



UNIVERSIDADE FEDERAL DO RIO DE JANEIRO
ESCOLA POLITÉCNICA
DEPARTAMENTO DE ENGENHARIA ELETRÔNICA E
DE COMPUTAÇÃO

Sistema de Monitoramento de Cisterna

Juan Lima

Lucas Rodrigues

Mateus Guina

12 de outubro de 2021

1 Introdução

Este relatório tem por intuito esclarecer ao leitor a motivação do desenvolvimento do protótipo Sistema de Monitoramento de Cisterna, apresentar as soluções encontradas e justificativas de implementações. Todo o desenvolvimento aqui relatado só foi possível devido a tecnologias de código aberto, o que possibilitou a customização e implementação do protótipo de maneira gratuita. O código fonte dessa aplicação pode ser encontrado no GitHub através do seguinte link: https://github.com/Coutjj/Projeto_Integrado.

Espera-se que, através do sistema de monitoramento idealizado, este projeto possa atender aos casos específicos em que não é possível realizar a automação de processos de abastecimento de cisterna através de boias. Além de fornecer meios para o acompanhamento do nível de água presente nos reservatórios.

2 Motivação

Algumas regiões do município de São Gonçalo enfrentam diversos problemas em relação ao abastecimento de água, o que força os moradores dessas localidades a utilizarem o sistema de cisternas. Além de não receberem o abastecimento diretamente em suas casas, o acesso à água, disponibilizado pela CEDAE (Companhia Estadual de Águas e Esgotos do Rio de Janeiro), é intermitente, não sendo possível determinar os instantes em que o recurso estará disponível.

Para vencer essa barreira e ter acesso à água, os moradores instalam, por conta própria, bombas hidráulicas nos pontos de fornecimento disponibilizados pela CEDAE. No entanto, dada a imprevisibilidade do abastecimento, esses dispositivos podem operar à seco, o que leva a queima dos mesmos. Para que a operação das bombas hidráulicas ocorra sem imprevistos, os moradores são obrigados a acompanhar todo o processo de utilização do equipamento e agir rapidamente caso a bomba opere sem água.

3 Objetivo

O objetivo deste projeto é desenvolver uma forma automatizada de acompanhar o processo de abastecimento de uma cisterna, disponibilizando ao usuário formas de obter informações como: nível atual de água e se a bomba está abastecendo ou não a cisterna. O sistema proposto irá avisar ao usuário sempre que um comportamento inesperado ocorrer,

como o caso em que a bomba está ligada, mas não está entrando água na cisterna. Com isso, esperamos poupar o usuário do ato de realizar checagens contantes durante o processo de abastecimento.

4 Metodologia

Este projeto será testado através de um protótipo, que tem por objetivo servir como prova de conceito da solução idealizada. Para desenvolver o dispositivo de monitoramento, faremos uso do Raspberry Pi 3B, um mini computador com excelente capacidade de processamento e que excede em muito as necessidades do projeto. Como o grupo já possui acesso a este mini computador, isso poupará custos e acelerará a prototipagem. O Raspberry monitorará eventos comentados a seguir e servirá como servidor para hospedar um bot, que interagirá com o usuário através do Telegram.

O nosso servidor e unidade de processamento receberá sinais de 4 sensores diferentes, sendo 3 deles sensores de nível, figura 1a , que indicarão a quantidade de água na cisterna, e um outro sensor, posicionado no cano de entrada da cisterna, que captará a entrada ou não de água neste reservatório. Ao contrário do sensor de nível, que está disponível comercialmente e pode ser facilmente encontrado, não foi possível comprar um sensor de detecção de água que fosse adequado às condições de desenvolvimento do projeto, com isso, desenvolvemos um sensor simples (figura 1b), que consiste em dois parafusos posicionados na saída do cano. Estando os parafusos sujeitos a uma diferença de potencial, espera-se que quando a água passe através deles haja a condução de uma corrente, que será detectada pelo Raspberry.



(a) Sensor de nível.

(b) Sensor de entrada de água.

Figura 1: Sensores utilizados no protótipo.

A escolha do Telegram como plataforma de contato com o usuário se deu em razão deste

mensageiro disponibilizar de forma gratuita uma API, que possibilita ao programador automatizar mensagens e até mesmo construir um programa que interaja com os inputs realizados pelo usuário na plataforma. O usuário realizará as operações de interação com o protótipo através de 3 comandos: “/status”, “/start” e “/stop”.

O comando “/status” possibilita ao usuário receber informações sobre o estado atual da cisterna. Através dele, como indicado na figura 2, obtemos informações sobre a porcentagem de água presente na cisterna, se a cisterna está ou não recebendo água e se o monitoramento do processo de abastecimento está ativado.

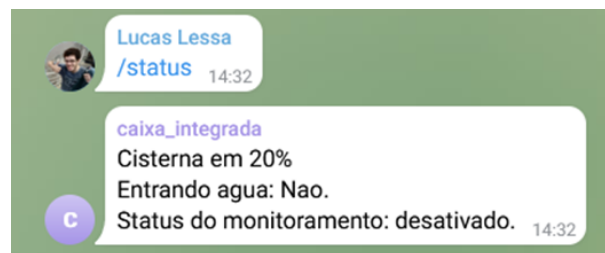


Figura 2: Comando “/status”.

A partir do comando “/start” é iniciado o processo de monitoramento da cisterna. Neste processo, uma rotina de checagem será estabelecida a cada 5s, verificando se ocorre o caso em que a bomba hidráulica esta ativada, mas não há entrada de água na cisterna. Caso esse evento ocorra, uma mensagem será enviada ao usuário, pedindo que ele execute a ação de desligar a bomba. Quando o usuário requisita o início do monitoramento, é enviada uma mensagem indicando que o processo ocorre normalmente e o estado atual da cisterna é atualizado, como pode ser verificado na figura 3 .

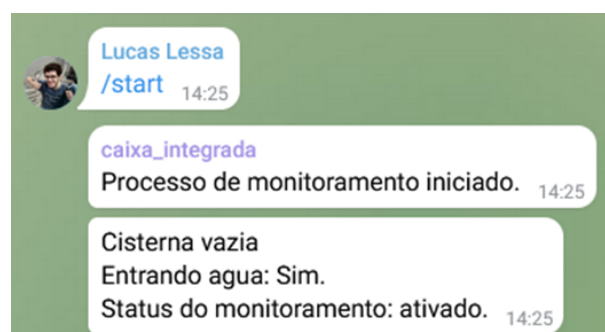


Figura 3: Comando “/start”.

A finalização do monitoramento é acionada a partir do comando “/stop”. Com isso, o sistema interrompe a rotina de checagem de entrada de água e avisa ao usuário que o monitoramento foi encerrado, figura 4 .



Figura 4: Comando “/stop”.

5 Construção do protótipo

De maneira geral, o diagrama do projeto está representado pela figura 5, onde temos representados a bomba (com uma distância de aproximadamente 130m da cisterna), os dois sensores conectados ao reservatório e o Raspberry que processa e envia os sinais provenientes dos sensores através do Telegram.

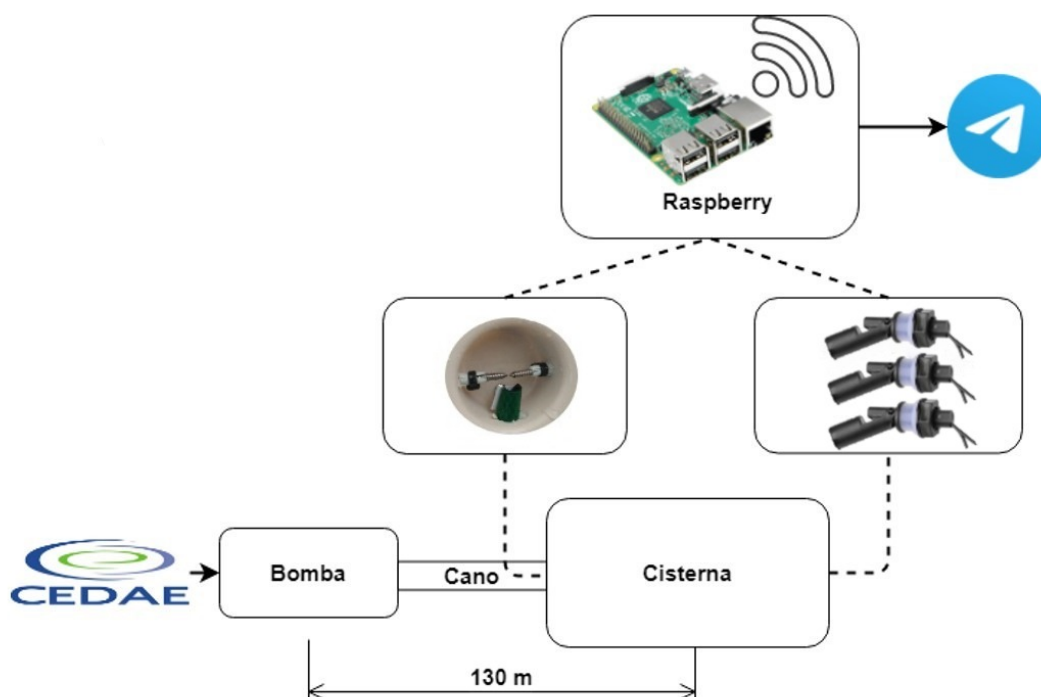


Figura 5: Diagrama geral.

Os sensores instalados na cisterna não apresentam o mesmo comportamento. O sensor de nível, figura 1a, possui um comportamento praticamente booleano, o que tornou o tratamento do sinal pelo Raspberry mais simples, através da função “DigitalInputDevice” (disponível na biblioteca gpiozero). Já para o sensor de presença de entrada de água, o tratamento do sinal de entrada não foi tão simples. Como nem sempre o fluxo de água é contante, a corrente que passa através dos parafusos oscilava, fazendo com que o sensor

fosse ativado e desativado por diversas vezes. Para resolver esse problema utilizamos a função `SmoothedInputDevice`, que pode realizar a média de um determinado número de amostras e assim verificar se o valor médio das amostras está acima de um nível que pode ser considerado como nível lógico alto.

Para receber o sinal proveniente do sensor de entrada de água, utilizamos uma porta OR do tipo CMOS. O ponto chave na hora de tratar esse sinal foi interpreta-lo como um divisor resistivo. Como a resistência apresentada pela água pode variar bastante, alguns valores de resistores foram testados, entre eles os valores de : $10\text{ K}\Omega$, $100\text{ K}\Omega$ e $1\text{ M}\Omega$. O resistor de $1\text{ M}\Omega$ foi o que apresentou o maior tensão na entrada da porta, resolvendo com folga o problema da variação de resistência da água. Na figura 6, temos um esquemático do circuito considerando apenas o sensor de água, a porta OR e a entrada do Raspberry.

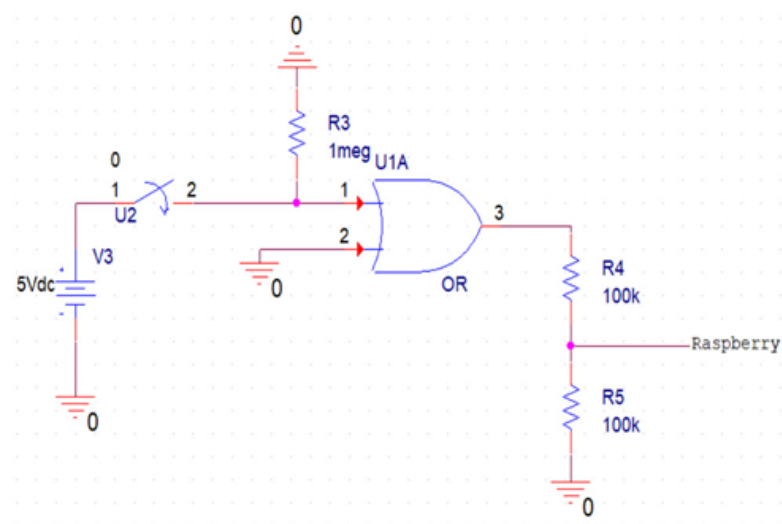


Figura 6: Esquemático do circuito para o sensor de presença de água.

A partir das considerações trazidas aqui, um protótipo do projeto foi construído, figura 7. Utilizamos um pequeno recipiente para simular a cisterna, uma torneira para fazer o papel da bomba hidráulica e com o auxílio de uma protoboard realizamos todas as conexões entre os sensores e o Raspberry.



Figura 7: Protótipo.

6 Justificativa

Este projeto se apresenta como auxílio ao caso específico de irregularidade de abastecimento de água enfrentado pelos moradores de São Gonçalo e tem por objetivo maior poupar os usuários da ação de constante checagem de operação de seus equipamentos de abastecimento de água.

Como uma primeira alternativa, foi idealizado que o sistema atual fosse hospedado em um formato de site, disponibilizando ao usuário blocos interativos para a realização de ações e visualização do status de operação. No entanto, chegou-se a conclusão de que o fluxo de informação não deveria ser esse. Como estamos tratando de um sistema de prevenção de danos, o usuário não deve ficar checando continuamente o site em busca de uma mensagem de falha e sim receber essa mensagem quando o evento ocorrer.

Um ponto a ser criticado no projeto é que o sistema não atua diretamente no acionamento e desligamento das bombas hidráulicas, mas isso se dá por uma barreira que excede as condições de desenvolvimento. A distância entre a cisterna e a bomba d'água é um dos principais fatores da não realização desta ação. Como podemos ver a partir da figura 8, a bomba d'água estaria posicionada a cerca de 130m dos sensores, em uma região ocupada e asfaltada. o que dificultaria a passagem de fios ou transmissão de RF. Além disso, a bomba fica posicionada em um local de acesso público, o que dificulta a instalação de equipamentos devido à ações de vandalismo e eventuais furtos, figura 9.



Figura 8: Distância entre a bomba e a central de monitoramento.



Figura 9: Ponto de operação da bomba hidráulica.

7 Conclusão

A construção e teste do protótipo ocorreram dentro dos limites de prazo estipulados e o objetivo geral, de criar um sistema capaz de auxiliar o usuário no processo de acompanhamento de abastecimento de uma cisterna, foi alcançado. Para alcançar o status de produto, é necessário dimensionar melhor o poder computacional necessário na atuação do servidor e central de monitoramento. Além disso, é preciso investigar a possibilidade de criar uma comunicação entre o Raspberry e a bomba hidráulica. Fornecendo assim, total conforto ao usuário, evitando que seja necessária a intervenção do mesmo para ativar e desativar a bomba.