diferem principalmente pela quantidade de I/Os disponíveis para uso externo e pelo seu tamanho. Os principais modelos podem ser vistos na Figura 7.

Figura 7 - Módulos com ESP8266.

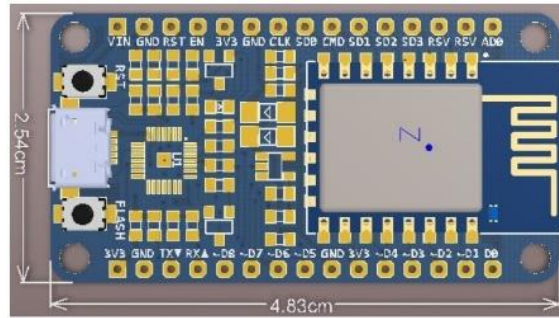


Fonte: (Hackstyle, 2016)

Será dada ênfase apenas ao módulo ESP-12E, que será usado neste trabalho por ser amplo e oferecer o maior número de I/Os.

O módulo ESP-12E, possui um processador capaz de integrar sensores e outras aplicações específicas usando suas portas GPIOs, e transmitir essas informações via *Wifi*. Por ser um chip bastante completo, requer pouquíssimos componentes na placa, o que torna a placa muito pequena ocupando pouco espaço no projeto, como mostrado na figura 8. (SYSTEMS,2016)

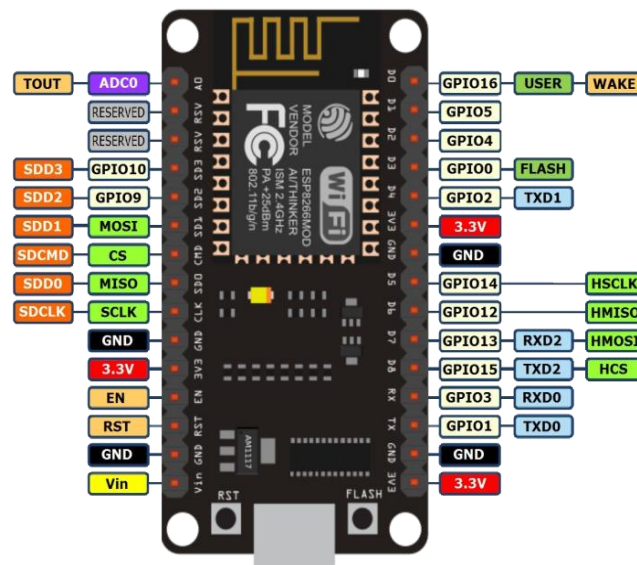
Figura 8 - Tamanho do Módulo ESP-12E.



Fonte: <http://blog.eletrogate.com/nodemcu-esp12-introducao-1/>

Abaixo, na Figura 9 temos uma imagem ilustrativa da ESP-12E detalhando o seu esquemático com a sua pinagem identificada.

Figura 9 - Pinagem e esquemático da ESP-12E.



Fonte: <http://blog.eletrogate.com/nodemcu-esp12-introducao-1/>

Onde, segundo (ESPRESSIF, 2018):

- ❖ **VIN** – Esse é o pino de alimentação externa (5,0V / 1A).
- ❖ **GND**– Pino de terra.
- ❖ **RST** – Pino de reset, é acionado em nível baixo (GND);
- ❖ **EN** – (Enable) ativa o módulo ESP-12 quando o nível for HIGH(3,3V).
- ❖ **3,3V** – Saída do regulador interno 3,3V – Para alimentar outros dispositivos.
- ❖ **CLK** – Interface SPI (clock) – pino SCLK (GPIO_6)

- ❖ **SD0** – Interface SPI (master in serial out) – pino MISO (GPIO_7)
- ❖ **CMD** – Interface SPI (chip select) – pino CS (GPIO_11)
- ❖ **SD1** – Interface SPI (master out serial in) – pino MOSI (GPIO_8)
- ❖ **SD2** – Pino GPIO_9 pode ser usado também para comunicação com SD Card (SDD2)
- ❖ **SD3** – Pino GPIO_10 – pode ser usado também para comunicação com SD Card (SDD3)
- ❖ **RSV** – Reservado (não use).
- ❖ **ADC0** – Pino de entrada do conversor analógico digital ADC de 10 bits. Tensão máxima de 3,3V (variação do valor digital – 0 a 1024).
- ❖ **D0** – Pino GPIO_16 pode ser usado para acordar (WAKE UP) o ESP8266 em modo sono profundo (Deep sleep mode).
- ❖ **D1** – Pino GPIO_5 – entrada ou saída.
- ❖ **D2** – Pino GPIO_4 – entrada ou saída.
- ❖ **D3** – Pino GPIO_0 é usado também para controlar o upload do programa na memória Flash. Está conectado no botão FLASH.
- ❖ **D4** – Pino GPIO_2 – UART_TXD1 quando carregando o programa na memória FLASH
- ❖ **D5** – Pino GPIO_14 pode ser usado em SPI de alta velocidade (HSPI-SCLK)
- ❖ **D6** – Pino GPIO_12 pode ser usado em SPI de alta velocidade (HSPI-MISO)
- ❖ **D7** – Pino GPIO_13 pode ser usado em SPI de alta velocidade (HSPI-MOSI) ou UART0_CTS.
- ❖ **D8** – Pino GPIO_15 pode ser usado em SPI de alta velocidade (HSPI-CS) ou UART0_RTS.
- ❖ **RX** – Pino GPIO_3 – U0RXD quando carregando o programa na memória FLASH.
- ❖ **TX** – Pino GPIO_1 – U0TXD quando carregando o programa na memória FLASH.

Depois de tomarmos conhecimento sobre o microcontrolador usado e o funcionamento de seus pinos, vamos falar sobre os sensores usados, começando pelo sensor de pH.

2.6 SENSOR DE pH

O Medidor de pH é formado por um eletrodo, que é uma espécie de sensor, conectado a um potenciômetro, que converte o valor de potencial do eletrodo em unidades de pH. No momento em que este eletrodo é submerso na amostra que vai ser analisada, como a água, por exemplo, ele produz milivolts que são transformados para a escala de pH. (blog.instrusul.com.br)

A escala padronizada de pH varia de 0 a 14. Quanto mais próximo do zero, maior a concentração de íons de hidrogênio (H⁺) e mais ácida é a solução. Mas se os valores são mais próximos do 14, a amostra é mais alcalina, pois se têm menor concentração de íons H⁺. A solução é considerada neutra quando o valor do pH for 7. (blog.instrusul.com.br)

Quando o pH da amostra estiver em 14, o potenciômetro disponibilizará uma tensão de 5V na saída Po. Quando o pH estiver em 0 terá 0V na saída e quando o pH estiver em 7 terá 2,5V, essa saída será lida pela porta analógica da ESP-12E que interpretará a tensão na porta e irá disponibilizar esse valor na escala de pH para os devidos fins.

Na figura 10 temos ilustrado o sensor de pH que será usado no projeto, o PH-4502C.

Figura 10 - Sensor de PH-4502C.



Fonte : <https://pt.aliexpress.com/item/1Set-Liquid-PH-Value-Detection-Regulator-Sensor-Module-Monitoring-Control-Meter-Tester-BNC-PH-Electrode-Probe/32805675619.html?spm=a2g0s.9042311.0.0.8c7eb90aEdyYpO>

Agora que falamos sobre o sensor que será usado para controlar o pH, vamos falar sobre outro sensor usado, o sensor para controlar a temperatura no aquário.

2.7 SENSOR DE TEMPERATURA

Manter a temperatura em uma faixa adequada conforme a espécie do peixe é um fator essencial para garantir a sobrevivência do mesmo, pois isso influencia tanto na alimentação, quanto na defesa imunológica dos peixes. Outro fator prejudicial decorrente da variação da temperatura é a proliferação de bactérias e algas na água.

Para a medição da temperatura da água, utilizou-se o sensor DS18B20, como mostrado abaixo na figura 11, que fornece de 9 a 12 bits de precisão. Além disso, possui internamente

um conversor A/D, que fornece a temperatura em formato digital por meio do barramento *one-wire* direto para o barramento do microcontrolador. O *one-wire* é um sistema de barramento que tem como característica fornecer dados de baixa velocidade, sinalização e sinal único de energia. Tem apenas dois cabos que são dados e GND e dispõe de um capacitor de 800 pF para armazenar carga e alimentar o dispositivo durante os períodos onde o cabo de dados estiver sendo usado para o tráfego de dados (WINGWIT, 2014).

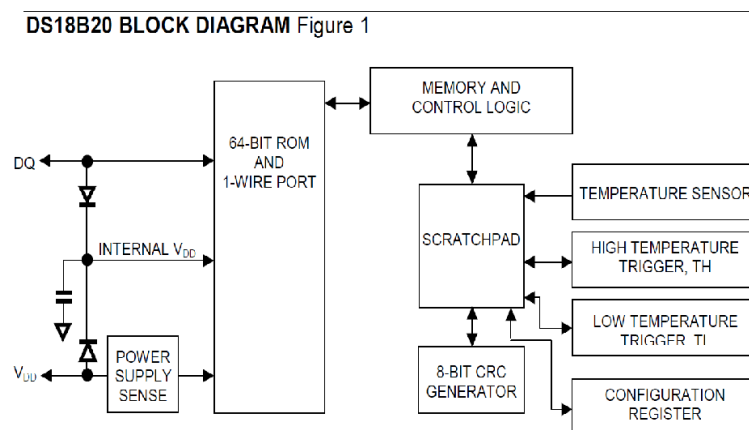
Figura 11 - Sensor DS18B20.



Fonte: <https://www.espruino.com/DS18B20>

O sensor é ligado por 3 fios, sendo dois para alimentação e um para troca de dados. A Figura 12 apresenta o diagrama em blocos do sensor DS18B20.

Figura 12 - Diagrama de blocos do sensor DS18B20.



Fonte: http://ferrarihk.en.seekic.com/product/integrated_circuits_ics/DS18B20.html

Uma vantagem do sensor utilizado é a de não requerer componentes externos. Cada dispositivo tem um único código serial de 64 bits armazenado em uma ROM onboard. O código serial de 64 bits é o ID do sensor, onde cada sensor possui um código único e serve para o endereçamento dos sensores no barramento one-wire. De acordo com informações contidas em seu datasheet ele é capaz de medir temperaturas de -55°C a $+125^{\circ}\text{C}$, possuindo uma precisão de $0,5^{\circ}\text{C}$ para temperaturas compreendidas em -10°C a $+85^{\circ}\text{C}$.

Depois de tornar o conhecimento da temperatura, vamos falar como mantê-la na faixa desejada, começando pelo aquecedor quando essa estiver abaixo da temperatura mínima desejada.

2.8 AQUECEDOR

Para fazer o aquecimento da água foi usado o aquecedor da Heater com potência máxima de 100 Watts, ilustrado na figura 13, o qual ficará submerso na água.

Este aquecedor é capaz de aquecer um aquário com capacidade de até 100 litros, sendo adequado ao presente trabalho, onde é utilizado um aquário de 67 litros. Quando solicitado, o aquecedor operará na potência máxima e quando atingir a faixa de temperatura especificada irá se desligar automaticamente.

Figura 13 - Aquecedor Heater-100W.



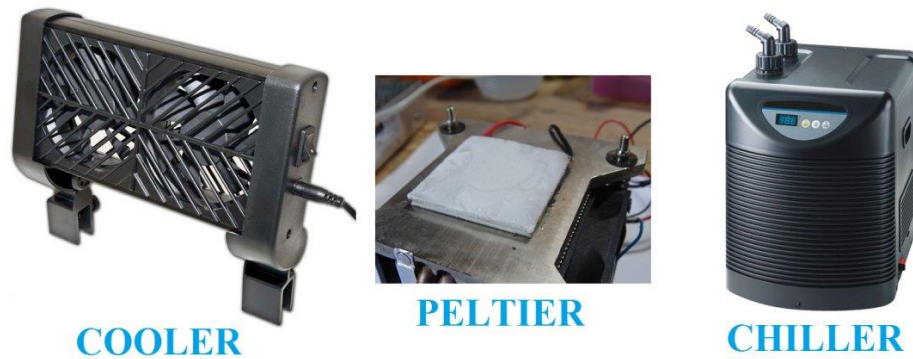
Fonte: <http://www.yellowfishaquarios.com.br/termostato-e-aquecedor-heater-100w-pr-646-400240.htm>

Assim como temos o aquecedor para realizar o aquecimento, devemos ter um dispositivo para realizar o resfriamento quando em temperaturas acima da faixa desejada.

2.9 RESFRIAMENTO

Para realizar o resfriamento de água existem basicamente 3 maneiras que estão ilustradas na figura 14.

Figura 14 – Formas de resfriamento.



Fonte: <http://www.aquaflux.com.br/conteudo/artigos/resfriamento-peltier-usar-ou-nao-no-controle-de-temperatura-de-um-aquario-1444353294.php>

Vamos falar primeiramente do equipamento mais eficaz para realizar a tarefa de resfriamento, o chiller, que consiste basicamente no mesmo funcionamento de um refrigerador, com a diferença que temos água do aquário passando onde seria o nosso congelador, porém se trata de um equipamento de alto valor e de um alto consumo de energia, seu custo está na faixa de 2 mil reais, o que já elimina ele para um projeto de um aquário residencial de pequeno porte.

O peltier se encontra como a solução de médio custo e eficiência, seu funcionamento consiste no efeito “Peltier”, que é a transferência de calor de um semicondutor para o outro quando uma corrente passa por eles, a placa esfria de um lado e esquenta do outro. Para usar no resfriamento bastaria passar a água do aquário pelo lado frio e assim nesse processo seria retirado uma pequena fatia de calor. Porém seu funcionamento é eficaz apenas para aquários de até 20 litros, o que não é nosso caso.

Os coolers utilizados em aquários são sistemas que utilizam o ar atmosférico em volta

do aquário para evaporar a água. O cooler em si não resfria a água do aquário de maneira nenhuma, o que retira o calor da água é o processo de evaporação que depende de uma série de fatores para ter resultado (aquaflux.com.br).

A água ao evaporar remove do aquário calor reduzindo a temperatura do sistema. É por esse sistema que nosso corpo mantém a temperatura ideal, através da transpiração, que, ao acontecer a evaporação do suor, o calor é dissipado e o corpo resfriado. É por isso também que sentimos frio ao passar molhado na frente do ventilador ou ao sair da piscina ou do mar (aquaflux.com.br).

O processo de evaporação retira calor da água na taxa de quase 600000 calorias para cada quilo de água evaporada (1 litro de água), o que quer dizer que se um litro de água evaporar de uma vez, um aquário de 600 litros de água (600000 gramas de água) abaixa 1°C (aquaflux.com.br).

Para o projeto então usaremos um cooler para essa tarefa, tendo em vista que não é um método de grande eficácia, ele estará em nosso projeto mais de forma representativa e temporária, ao encontrar uma maneira mais eficaz e de baixo custo bastará plugar na saída que está destinada ao cooler que funcionará da mesma maneira, pois toda a lógica e hardware já estarão prontos.

A figura 15 ilustra um cooler similar ao que será usado no projeto.

Figura 15 – Cooler para Refrigeração.



Fonte: <https://www.autocorerobotica.com.br/mini-ventilador-cooler-80x80mm-12v>.

O próximo tópico a ser tratado é sobre como será realizada a manutenção da água, para

que essa se mantenha sempre limpa, para isso usaremos um filtro externo.

2.10 FILTRO MECÂNICO EXTERNO

Um filtro mecânico é afixado na borda do aquário, e capta a água do aquário por meio de uma bomba eletromagnética de sucção. A água entra numa câmara, passa por uma manta acrílica virgem e outra impregnada de carvão ativado antes de retornar para o aquário. Este filtro é muito utilizado, e com grande eficiência, na retirada de substâncias químicas e de partículas sólidas em suspensão na água.

Este tipo de filtro deve ser deixado em funcionamento constante no aquário, ele é capaz de filtrar aproximadamente 378 litros de água por hora quando funcionando na potência máxima.

Figura 16 - Filtro Externo.



Fonte: https://produto.mercadolivre.com.br/MLB-953072750-aquaclear-110-aquario-de-energia-filtro-para-60-110-galo-_JM

Outro fator fundamental é a oxigenação da água, que no projeto será usado um pequeno compressor de ar para essa tarefa de grande importância.

2.11 COMPRESSOR DE AR

O organismo dos peixes passa a todo tempo por um processo de troca de gases, em que o peixe absorve o oxigênio e libera CO₂, o famoso gás carbônico. Nesse caso, o compressor de ar, além de oxigenar todo o ambiente em que vivem os peixes, também ajuda a eliminar o gás

carbônico do aquário. Seu funcionamento consiste basicamente em retirar o ar da atmosfera e empurrar para dentro do aquário.

Nesse projeto será utilizado o AIR PUMP AC-1000 da RESUN, que tem uma potência de 2,5 Watts, e é capaz de produzir uma pressão de 0,012MPa. Seu funcionamento assim como o Filtro Mecânico Externo deve ser constante.

Figura 17 - Compressor de Ar.



Fonte: <http://www.tommiland.eu/03816-unionstar-ac-1000-air-compressor-2-5w-ean10472-skup168.ph>

O próximo tópico a ser tratada é como será realizada a alimentação automática no aquário.

2.12 ALIMENTADOR AUTOMÁTICO

Para realizar a alimentação de maneira automática no aquário foi desenvolvido manualmente pelo autor, um sistema que utiliza um motor de toca fitas de um rádio antigo como mostrado na figura 18, esse motor é alimentado com 12 volts.

Figura 18 - Motor de Toca Fitas.



Fonte: Arquivo pessoal

O alimentador foi desenvolvido no interior de um recipiente de metal, acoplado com o motor ao fundo desse recipiente, para realizar o escoamento da ração até o orifício onde será controlado a passagem da ração foi utilizado um gargalo de um litro pet, esse orifício está tampado com uma espécie de tampão que é controlado pelo motor, seu funcionamento consiste da seguinte maneira, toda vez que o motor é acionado por um curto período de tempo, o tampão gira até um determinado ponto e quando desligado o acionamento, ele retorna ao ponto inicial por meio de um elástico que o obriga a retornar. As figuras 19 e 20 ilustram o alimentador.

Figura 19 – Vista inferior do Alimentador.



Fonte: Arquivo pessoal.

Figura 20 - Vista superior do Alimentador.



Fonte: Arquivo pessoal.

Nessas duas passagens de ida e volta irá cair uma certa quantidade de ração, logo precisamos observar a quantidade da ração despejada em uma passagem de ida e volta e com base no consumo total dos peixes do aquário determinamos quantas vezes será necessário acionar o motor para realizar a alimentação na proporção correta.

Para realizar a alimentação de maneira automática com a ESP-12E precisamos ter um módulo relé para fazer o acionamento, pois como citado anteriormente a alimentação do motor é de 12V e o microcontrolador tem saídas de no máximo 3,3V nas portas digitais. Também será preciso saber o horário para poder realizar a alimentação, e para isso será usado um módulo RTC. Ambos itens serão explicados a seguir.

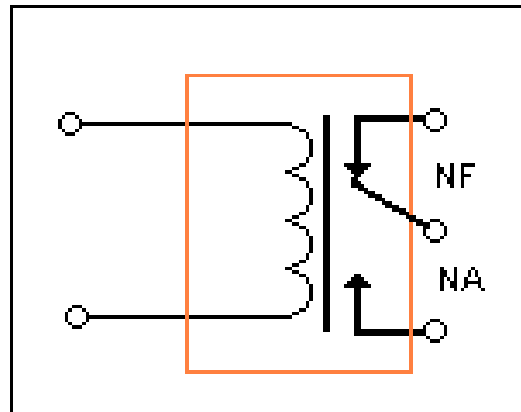
2.13 MÓDULO RELÉ

São amplamente utilizados em automação predial, sistemas de geração, transmissão e distribuição de energia elétrica e em máquinas e equipamentos em geral. São compostos por um eletroímã em forma de bobina; uma armadura metálica, que possa ser atraída pelo campo magnético criado pelo eletroímã; uma mola e um conjunto de contatos elétricos que serão abertos e fechados ou comutados conforme a configuração e o sinal recebido pelo circuito. (osetoreletrico.com.br)

Os relés são dispositivos eletromecânicos que controlam circuitos por meio de pequenas correntes ou tensões, funcionando como um interruptor liga/desliga, fechando ou abrindo conexões. Ao receber corrente, a bobina cria um campo magnético que atrai o contato fechando ou abrindo o circuito e alternando de volta para o estado inicial quando esse campo é cessado.

A figura 21 mostra o esquemático de funcionamento de um relé, onde NF é normalmente fechado e NA é normalmente aberto.

Figura 21 - Esquemático de funcionamento de um Relé.



Fonte: <https://athoselectronics.com/rele/>

Quando a corrente elétrica percorre a bobina e gera o campo magnético, a armadura é atraída por essa força e alterna a posição dos contatos dependendo da posição inicial do relé. Se a corrente é interrompida, o campo se anula e a mola faz com que os contatos voltem à posição inicial. (osetoreletrico.com.br)

Figura 22 - Módulo Relé de 4 Canais.



Fonte: <https://www.botnroll.com/pt/digital/453-modulo-rele-8-canais-5v-em-linha.html>

2.14 MÓDULO RTC DS3231

O Real Time Clock (RTC) DS3231 é um relógio de tempo real de alta precisão e baixo consumo de energia. Em sua placa vem embutido um sensor de temperatura e um cristal oscilador para melhorar sua exatidão. (filipeplop.com.br)

Este módulo DS3231 é capaz de fornecer informações como segundos, minutos, hora, dia do mês, dia da semana, mês e ano. Correções como meses com menos de 31 dias e anos bissextos são corrigidos automaticamente e pode operar tanto no formato 12 horas como 24

horas. (filipeflop.com.br)

Em caso de falha de energia o DS3231 automaticamente aciona a bateria que acompanha o módulo para evitar perda de dados. Endereço e informações são transferidas via protocolo I2C. (filipeflop.com.br)

O módulo RTC DS3231 é fabricado por diversas empresas, no projeto estaremos usando o da marca keystudio.

Figura 23 - Módulo RTC DS3231.



Fonte: <http://www.keystudio.com/keystudio-clock-module-for-arduino.html>

Este módulo será usado para podermos programar horários definidos pelo usuário para ser realizada a alimentação e a iluminação do aquário.

2.15 FITA LED 3528 SMD

Como o nome indica, o diodo emissor de luz (LED) é aquele que emite luz visível ou invisível (infravermelha) quando energizado. Em qualquer junção p-n polarizada diretamente, existe, dentro da estrutura e principalmente próximo da junção, uma recombinação de lacunas e elétrons. Essa recombinação exige que a energia do elétron livre não ligado seja transferida para outro estado. Em todas as junções p-n semicondutoras, uma parte dessa energia será liberada na forma de calor e outra parte, na forma de fótons (BOYLESTAD; NASHELESKY, 2013)

No projeto usaremos o LED 3528 que são chamados assim pois a dimensão do chip é de 3,5mm x 2,8mm. Esses chips contêm apenas 1 LED.

Figura 24 - Fita LED 3528 SMD.



