



DESENVOLVIMENTO DE UM CONTROLADOR AUTOMÁTICO PARA SISTEMAS DE IRRIGAÇÃO POR GOTEJAMENTO

Eduardo Teles Dias Filgueira*¹, Antônio Marcos de Melo Medeiros¹, Bruno Quirino de Oliveira¹,
Fabrícia Neres Borges¹

¹PUC Goiás – Pontifícia Universidade Católica de Goiás

Resumo - Este trabalho teve como objetivo elaborar o protótipo de um sistema de irrigação por gotejamento, bem como projetar o controlador para automatizá-lo. O sistema de controle proposto utilizou a plataforma embarcada ESP32, a qual possui suporte à rede WiFi, para realizar a leitura dos sensores de umidade do solo, umidade do ar e temperatura. Para monitorar o sistema de irrigação construiu-se uma interface gráfica na plataforma Node-RED, ferramenta visual para conectar dispositivos de IoT (Internet das Coisas). A calibragem do sensor de umidade do solo indicou uma faixa de valores que, convertidos em porcentagem de umidade, variam de 30% com o sensor inserido em solo seco até 98%, em solo saturado. No software de controle, definiu-se que valores inferiores ou iguais a 48% denotariam solo seco e, de 49% até 93%, solo úmido. Após os testes, concluiu-se que, tanto o protótipo quanto o controlador funcionaram conforme o projetado.

Palavras-Chave - Automação, Controle, Gotejamento, Irrigação.

DEVELOPMENT OF AN AUTOMATIC CONTROLLER FOR DRIP IRRIGATION SYSTEMS

Abstract - This work aimed to develop the prototype of a drip irrigation system, as well as to design the controller to automate it. The proposed control system used the embedded platform ESP32, which has support for the WiFi network, to read the sensors for soil moisture, air humidity and temperature. To monitor the irrigation system, a graphical interface was built on the Node-RED platform, a visual tool for connecting IoT (Internet of Things) devices. The calibration of the soil moisture sensor indicated a range of values that, converted into percentage of moisture, vary from 30% with the sensor inserted in dry soil to 98%, in saturated soil. In the control software, it was defined that values less than or equal to 48% would denote dry soil and, from 49% to 93%, moist soil. After the tests, it was concluded that both the prototype and the controller worked as designed.

*eduardoteles999@gmail.com

Keywords - Automation, Control, Drip, Irrigation.

I. INTRODUÇÃO

A irrigação é uma forma de prover água de maneira artificial a uma determinada cultura, fazendo com que esta produza de forma adequada [1]. A definição moderna de irrigação não se restringe somente ao combate à seca, abrangendo um conceito mais amplo de agronegócio, no qual a irrigação é uma estratégia fundamental para elevar a produtividade agrícola de forma sustentável [2].

Existem diversos tipos de sistemas de irrigação, os quais podem ser categorizados da seguinte forma: irrigação por aspersão, irrigação localizada, irrigação por superfície e irrigação subterrânea [3]. Falando-se especificamente sobre o método localizado, este pode ainda ser dividido em duas técnicas: a irrigação localizada por microaspersão e por gotejamento.

A irrigação localizada é definida como a aplicação de água de forma pontual na região do entorno das raízes das plantas. A água é conduzida a baixa pressão por meio de tubulações e aplicada ao sistema radicular da cultura através de emissores (que variam conforme o tipo de irrigação) [4]. Esse tipo de irrigação é caracterizado como de alta frequência, uma vez que a umidade do solo é mantida próxima da capacidade de campo e os turnos de rega são curtos (geralmente de 1 a 4 dias) [4].

No Brasil, a agricultura é um dos motores da economia e a irrigação vem ganhando espaço ao longo das décadas, no contexto da agricultura de precisão. O país está entre os 10 com maior área equipada para irrigação no mundo, com cerca de 6,95 milhões de hectares [5]. Além disso, a atividade que mais consome recursos hídricos no Brasil é a irrigação, sendo responsável por 46% das retiradas em corpos hídricos e por 67% do consumo de água [5].

Para garantir a eficiência de um sistema de irrigação é necessário um uso racional e preciso de água e energia, bem como um manejo adequado do sistema, levando-se em consideração as necessidades da cultura em todas as fases de cultivo [2]. Acrescenta-se a isso a importância da preservação ambiental, com o objetivo de manter a integridade dos cursos d'água. Nesse cenário, a automação e o controle e monitoramento de variáveis podem contribuir não apenas com a automatização do ato de irrigar, mas também servir como ferramenta para facilitar e otimizar o manejo.

Com base nesses conceitos, este trabalho teve o objetivo de desenvolver um controlador para automatizar um sistema de irrigação por gotejamento. O controlador proposto utilizou o sistema embarcado ESP32 (o qual possui suporte à rede WiFi e Bluetooth embutidos), para realizar a leitura dos sensores de umidade do solo, umidade do ar e temperatura. Para monitorar o sistema de irrigação, foi projetada uma interface gráfica na plataforma Node-RED (ferramenta visual para conectar dispositivos de IoT).

II. FUNDAMENTAÇÃO TEÓRICA

A. Irrigação localizada

A irrigação localizada consiste em aplicar água na faixa de solo próxima do sistema radicular da planta [4]. Essa aplicação é feita a baixa pressão, porém em alta frequência, com o intuito de garantir maior eficiência e menor consumo de água, pois a área molhada é menor, o que reduz as perdas por evaporação [6].

Existem dois tipos de irrigação localizada: microaspersão e gotejamento. Ambos se diferenciam apenas quanto ao tipo de emissor utilizado. Em linhas gerais o sistema localizado é composto pelas seguintes partes componentes [4]:

- Conjunto motobomba, responsável por bombear a água da fonte de captação até a área a ser irrigada.
- Tubulações principais (geralmente de PVC – Policloreto de Vinila) e de derivação (polietileno ou PVC).
- Tubulações laterais, nas quais estão inseridos os emissores (gotejadores, microaspersores, difusores, microtubos tipo spaghetti), sendo que estes podem ser de encaixe manual (como os já citados) ou inseridos no momento da fabricação do tubo (no caso dos tubos gotejadores, mangueiras gotejadoras e mangueiras exudantes).
- Conexões, que são elementos responsáveis por unir mecanicamente as tubulações, sendo em geral peças de polietileno e PVC. A variedade de conexões permite que os tubos sejam instalados em diversos tipos de configurações, além de possibilitar a inclusão de uma ampla sorte de acessórios necessários ao sistema. Exemplos de conexões são as curvas, têes, adaptadores, uniões, redutores e conectores iniciais.
- Sistema de filtragem: elemento importante em uma irrigação localizada, pois como os emissores possuem um diâmetro pequeno, a entrada de impurezas na tubulação pode entupi-los. Os tipos mais comuns são os filtros de areia, filtro de tela, filtro de discos e os hidrociclones.
- Sistema de automação: inclui a parte de acionamento do motor da bomba, sensores (umidade do solo, umidade do ar, sensor de chuva e sensor de velocidade do vento), válvulas solenoides (que possibilitam a divisão da área irrigada em setores) e controladores (sistemas microprocessados que permitem controlar a irrigação de acordo com parâmetros preestabelecidos do sistema). A principal função da automação é reduzir a mão-de-obra e aumentar a eficiência.
- Sistema de fertirrigação: é um componente opcional que possibilita adicionar fertilizantes (e outros produtos

químicos) e dilui-los na água da irrigação. Essa é uma forma de otimizar o processo de fertilização e economizar mão-de-obra, pois o fertilizante diluído é conduzido até a raiz da planta.

III. MATERIAIS E MÉTODOS

A. Sistema embarcado ESP32

Neste trabalho foi usada a plataforma microcontrolada NodeMCU, modelo DevKit, de 38 pinos, equipada com o microcontrolador ESP32-WROOM32. [7] O NodeMCU ficou a cargo de receber a leitura dos sensores, executar o firmware de automação e acionar os atuadores. O microcontrolador ESP32 é amplamente utilizado em projetos de automação e IoT e possui como características destacadas a comunicação WiFi e Bluetooth integradas. [7]

B. Raspberry Pi 3

A Raspberry Pi 3 modelo B é um minicomputador bastante versátil, do qual destaca-se o processador quad core, memória RAM (Random Access Memory - Memória de Acesso Aleatório) de 1 GB e conexão à internet via cabo ou WiFi. [8] Neste projeto, a referida placa foi utilizada como um mini servidor de arquivos e para estabelecer comunicação entre a interface de usuário e a placa NodeMCU.

C. Protocolo MQTT

O MQTT (Message Queue Telemetry Transport) é um protocolo utilizado em comunicação de dispositivos IoT (Internet of Things). Possui como características principais a leveza e flexibilidade, além do fato de suportar comunicação assíncrona (o emissor e o receptor estão desacoplados, ou seja, supondo uma rede de sensores: os dispositivos enviam leituras e permitem que a rede descubra o melhor caminho e sincronização para entregar a informação aos dispositivos e serviços de destino) [9].

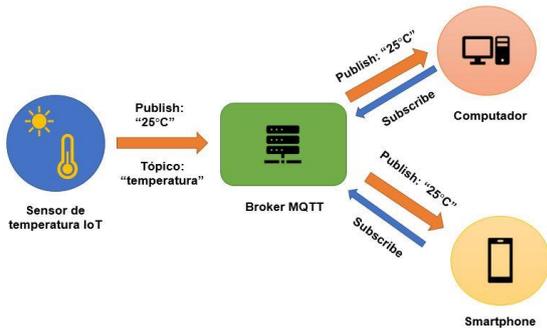
O MQTT utiliza um modelo de publicação/assinatura (publish/subscribe). Neste método dois tipos de entidades são definidos: o message broker e os clientes. [9] O broker é um servidor que recebe as mensagens de todos os clientes e roteia essas informações para os clientes de destino relevantes [9]. Um cliente é qualquer ente que possa se comunicar com o broker (por exemplo, um sensor IoT) [9].

O esquema de publish/subscribe funciona da seguinte forma:

- O cliente conecta-se ao broker, podendo assinar qualquer tópico (instância na qual informações podem ser publicadas e obtidas);
- O cliente então publica uma mensagem no tópico desejado, enviando ao servidor a mensagem e o tópico no qual o conteúdo deve ser publicado;
- O broker, dessa forma, encaminha a mensagem a todos os clientes assinantes do referido tópico.

As mensagens organizadas por tópicos permitem que se especifique quais clientes podem interagir com determinadas mensagens [9]. A figura 1 ilustra, de forma resumida, o modelo de publicação/assinatura do MQTT.

Figura 1. Modelo de publish/subscribe.



D. Node-RED

O Node-RED é uma ferramenta visual que possibilita a integração de dispositivos de IoT, hardwares, serviços online e APIs (sigla em inglês para Interface de Programação de Aplicações) [10].

Para se instalar a dependência Node-RED, que pode ser entendida como um pacote de módulos, é necessário que o Node.js já esteja instalado. Este último é, em linhas gerais, um node pré-compilado que permite a programação em Javascript. Todo esse procedimento pode ser executado em qualquer sistema operacional [10].

O Node-RED provê um ambiente de programação baseado em browser, podendo ser utilizado em qualquer tipo de navegador. Esse recurso permite que se crie lógicas de programação baseadas na interconexão de nós (nodes), que basicamente são blocos de código que implementam algum tipo de funcionalidade [10].

O Node-RED contém cerca de 40 nós, os quais proveem um amplo leque para as mais variadas utilidades. Além disso, é possível adicionar outros nós, oriundos de bibliotecas existentes nas comunidades, e até mesmo criar os próprios nodes [10].

E. Sensores, atuadores e componentes hidráulicos

Para monitorar as principais grandezas climáticas foi utilizado o sensor de temperatura e umidade do ar DHT22. A umidade do solo foi medida usando-se um sensor capacitivo e para detectar a presença de chuva, usou-se o módulo sensor de chuva yl-83. Na montagem do protótipo de irrigação foram usados os seguintes componentes, enumerados na figura 2.

Figura 2. Conexões hidráulicas em PVC.



Na figura 2 os números correspondem a um tipo de conexão específica, a saber: (1) 02 têes 1/2" (polegada) x 1/2" x 3/4" e 01 tê 1/2" x 1/2" x 1/2"; (2) 04 plugues roscáveis 1/2"; (3) 02 curvas 1/2" x 3/4"; (4) 04 conectores 1/2" roscáveis; (5) 03 adaptadores roscáveis 1/2"; (6) 05 adaptadores roscáveis 3/4";

(7) 04 conectores 1/2" com rosca fêmea; (8) 04 adaptadores 3/4" com rosca fêmea .

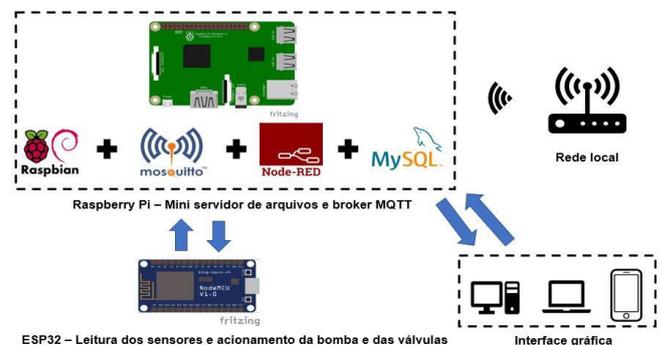
Todas essas conexões, juntamente com uma mangueira flexível de jardim de 1/2 polegada, formou o circuito hidráulico do sistema de irrigação, o qual foi distribuído em quatro setores. Na linha lateral de cada setor foram instalados gotejadores reguláveis. Para este projeto escolheu-se o gotejador Agrojet GA-2, que possui regulagem individual de vazão (1 a 20 l/h) e pressão de trabalho de 6 a 30 mca [11].

Para realizar a distribuição de água utilizou-se uma bomba de aquário 12V CC com motor brushless (sem escovas), 4,2 W e vazão de 240 l/h. Para comandar os setores de irrigação foram usadas 4 válvulas solenoides 12V, uma para cada setor. Para fazer a interface entre o circuito de controle e os atuadores foram utilizados dois módulos relé de 5V: um módulo de 01 canal para acionar a bomba e um módulo de 04 canais para comandar as válvulas.

IV. DESENVOLVIMENTO

A figura 3 retrata as partes componentes do controlador de irrigação, bem como as tecnologias usadas para concebê-lo. O sistema é composto por uma Raspberry Pi, na qual está instalado o broker MQTT, o Node-RED e o banco de dados MySQL. O ESP32 faz parte do circuito que recebe dados dos sensores e aciona os atuadores. O protocolo MQTT permite a comunicação entre esses dois dispositivos citados. Através de uma página web criada no Node-RED foi possível monitorar as variáveis coletadas e o estado dos atuadores do sistema de irrigação, bem como comandá-los.

Figura 3. Esquema de funcionamento do sistema.



Para implementar o broker MQTT foi utilizada a Raspberry Pi e nela instalou-se o sistema operacional Raspbian, que é uma distribuição Linux dedicada a esse tipo de sistema embarcado. Antes de proceder com a referida instalação é necessário que a Raspberry esteja provida de um cartão de memória (de preferência um do tipo classe 10, com no mínimo 4GB de armazenamento) para comportar o operacional, e alimentada por uma fonte 5V que forneça uma corrente mínima de 500 mA (recomendável: 2,5 A) [8].

Com o Raspbian pronto para uso, procedeu-se com a instalação dos demais softwares: Mosquitto™ (para criar o broker MQTT), Node-RED (ferramenta gráfica para elaboração da lógica de controle e criação da interface de usuário), e MySQL™ (software para criar e gerenciar sistemas de bancos de dados relacionais). Todos os componentes

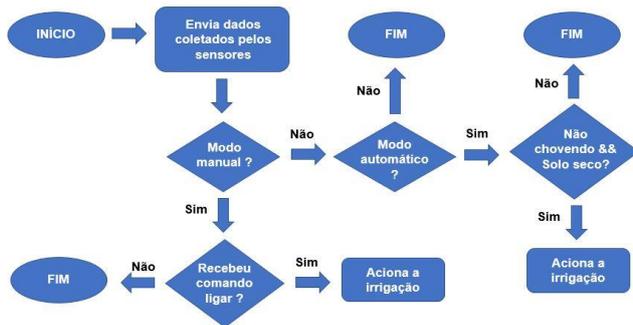
mostrados na figura 3 estão conectados à rede local e para o broker foi definido um endereço de IP (Internet Protocol) fixo.

A plataforma embarcada ESP32 é responsável por fazer a leitura dos sensores: umidade e temperatura do ar, umidade do solo e sensor de chuva. O microcontrolador envia os dados coletados para o broker MQTT e dele recebe os comandos para ligar ou desligar o sistema de irrigação (bomba e válvulas).

A lógica que diz respeito às tarefas executadas pela ESP32 foi implementada no IDE (Integrated Development Environment) do Arduino, sendo necessário a instalação de um plugin para carregar o programa para a placa NodeMCU e a adição de bibliotecas para realizar a comunicação via MQTT. A figura 4 representa, de forma resumida, o ciclo de operação do código executado pelo ESP32.

No código representado pela figura 4, o ESP32 se inscreve nos tópicos “temperatura”, “umidade”, “umidadeSolo” e “chuva” para receber as informações advindas dos sensores. O NodeMCU atua como Publisher nos tópicos “bomba”, “setor1”, “setor2”, “setor3” e “setor4” publicando os comandos para ligar/desligar a bomba e abrir/fechar as válvulas que comandam os setores 1, 2, 3 e 4.

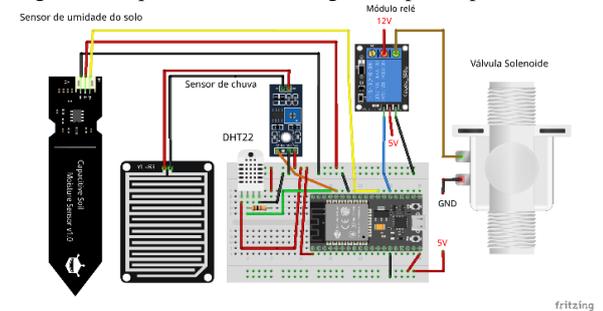
Figura 4. Fluxograma ESP32.



A lógica de controle foi construída com base em dois modos de operação: manual e automático. Na primeira forma, o usuário, por meio da interface gráfica, pode comandar o sistema, controlando o estado da bomba e das válvulas. No modo automático, o ESP32 recebe os valores referentes aos sensores de umidade do solo e sensor de chuva. Caso o dado obtido pelo tópico “chuva” esteja indicando um valor que denota ausência de precipitação e, ao mesmo tempo, a informação referente ao tópico “umidadeSolo” represente um valor classificado como solo seco, o NodeMCU liga a bomba e as válvulas.

Na opção automática, após a bomba ser ligada, foi programado o acionamento em sequência das válvulas, cada uma em um intervalo de um segundo após a outra. Quando o sistema detectar que há presença de chuva ou solo úmido, a bomba é desligada e, em seguida, cada uma das válvulas é fechada. A figura 5 ilustra o esquemático do circuito de acionamento do protótipo, mostrando as ligações dos sensores ao ESP32.

Figura 5. Esquemático da montagem do protótipo.



V. RESULTADOS

No ambiente de programação Node-RED, ao instalar a biblioteca node-red-dashboard, foi possível criar interfaces gráficas que podem rodar em qualquer tipo de navegador web. A figura 6 ilustra a interface criada para monitorar o sistema de irrigação.

Figura 6. Interface gráfica para controle e monitoramento.



A figura 6 retrata a interface de monitoramento e controle do protótipo, nela estão dispostos cinco quadros: “Temperatura”, “Umidade do ar”, “Umidade do solo”, “Status da irrigação” e “Acionamento”. O quadro “Temperatura” é composto por um mostrador, indicando o valor de temperatura em tempo real (em °C), e pelos indicadores das temperaturas mínima, máxima e média (em °C) que foram coletadas do banco de dados. O quadro “Umidade do ar” abriga o mostrador de umidade do ar (em %) e o indicador do sensor de chuva. O quadro “Umidade do solo” é composto pelo mostrador de umidade do solo, que indica o valor em percentual.

O quadro “Status da irrigação” mostra o estado da bomba e das válvulas, bem como se o modo automático está ativo ou não. Por fim, o quadro “Acionamento” é formado pelos botões de comando dos referidos atuadores e pelo botão de ligar/desligar o modo automático.

A figura 7 mostra a montagem do protótipo do sistema de irrigação. As quatro linhas laterais do sistema possuem dois metros de comprimento cada uma e ao longo delas foram inseridos os gotejadores reguláveis. Cada um dos quatro setores de irrigação é comandado por uma válvula solenoide 12 V

Figura 7. Protótipo do sistema de irrigação.



O sensor capacitivo foi calibrado antes de ser instalado em campo. Para fazer a calibração usou-se um recipiente de 300 ml de capacidade, contendo terra seca, e foi-se adicionando amostras de 10 ml de água até o solo atingir a saturação.

No processo de calibragem do sensor de umidade do solo utilizou-se o monitor serial do Arduino IDE para acompanhar os valores obtidos. O ESP32 possui 2 ADCs (Conversores Analógico-Digitais) de 12 bits cada, representando valores digitais de saída na faixa de 0 a 4095. Uma observação importante a ser feita é que, quando se usa a comunicação WiFi, apenas um módulo A/D pode ser usado (o ADC1, isso porque os canais do ADC2 são usados para alimentar o módulo WiFi) [7].

Após o processo de calibração, obteve-se as seguintes faixas de medição (representadas pelo valor digital apresentado no serial monitor): 3451 – 3429 com o sensor no ar, 3087 – 3055 no solo seco, 2147 – 2111 com 20 ml de água adicionada, 1917 – 1903 após adicionar mais 10 ml, 1791 – 1754 depois de inserir mais 10 ml (40 ml no total), 1721 – 1715 ao totalizar 50 ml, caracterizando solo saturado. Colocou-se mais 50 ml de água no recipiente (totalizando 100 ml), fazendo com que o volume de água ultrapassasse o de solo, e obteve-se um intervalo de leitura de 3451 – 1483, concluindo-se que, quanto mais úmido está o solo, menor é o valor representado no monitor serial.

Para mostrar o valor de umidade do solo em tempo real na interface de usuário, usou-se a função map do IDE Arduino para converter o intervalo 0 a 4095 do conversor A/D do

ESP32 em uma escala percentual de 1 a 100. A função map implementa a fórmula mostrada na equação 1 [12]:

$$res = (val - x_0) * \frac{(y - y_0)}{(x - x_0)} + y_0 \quad (1)$$

Onde:

val	-	Valor (representado na escala anterior) a ser convertido para a escala desejada.
y	-	Valor máximo da nova escala.
y ₀	-	Valor mínimo da nova escala.
x	-	Valor máximo da escala anterior.
x ₀	-	Valor mínimo da escala anterior.
res	-	Resultado da conversão.

Realizou-se um novo procedimento de teste com o sensor de umidade do solo, para se obter o intervalo de medição dessa vez em percentual de umidade. Os valores obtidos foram os seguintes: 30% a 48%, em toda a faixa de valores que denota solo seco; 49% a 93% com o solo úmido; 94% a 98% com o solo saturado. Com isso, conclui-se que o range de leitura do sensor varia de 30% (solo seco) a 98% (solo saturado). Na programação feita para controlar o sistema de irrigação, definiu-se que o sistema ligaria quando detectados valores inferiores ou iguais a 48% e desligaria quando se alcançasse 93% de umidade.

Com o protótipo de irrigação montado efetuou-se um teste para verificar o intervalo de tempo até se atingir o percentual

mínimo para acionar a irrigação, bem como o tempo até se alcançar o limite de umidade no qual o sistema é desligado. A figura 8 representa a variação do percentual de umidade do solo em relação ao tempo.

Figura 8. Umidade do solo x Tempo.



A medição durou cerca de 1 hora, o sistema manteve-se ligado por aproximadamente 10 minutos até atingir o limite máximo de umidade definido (93%), quando a irrigação desligou. Após isso a umidade do solo manteve-se praticamente constante até o fim do experimento, estabilizando-se em cerca de 88%.

VI. CONCLUSÕES

O desenvolvimento do controlador automático, bem como do protótipo do sistema de irrigação por gotejamento funcionou adequadamente, conforme projetado. A interface de usuário mostrou os dados de forma correta, permitindo visualizar em tempo real os dados coletados pelos sensores e acionar remotamente a irrigação. A calibração do sensor de umidade do solo permitiu que se estabelecesse uma faixa de medição que, convertida em percentual de umidade, varia de 30% a 98%. O teste do protótipo permitiu acompanhar a variação de umidade do solo em relação ao tempo, além de verificar que o sistema esteve apto a irrigar um pequeno jardim residencial de forma autônoma.

O sistema embarcado ESP32 possui como vantagem o módulo WiFi embutido, permitindo comunicação remota com outros dispositivos conectados na rede. Contudo, possui como limitação o fato de se poder usar somente um dos módulos de conversão A/D quando o sistema WiFi está ativo. O protocolo MQTT apresentou-se bastante eficaz na comunicação e transmissão dos dados coletados pelos sensores, com o broker implementado na Raspberry Pi. O ambiente Node-RED cumpriu importante papel na confecção da interface de usuário e na programação das funcionalidades do sistema, apresentando facilidade de uso, flexibilidade e integração.

Como proposta para futuros trabalhos, tem-se a inclusão de um pluviômetro e a adaptação do controlador para operar em lâminas de irrigação variáveis. Isso pode aumentar consideravelmente o investimento inicial, já que o pluviômetro é mais caro do que os sensores simples usados neste projeto, entretanto com ele pode-se contabilizar o quanto choveu e não apenas a presença de chuva, possibilitando o ajuste da lâmina de irrigação.

A adição de mais sensores de umidade do solo ao longo de cada linha lateral é um melhoramento que possibilita monitorar o balanço hídrico em cada setor, contribuindo para uma automação mais eficiente. Além disso, sugere-se também a instalação de um módulo fotovoltaico para geração de energia, com a finalidade de alimentar o sistema, fazendo com que este opere de forma autossuficiente.

REFERÊNCIAS

- [1] Braga, M. B.; Calgaro, M. “*Sistema de produção de melancia - Irrigação*”. Embrapa Semiárido - Sistemas de Produção, 6. ISSN 1807-0027. Versão Eletrônica. Ago/2010. Acedido em 25 de setembro de 2019, em: <https://sistemasdeproducao.cnptia.embrapa.br/FontesHT/ML/Melancia/SistemaProducaoMelancia/irrigacao.htm>
- [2] Bernardo, S.; Soares, A. A.; Mantovani, E. C. “*Manual de Irrigação*”. Editora UFV, 8ª ed, Viçosa – MG, 2008. 625 p.
- [3] Frizzone, J. A. “*Os métodos de irrigação*”. Notas de Aula da Disciplina LEB 1571 - Irrigação. ESALQ/USP, 2017. Departamento de Engenharia de Biosistemas - Curso de Engenharia Agrônoma, Piracicaba, SP.
- [4] Biscaro, G. A.; Gomes, E. P.; Cruz, R. L.; Geisenhoff, L. O.; Oliveira, A. C. “*Sistemas de Irrigação localizada*”. Editora UFGD, Dourados - MS, 2014. 256 p.
- [5] Agência Nacional de Águas (Brasil). “*Atlas Irrigação: uso da água na agricultura irrigada*”. ISBN 978-85-8210-051-6. Brasília - DF: ANA, 2017. 86 p.
- [6] Esteves, B. S.; