

**UNIVERSIDADE FEDERAL DE SANTA MARIA
CENTRO DE TECNOLOGIA
CURSO DE ENGENHARIA DE CONTROLE E AUTOMAÇÃO**

**SISTEMA DE MEDIÇÃO E MONITORAMENTO DO
CONSUMO DE ÁGUA DURANTE O BANHO COMO
FORMA DE CONSCIENTIZAÇÃO**

**RELATÓRIO DA DISCIPLINA DE MICROCONTROLADORES PARA
AUTOMAÇÃO**

Prof. Robinson Figueiredo de Camargo

Bruno Gabriel Flores Sampaio

**Santa Maria, RS, Brasil
2020**

RESUMO

Com as crescentes notícias acerca do uso de forma inadequada da água, uma das preocupações de toda comunidade científica e industrial mais conscientes, está sobre como reduzir o consumo de água através da conscientização das pessoas e criação de aparelhos que consigam economizar os recursos hídricos. Como forma de conscientização a Organização das Nações Unidas (ONU) criou no dia 22 de março de 1992, o Dia Mundial da Água, divulgando a Declaração Universal dos Direitos da Água, entretanto, pouca coisa mudou de lá para cá, pois, as pessoas não tem consciência de quantos litros de água elas gastam por dia e de que forma elas gastam esses recursos.

Como forma de conscientização, nesse projeto será proposto a criação de um dispositivo que mede o volume de água gasto durante um banho e, que assim, a pessoa que esteja fazendo o uso do chuveiro, possa acompanhar esse consumo durante o processo. Acredita-se que com a visualização do volume total de água gasto durante um banho, o usuário tome consciência de quantos litros de água ele está desperdiçando, muitas vezes com coisas desnecessárias, e assim, o seu tempo de banho seja reduzido, economizando água consequentemente.

Sabe-se da importância da conscientização a respeito do uso da água e é necessário ter-se os órgãos de apoio ao nosso lado, para que se consiga usar a água plenamente e assim deixar-se um legado positivo para as futuras gerações. Com cada um fazendo a sua parte e economizando, na medida do possível, conseguiremos manter um mundo mais saudável e sem o medo de como será no futuro, tudo graças a tecnologia. A água é um bem de todos e deve ser respeitada.

SUMÁRIO

1. INTRODUÇÃO	7
2. OBJETIVO GERAL	8
3. OBJETIVO ESPECÍFICO.....	8
4. ESPECIFICAÇÕES.....	9
4.1 Sensor de fluxo.....	9
4.2 Microcontroladores Arduino	10
4.3 Módulo Transmissor e Receptor RF 433MHz FS1000A / MX-RM-5V.....	11
4.4 Display LCD 16x2	12
4.5 Alimentação	13
5. DESENVOLVIMENTO	14
5.1 HARDWARE.....	15
5.1.1 Transmissor	15
5.1.2 Receptor	19
5.1.3 Estimativa de manutenção.....	22
5.1.4 Conclusão de Hardware	23
5.2 SOFTWARE	24
5.2.1 Fluxograma do dispositivo Transmissor	25
5.2.2 Fluxograma do dispositivo Receptor.....	26
5.2.3 Comentário acerca do software	27
5.3 Protótipo e projeto final.....	27
CONCLUSÃO	28
BIBLIOGRAFIAS.....	31
6. APÊNDICE	33
6.1 Código do dispositivo Transmissor.....	33
6.2 Código do dispositivo Receptor	35
6.3 Modelagem 3D da carcaça do dispositivo Receptor	38

LISTA DE FIGURAS

Figura 1: Sensor de fluxo YF-S201	10
Figura 2: Arduino UNO a Direita e Arduino Nano a Esquerda	11
Figura 3: Módulo receptor RF a esquerda e módulo transmissor RF a direita.....	12
Figura 4: Display LCD 16x2	13
Figura 5: Arduino NANO e pinagens (Retirado de www.filipeflop.com)	16
Figura 6: Transmissor FS1000A visto de cima	17
Figura 7: Sensor de fluxo YF-S201	18
Figura 8: Fonte de alimentação 9V do transmissor	19
Figura 9: Módulo receptor MX_RM_05	20
Figura 10: Cores RGB (Red, Green, Blue).....	21
Figura 11: Anodo Comum e Catodo Comum.....	22
Figura 12: Modelagem 3D da carcaça do dispositivo Receptor	27
Figura 13: Protótipo dispositivo Transmissor	30
Figura 14: Protótipo dispositivo Receptor	30
Figura 15: Código Dispositivo Transmissor – Cabeçalho	33
Figura 16: Código Dispositivo Transmissor - Função <i>setup</i>	33
Figura 17: Código Dispositivo Transmissor - Função <i>Loop</i>	34
Figura 18: Código Dispositivo Transmissor - Função Controle para resetar	34
Figura 19: Código Dispositivo Receptor – Cabeçalho	35
Figura 20: Código Dispositivo Receptor - Função <i>setup</i>	36
Figura 21: Código receptor - função <i>loop</i>	37
Figura 22: Mensagem do display	38
Figura 23: Modelagem - vista ortogonal e frontal	38
Figura 24: Modelagem - vista ortogonal esquerda e direita	39

LISTA DE TABELAS

Tabela 1: Tabela de componentes do transmissor	16
Tabela 2: Tabela de componentes do receptor	19
Tabela 3: Pinos do LCD	21
Tabela 4: Tabela de alcance do transmissor/receptor em 1Hz	23
Tabela 5: Gastos anuais e economia de luz com o dispositivo operando – Parte 1	29
Tabela 6: Gastos anuais e economia de luz com o dispositivo operando - Parte 2	29

LISTA DE EQUAÇÕES

Equação 1: Taxa de fluxo YF-S201	9
Equação 2: Dimensionamento de antenas de RF.....	17

1. INTRODUÇÃO

Pesquisas acerca do uso indiscriminado da água no planeta Terra vem sendo feitas a muito tempo, e as estimativas são sempre preocupantes, a água está acabando. Segundo Claudio Angelo, no seu livro “O Aquecimento Global” (ANGELO, s.d), em até 2025, cerca de 2,7 bilhões de pessoas no mundo podem sofrer com a falta de água potável de alguma forma, principalmente nas regiões em que a escassez de água já é um problema hoje. O Brasil ainda não sente esse efeito em grande parte do país, principalmente por possuir grandes reservas de água distribuídas por grandes extensões do seu território, no entanto, isso nos traz grandes responsabilidades acerca do seu uso.

Um levantamento feito a respeito do consumo urbano de água no Brasil, relacionou aspectos demográficos e o consumo residencial e um dos aspectos descobertos foi que, apesar da diminuição do crescimento no número de pessoas, houve um crescente aumento no volume de água gasto pelas residências (CARMO, 2014). Um dos fatores para o aumento do volume de água gasto pelas famílias, certamente se dá ao fato de muitas tecnologias terem sido desenvolvidas para aumentar o bem-estar das pessoas. Com o comodismo criado por essas novas tecnologias, o desperdício dos recursos hídricos aumentaram como consequência da facilidade de se usar tais dispositivos, pois agora temos o prazer de conseguir tomar banhos mais quentes nos dias mais frios, regar os quintais quase que diariamente e manter as calçadas e carro sempre limpos, tudo a custo de “um pouquinho” mais de água.

Pensando no âmbito de conscientização, é difícil se ter consciência de um problema, quando não sabemos o impacto das nossas ações para o agravamento dos mesmos. Muitas vezes em atividades que julgamos inocentes, estamos causando um desperdício que poderia ser contornado se tivéssemos noção dos nossos gastos e desperdícios. Tarefas de higienização pessoal ou de bens, como limpeza carros e casa, são essenciais e não podemos deixar de usufruir dos comodismos que temos, no entanto, precisamos sim saber dos nossos gastos e de como isso afeta nossa vida e a vida de todos no planeta Terra, direta e indiretamente.

Portanto o primeiro passo para isso, é criar um sistema de medição de vazão em sistemas hídricos que sejam de fácil implementação e que permitam a visualização em tempo real do consumo individual para os dispositivos de consumo hídricos já existentes. Dentro do ambiente residencial podem-se fazer a identificação de vários pontos onde esses dispositivos poderiam ser instalados, como por exemplo, em tubulações de mangueiras de jardim, caixa

d'água, torneiras e tanques, chuveiros e máquinas de lavar, salvando dezenas de litros de águas por dia.

2. OBJETIVO GERAL

Como forma de tentar conscientizar a população, uma das formas se dá com a visualização dos gastos que esse volume de recursos hídricos traz, gerando uma economia espontânea. A fim de sanar isso, será criado um dispositivo capaz de fazer a visualização desses gastos, propondo a criação de um dispositivo que consiga fazer a leitura do volume de água gasto durante um banho, mostrando-o durante o processo, a fim de que assim, a pessoa que esteja usando-o, consiga ver em tempo real, quantos litros de água ela está gastando enquanto ensaboa o corpo ou lava os cabelos com o chuveiro ligado, por exemplo.

Buscando um meio de conseguir fazer essa medição, nada mais prático do que fazer o uso de um microcontrolador comercial de código aberto, que possa ser replicado com certa facilidade e fazendo o uso de sensores, sendo possível prever com considerável precisão os gastos de água e através de um *display*, tornamos esses valores visíveis a quem estiver no chuveiro. Esse mesmo sistema poderia ser expandido nos demais sistemas hidráulicos da casa, como mangueiras e máquinas de lavar, como citados anteriormente. Por isso a preocupação de criar um sistema descentralizado de sensoramento e que consiga interligar todos os sistemas em um único ponto, a fim de mostrar onde toda a água está sendo usado e em que quantidade.

Esse seria um sistema ideal e que toda residência já deveria possuir, visto os impactos positivos no consumo consciente de água que esses dispositivos trariam. Portanto, para atingirmos tal objetivo, devemos criar um sistema de fácil aplicação, com um bom custo benefício e que seja confiável.

3. OBJETIVO ESPECÍFICO

Dentro os objetivos deste trabalho, os mais específicos são:

1. Medir e monitorar o volume de água por segundo (vazão) de um sistema hidráulico, com precisão.
2. Fazer a transmissão dos dados de vazão coletados pelos sensores, para outro sistema independente, totalmente via Rádio Frequência (RF).

3. Garantir integridade na comunicação. Receber os quadros com a certeza de que os dados recebidos foram os mesmos enviados pelo transmissor.
4. Mostrar a vazão e o volume de água gastos durante o uso do sistema, juntamente de quanto tempo o sistema esteve ligado e gerando consumo durante o processo.
5. Ser capaz de saber quando o sistema está ligado ou não, sem a necessidade de um botão que ligue os dispositivos toda vez que uma nova medição seja requerida.

4. ESPECIFICAÇÕES

Os sensores, módulos e microcontroladores utilizados para a execução desse projeto, foram escolhidos de acordo com a familiarização dos mesmos para o desenvolvimento do projeto, custo benefício e acessibilidade. Será dada uma breve explicação de como funciona cada um dos elementos base que serão utilizados no projeto, tendo como objetivo de familiarizar o leitor às formas de funcionamento dos dispositivos. Assim, quando for mostrada as formas de desenvolvimento do projeto, ficará mais fácil de entender o papel de cada peça e como cada um deles funciona.

4.1 Sensor de fluxo

O sensor de fluxo utilizado para o projeto é o sensor *YF-S201*. Esse sensor usa a força mecânica da água que flui do encanamento para dentro dele, girando pares de hélices que se encontram dentro do sensor. Ao girar as hélices, o sensor cria um pulso magnético que pode ser captado através do efeito *hall*. Esses pulsos definem o fluxo da água que escoar sobre o sensor e possuir uma taxa definida pela Equação 1.

$$\text{Equação 1: Taxa de fluxo YF-S201} \\ \text{Pulso (Hz)} = 7.5 \times \text{Taxa do fluxo 'Q' [L / min]}$$

Isso significa que a cada pulso dado por segundo, o fluxo de água é proporcional a 7.5 vezes o valor de 1 litro de água por minuto, gerando uma taxa linear de variação do fluxo pelo número de giros da hélice, sendo fácil de se predizer essa taxa de vazão. De acordo com a folha

de dados do sensor (*datasheet*). Taxa de erro que o sensor pode atingir é de 1 a 3% tanto para cima, quanto para baixo. Além de tudo isso, ele também é considerado um dos sensores com maior custo benefício no mercado para esse tipo de aplicação em domótica e *IoT*, utilizando o efeito *hall* (J. Lalnunthari and H. H. Thanga, 2017). A Figura 1 mostra as dimensões do sensor retiradas diretamente dessa folha de dados.



Figura 1: Sensor de fluxo YF-S201

4.2 Microcontroladores Arduino

Arduino é uma empresa de hardware livre que fabrica placas com um microcontrolador e um *IDE* (*Interface de desenvolvimento Integrada*) com as que qualquer usuário pode criar projetos de todo tipo. A ideia é que os usuários, tanto os avançados como os novatos, possam montar uma placa próprias, criando um design único para suas necessidades e usá-la livremente sem ter que comprar uma placa pré-fabricada (GONZÁLEZ, 2016). Dessa forma, a empresa conta com milhares de desenvolvedores que não medem esforços para ajudar os mais novos no ramo, tornando assim a plataforma mais acessível e confiável.

Contando com isso, escolheu-se um microcontrolador que fosse acessível e que ao mesmo tempo conseguisse atender as demandas do projeto. No site da Arduino consegue-se ter todas as informações a respeito dos produtos que a empresa oferece e suas especificações completas (ARDUINO).

Para o projeto, serão necessárias duas controladoras, uma para fazer a captação dos dados do fluxo do chuveiro, por meio do sensor citado anteriormente e outra para fazer o processamento desses dados. Pensando que a primeira controladora estará acoplada no encanamento e sujeita a condições adversas de temperatura e umidade, foi escolhido o Arduino Nano, devido as suas dimensões físicas reduzidas, enquanto para fazer o processamento dos dados, escolheu-se o Arduino UNO, por ser mais versátil e compatível com as necessidades do projeto, além de ser bastante manejável. A Figura 2 mostra a diferença de tamanho entre o Arduino UNO, a Direita e o Nano, a Esquerda.

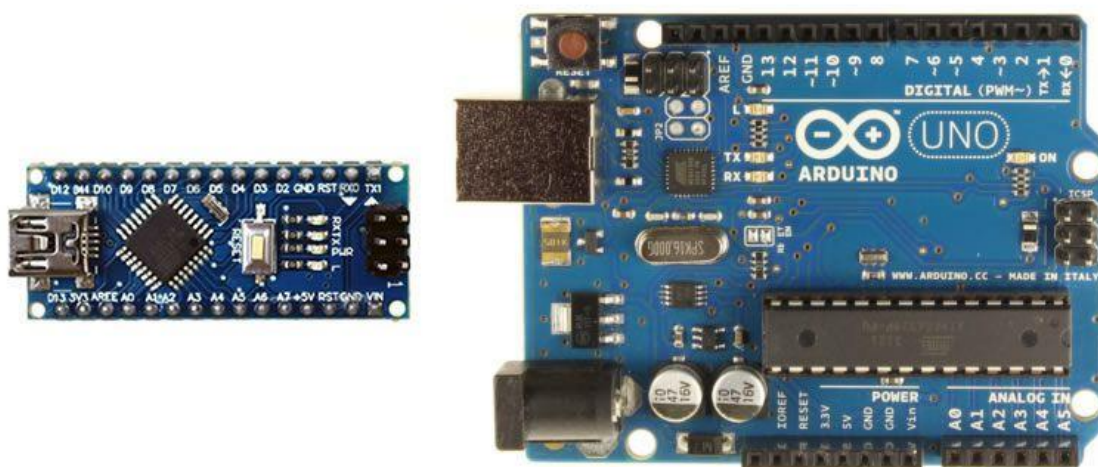


Figura 2: Arduino UNO a Direita e Arduino Nano a Esquerda

4.3 Módulo Transmissor e Receptor RF 433MHz FS1000A / MX-RM-5V

Para se fazer a transmissão dos dados, será utilizado um par de transmissores de rádio frequência, que operam no espectro de 433MHz. O transmissor é do modelo *FS1000A* e o receptor é do modelo *MX-RM-5V*. Esses módulos operam com modulação *ASK (Amplitude Shift Keying)*, ao qual através de uma onda portadora de alta frequência, transmitem dados de forma digital a uma boa distância em metros, motivo esse pelo interesse nos módulos. Na Figura 3 encontra-se a foto de um par do módulo, estando o transmissor a direita e o receptor a esquerda.

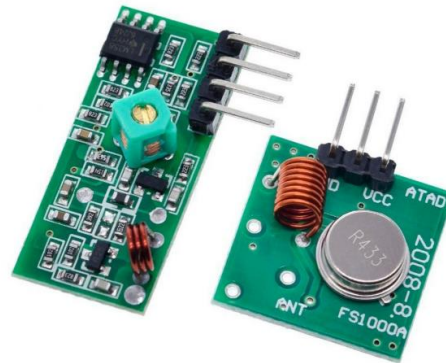


Figura 3: Módulo receptor RF a esquerda e módulo transmissor RF a direita

O sistema será do tipo *simplex*, ou seja, haverá comunicação somente em um sentido. O transmissor envia os dados para o receptor e não há inversão. Para garantir a segurança na entrega dos dados, haverá um *FCS* (*Frame Check Sequence*) para a detecção de erros. Dessa forma, se um dado esperado não for recebido ou então não chegar inteiro, se a checagem de erro falhar, o sistema não irá contabilizar esses dados distorcidos ou inexistentes, garantindo que não seja mostrado ao usuário, informações erradas ou então que o sistema falhe por falta de coesão ou precisão.

O sistema receptor/transmissor apresenta boa confiança, uma vez que já foi usado em artigos acadêmicos para fazer a transmissão de dados de diagnósticos de máquinas e dispositivos de sensoriamento (PRAZNOWSKI, Krzysztof & AUGUSTYNOWICZ, Andrzej, 2018) além de possui muitos materiais disponíveis na Web, dentro da comunidade *maker*.

4.4 Display LCD 16x2

A fim de acontecer uma relação entre o usuário e sistema, será incorporado uma *IHM* (interface homem máquina) em forma de *display*. O *display* em questão é o *LCD* (*Display* de Cristal Líquido) que cria uma interface com o usuário, onde será impresso todas as informações que coletaremos e que julgamos necessárias. O *LCD* possui 16 colunas e 2 linhas, com luz de fundo (*backlight*) verde e letras na cor preta. Para conexão, são 16 pinos, dos quais usamos 12 para uma conexão básica, já incluindo as conexões de alimentação, *backlight* e contraste (Figura 4).

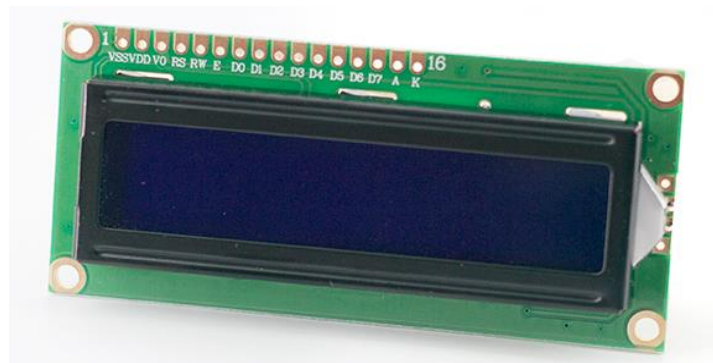


Figura 4: Display LCD 16x2

4.5 Alimentação

Para conseguir alimentar o sistema, será necessário se levar em consideração alguns aspectos, como o local onde será implementado, quais as condições desse ambiente e quais os recursos disponíveis ali. Como o sistema, a primeiro momento, será dividido em duas partes, temos que levar em conta dois locais de instalação e, portanto, temos duas situações.

A primeira e mais simples das duas, será o local onde estará a interface de comunicação com o usuário. Esta estará disposta na altura dos olhos, dentro do local de banho e por se tratar de um local bastante úmido e com riscos de choques, de acordo com a NBR5410, não é permitido tomadas ou qualquer outra fonte de energia a menos de 60 centímetros de boxes ou locais de banho (ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS, 2004, pag 170). Portanto, o sistema deverá utilizar uma bateria de 9V que consiga suprir a sua demanda energética.

Para o segundo caso, já se pode usar mais artifícios, uma vez que, apesar do local inapropriado, onde há um chuveiro, haverá também uma fonte de alimentação bastante forte, capaz de suprir o pequeno sistema. Nesse ponto, irá se referir ao sistema, sendo composto pelo Arduino, o sensor de fluxo e o módulo transmissor, portanto, não há a necessidade de uma fonte de energia grande ou então, o risco de sobrecarga da rede, uma vez que o gasto energético do sistema é quase nulo, quando comparado ao da instalação de um chuveiro. Nesse caso, será utilizado uma fonte de alimentação 12V própria que será conectada diretamente na rede, utilizando os próprios conectores do chuveiro ou então outro conector comercial do mesmo padrão.

5. DESENVOLVIMENTO

De forma resumida, será utilizado um sensor de fluxo, acoplado em série ao encanamento dos dispositivos ao qual fará o sensoriamento, o dispositivo em questão será o chuveiro. O funcionamento do Sensor de Fluxo de água baseia-se em uma turbina estilo hélice que gira conforme o fluxo de água e detecta a quantia de voltas dadas, gerando um sinal de pulso modulado *pwm* (*pulse width modulation*). Dessa forma, usando um microcontrolador da família Arduino, ele será capaz de detectar qual o volume de água que está fluindo pelo encanamento e assim, fazer o monitoramento.

O microcontrolador utilizado será da família Arduino devido ao fácil acesso dos pinos de leitura digital e *pwm*. Além da medição do fluxo de água, o microcontrolador, também será utilizado para fazer a transmissão dos dados coletados com o sensor para outro microcontrolador da mesma família, a fim de conseguir fazer pontos independentes de monitoramento, com um único ponto de recepção para controle dos gastos. Dessa forma tem-se o controle do sistema, dividido em dois pontos independentes, interligados de forma virtual e controlados pelos Arduinos de forma que a programação de cada um seja feita conforme as necessidades dele e assim, tornando o sistema modular, sendo bastante fácil de se fazer a expansão do sistema para mais dispositivos dispersos sob a casa.

O microcontrolador UNO assim como o Nano, possuem características muito similares, exceto pelas dimensões físicas. No entanto, as demais diferenças são irrelevantes para o projeto, uma vez que os microcontroladores são suficientes para o projeto, possuindo pinos digitais suficientes e entradas e saídas *pwm* que são os requisitos mais importantes. A programação de ambos pode ser feita na mesma *IDE* e não possuem diferença de linguagem alguma.

Pensando na transmissão dos dados coletados pelo sensor de Fluxo, a primeira ideia que saltou aos olhos, foi fazer via transmissão em rádio frequência. Existem módulos desse tipo que podem ser usados com o microcontrolador Arduino, em espectros de frequências legais na faixa de 433MHz. Tais módulos, apesar de conseguirem, não precisam possuir alcance maior que alguns metros e podem usar antenas compactas para isso.

Para fazer a ponte entre o sistema e o usuário, fez-se necessário o uso de uma *IHM* simples, que pode ser um pequeno *display* sem muitas funções, além da função de mostrar o registro do consumo de água e o desperdício gerado. Dessa forma, tem-se um sistema completo

e bastante simples, com possibilidade de expansão e que consegue obter todos os objetivos propostos.

Para demonstração das etapas de desenvolvimento do sistema de forma completa, foi necessário se separar em dois grandes tópicos, sendo o primeiro tópico, a parte de **Hardware**, onde será montado o sistema físico do projeto e através da folha de dados dos dispositivos, feito o dimensionamento do sistema para que consiga operar conforme o esperado e com a máxima redução de erros. Enquanto o segundo tópico, fala a respeito da implementação de **Software** do sistema, abordando pontos importantes para o funcionamento pleno do sistema lógico. O código utilizado pelos sistemas estará disponível no apêndice do documento.

5.1 HARDWARE

A primeira coisa feita após a definição das dores que o sistema iria sanar e como ele iria fazer, foi a escolha das peças que comporiam o sistema como um todo. Cada sensor ou dispositivo usado, foi escolhido com alguma tarefa em particular que ele deveria solucionar e com isso, criar o sistema como uma peça única. Cada peça tem uma particularidade sua e funciona com algum protocolo de operação próprio, portanto, a familiarização com os componentes utilizados, foi um fator importante para a sua escolha, mais até que o seu preço ou desempenho. Dessa forma, outros dispositivos podem ser substituídos em réplicas ou adaptações desse projeto, desde que cumpram com as mesmas funções dos dispositivos originalmente escolhidos.

O sistema está plenamente modularizado, de tal forma que é possível se analisar componente por componente, individualmente, sem que se perca precisão da ideia.

5.1.1 Transmissor

Para o sistema de transmissão dos dados, estão inclusos os componentes listados na Tabela 1.

5.1.1.2 Módulo transmissor FM

O Módulo transmissor *FS1000A* é o responsável pela transmissão dos dados coletados pelo sensor de fluxo e manipulados pelo Arduino UNO. O módulo possui 3 pinos (Figura 6), sendo eles indicados como (1) *DATA*, (2) *VCC* e (3) *GND*, significando respectivamente, pino de transmissão, alimentação positiva de 5V a 12V e aterramento. O módulo opera com modulação *ASK*, ou seja, quando recebe tensão de 5V no pino de dados ele envia uma série de ondas eletromagnéticas pelo ar, ao cessar a alimentação nesse pino, ele para de transmitir. Essa operação *ON-OFF* é totalmente feita via *software*, basta o módulo estar conectado em algum pino digital do microcontrolador.

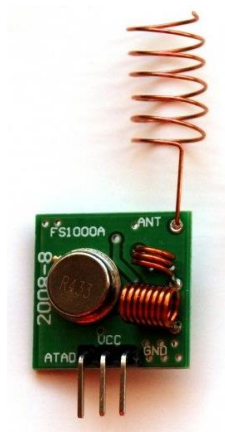


Figura 6: Transmissor FS1000A visto de cima

O módulo transmissor opera em um espectro eletromagnético de 433MHz, tornando-o um transmissor omnidirecional, ou seja, ele envia sinais eletromagnéticos em todas as 3 dimensões com o seu epicentro na antena. Em geral, esse tipo de transmissão costuma ser mais eficaz, ou seja, costuma a enviar sinais a distancias maiores, de acordo com a tensão aplicada nos pinos de alimentação *VCC*, podendo chegar até 200 metros em campo aberto quando acoplado uma antena eficaz.

Será utilizado no transmissor, uma antena dimensionada de 17.30cm utilizando um fio de cobre rígido 24awg como segue a Equação 2.

Equação 2: Dimensionamento de antenas de RF

$$\lambda \cdot f = c$$

A Equação 2 é conhecida mundialmente e imutável. Composta por λ sendo o comprimento da onda eletromagnética que queremos encontrar, f a frequência de operação de transmissão e c a velocidade da luz, uma constante que vele aproximadamente 300Kkm/s. Aplicando os valores de $f = 433.000.000 \text{ Hz}$ e $c = 299\,792\,458 \text{ m/s}$, obtemos o valor aproximado de 0,6923m. No entanto, o esperado de uma antena é que ela tenha pelo menos o

comprimento de múltiplos de $1/4$ de λ . Nesse caso $0,6923\text{m}/4 = 0,17309\text{m}$ ou 17,30cm. No entanto, visando melhor qualidade, iremos usar um múltiplo de $1/2$ de λ enrolado em forma helicoidal de 5mm de diâmetro interno e mantendo o passo o mais uniforme possível (HALLIDAY, David. RESNICK, 2008).

5.1.1.3 Sensor de fluxo

O sensor de fluxo *YF-S201*, possui um diâmetro interno de $1/2''$ e pode ser acoplado diretamente no encanamento, com o uso de luvas. O sensor ao girar a hélice interna com o movimento mecânico água, irá gerar pulsos magnéticos que através do efeito *hall*, geraram pulsos que poderão ser captados em forma de interrupção no microcontrolador.



Figura 7: Sensor de fluxo YF-S201

Na figura 7 podemos ver a estrutura interna do sensor e as respectivas pinagens, sendo elas (1) *GND*, (2) *VCC* e (3) *OUT*, responsáveis respectivamente pela alimentação negativa, positiva e pela saída dos pulsos. Os pulsos são gerados pelo pequeno ímã no eixo de rotação da turbina. Esse sensor pode ser visto sendo usado no artigo de C. Rajurkar usado em um projeto de *IoT* (*Internet of things*) no gerenciamento do consumo de água em residências (C. Rajurkar, 2017).

5.1.1.4 Fonte de energia

A fonte utilizada, será uma fonte chaveada com entrada 110 ~ 220v e saída de 9V, com capacidade de até 1A (Figura 8). A fonte será ligada diretamente na rede elétrica, de tal forma que o sistema consiga sempre estar ligado, evitando assim o contato com o dispositivo, para que não torne o método de ligar e desligar o sistema, algo cansativo e desencoraje o uso.

A ponteira da fonte de alimentação é do tipo P4, no entanto, haverá um estágio de adaptação para que o conector encaixe no Arduino NANO, uma vez que ele deverá ser ligado diretamente nos pinos de alimentação *Vin* na própria placa (*onboard*) e ser devidamente isolado, evitando riscos de choque elétricos por ser exposto em ambientes muito úmidos.



Figura 8: Fonte de alimentação 9V do transmissor

5.1.2 Receptor

Para o sistema de recepção dos dados vindos do transmissor, foram utilizados:

Tabela 2: Tabela de componentes do receptor

Nome do componente	Preço médio	Descrição	Modelo
Arduino UNO	R\$ 50,00	Microcontrolador	ATMega328p
Receptor RF 433Mhz	R\$ 5,00	Receptor dos sinais RF	MX_RM_05
Display LCD 16x2	R\$ 20,00	Interface supervisória	16x2 <i>backlight</i> verde
Led RGB	R\$ 1,00	Indicador luminoso	Anodo comum 5mm
Botões	R\$ 2,00	Configurações extra	#
Total	R\$ 78,00	#	#

5.1.2.1 Arduino UNO

Da mesma forma que o transmissor, o receptor usará um Arduino, porém dessa vez do modelo UNO, a pesar de ser o mesmo chip microcontrolador ATmega328P, a placa é um pouco diferente, como mostra a Figura 2.

O receptor usará um Arduino UNO para controlar os sinais recebidos pelo receptor de RF e processá-los, assegurando que os dados sejam recebidos de forma íntegra e assim, passando tais informações para a interface do *display* e o *led RGB*. O Arduino será o cérebro por trás de todo processo de controle e atuará como um agente regulador, sendo o responsável pela alimentação e manipulação de todo o sistema.

5.1.2.2 Receptor RF 433MHz

O receptor RF do modelo *MX_RM_05* será o responsável pela recepção dos quadros recebidos do transmissor. Da mesma forma que o transmissor possui uma antena de 17,3cm para enviar os quadros, o receptor também irá possuir uma antena para receber as ondas eletromagnéticas que carregam toda informação útil, evitando que pedaços se percam no caminho ou que não consigam ser captados. O controle de integridade dos quadros recebidos, porém, não será processado pelo módulo receptor, mas sim pelo microcontrolador.

Na Figura 9 vemos o receptor descrito acima, junto de sua antena.

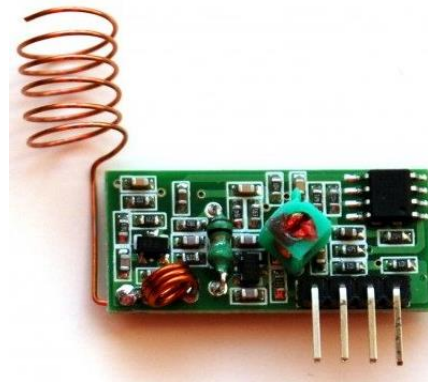


Figura 9: Módulo receptor MX_RM_05

A pesar de possuir 4 pinos, o receptor só utiliza 3 deles, uma vez que os pinos 2 e 3 estão interligados e representam os pinos de dados. Os pinos são (1) *GND*, (2,3) *Dados* e (4) *VCC* vistos da esquerda para a direita na Figura 9. Ao contrário do transmissor, o receptor só opera em faixas de tensão de 5V.

5.1.2.3 Display LCD 16x2

O *display* de 16 colunas e 2 linhas, funciona seguindo os protocolos *SPI(serial peripheral interface)* através de uma série de registradores internos que fazem todo o controle do que será impresso e onde dentre as colunas e linhas será impresso. Na Figura 4 vemos os pinos do *display* em um total de 16 pinos, no entanto, ao contrário do que pode se pensar, apenas 4 deles são pinos de dados. Na Tabela 3 podemos ver qual a pinagem, em ordem crescente, do *display* e para que servem.

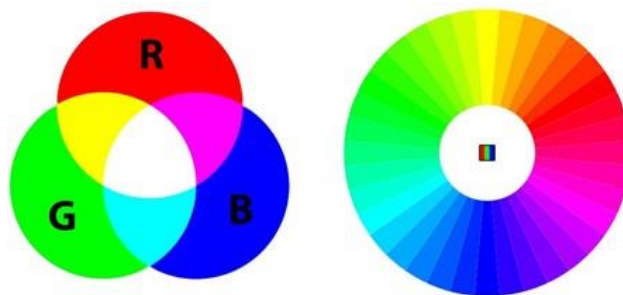
Tabela 3: Pinos do LCD

Pino LCD	Nome	Função
1	Vss	Alimentação positiva (5V)
2	Vdd	Alimentação negativa (<i>ground</i>)
3	V0	Ajuste de brilho do <i>display</i>
4	RS	<i>Reset</i>
5	R/W	<i>Read / Write</i>
6	EN	<i>Enable</i>
7 ~ 10	D0~D3	Pinos de dados para função <i>Read</i> (Não usados)
11~14	D4~D7	Pinos de dados para função <i>Write</i> (Usados)
15 e 16	Anodo e Catodo	Pinos de ajuste de LED <i>background</i> (Não usados)

Para o projeto, serão utilizados os pinos de dados *D4*, *D5*, *D6* e *D7*, para a função de *Write* no *display*, além dos pinos *RS* e *EN*, importantes para o controle de escrita na tela. Além dos pinos digitais, usaremos um pino analógico do microcontrolador para fazermos o ajuste de *V0*, sem o uso de um potenciômetro externo. *Vss* e *Vdd* serão conectados na alimentação do microcontrolador, sendo o positivo e o *ground*, respectivamente.

5.1.2.4 Led RGB e Botões

Será utilizado também um *led RGB* como forma de aviso luminoso ao usuário do dispositivo. O *led RGB* nada mais é que três *leds* em um só encapsulamento. Ele é formado por um *led* vermelho (R de *Red*), um verde (G de *Green*) e um azul (B de *Blue*) Figura 10. Por ser um *led* multicolorido, podemos variar sua cor e brilho, a fim de gerar efeitos de urgência e para chamarmos atenção ao dispositivo.

**Figura 10: Cores RGB (Red, Green, Blue)**

O *led* em questão é do tipo Catodo Comum, o que significa que ele possui seu pino de alimentação Negativo, comum aos demais *leds*. Portanto, para acendermos determinada cor,

precisamos enviar um sinal com tensão positiva para os respectivos pinos. Na Figura 11 vemos os dois tipos de *leds* disponíveis, sendo eles do tipo Catodo comum e Anodo comum, um é o oposto do outro.

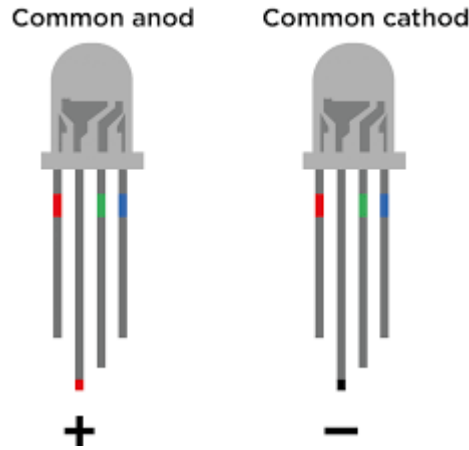


Figura 11: Anodo Comum e Catodo Comum

Para ligar o *led*, cada cor possui uma tensão de operação, sendo elas 2V para a cor vermelha, 3.2 para as cores Azul e Verde. Para usá-lo com o microcontrolador, faz-se necessário 4 pinos analógicos para que assim, consiga-se entregar as tensões necessárias para cada um dos pinos sem precisar fazer o uso de resistores.

5.1.3 Estimativa de manutenção

Como o dispositivo eletrônico vai estar instalado em um ambiente bastante úmido por natureza, é importante que medidas de combata ao desgaste desses dispositivos e falhas contra eventuais queimas prematuras. Um sistema de isolamento é essencial para que a umidade não entre em contato com os circuitos eletrônicos, tal isolamento pode ser feito com a criação de uma caixa de proteção que junte as partes sensíveis e deixem expostas somente as partes que não podem ser guardadas. Havendo um sistema de isolamento eficiente, a vida útil do sistema se estenderá por mais tempo, caso contrário, falhas devido a oxidações ou curtos-circuitos podem acontecer, desgastando as peças e assim, havendo necessidade de troca de equipamentos.

5.1.4 Conclusão de Hardware

Como o maior problema do dispositivo está na comunicação RF entre transmissor e receptor, é fundamental que estes estejam alinhados para que evite que quadros sejam perdidos no ambiente e assim, o sistema perca qualidade. Para isso foi feito um teste como mostra a Tabela 4. Esse teste visou avaliar a qualidade da transmissão e a taxa de perdas no caminho, avaliando quantos quadros foram enviados e quantos recebidos, alterando-se a distância em linha reta entre eles e após, adicionando obstáculos no caminho.

Tabela 4: Tabela de alcance do transmissor/receptor em 1Hz

Distância	Obstáculos	Quadros enviados	Quadros recebidos	Taxa de sucesso
1m	Não	60	60	100%
5m	Não	60	60	100%
10m	Não	60	60	100%
15m	Não	60	59	98%
20m	Não	60	30	50%
1m	Sim	60	60	100%
5m	Sim	60	59	98%
10m	Sim	60	28	46%
15m	Sim	60	0	0%
20m	Sim	60	0	0%

Pode-se perceber então, que o transmissor é eficaz para transmitir dados para distâncias de até 15 metros em linha reta, sem obstáculos. No entanto, adicionando paredes e mais elementos domésticos, o seu alcance cai para até 5m com eficiência, mas consegue captar até 10m, recebendo quase a metade dos quadros que são enviados.

Diante disso, com uma proporção de sucesso de quase 100%, tratando de ambientes abertos em distância de 10m e com obstáculos a 5m, considera-se que mesmo a nível de protótipo, com os contatos elétricos longe do ideal, o sistema se mostrou muito eficiente, portanto, será capaz de realizar as tarefas para o qual foi desenvolvido, uma vez que os dispositivos transmissores e receptores não irão operar em distancias maiores que 10m.

5.2 SOFTWARE

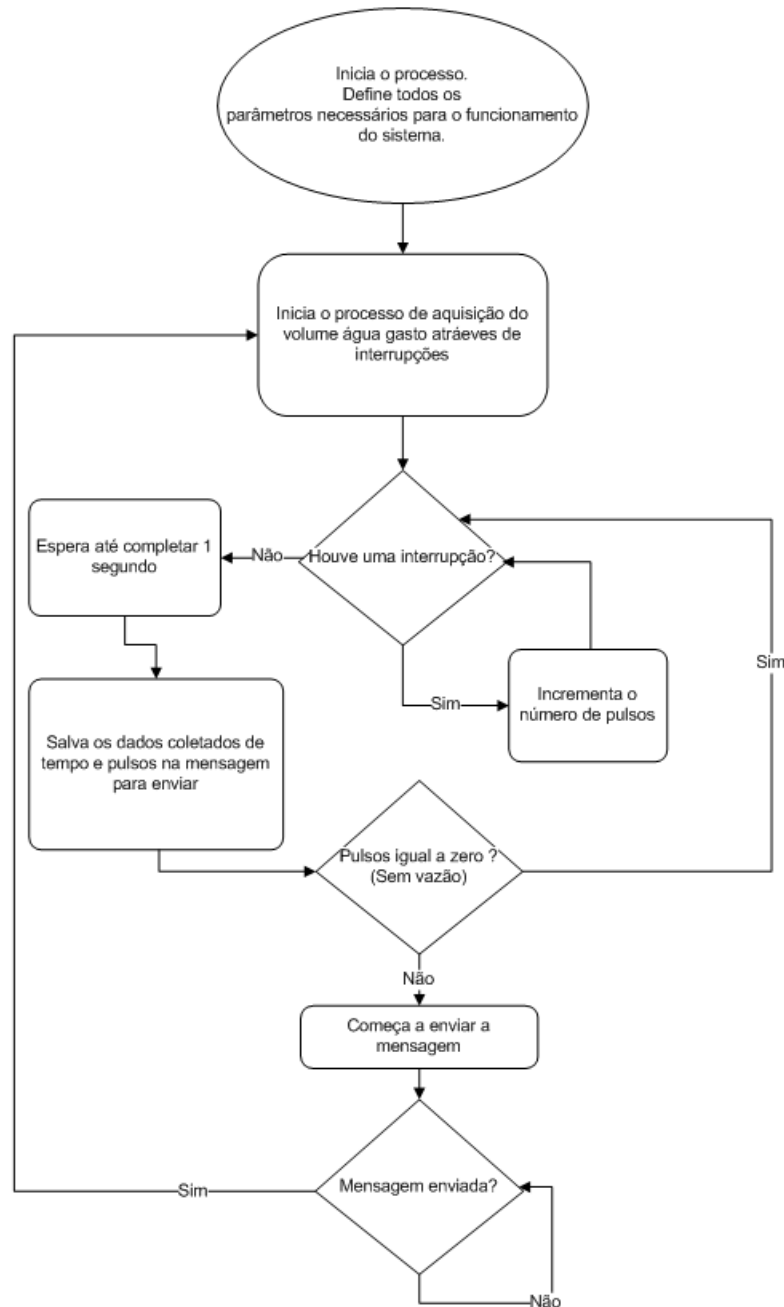
Para a análise de *Software*, irá ser destinado um tempo especial para olhar de perto, como cada um dos componentes do sistema Transmissor e Receptor irão trabalhar em conjunto, eletricamente, para que o sistema funcione plenamente. O cérebro do sistema serão os microcontroladores da família Arduino, que possuem a capacidade de serem programáveis e assim, adaptáveis, às diversas aplicações. O Arduino será tratado como unidade central e os sensores e atuadores, serão os periféricos do sistema.

Para a programação do sistema, a plataforma Arduino possui uma *IDE* próprio, um ambiente de desenvolvimento integrado criado para facilitar a programação das diversas plataformas de desenvolvimento, usando a linguagem de programação C/C++. Essa *IDE* pode ser instalada diretamente no seu computador pessoal ou então ser acessada diretamente no site da Arduino.cc de forma online (ARDUINO, 2020).

A programação Arduino é feita em alto nível se comparado à linguagem de máquina, a linguagem a qual o microprocessador irá interpretar, geralmente feita utilizando a linguagem de programação PASCAL. No entanto é vista como uma linguagem de baixo nível quando comparada as demais linguagens de programação que podem ser usadas, pois aproxima-se da máquina, podendo lidar diretamente com registradores e contadores do chip microprocessador.

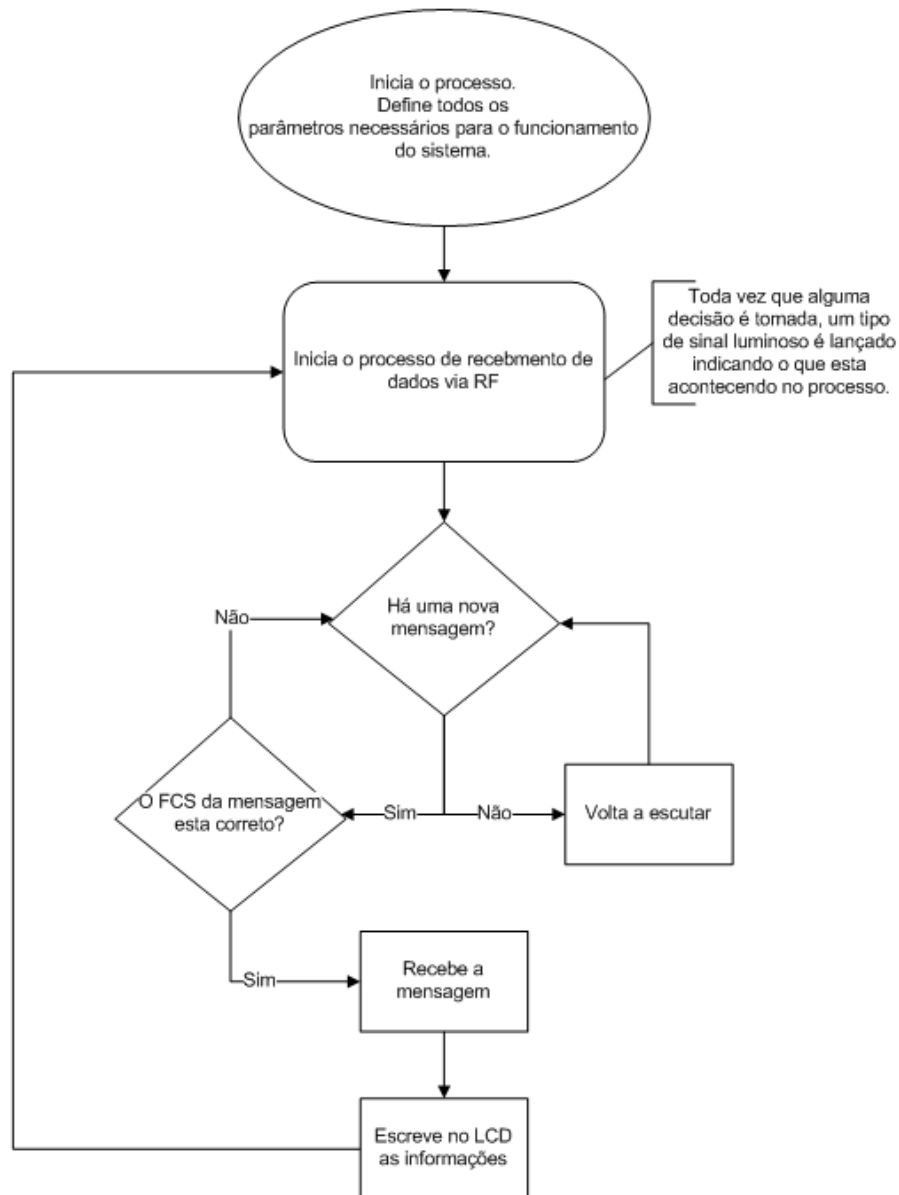
O código está escrito em linguagem C/C++ e pode ser encontrado, completo, no apêndice, no entanto, é necessário se ter um conhecimento básico de algoritmos de programação. Nesse tópico iremos abordar somente o fluxograma do sistema, de forma bastante simplificada, uma vez que o código completo está disponível ao final do documento e não é de interesse geral.

5.2.1 Fluxograma do dispositivo Transmissor



Vendo o fluxograma do dispositivo transmissor, percebe-se que o sistema é bastante simples, apenas faz a aquisição dos dados durante um segundo e logo após, os envia utilizando o módulo RF mencionado. Em poucas etapas, temos os dados de tempo e quantidade de pulsos dados pelo sensor de fluxo por segundo sendo enviados. Caso o número de pulsos seja igual a zero, significa que o sistema não está consumindo água e, portanto, não envia dado algum.

5.2.2 Fluxograma do dispositivo Receptor



Para a recepção dos quadros enviados, faz-se duas verificações. A primeira se há uma nova mensagem ou não, caso haja, fazemos a segunda verificação, caso não, fica escutando até encontrar uma nova mensagem. A segunda verificação acontece quando se encontra uma nova mensagem, assim, verifica-se a mensagem, se chegou íntegra ou não, verificando seu *FCS*, caso esteja, então recebe-se a mensagem e escreve no *display*. Após feito isso, volta a escutar novas mensagens.

Depois de um tempo predeterminado sem receber quadros novos, o sistema reinicia e entra em modo de economia de energia, para poupar bateria, uma vez que os banhos valem cerca de 2% do dia e, em todo resto do tempo, o dispositivo permanecerá sem utilidade.

5.2.3 Comentário acerca do software

Dessa forma, pode-se perceber com a ajuda dos fluxogramas, que diferente do imaginado, à primeira vista, a coleta e transmissão de dados obtidos dos sensores pode ser feita de uma forma muito simples e sem a necessidade de se ter um conhecimento avançado em programação. Esse é um fator importante para a réplica do projeto para os interessados e entusiastas em domótica e *IoT*.

Acrescenta-se também que o código fonte do projeto estará disponível de forma gratuita e aberta através da plataforma GitHub, no seguinte endereço Web: <https://github.com/iOsnaaente/Medicao-de-consumo-de-agua>.

5.3 Protótipo e projeto final

O projeto até este ponto foi desenvolvido como um protótipo, sendo assim, as imagens apresentadas durante o documento e as imagens do apêndice foram todas coletadas durante o processo de criação e desenvolvimento, sem levar em considerações aspectos importantes da aplicação real do projeto. Como já mencionado, ambos dispositivos estarão dispostos em ambientes agressivos para eletrônicos, desta forma, eles precisam de proteção contra umidade e agentes corrosivos, como resíduos de sabão e químicos comuns encontrados em banheiros.

Como a ideia do projeto é fazer um produto viável e funcional, foi projetado uma carcaça para o dispositivo receptor, que estará em contato direto com todos os agentes corrosivos citados. Ela foi idealizada, para ser impressa em material ABS com uma impressora 3D. Essa carcaça pode ser vista na Figura 12 abaixo.

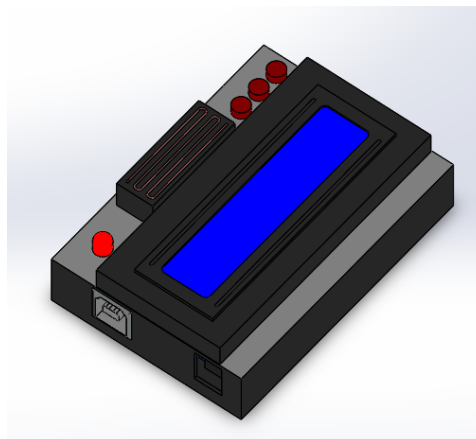


Figura 12: Modelagem 3D da carcaça do dispositivo Receptor

O modelo de carcaça foi desenvolvido para proteger a parte eletrônica, além da bateria que será usada juntamente com o dispositivo, dessa forma, o microcontrolador e todos os demais elementos eletrônicos estarão isolados do ambiente agressivo do banheiro, deixando apenas visível o *display* LCD, a antena e os botões. Na Figura 12 pode-se ver que ela foi projetada para alocar 3 botões, sendo cada um deles usado para uma das 3 funcionalidades de menu, já apresentadas no *software*, além de um *led RGB* e a antes acoplada na parte superior central.

Veja as Figuras 23 e 24 onde mostra, de outras vistas, a carcaça, no apêndice.

CONCLUSÃO

O desenvolvimento do projeto está a nível de protótipo. A montagem dos dois dispositivos (Transmissor e Receptor) foram feitas a nível de bancada e estão totalmente expostos como mostram a Figura 13 e Figura 14. Considerando o dispositivo como um produto final, ainda existem etapas adicionais para a implementação plena desse produto como a criação de uma placa de circuitos impressa com implementação de um microcontrolador *onboard* do tipo *standalone* e, redesenho da carcaça para se ajustar às mudanças.

O Arduino é um microcontrolador de código aberto (*opensource*), portanto, é comum a criação de placas pessoais e especializadas para serviços específicos. Para o dispositivo em questão, seria muito interessante a criação de uma placa própria, que cumpra as funções requisitadas e que tenham os módulos e dispositivos integrados em uma mesma placa. As vantagens de se fazer tal placa são várias, como, otimização de espaço físico, segurança de contatos elétricos, isolamento apropriado contra oxidações resultantes do ambiente úmido e facilidade de manutenção.

No entanto, se limitando a nível de protótipo, nosso projeto cumpriu bem os requisitos propostos por ele e mostrou bastante eficaz na aquisição dos dados de volume de água e comunicação a longas distâncias com boa eficiência e entrega integral de quadros, conseguindo comunicar o sistema em dois pontos distintos, tratando o sistema como um único sistema centralizado.

Além do mais, podemos ver que não houve um investimento astronômico para fazermos a aquisição de todos os componentes, se levando em conta os benefícios a longo prazo e pensando na economia de água e também elétrica. O gasto médio para se comprar e montar o

circuito receptor foi de R\$ 91 e para o circuito transmissor foi de R\$ 78, dando ao total um gasto de R\$ 169, sem levantar gastos para a impressão da carcaça. À primeira vista parece que há a necessidade de um investimento bastante gordo para algo que “não há necessidade”.

No entanto, quando analisamos a Tabela 5, usada para calcularmos o gasto de um chuveiro de potência média entre 5.5kWh e 7.5kWh e, presumindo que o tempo de banho de 3 casas com números de pessoas diferentes e tempos de banho diferentes, analisamos. Efeito da conta de luz em reais sem o uso do sistema e com o uso do sistema, presumindo que ele reduza o tempo de banho em 33% do tempo sem o sistema. O sistema se paga em questão de meses, considerando a tarifa de R\$ 0.56 a cada kWh que foi cobrada no mês de junho de 2020 na região centro-oeste do rio grande do sul - Brasil. (ANEEL, 2020).

Tabela 5: Gastos anuais e economia de luz com o dispositivo operando – Parte 1

Casa	Potencia kWh	Nº pessoas	Tempo banho	Tempo total da casa	Tempo com dispositivo	Custo kWh	Gasto kWk antes	Gasto kWk depois
Casa1	6.5	4	15	60	20	R\$0,56	6.50	2.15
Casa2		3	10	30	10		3.25	1.07
Casa3		2	10	20	6.5		2.17	0.72
Total		9		110	36.5		11.91	3.93

Tabela 6: Gastos anuais e economia de luz com o dispositivo operando - Parte 2

Diferença em kW	Valor Diferença em um dia	Em uma semana	Em um mês	Em um ano	Gasto no ano sem o sistema
4.35	R\$ 2,44	R\$ 17,07	R\$ 73,16	R\$ 890,15	R\$ 2.145,00
2.18	R\$ 1,22	R\$ 8,54	R\$ 36,60	R\$ 445,10	R\$ 1.072,50
1.45	R\$ 0,81	R\$ 5,70	R\$ 24,40	R\$ 296,70	R\$ 715,00
7.98	R\$ 4,47	R\$ 31,30	R\$ 134,10	R\$ 1.621,96	R\$ 3.932,50

Nas condições da hipótese levantada, teríamos uma economia de luz de aproximadamente R\$ 1.620 em um ano, somando as três casas, com diferentes números de pessoas e diferentes tempos de banho e potencias de chuveiro. Com essa economia, seria suficiente para criarmos mais 10 sistemas como esse e assim sucessivamente, gerando uma grande economia na conta de luz, ajudando no desperdício de água e ajudando o meio ambiente de uma forma voluntária.

Além da economia em energia elétrica, fator pelo qual o sistema funcionará, haverá a economia direta do volume em litros de água que são desperdiçados diariamente de forma inconsciente. Na mesma proporção que economizaríamos 30% na conta de luz no final do dia, teremos a mesma proporção na economia de água, sendo assim, em um banho de 10 minutos, que são gastos em média 100 litros de água, teríamos uma economia de 30 litros por banho no dia. Usando o total de 9 pessoas da Tabela 5 e 6, podemos assumir que haveria uma economia de 270 litros de água por dia, 8100 litros por semana e 97.200 litros por ano. Economia essa que se dará de forma espontânea, sem o pesar ou a dor de ter que reduzir seu tempo, prazeroso, de banho, já que o bolso pesa mais que a consciência em muitos casos.

Tendo atingido os objetivos visados no momento da criação do projeto, finalizo demonstrando extrema alegria com os resultados adquiridos e pensando no efeito positivo que poderia alcançar com difusão desse método de economia voluntário. O meio ambiente é um espaço dividido com todos nós e cuidar dele é um dever que temos com as gerações que virão depois de nós. Encerro com as Figuras 13 e 14, onde temos imagens do dispositivo em bancada, do transmissor e receptor.

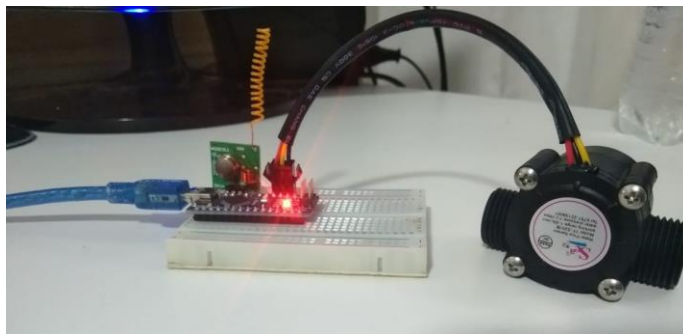


Figura 13: Protótipo dispositivo Transmissor

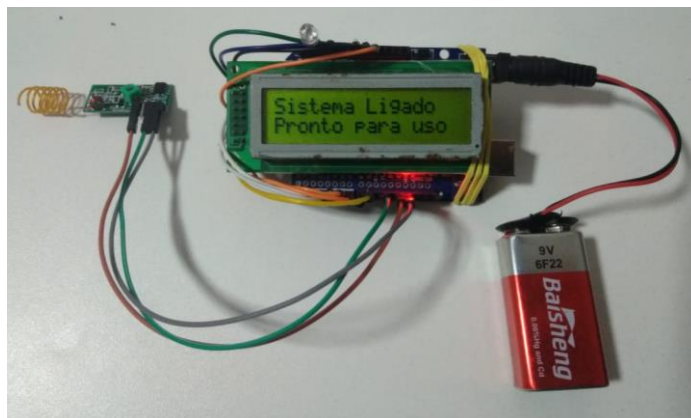


Figura 14: Protótipo dispositivo Receptor

BIBLIOGRAFIAS

- [1] ANGELO, Claudio; et al. **Governo do estado do Paraná: Uma previsão catastrófica marca o colapso da água no mundo para o ano 2025**, s.d. Disponível em: < http://www.diaadiaeducacao.pr.gov.br/portals/roteiropedagogico/publicacao/2519_A_era_da_falta_de_agua.pdf >. Acesso em: 15 de maio de 2020.
- [2] CARMO, Roberto Luiz do; DAGNINO, Ricardo de Sampaio; JOHANSEN, Igor Cavallini. **Transição demográfica e transição do consumo urbano de água no Brasil**. Rev. bras. estud. popul., São Paulo, v. 31, n. 1, p. 169-190, Jun 2014.
- [3] GONZÁLEZ, Gabriela. Blogthinkbig.com. **Por que Arduino é útil e o que é possível criar com ele?**. 5 de maio de 2016; Disponível em < <http://br.blogthinkbig.com/> >. Acesso em 22 de maio de 2020.
- [4] ARDUINO. **Página inicial**. Disponível em: < <https://www.arduino.cc/> >. Acesso em: 16 de maio de 2020.
- [5] ARDUINO REFERENCIAS. **Referencias**. Disponível em: < <https://www.arduino.cc/reference/pt/> >. Acesso em: 06 de jun de 2020.
- [6] ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **NBR 5410:2004 - Instalações elétricas de baixa tensão**. Rio de Janeiro, 2004, p. 170.
- [7] HALLIDAY, David. RESNICK, Robert e WALKER, Jearl. **Fundamentos de Física: Gravitação, Ondas e Termodinâmica**. vol.2: 8ª edição, 2008. p258 - A Velocidade de uma Onda Progressiva Eq. 16-13.
- [8] C. Rajurkar, S. R. S. Prabakaran and S. Muthulakshmi, **IoT based water management**, 2017 International Conference on Nextgen Electronic Technologies: Silicon to Software (ICNETS2), Chennai, 2017, pp. 255-259.

- [9] J. Lalnunthari and H. H. Thanga, **Dependence of hall effect flow sensor frequency on the attached inlet and outlet pipe size**, 2017 IEEE International Conference on Consumer Electronics-Asia (ICCE-Asia), Bangalore, 2017, pp. 56-60.
- [10] ANEEL, **Ranking de tarifas por região**. Disponível em: <<https://www.aneel.gov.br/ranking-das-tarifas>>. Acesso em 08 de jun de 2020.
- [11] DELFINO, Carlos. **HelloWorld - Documentação da biblioteca VirtualWire**; Disponível em: <<http://carlosdelfino.eti.br/helloworldarduino/VirtualWire/>> Acesso em 15 de jun de 2020.
- [12] PRAZNOWSKI, Krzysztof & AUGUSTYNOWICZ, Andrzej. **Wireless signal transmission for diagnostics of machines and devices**. AIP Conference Proceedings. 2029. Out de 2018.
- [13] PJRC, Eletronic Projects Components Available Worldwide. **LiquidCrystal Library**; Disponível em <https://www.pjrc.com/teensy/td_libs_LiquidCrystal.html> Acesso em 15 de jun de 2020.

6. APÊNDICE

6.1 Código do dispositivo Transmissor

A partir da Figura 15 até a Figura 18, teremos imagens do código usado no dispositivo transmissor. Ele está comentado em linha de código e pode ser visto abaixo:

```

1 // Para usar a transmissão RF 433Mhz
2 #include <VirtualWire.h>
3
4 const int transmit_pin = 11;
5 const int sensorFluxo = 2 ;
6
7 const int VCC = 10;
8 const int GND = 9 ;
9
10 char mensagem[4]; // UIN8_T (Pulsos, segundos, minutos, horas, controle)
11 void incPulso(); // Para sensor de fluxo - interrupções
12
13
14 // Variaveis para controle
15 unsigned long int Time2Reset = 0;
16 unsigned long int TempoCompensado = 0;
17 unsigned long int Tempo_1 = 0;
18
19 bool flagStopSend = false;
20 bool flag2reset = false;
21
22
23 // Para montar a mensagem
24 uint8_t contaPulso = 0;
25 uint8_t Seg = 0;
26 uint8_t Min = 0;
27 uint8_t Hor = 0;
28

```

Figura 15: Código Dispositivo Transmissor – Cabeçalho

```

30 void setup() {
31
32 // Para usar o fs1000a nas portas digitais
33 pinMode(VCC, OUTPUT);
34 pinMode(GND, OUTPUT);
35 digitalWrite(VCC, HIGH);
36 digitalWrite(GND, LOW);
37
38 //Iniciar as configurações de transmissão e recepção RF
39 vw_set_tx_pin(transmit_pin); // Pino de Transmissão RF
40 vw_setup(2000); // Bits por segundo
41
42 // Configurações do sensor de fluxo
43 // configuração do pino 2 (sinal do sensor de efeito hall, sensor de fluxo de água)
44 pinMode(sensorFluxo, INPUT_PULLUP);
45 //Configura o pino 2(Interrupção 0) interrupção
46 attachInterrupt(0, incPulso, RISING);
47 delay(100);
48 }

```

Figura 16: Código Dispositivo Transmissor - Função *setup*

```

51 void loop() {
52
53     contaPulso = 0; //Zera a variável
54
55     delay (1000 - TempoCompensado); // Aguarda 1 segundo - tempo de looping
56
57     // Variavel de sincronia
58     Tempo_1 = millis();
59
60     // Monta a mensagem para transmissão
61     mensagem[0] = contaPulso;
62     mensagem[1] = Seg;
63     mensagem[2] = Min;
64     mensagem[3] = Hor;
65
66     if (!flagStopSend){
67
68         // Média das leituras obtidas a cada 1 minuto
69         Seg++;
70         if (Seg == 60) {
71             Min++;
72             if (Min >= 60){
73                 Min = 0;
74                 Hor++;
75             }
76             Seg = 0;
77         }
78
79         // Envia a mensagem via RF (Não garante recebimento)
80         digitalWrite(LED_BUILTIN, HIGH);
81         vw_send((uint8_t *)mensagem, sizeof(mensagem));
82         vw_wait_tx();
83         for(int i=0; i<sizeof(mensagem); i++)
84             digitalWrite(LED_BUILTIN, LOW);
85     }
86
87     controle2reset();
88
89     TempoCompensado = millis() - Tempo_1;
90 }

```

Figura 17: Código Dispositivo Transmissor - Função *Loop*

```

105 // Veriifica se esta havendo fluxo e controla função reset
106 void controle2reset(){
107     if(!contaPulso){
108         flagStopSend = true;
109
110         if(!flag2reset){
111             Time2Reset = millis();
112             flag2reset = true;
113         }
114         if (millis() - Time2Reset > 6000) // 10 minutos
115             Seg = Min = Hor = 0;
116
117     }else{
118         flagStopSend = false;
119         flag2reset = false;
120         Time2Reset = 0;
121     }
122 }

```

Figura 18: Código Dispositivo Transmissor - Função Controle para resetar

6.2 Código do dispositivo Receptor

A partir da Figura 19 até a Figura 21, teremos imagens do código usado no dispositivo receptor. Ele está comentado em linha de código e pode ser visto abaixo:

```

1  // Para usar a transmissão RF 433Mhz
2  #include <VirtualWire.h>
3
4  // Para usar o Display LCD
5  #include <LiquidCrystal.h>
6
7  // Seta o número máximo de bytes para serem enviados
8  #define MAX_BUFF 10
9
10 // Tamanho do buffer
11 uint8_t buff_len = MAX_BUFF;
12 // Criação de um Buffer para receber as mensagens do transmissor
13 uint8_t buff[MAX_BUFF];
14
15 // Pino usado para o RX
16 const int receive_pin = 11;
17
18 // Variáveis de inatividade
19 unsigned long int time2reset = 0;
20 bool flag2reset = 0;
21
22 // Inicia a interface do display LCD com os pinos
23 const int RS=9, EN=8, D4=7, D5=6, D6=5, D7=4;
24 LiquidCrystal lcd(RS, EN, D4, D5, D6, D7);
25
26 // Para as indicações luminosas LED RGB - Catodo comum
27 const int VERM = A1; // Controle da luz vermelha
28 const int CONT = A2; // Controle Vdd
29 const int VERD = A3; // Controle da luz amarela
30 const int AZUL = A4; // Controle da luz verde
31
32 // Constantes para controle
33 uint8_t *pulsos = &buff[0];
34 uint8_t *Seg = &buff[1];
35 uint8_t *Min = &buff[2];
36 uint8_t *Hor = &buff[3];
37
38 // Botão de controle de menu
39 const int BOTA01 = 2;
40 const int BOTA02 = 3;
41 int menu = 0;
42
43 // Variáveis importantes para a medição
44 float vazao = 0.0;
45 float miliLitros = 0.0;
46 float Litros = 0.0;
47

```

Figura 19: Código Dispositivo Receptor – Cabeçalho

```

48 ▼ void setup() {
49
50     // Para ligar o Vcc do Receptor nos pinos 13
51     pinMode(13, OUTPUT);
52     digitalWrite(13, HIGH);
53
54     // Controle de luzes
55     pinMode(VERM, OUTPUT);
56     pinMode(VERD, OUTPUT);
57     pinMode(AZUL, OUTPUT);
58     pinMode(CONT, OUTPUT);
59
60     // Evitar queimar o LED
61     analogWrite(CONT, LOW);
62
63     // Potenciometro para o LCD
64     pinMode(A0, OUTPUT);
65     analogWrite(A0, 120);
66
67     // Setando as configurações do LCD * 16 colunas 2 linhas
68     lcd.begin(16, 2);
69
70     // Printando uma mensagem de boas vindas no LCD.
71     lcd.setCursor(0,0);
72     lcd.print("Sistema Ligado");
73     lcd.setCursor(0,1);
74     lcd.print("Pronto para uso");
75
76     // Define os botões com interrupção
77     // Controle de menu
78     pinMode(BOTA01, INPUT_PULLUP);
79     attachInterrupt(0, incMenu, FALLING);
80     // Reseta medição
81     pinMode(BOTA02, INPUT_PULLUP);
82     attachInterrupt(1, reset, FALLING);
83
84     //Iniciar as configurações de transmissão e recepção RF
85     vw_set_rx_pin(receive_pin); // Pino de Recepção RF
86     vw_setup(2000);             // Bits por segundo
87     vw_rx_start();              // Inicia a recepção de dados RX (Receive)
88
89     // Sinalização de inicialização
90     piscaCores(100);
91 }

```

Figura 20: Código Dispositivo Receptor - Função *setup*

```

94 void loop() {
95
96     // Espera nova mensagem por 1 segundo (1000 milissegundos);
97     vw_wait_rx_max(1000);
98
99     // Recebe a mensagem no Buff e o tamanho BuffLen
100     if(vw_have_message()){
101         if (vw_get_message(buff, &buff_len)){
102
103             vazao = *pulsos/7.5;
104             mililitros = vazao / 60;
105             Litros += mililitros;
106
107             if (menu == 0){
108                 lcd.clear();           // Limpa a tela antes de nova escrita
109                 lcd.setCursor(0, 0);   // Setar a posição de escrita no LCD
110                 lcd.print("L/min:");   // Printar a dimensão
111                 lcd.setCursor(12, 0);  // Seta a posição de escrita no LCD
112                 lcd.print(vazao);
113                 printTempo();
114             }
115             else if (menu == 1){
116                 lcd.clear();           // Limpa a tela antes de nova escrita
117                 lcd.setCursor(0, 0);   // Setar a posição de escrita no LCD
118                 lcd.print("Litros:");   // Printar a dimensão
119                 lcd.setCursor(12, 0);  // Seta a posição de escrita no LCD
120                 lcd.print(Litros);
121                 printTempo();
122             }
123             else{
124                 // Opção de menu 3 - Clear() -> Limpa a tela deixando-a
125                 // com o visor desligado
126                 lcd.clear();
127             }
128         }
129     }
130
131     else{
132         //RECEBENDO SINAL ERRADO, FAZER AS MEDIAS E CONTORNAR
133         piscadela(1,0,0,100);
134         *Seg>60 ? *Seg=0 : *Seg++;
135         printTempo();
136     }
137
138     flag2reset = false;
139     time2reset = 0;
140
141 }
142
143 else{
144     piscadela(1,0,1,100);
145     // Não esta recebendo sinal - sinaliza
146     if(!flag2reset)
147         time2reset = millis();
148     flag2reset = true;
149
150     // Se o tempo inativo for maior que 10 minutos - desliga
151     if(millis()-time2reset>60000)
152         lcd.clear();
153     Litros = 0;
154 }
155
156 delay(1);
157 }
158
159 /* Definem os 3 estados de menu
160 0 = Vazão [litros / minuto]
161 1 = Litros
162 2 = Menu desligado
163 */
164 void incMenu(){
165     menu == 2 ? menu = 0 : menu++ ;
166     delay(10);
167 }

```

Figura 21: Código receptor - função loop

A Figura 22 mostra um exemplo de mensagem recebida pelo dispositivo receptor, a nível de protótipo.

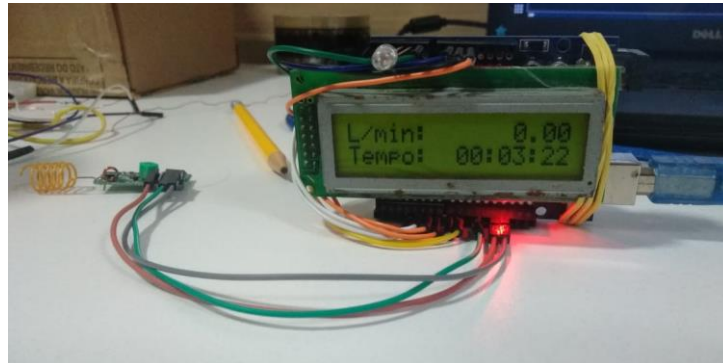


Figura 22: Mensagem do display

Os códigos completos podem ser acessados em: <https://github.com/iOsnaaente/Medicao-de-consumo-de-agua> e estão disponíveis para serem replicados e adaptados em novos projetos.

6.3 Modelagem 3D da carcaça do dispositivo Receptor

Nas Figuras 23 e 24 temos as vistas da modelagem da carcaça feita em *software* de modelagem 3D.

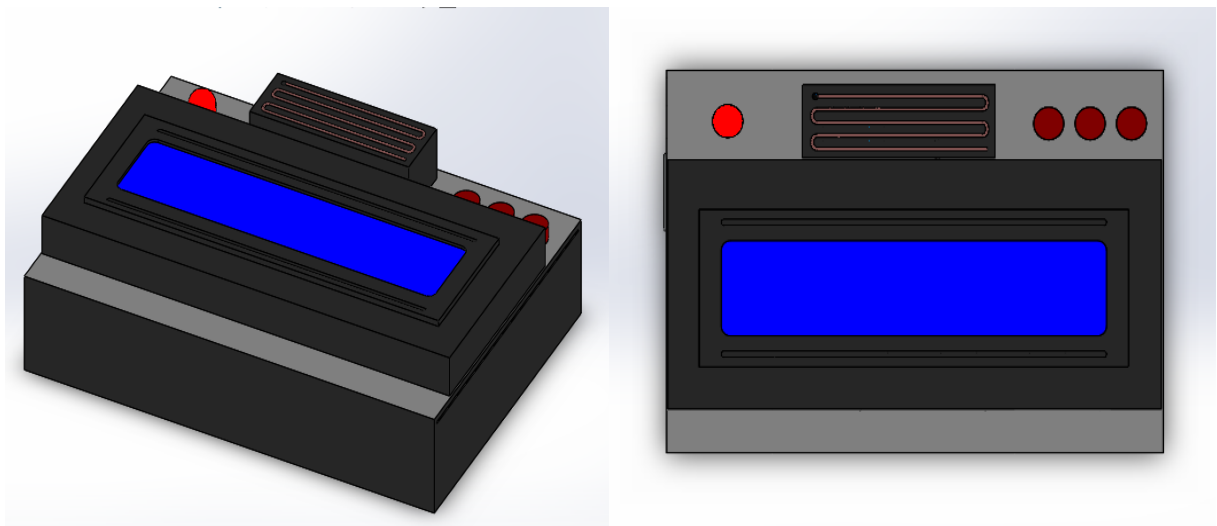


Figura 23: Modelagem - vista ortogonal e frontal

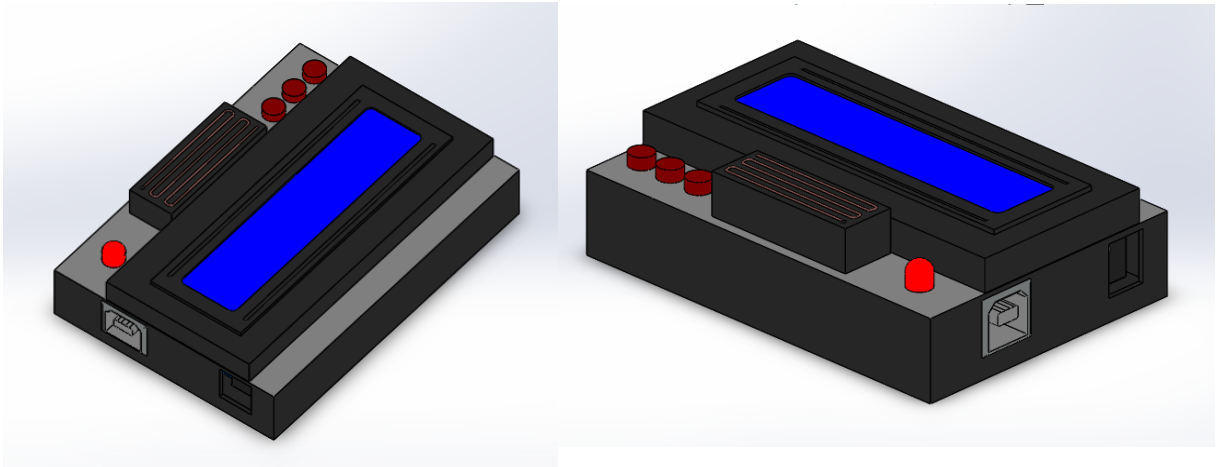


Figura 24: Modelagem - vista ortogonal esquerda e direita