Fonte: <http://www.baudaeletronica.com.br/fita-de-leds-5m-rgb-3528-a-prova-d-agua-controle-24-teclas-fonte-2a.html>

A fita usada possui iluminação em 3 cores, vermelho, verde e azul, sendo um LED destinado a cada cor, organizados na fita de maneira sistemática e alternada, possibilitando assim uma variação na iluminação e uma combinação de cores ao ser possível ligada duas ou as três cores ao mesmo tempo.

Para realizar o acionamento das cores será preciso a utilização de transistores do tipo NPN, para que seja possível acionar a cor diretamente através do microcontrolador, assim como o alimentador automático a fita LED também usa uma alimentação de 12 volts, logo também será preciso utilizar uma fonte para alimentação. Primeiramente vamos falar do transistor que fará o acionamento das cores dos LEDs.

2.16 TRANSISTOR

De maneira simples, os transistores são considerados como uma espécie de interruptor. Dependendo da sua polarização, o transistor pode trabalhar de diversas maneiras. Estes dispositivos são de extrema importância para eletrônica, podemos destacar os dois tipos principais e mais conhecidos os transistores de junção bipolar de sinal NPN e PNP.

O transistor é um dispositivo semicondutor de três camadas que consiste em duas camadas de material do tipo n e uma do tipo p ou em duas camadas do tipo p e uma do tipo n. O primeiro é denominado transistor NPN o outro, transistor PNP (BOYLESTAD; NASHELESKY, 2013).

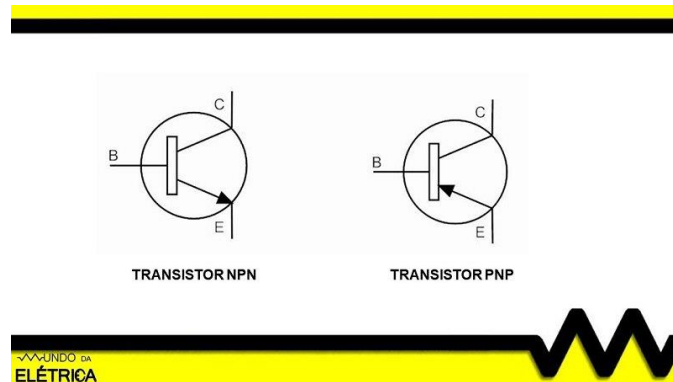
Os transistores são fabricados de diferentes formas, mas a maioria deles são fabricados apresentando três terminais diferentes, sendo elas:

- B = Base, principal responsável pela ativação do transistor.

- C = Coletor.
- E = Emissor.

Na figura 25 ilustramos as duas formas de configurações.

Figura 25 - Configurações do transistor.



Fonte: <https://www.mundodaeletrica.com.br/como-funciona-um-transistor-e-qual-a-sua-aplicacao/>

No projeto usaremos transistores do tipo NPN, portanto daremos ênfase somente a esta configuração.

2.16.1 Transistor NPN

O transistor NPN possui suas junções NP uma voltada contra a outra, com o cristal P (positivo) para as costas da outra junção, formando, então, o transistor de junção bipolar (TJB) NPN. Esse é acionado com carga positiva em relação ao emissor. (embarcados.com.br)

Será usado esse transistor para controlar a passagem do polo neutro da alimentação, ou seja:

- O coletor estará conectado ao dispositivo que deseja se acionar;
- A base estará ligada ao microcontrolador que fará o acionamento do transistor;
- No emissor ligaremos o polo neutro da fonte de alimentação.

O dispositivo que irá funcionar usando o transistor como interruptor deve estar previamente alimentado com a tensão positiva de acionamento, no caso será a fita LED, ficando dependente apenas da tensão neutra (0V) para funcionar, como vimos antes, a diferença da tensão na base (3,3V) para a tensão no emissor (0V), fará com que o dispositivo conectado a

este transistor receba a tensão de 0V.

No projeto usaremos o transistor TIP120 ilustrado da figura 26 que se trata de um transistor do tipo NPN como explicado anteriormente.

Figura 26 - TIP120.



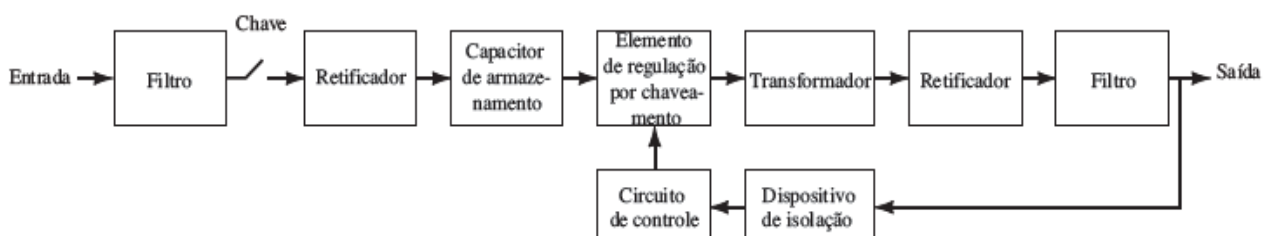
Fonte: <https://www.el-component.com/bipolar-transistors/tip120>

2.17 FONTE DE ALIMENTAÇÃO

A entrada CA é conectada ao circuito por uma série de condicionadores e filtros de linha que removem o ruído elétrico. A entrada é então retificada e levemente filtrada. A alta tensão CC é chaveada em uma taxa de aproximadamente 100 kHz. A taxa e a duração do chaveamento são controladas por um circuito integrado especial. Um transformador de isolamento liga a tensão CC chaveada a um circuito de retificação e filtragem. A saída da fonte de alimentação é realimentada no circuito integrado de controle. Monitorando a saída, o CI pode regular a tensão de saída (BOYLESTAD; NASHELESKY, 2013).

A fonte de alimentação tem como função transformar a energia elétrica que chega na nossa tomada em forma de corrente alternada em energia elétrica de corrente contínua.

Figura 27 - Diagrama em blocos da fonte de alimentação chaveada.



Fonte: (BOYLESTAD; NASHELESKY, 2013).

No nosso projeto usaremos uma fonte chaveada que regula a tensão para 12V e disponibiliza uma corrente de até 3 amperes, gerando uma potência de 36 watts, a figura 28 ilustra a fonte usada no projeto.

Figura 28 - Fonte 12 volts.



Fonte: Arquivo pessoal.

2.18 NÍVEL DE ÁGUA

O sensor de nível é uma pequena bóia com um contato elétrico encapsulado dentro dele, funciona basicamente como um interruptor, permitindo ou não a passagem de corrente elétrica por entre seus contatos. Fixado na lateral de um reservatório ou mesmo a um sistema de canos inserido dentro deste reservatório, pode ser usado como um contato aberto ou fechado para o comando elétrico de uma bomba. (mundodaeletrica.com.br)

Para cuidarmos desse pequeno detalhe, mas de grande importância, usaremos o sensor P45 ilustrado na figura 29.

Figura 29 - Sensor de Nível P45.



Fonte: <https://www.dx.com/pt/p/pp-liquid-water-level-sensor-right-angle-float-switch-p35-white-436668#.W8UDjWhKi00>

2.19 DISPLAY LCD

O LCD (Display de Cristal Líquido) é um dispositivo gráfico muito poderoso na criação de interfaces com o usuário. Amplamente utilizado nos mais diversos tipos de projetos, esses displays são formados por uma fina camada de cristal líquido entre duas placas de vidro, com uma fonte de luz fluorescente ou de LEDs por baixo de toda essa estrutura. A formação de caracteres e imagens ocorre devido ao fato do cristal líquido, naturalmente transparente, se tornar opaco ao receber uma carga elétrica, impedindo a passagem de luz. (portal.vidadesilicio.com.br)

Usaremos um display 16x2 no nosso projeto, que será usado para mostrar informações essenciais na tela, como pH, nível de água, temperatura, data, hora e status da conexão Wi-Fi.

Figura 30 - Display LCD 16x2.



Fonte: <http://www.robot-electronics.co.uk/products/lcd-displays/lcd-only/16x2-green-display.html>

Para o presente projeto foi necessário além das portas que a ESP-12E oferece, então foi usado como alternativa um registrador de deslocamento para expandir as portas lógicas digitais

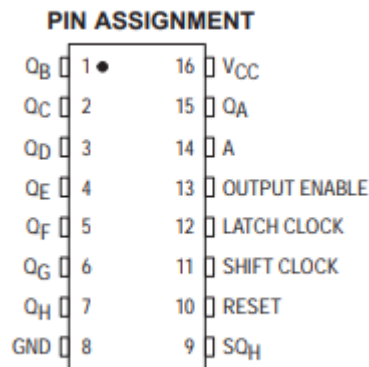
de saída.

2.20 REGISTRADOR DE DESLOCAMENTO MC74HC595

O MC74HC595 consiste de um registrador de deslocamento de oito bits, ele nos dá oito saídas digitais e ocupa apenas três para receber os dados, nos deixando então com um aumento de cinco portas lógicas.

Na figura 31 está o esquemático da pinagem.

Figura 31 – Pinagem MC74HC595.



Fonte: <http://pdf1.alldatasheet.com/datasheet-pdf/view/12198/ONSEMI/74HC595.html>

Os pinos utilizados para configurar ele são:

- GND(8)
- VCC(16)
- Latch(12)
- Clock(11)
- QA(15)

Para ser usado como expensor de portas iremos usar a função “shiftout” que irá setar os 8 pinos de saídas, considerando que zero é desligado e um ligado.

Exemplo:

```
shiftOut(dados, clockPin, LSBFIRST, 00110000);
```

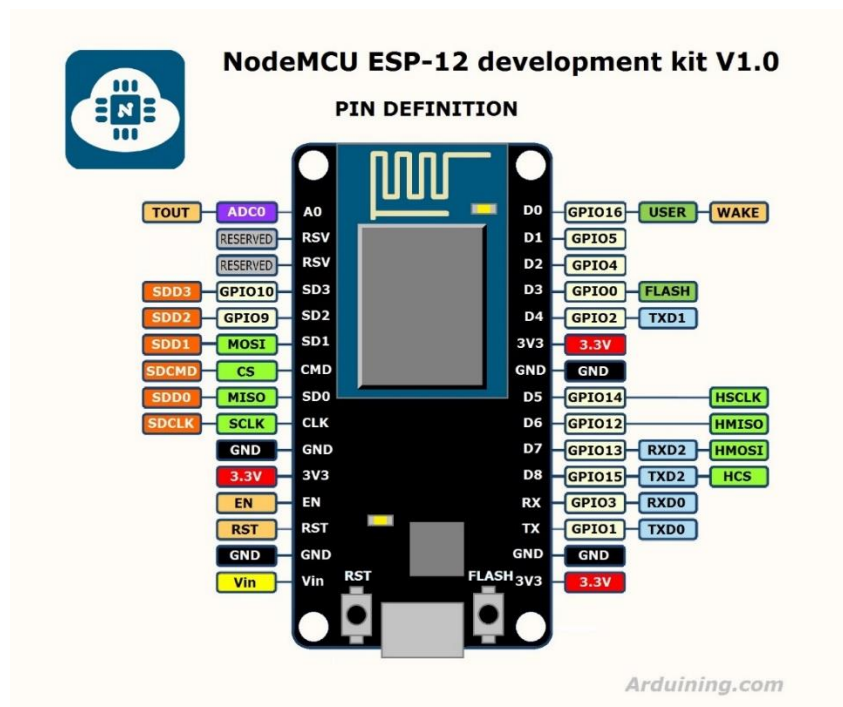
No exemplo estamos deixando as saídas 2 e 3 ligadas e as restantes desligadas.

3 DESENVOLVIMENTO

3.1 DISPOSITIVOS LIGADOS À ESP-12E

Primeiramente foi realizado uma apuração de todos os dispositivos que seriam usados no projeto e atribuído um pino que os atenda, abaixo vamos trazer a disposição dos pinos com seus respectivos periféricos, na figura 32 temos o esquemático da ordem dos pinos no microcontrolador.

Figura 32 - Disposição dos pinos no Microcontrolador.



Fonte: <http://arduinoamue.te.blogspot.com/2015/12/>

Na parte esquerda da placa temos:

- **A0** – Pino utilizado como entrada analógica ao qual está ligado o sensor de pH.
- **GND** – Pino ligado ao neutro da fonte de alimentação externa.
- **VIN** – Pino ligado à fase da fonte de alimentação externa.

O restante dos pinos da esquerda não foi usado no projeto por se tratarem de pinos de

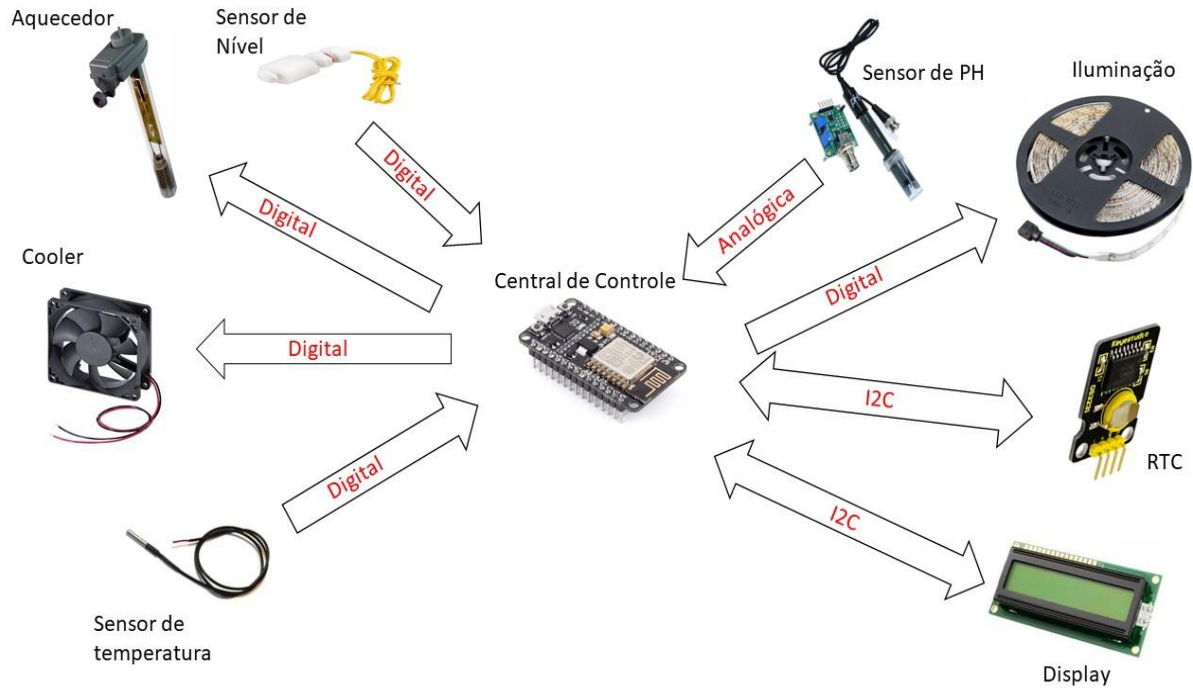
uso interno e/ou reservados.

Na parte direita da placa temos:

- **D0** – Pino utilizado para função de controle da fonte de 12V.
- **D1** – Pino usado na função padrão SCL para comunicação I2C com o display e módulo RTC.
- **D2** – Pino usado na função padrão SDA para comunicação I2C com o display e módulo RTC.
- **D3** – Pino utilizado como saída para clock do expensor de portas 74HC595.
- **D4** – Pino utilizado como saída para latch do expensor de portas 74HC595.
- **3V3** – Pino Utilizado para realizar a alimentação de 3,3V de componentes.
- **GND** – Pino Utilizado para realizar a alimentação neutro de 3,3V de componentes.
- **D5** – Pino utilizado como saída para dados do expensor de portas 74HC595
- **D6** – Pino utilizado como entrada para o sensor de temperatura.
- **D7** – Pino utilizado como entrada para o sensor de nível.
- **D8** – Pino utilizado como saída para controlar a alimentação.
- **RX** – Não utilizado.
- **TX** – Não utilizado.

Na figura 33 temos uma demonstração ilustrada, onde seta saindo do controlador é saída e entrando é entrada.

Figura 33 - Dispositivos ligados à ESP-12E.



Fonte: Arquivo pessoal.

Vamos começar falando sobre o sensor que está ligado na porta analógica A0, que desempenha uma função essencial, o sensor de pH.

3.2 CALIBRAGEM DO SENSOR DE pH

Para utilizar o sensor de pH precisamos basicamente de duas coisas, um código que faça a conversão do valor lido em volts na porta analógica A0 da ESP e adaptação na saída do sinal para a faixa desejada, pois a faixa de sinal que sai do potenciômetro chega em 5V ultrapassando a faixa de leitura da ESP-12E que é até 3,3V.

Foi utilizado o seguinte código para leitura e conversão desse valor.

Figura 34 - Código utilizado para mensurar o pH.

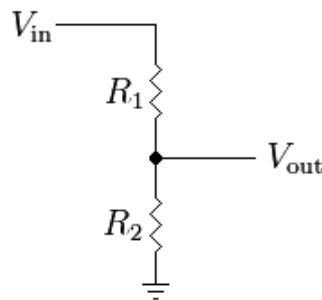
```
// ---- VERIFICAR O PH DA ÁGUA ----
int medida = analogRead(ph_pin);
double volts = ((5 / 1024.0) * medida);

ph = 7 + ((2.5 - volts) / 0.18); //transformando medidas em volts para medida em pH
```

Fonte: Arquivo pessoal.

O microcontrolador usado, consegue ler sinais numa faixa até 3,3 volts em sua porta analógica, para adequar isso foi preciso implementar um divisor resistivo na saída do sinal, como mostrado na figura abaixo.

Figura 35 - Divisor resistivo.



Fonte: <http://tabela-resistor.blogspot.com/2013/06/divisor-resistivo.html>

Usando a equação da figura 36, chegamos que $R_1/R_2 = 1.7/3.2$ para obtermos a faixa de tensão desejada de 3.3, considerando $V_{in} = 5V$ que é o sinal máximo quando o pH estiver próximo à 14.

Figura 36 - Equação Divisor Resistivo.

$$V_{out} = \frac{R_2}{R_1 + R_2} \cdot V_{in}$$

Fonte: <http://tabela-resistor.blogspot.com/2013/06/divisor-resistivo.html>

A partir da leitura do pH cria-se um alerta sempre que este estiver fora da faixa permitida, assim o proprietário deve tomar as providências necessárias para normalizar o pH. Esse alerta será despertado no broker de comunicação do protocolo MQTT, que por sua vez criará alertas sonoros e também enviará uma mensagem para o aplicativo TELEGRAM, alertando. Além disso, um alerta também será indicado no display, o qual indicará o valor do pH.

3.3 INSTALAÇÃO DO DISPLAY E MÓDULO RTC

Tanto o display quanto o módulo RTC utilizam o protocolo I2C para se comunicar com o microcontrolador, como explicado acima os pinos para essa comunicação estão por padrão em D1 e D2, que correspondem respectivamente a SCL e SDA.

Para instala-los foi realizada a alimentação dos mesmos e realizada a ligação em D1 e D2 para que se possa estabelecer a comunicação I2C.

Abaixo na figura temos o código usado para leitura de dados do módulo RTC.

Figura 37 - Código utilizado para mensurar a temperatura.

```
// ---- FUNÇÃO PARA LER HORÁRIO DO RTC ----
void horario()
{
    // ---- LER DADOS DO RTC ----
    DateTime now = rtc.now();
    segundo = now.second();
    minuto = now.minute();
    hora = now.hour();
    dia = diadasemana[now.dayOfTheWeek()];
    data = now.day();
    mes = now.month();
    ano = now.year();
}
```

Fonte: Arquivo pessoal.

Assim como nossos relógios eventualmente podem se atrasar ou adiar, nosso módulo RTC também está sujeito a tal, para melhorar a precisão do horário foi criada uma função para atualizar a hora com um servidor NTP, ou seja, sempre que o nosso microcontrolador estiver conectado a internet será chamada uma função periodicamente para atualização do horário do nosso RTC, abaixo na figura está o código usado.

Figura 38 - Código utilizado para atualizar o horário.

```

422 // ---- FUNÇÃO PARA ATUALIZAR O HORÁRIO DO RTC PELO NTP ----
423 void att_hr()
424 {
425
426     ntp.forceUpdate();
427     hr_ntp = ntp.getFormattedTime();
428
429     //dia = diadasemana[ntp.getDay()];
430     hora = ntp.getHours() + 1;
431     if(hora==24)
432     hora =00;
433     minuto = ntp.getMinutes();
434     segundo = ntp.getSeconds();
435     DateTime now = rtc.now();
436     data = now.day();
437     mes = now.month();
438     ano = now.year();
439
440     rtc.adjust(DateTime(ano, mes, data, hora, minuto, segundo));
441 }

```

Fonte: Arquivo pessoal.

O principal motivo por utilizarmos o módulo RTC é para realizar a alimentação no devido horário, e como a alimentação utiliza um motor de 12V será necessário o uso de uma fonte.

3.4 FONTE 12V

Seu uso no projeto está ligado aos seguintes periféricos:

- Alimentador;
- Iluminação;
- Cooler de resfriamento.

Sendo que quando nenhum desses estiver usando a fonte, ela permanecerá desligada para evitar o desperdício de energia e também reduzir o calor interno no equipamento.

O código que verifica se a fonte não está sendo usada por nenhum equipamento está ilustrado na figura abaixo.

Figura 39 - Código verificar se a fonte está sendo ocupada.

```
// ---- FUNÇÃO PARA VERIFICAR SE A FONTE ESTÁ SENDO OCUPADA ----
void verificafonte()
{
  boolean t=0;
  if((controlador[1] == '1') || (controlador[2] == '1') || (controlador[5] == '1') || controlador[3] == '1')
  {
    t=1;
  }
  if(!t){
    digitalWrite(fonte, 0);
  }
}
```

Fonte: Arquivo pessoal.

Seu funcionamento consiste em sempre que alguma dessas funções necessitar utilizar a fonte, elas façam a requisição para ligar, e ao final é chamada essa função para verificar se algum outro dispositivo também está utilizando, caso contrário a fonte é desligada.

Para o controle dessa fonte com a ESP-12E é necessário utilizar um relé, pois a alimentação da fonte é 220V.

3.5 MÓDULO RELÉ

O módulo relé usado no projeto dispõem quatro saídas, as quais serão usadas da seguinte maneira:

- Saída 1: Acionamento da fonte 12V.
- Saída 2: Acionamento do aquecedor.
- Saída 3: Acionamento da alimentação.
- Saída 4: Acionamento do cooler.

Para realizar o acionamento de uma saída basta que um sinal de tensão aproximada de 3V chegue na respectiva entrada e a saída se acionará, alimentando assim o dispositivo a ela conectada.

A saída 1 do relé já foi explicada anteriormente, agora vamos tratar de como é realizado o acionamento do aquecedor e do cooler pela ESP-12E, mas antes disso vamos verificar como é realizada a mensuração da temperatura, pois para ligar o aquecedor ou o cooler antes precisamos saber a temperatura.

3.6 CONFIGURAÇÃO DO SENSOR DE TEMPERATURA

Para realizar a instalação do sensor DS18B20 foi realizada a alimentação do mesmo na tensão de 3,3V e ligado o outro terminal para leitura no pino D6 da ESP-12E, a partir disso basta programar o trecho de código utilizado para leitura da temperatura.

Figura 40 - Código utilizado para mensurar a temperatura.

```
// ---- MENSURANDO A TEMPERATURA ----
sensors.requestTemperatures();
temp = (sensors.getTempCByIndex(0));
```

Fonte: Arquivo pessoal.

A partir que temos o conhecimento da temperatura podemos ligar/desligar o aquecedor/cooler quando houver a necessidade para tal. A temperatura mínima e máxima desejada no aquário também pode ser ajustada através do broker, abaixo temos a figura que ilustra esse ajuste.

Figura 41 – Tela do broker responsável por ajustar valores de temperatura.

Controle de Temperatura

Ajuste:
VÁLIDO

Ajuste de Temperatura

Temperatura Máxima ●

Temperatura Mínima ●

Valores Gravados em Aquadroit

Temp. Máxima:	30,00
Temp. Mínima:	22,00

Fonte: Arquivo pessoal.

Onde valores gravados em Aquadroit são aqueles que em caso do usuário não configurar nada eles são os valores padrão, assim que o usuário muda, eles automaticamente mudam.

Nas figuras 42 e 43 temos a lógica para ligar/desligar o aquecedor e resfriador respectivamente.

Figura 42 – Lógica para ligar/desligar o aquecedor.

```
if (temp < tempMin)
{
    controlador[3] = '1';
    ctrl = strtol(controlador, NULL, 2);
    digitalWrite(latchpin, LOW);
    delay(100);
    shiftOut(dados, clockPin, LSBFIRST, ctrl);
    digitalWrite(latchpin, HIGH);
} else {
    controlador[3] = '0';
    ctrl = strtol(controlador, NULL, 2);
    digitalWrite(latchpin, LOW);
    delay(100);
    shiftOut(dados, clockPin, LSBFIRST, ctrl);
    digitalWrite(latchpin, HIGH);
}
```

Fonte: Arquivo pessoal.

Figura 43 - Lógica para ligar/desligar o resfriador.

```

if (temp > tempMax)
{
    controlador[4] = '1';
    ctrl = strtol(controlador, NULL, 2);
    digitalWrite(latchpin, LOW);
    delay(100);
    shiftOut(dados, clockPin, LSBFIRST, ctrl);
    digitalWrite(latchpin, HIGH);
} else {
    controlador[4] = '0';
    ctrl = strtol(controlador, NULL, 2);
    digitalWrite(latchpin, LOW);
    delay(100);
    shiftOut(dados, clockPin, LSBFIRST, ctrl);
    digitalWrite(latchpin, HIGH);
}

```

Fonte: Arquivo pessoal.

Agora nos resta explicar o último item ligado no módulo relé, o alimentador.

3.7 ALIMENTADOR AUTOMÁTICO

Ligado na saída 3 do módulo relé, será acionado de duas diferentes maneiras:

- ❖ Alimentação automática em horários estabelecidos pelo usuário;
- ❖ Alimentação manual através do broker.

Assim como o usuário pode definir os horários da alimentação, também é possível determinar a quantidade de ração que será dada em cada refeição, esse controle é realizado através da quantidade de vezes que se acionará o motor em cada horário da alimentação. Para ajustar isso basta ir até o broker e ajustar essas configurações.

Abaixo segue a tela do broker responsável por manipular esses valores.

Figura 44 – Tela do broker responsável por manipular valores da alimentação.

Horários para Alimentação

Ajuste da Alimentação

1º-Hora	▼ 0 ▲
1º-Minutos	▼ 0 ▲
2º-Hora	▼ 0 ▲
2º-Minutos	▼ 0 ▲
Quantidade de Porções	▼ 1 ▲

Valores Gravados em Aquadroit

Horas 1º Alimentação:	10
Minutos 1º Alimentação:	30
Horas 2º Alimentação:	18
Minutos 2º Alimentação:	30

ALIMENTAR

Fonte: Arquivo pessoal.

Assim como no caso da temperatura, aqui também temos os valores de alimentação pré-definidos, como o ideal é realizar no máximo duas alimentações diárias para os peixes, foi restringido a agendar apenas dois horários para alimentação, caso o usuário decida por colocar um só, basta deixar o horário um igual ao horário dois, lembrando que também é possível realizar a alimentação manual a qualquer horário, bastando pra isso pressionar o botão alimentar.

O campo quantidade de porções é referentes a quantas vezes será acionado o motor em cada refeição, também tem seu valor pré-definido que é três.

Abaixo temos os trechos de códigos usados para efetivamente realizar a alimentação no microcontrolador.

Figura 45 - Código utilizado realizar a alimentação automática.

```

// ---- FUNÇÃO PARA REALIZAR A ALIMENTAÇÃO
void alimentacao()
{
    digitalWrite(fonte, 1);
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Alimentando");
    lcd.setCursor(4, 1);
    lcd.print(">(((*>..");

    for(int y=0; y < vezes ; y++)
    {
        digitalWrite(alimentar, 1);
        lcd.setCursor(12 + y, 0);
        lcd.print(".");
        delay(800);
        digitalWrite(alimentar, 0);
        delay(500);
    }

    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Alimentado!!");
    lcd.setCursor(4, 1);
    lcd.print(">((((*>");
    lcd.setCursor(14, 0);
    lcd.print(":D");

    alimentou = minuto + 5;
    if(alimentou > 59){
        alimentou = alimentou - 60;
    }
    passa = false;
    verificafonte();
}

```

Fonte: Arquivo pessoal.

O código da figura 45 descreve a alimentação da maneira automática, o trecho responsável pela alimentação manual está dentro da função comando, que recebe informações vinda do broker, será explicado mais detalhadamente a seguir abaixo na figura 46 mostramos parte do código responsável pela alimentação manual que está dentro da função comando.

Figura 46 - Código utilizado realizar a alimentação manual.

```
if (cmd == "alimentar") {  
    digitalWrite(fonte, 1);  
    delay(300);  
    digitalWrite(alimentar, 1);  
    delay(800);  
    digitalWrite(alimentar, 0);  
    verificafonte();  
}
```

Fonte: Arquivo pessoal.

O próximo tópico a ser tratado é a iluminação.

3.8 ILUMINAÇÃO

Para a iluminação utilizaremos a fita LED que possui três cores ordenadas alternadamente, onde cada cor é ligada independente da outra. A fita LED possui 4 terminais, sendo um deles para a alimentação positiva e os outros três para acionar cada cor, sendo assim será deixada o terminal positivo já alimentado na fonte de 12V e para cada um dos outros terminais utilizaremos um transistores do tipo NPN para fazer o acionamento.

O comando para iluminação virá do broker, para isso criamos um tópico no broker chamado de comando, e na ESP-12E realizamos a subscrição nesse tópico, sendo assim sempre que houver algum dado novo nesse tópico o broker o enviará para a ESP-12E que por sua vez terá que fazer o processamento do mesmo, nesse mesmo tópico também será publicado a alimentação manual quando esta for acionada pelo usuário. Na figura abaixo temos um trecho de código responsável por processar as informações recebidas nesse tópico, a mesma lógica se repete para as outras duas cores e a lógica de acionamento da alimentação está ilustrada na figura 47.

Figura 47 – Parte do código de interpretação do tópico comando.

```
// ---- FUNÇÃO PROCESSAR AS INFORMAÇÕES RECEBIDAS NO TÓPICO COMANDO ----
void controle(String cmd)
{
  if (cmd == "verde/1") {
    digitalWrite(fonte, 1);
    controlador[1] = '1';
    ctrl = strtol(controlador, NULL, 2);
    digitalWrite(latchpin, LOW);
    delay(100);
    shiftOut(dados, clockPin, LSBFIRST, ctrl);
    digitalWrite(latchpin, HIGH);
  }

  if (cmd == "verde/0") {
    controlador[1] = '0';
    ctrl = strtol(controlador, NULL, 2);
    digitalWrite(latchpin, LOW);
    delay(100);
    shiftOut(dados, clockPin, LSBFIRST, ctrl);
    digitalWrite(latchpin, HIGH);
    verificafonte();
  }
}
```

Fonte: Arquivo pessoal.

Figura 48 – Tela do broker responsável por controlar a iluminação.



Fonte: Arquivo pessoal.

Para a iluminação também foi utilizada um módulo no Node-RED, a node-red-contrib-schedex, onde foi informado a ela nossas coordenadas geográficas, e então selecionada algumas funções em decorrer da posição do sol.

Utilizamos a função “sunrise”, que quer dizer que quando a borda superior do sol estiver aparecendo no horizonte ela disparará algo conectado a ela, no nosso caso ela irá ligar a iluminação na cor verde automaticamente.

Outra função utilizada foi a “sunriseEnd”, que é quando a borda inferior do sol estiver aparecendo no horizonte, então ligaremos a cor azul.

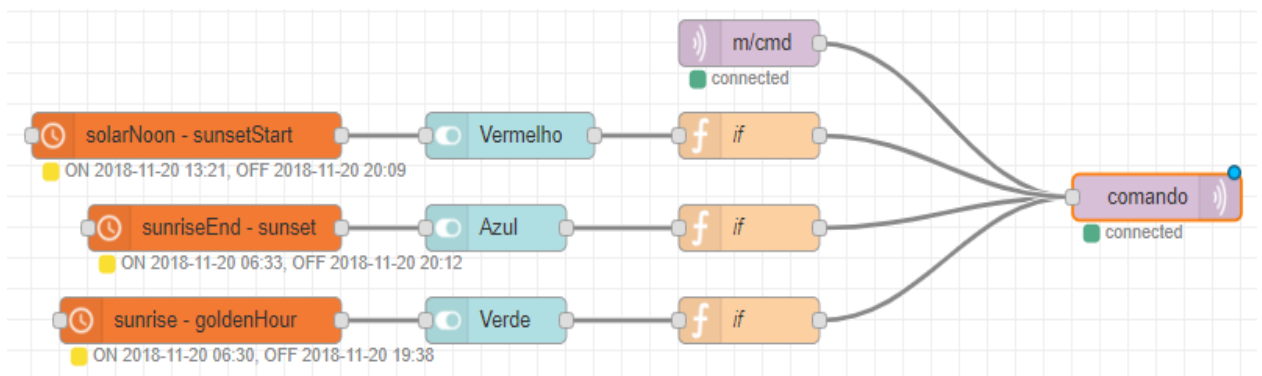
Para realizarmos por fim o ligamento da cor vermelho usaremos a função “solarNoon” que é o horário do dia que o sol está com o brilho máximo.

Para realizarmos o desligamento da iluminação foi utilizada funções dessa biblioteca também, vamos começar o desligamento pela cor verde usando a função “goldenHour”, que é o horário do dia que a iluminação começa a ficar semelhante a cor dourada.

Para desligarmos a cor vermelha usaremos a função “sunsetStart” que é o momento em que a borda inferior do sol “toca” o horizonte.

Uma última função utilizada desse módulo foi a “sunset” que é o horário em que o sol desaparece abaixo do horizonte, e a iluminação se acaba por inteiro, nesse instante será realizado o desligamento total da iluminação desligando a cor azul.

Figura 49 – Tela de programação do broker responsável por controlar a iluminação.



Fonte: Arquivo pessoal.

A utilização desse módulo e essas funções tem como objetivo principal simular o amanhecer/entardecer do dia da melhor e mais precisa forma, e também para realizar o acionamento e desligamento da iluminação quanto o dono não se lembrar ou não estiver em casa para fazer, pois os peixes necessitam da iluminação diariamente.

Depois de citar cada item ligado ao módulo relé, vamos falar de outro sensor importante, o de nível de água.

3.9 NÍVEL DE ÁGUA

Com apenas 2 fios nesse sensor, ele é de simples instalação, para realizar a instalação foi ligado um fio na tensão de 3,3V e outro ao pino D7, que é o responsável por verificar o nível através do seguinte código.

Figura 50 – Código responsável pela leitura do nível de água.

```
// ---- VERIFICANDO NÍVEL DE ÁGUA ----  
if (digitalRead(sensorNivel) == 1)  
{  
  nivel = "BAIXO";  
  
} else {  
  nivel = "NORMAL";  
}
```

Fonte: Arquivo pessoal.

Da mesma maneira que o pH, o nível de água cria um alerta sempre que este estiver no nível baixo, alertando o usuário que deve repor a água da maneira correta.

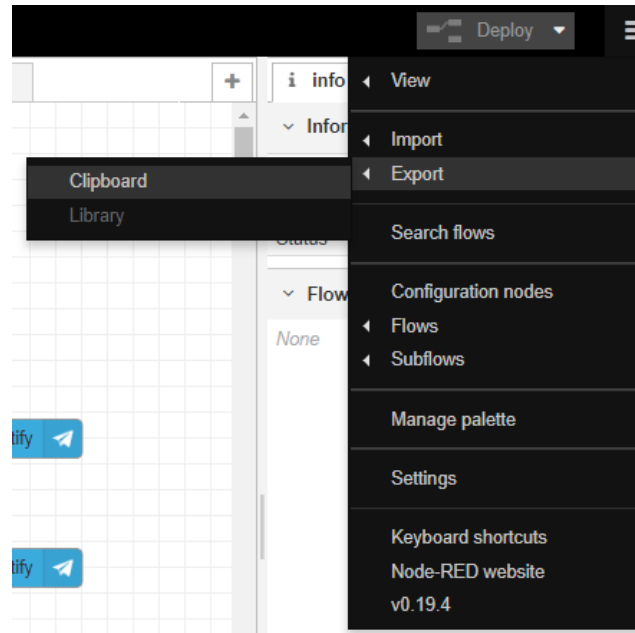
Vamos falar sobre a interface criada no broker utilizando o Node-RED e suas funções.

3.10 Node-RED e BROKER MQTT

Existem inúmeros brokers gratuitos disponíveis para uso, assim como existem os pagos, as maneiras que podemos hospedar o broker foi exemplificado na figura 3, página 18. Dos públicos, existem diversas opções, os mais conhecidos são o da eclipse, IBM Bluemix e HiveMQ, estes disponibilizam uma franquia de dados mensais de maneira gratuita e uma limitação no número de clientes, o que para uso particular já é o suficiente nos três casos. Aqui estaremos utilizando o Node-RED de maneira local, instalado em um computador, pois é a melhor maneira de usar enquanto está em fase de desenvolvimento, depois de programado basta exportar o código para um broker na nuvem para se ter acesso universal através da internet.

Nas figuras 51 e 52 mostramos como o processo de importar o código do broker de um servidor para outro é rápido e fácil.

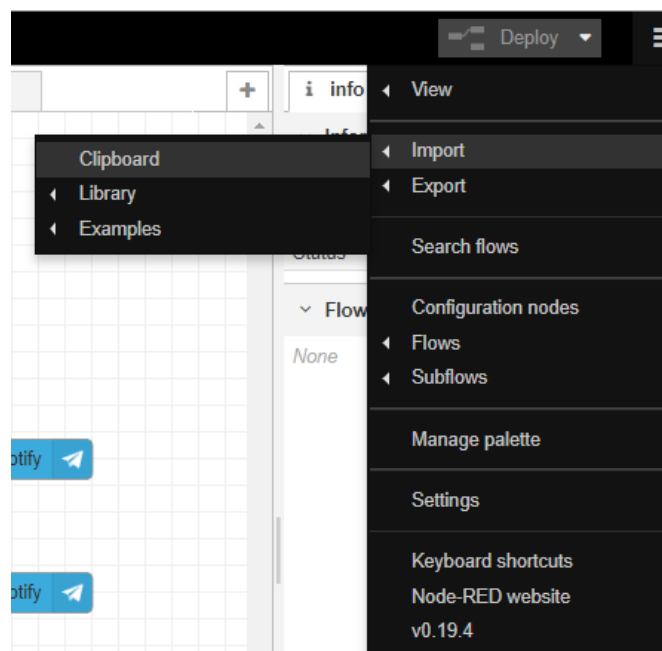
Figura 51 – Maneira para exportar um código.



Fonte: Arquivo pessoal.

Depois de clicar em clipboard, basta copiar o conteúdo e ir até o broker que deseja importar e realizar um processo semelhante ao mostrado acima, o qual está ilustrado na figura abaixo, ao clicar em clipboard pasta colar o conteúdo e salvar.

Figura 52 – Maneira para importar um código.



Fonte: Arquivo pessoal.

Para utilizarmos qualquer equipamento digital é interessante possuir uma interface amigável, simples e robusta, antes de citarmos cada interface, vamos mostrar abaixo algumas capturas de tela do broker.

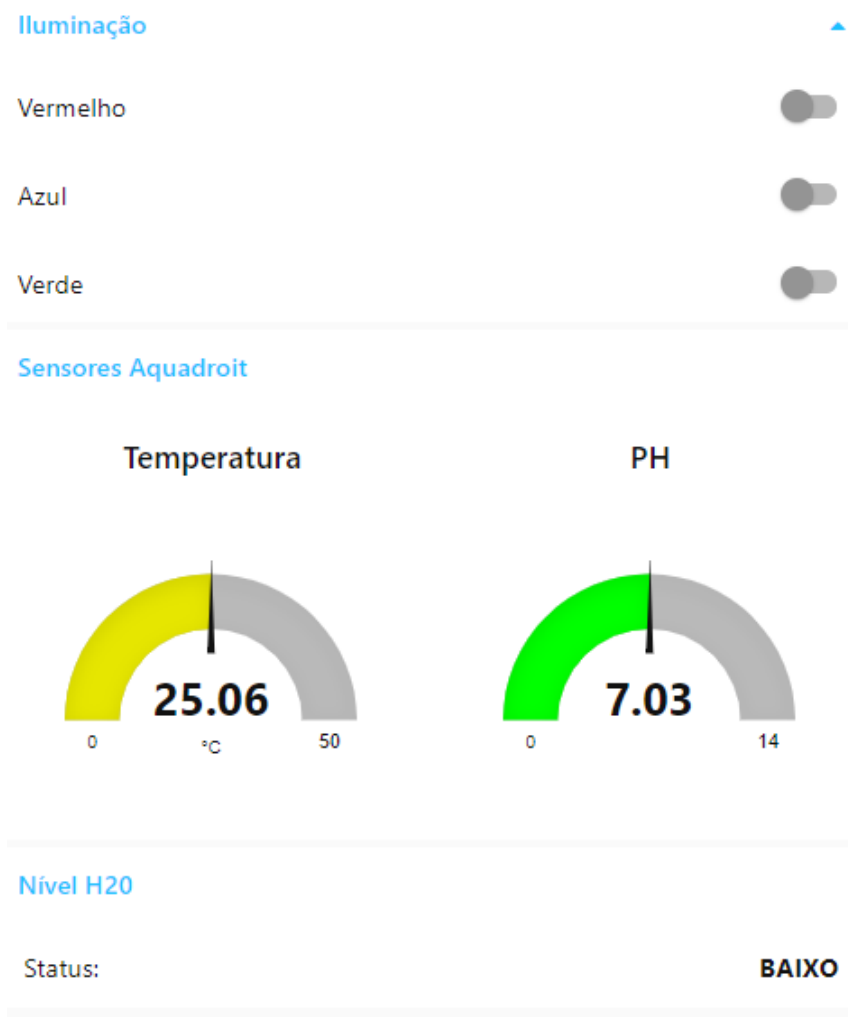
Figura 53 – Menu de navegação disponibilizado do lado superior esquerdo.



Fonte: Arquivo pessoal.

De fácil navegação, disponibiliza apenas 3 itens, sendo o primeiro a tela principal, onde estão os principais dados e comandos, abaixo na figura está ilustrada uma captura de tela realizada.

Figura 54 – Tela de Início no broker.



Fonte: Arquivo pessoal.

Na tela Central de controle temos a seguinte interface, está dividida em 2 imagens para melhor visualização.

Figura 55 – Tela Central de Controle no broker parte 1.

Horários para Alimentação

Ajuste da Alimentação

1º-Hora	▼ 0 ▲
1º-Minutos	▼ 0 ▲
2º-Hora	▼ 0 ▲
2º-Minutos	▼ 0 ▲
Quantidade de Porções	▼ 1 ▲

Valores Gravados em Aquadroit

Horas 1º Alimentação:	10
Minutos 1º Alimentação:	30
Horas 2º Alimentação:	18
Minutos 2º Alimentação:	30

ALIMENTAR

Fonte: Arquivo pessoal.

Figura 56 – Tela Central de Controle no broker parte 2.

Controle de Temperatura

Ajuste:
VÁLIDO

Ajuste de Temperatura

Temperatura Máxima ● —————

Temperatura Mínima ● —————

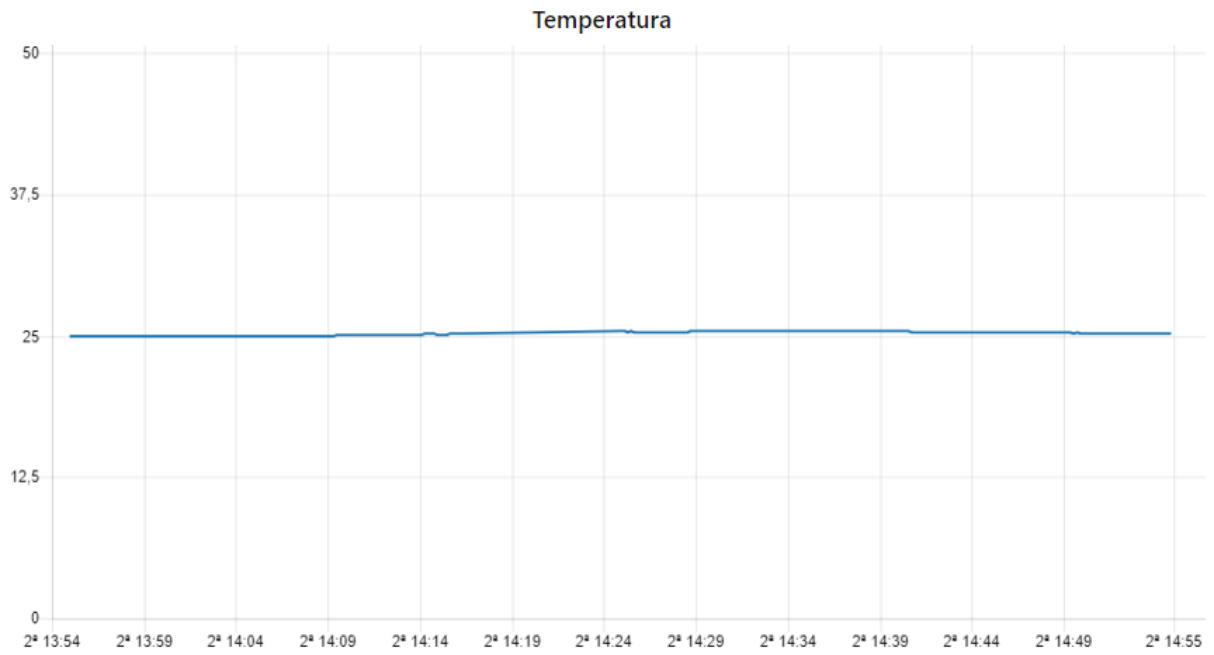
Valores Gravados em Aquadroit

Temp. Máxima:	30,00
Temp. Mínima:	22,00

Fonte: Arquivo pessoal.

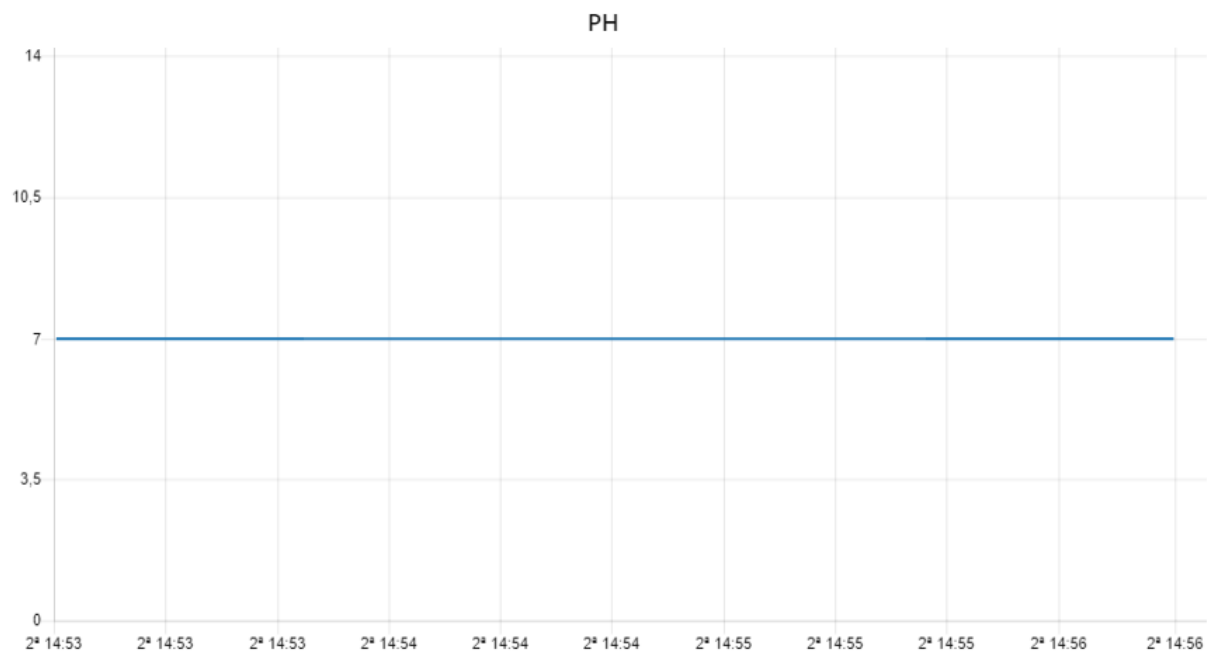
Por fim temos a interface de Gráficos onde é mostrado a variação do pH e temperatura no decorrer do dia, assim como a tela de central de controle também será dividida em duas figuras para melhor visualização.

Figura 57 – Tela Gráficos no broker parte 1.



Fonte: Arquivo pessoal.

Figura 58 – Tela Gráficos no broker parte 2.



Fonte: Arquivo pessoal.

Nos resta agora apresentar o aplicativo utilizado no projeto, trata-se de um aplicativo já

existente, desenvolvido especificamente para comunicação MQTT, disponível para download na Play Store no caso de sistemas Android, existem também aplicativos para iPhones disponíveis para download na App Store, aqui daremos ênfase somente ao aplicativo para o sistema Android escolhido, o qual está descrito a seguir.

3.11 APLICATIVO MQTT DASH

Disponível para download na Play Store, este foi o aplicativo escolhido entre diversos testados, pois é um aplicativo de baixo consumo de armazenamento e requer pouco hardware do aparelho, possibilitando dessa maneira ser instalado em um grande faixa de dispositivos e não somente nos mais sofisticados. Na figura abaixo está ilustrada seu ícone na Play Store.

Figura 59 – Aplicativo MQTT Dash.

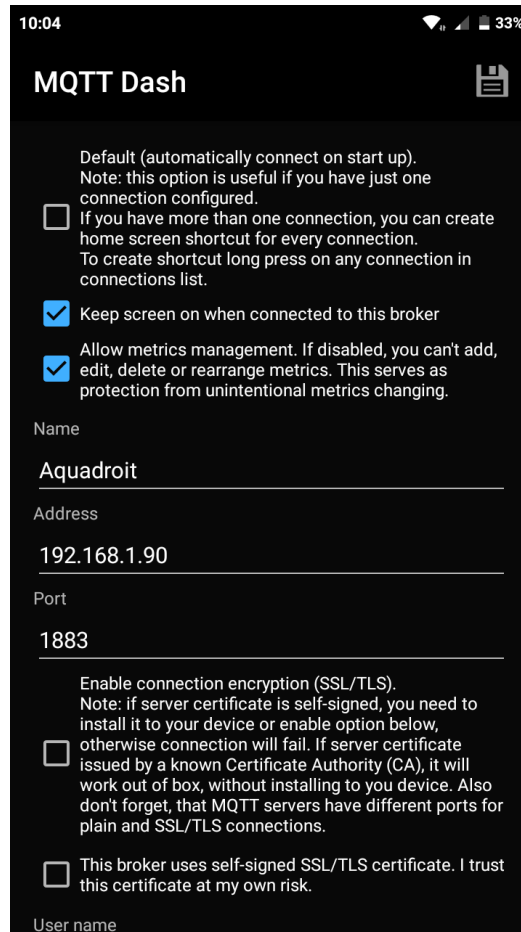


Fonte: <https://play.google.com/store/apps/details?id=net.routix.mqttdash>.

Após realizar o download e instalação do mesmo, colocamos um nome para esse broker que desejamos e então configuramos o endereço do nosso broker, como está sendo utilizado um broker local, colocaremos então o IP do computador utilizado, e a porta 1883 que é a porta padrão para uso local.

Abaixo a figura que ilustra a configuração inicial do aplicativo.

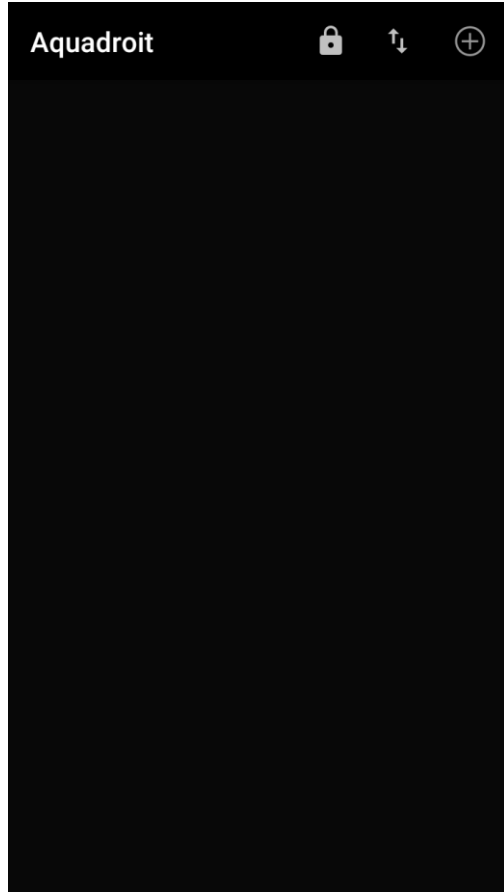
Figura 60 – Captura de tela 1 MQTT Dash.



Fonte: Arquivo pessoal.

A seguir vamos para uma tela em que temos apenas na parte superior direita três opções, devemos então clicar no símbolo de “adição” e adicionar os tópicos que desejamos nos inscrever e os tópicos que queremos publicar. A figura abaixo ilustra a tela.

Figura 61 – Captura de tela 2 MQTT Dash.



Fonte: Arquivo pessoal.

4 RESULTADOS

Esta seção mostrará os resultados da comunicação do broker MQTT para manipulação dos valores com a ESP-12E, com o aplicativo TELEGRAM que foi integrado ao sistema para utilização como alerta e com o aplicativo descrito na seção 3.11, e como ele ficou depois de configurado todos os botões.

4.1 COMUNICAÇÃO ENTRE A ESP-12E E BROKER MQTT

Essa comunicação como foi vista está dividida em duas funções, a de leitura do tópico no broker pela ESP quando o broker receber novas informações vinda da interface do Node-RED, ou então do aplicativo utilizado, e a função da publicação de valores no broker. Na figura 62 temos a função void loop utilizada na ESP, que será a rotina que nosso microprocessador realizará.

Figura 62 – Função de rotina do microprocessador.

```

void loop()
{
  ArduinoOTA.handle(); //Handle é descritor que referencia variáveis no bloco de memória
  if(WiFi.status() != WL_CONNECTED){
    conectarWifi();
  }
  if (!client.connected() && WiFi.status() == WL_CONNECTED) { //Se o cliente não estiver conectado, então
    reconectarMQTT();
  }
  client.loop();

  horario(); //lê horario do módulo RTC |

  if( (hora == hr1prg && (minuto >= min1prg && minuto < (min1prg+3)) && passa) || (hora == hr2prg && (minuto >= min2prg && minuto < (min2prg+3)) && passa) )
  {
    alimentacao();
  }else if(minuto < 55 && minuto > alimentou){
    passa = true;
  }

  aferirSensores();
  if(WiFi.status() == WL_CONNECTED) //Caso esteja conectado ai wifi publica no broker e atualiza a hora
  {
    att_hr();
    publicarDados();
  }

  imprimir();
}

```

Fonte: Arquivo pessoal.

Na função de rotina, temos que a publicação dos dados é periódica, ou seja após a leitura dos sensores e se o wi-fi estiver conectado é realizada a publicação no broker, então nosso broker recebe periodicamente essas informações, e quanto a leitura de tópicos subscritos, a função cliente.loop que verifica de maneira periódica se existem novos dados no broker nos tópicos que estamos subscritos, caso afirmativo chamamos a função de leitura.

O ponto que queremos chegar é que a ESP possui uma rotina e que leva um determinado tempo tanto para ela verificar se existem novos dados no broker, quanto para publicar, esse tempo de rotina da ESP é o tempo que somado a velocidade de conexão da rede, nos dá o tempo para comunicação entre broker e microcontrolador.

Por esse motivo a função void loop deve ser a mais pequena possível, caso contrário o tempo de comunicação seria enorme, nosso código utilizado na ESP possui aproximadamente oitocentas linhas, enquanto a rotina principal ficou com vinte e sete linhas.

4.2 COMUNICAÇÃO DO BROKER COM O TELEGRAM

Para realizar a comunicação com o telegram foi necessário criar um “bot” no telegram para criarmos uma aplicação que possa enviar mensagens diretamente ao aplicativo através do nome do bot criado e do chatId disponibilizado posteriormente, já no Node-RED foi instalado um módulo e configurado com os parâmetros do telegram gerados ao criar o “bot”, na figura 63 temos o tela de configuração do Node-RED com o parâmetros do telegram.

Figura 63 – Configuração do Telegram no Node-RED.

The image shows a configuration window titled "Edit notify node" with three buttons at the top: "Delete", "Cancel", and "Done". Below the buttons is a section labeled "node properties" with a dropdown arrow. The configuration fields are as follows:

- Name:** A text input field containing the word "Name".
- Bot:** A dropdown menu showing "aquadroit_bot" with a small edit icon to its right.
- Chat ID:** A text input field containing the number "455740179".
- Parse Mode:** A dropdown menu showing "Plain Text".
- Message:** A text input field containing the text "Alerta! o pH está alto!".

Fonte: Arquivo pessoal.

No caso da imagem acima, esse alerta será acionado quando o pH estiver acima do nível ideal, temos mensagens de alerta do telegram para as seguintes situações:

- ❖ pH acima do limite;
- ❖ pH abaixo do limite;
- ❖ limite de temperatura máxima definida é menor que a mínima;
- ❖ nível de água baixo.

Para todas as situações acima, é criado o alerta no telegram pois a única maneira de normalizar essas situações é manualmente pelo usuário, então é importante alertá-lo das mais diversas maneiras.

Abaixo na figura temos um exemplo de uma mensagem alerta recebida no aplicativo

telegram.

Figura 64 – Mensagem alerta no telegram.



Fonte: Arquivo pessoal.

O telegram permanecerá recebendo mensagens de alerta até que a situação se normalize, podemos cadastrar diversos dispositivos para receber esses alertas, nele também poderia ter sido configurado mensagens para receber outras informações como temperatura, pH, entre outros, assim como poderíamos mandar mensagens ao broker para realizar o acionamento da iluminação, alimentar, entre outros, mas o objetivo do telegram no sistema é apenas receber alertas, para a configuração de valores e recebimentos de dados estamos utilizando o aplicativo descrito na seção 3.11.

Para essas situações também é criado um alerta sonoro no broker, onde é reproduzida uma frase alertando sobre a ocorrência, esse alerta sonoro no broker assim como no telegram permanece até que a situação esteja normalizada.

4.3 INTERFACE DO APLICATIVO MQTT DASH

No aplicativo foram colocadas apenas as funções mais cotidianas, deixando as configurações de regulagem de temperatura mínima, máxima, horários de alimentação, quantidade de ração de cada alimentação apenas para configuração no navegador, pois estas geralmente irão ser configuradas apenas uma vez e não precisam estar presentes no aplicativo, isso iria gerar mais uma poluição visual do que uma funcionalidade.

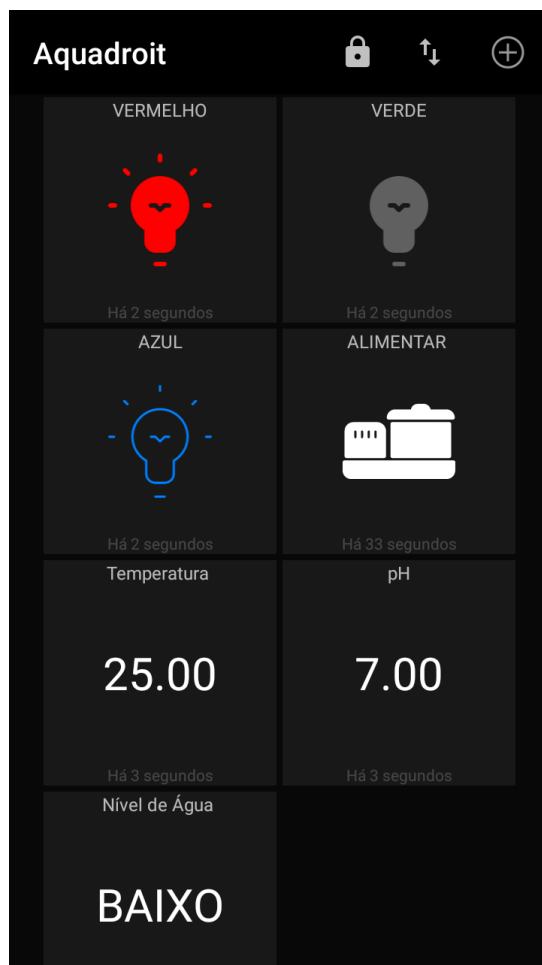
Na figura abaixo ilustramos como ficou a interface.

Figura 65 – Interface MQTT Dash.



Os três primeiros botões representam a iluminação da ESP, nele podemos fazer o acionamento/desligamento das mesmas assim como recebemos os status dela periodicamente, na figura 66 mostramos uma situação em que a luz vermelha e azul estão ligadas e a verde desligada.

Figura 66 – Captura de tela MQTT Dash.



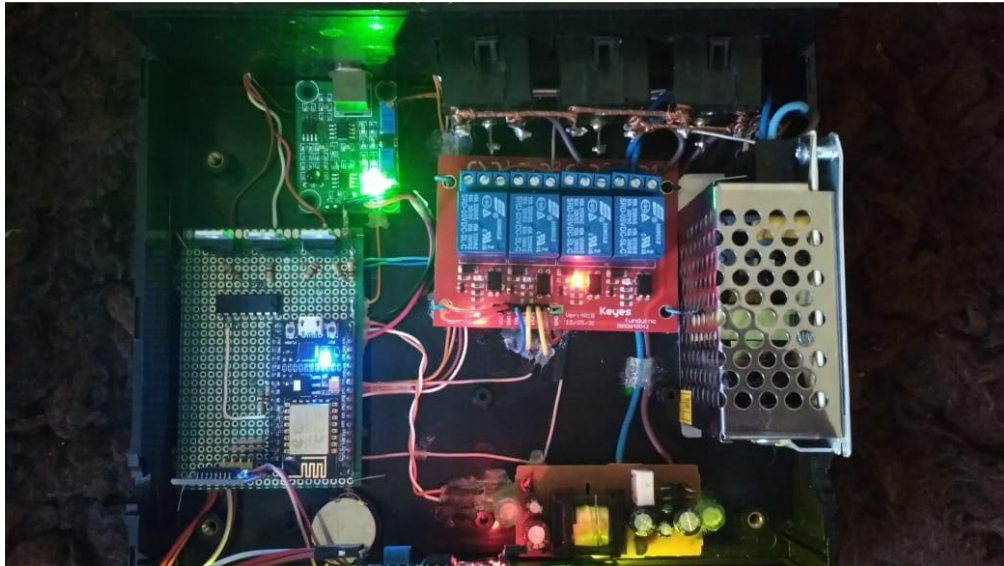
Fonte: Arquivo pessoal.

O botão alimentar realiza a alimentação de maneira manual ao ser pressionado e os três últimos servem para visualização dos valores obtidos periodicamente nos sensores.

4.4 PROJETO FISICO AQUADROIT

Por fim iremos mostrar como ficou nosso projeto ao final da montagem dos componentes.

Figura 67 – Vista Superior Aquadroit.



Fonte: Arquivo pessoal.

Figura 68 – Vista traseira Aquadroit.



Fonte: Arquivo pessoal.

Aqui na parte traseira temos três tomadas, sendo elas para o filtro externo, compressor de ar e aquecedor respectivamente na ordem esquerda para direita, os demais fios são para leitura dos sensores e iluminação.

Figura 69 – Vista frontal Aquadroit.



Fonte: Arquivo pessoal.

5 CONCLUSÃO

A tecnologia está em constante evolução e isso faz com que soframos mudanças na maneira que interagimos. Esse avanço revela a cada dia novas formas de fazermos tarefas cotidianas de maneiras mais simples e eficientes.

A internet das coisas traz muitos benefícios para a sociedade, auxiliando na realização das tarefas cotidianas, e também integrando cada vez mais todos os dispositivos presentes na residência.

No presente projeto foi apresentado o desenvolvimento de um sistema para integrar o controlar o aquário de maneira remota, seja por computador, celular ou então qualquer outro dispositivo que seja capaz de manter uma conexão à internet.

De acordo com os resultados obtidos durante os testes, conclui-se que o sistema se apresentou eficiente, permitindo de maneira simples a conexão com a internet e uma rápida e eficaz troca de informações com o broker, durante a sessão de testes não houve sequer uma perda de informações nessa troca de mensagens.

O sistema aqui desenvolvido pode ser usado como referência e inspiração para o desenvolvimento de sistemas similares, pois assim como no projeto, que o objetivo visa facilitar a tarefa de quem possui aquário, e principalmente a melhoria da qualidade de vida dos peixes presentes nele, toda aplicação desenvolvida que tem como objetivo beneficiar de alguma maneira a sociedade, a natureza ou a relação entre ambos é uma ótima maneira de se aplicar os conhecimentos adquiridos dentro de sala de aula na sociedade.

REFERÊNCIAS

- AGRAWAL, S.; VIEIRA, D. A survey on internet of things. **Abakós**, v. 1, n. 2, p. 78–95, 2013.
- LIMA, W. D. A internet das coisas. **TECNOLOGIAS EM PROJEÇÃO**, v. 8, n. 2, p. 67–78, 2017.
- TEIXEIRA, F. A.; PEREIRA, F.; VIEIRA, G.; MARCONDES, P.; WONG, H. C.; NOGUEIRA, J. M. S.; OLIVEIRA, L. B. **Siot – defendendo a internet das coisas contra exploits**. In: Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (SBRC). Florianópolis: SBRC, 2014. Acesso em 8 set. 2018. Disponível em: <<http://sbrc2014.ufsc.br/anais/files/trilha/ST14-1.pdf>>.
- MANCINI, M. **Internet das coisas: História, conceitos, aplicações e desafios**. 2017.
- TANENBAUM, A. S. **Redes de computadores**. 4ª. ed. [S.l.]: Campus, 2003.
- MARTINS, B. D.; COSTA, R. M. B.; QUEIROZ, R. L. de. **IEEE 802.11 a,b,g,n**. 2013. Acesso em 10 out. 2018. Disponível em: <https://www.gta.ufrj.br/grad/13/_1/80211abgn/IEEE_802.11n.html>.
- <https://www.ibm.com/developerworks/br/library/iot-mqtt-why-good-for-iot/index.html>. Acesso em: 15 out. 2018.
- <https://nodered.org/#get-started>. Acesso em: 15 out. 2018.
- GIMENEZ, Salvador. **Microcontroladores 8051** 3ª. Ed. São Paulo: Pearson Education do Brasil, p. 4, 2005.
- JUCÁ, Sandro. **Apostila de Microcontroladores PIC e Periféricos**. Disponível em <http://www.ebah.com.br/content/ABAAAAMX0AH/apostila-microcontroladores-pic-perifericos.html>. Acesso em: 12 de outubro, 2018.
- PEREIRA, F. **Microcontroladores PIC: Programação em C**. 7ª. Ed. São Paulo: Érica, 2007.
- GROOVER, Mikell P. **Automation, Production Systems and Computer Integrated Manufacturing Systems**. 2ª Ed. Prentice Hall, 2000.
- ESPRESSIF. **ESP8266EX Datasheet Versão 5.8**. 2018. Acesso em 09 out. 2018. Disponível em: <https://www.espressif.com/sites/default/files/documentation/0a-esp8266ex_datasheet_en.pdf>.
- <http://blog.instrusul.com.br/como-funciona-medidor-de-ph/>. Acesso em 18 out. 2018.
- Hackstyle. **ESP8266 – Configuração básica**. 2016. Acesso em 02 out. 2018. Disponível em: <<https://hackstyle.org/>>.
- CURVELLO, A. **APRESENTANDO O MÓDULO ESP8266: Comunicações**. 2015.

Acesso em 09 out. 2018. Disponível em: <<https://www.embarcados.com.br/modulo-esp8266/>>.

WINGWIT. 1-Wire Bus Protocol. Disponível em:

<http://pt.wingwit.com/Ferragens/networkequipment/46083.html#.U967TPldU_U>. Acesso em: 10 out. 2018.

<http://www.aquaflux.com.br/conteudo/artigos/resfriamento-peltier-usar-ou-nao-no-controle-de-temperatura-de-um-aquario-1444353294.php>. Acesso em: 10 out. 2018.

<https://www.osetoreletrico.com.br/reles-e-contatores/>. Acesso em 05 de nov. 2018.

<https://www.filipeflop.com/produto/real-time-clock-rtc-ds3231/>. Acesso em 07 de nov. 2018.

BOYLESTAD, ROBERT; NASHELESKY, LOUIS. Dispositivos Eletrônicos e Teoria de Circuitos. 11º Ed. São Paulo: Pearson Education do Brasil, p. 36-37, 2013.

<https://www.embarcados.com.br/transistor-pnp-canal-eletronica-facil/>. Acesso em 10 nov. 2018.

<https://www.mundodaeletrica.com.br/control-de-nivel-de-reservatorio/>. Acesso em 5 out. 2018.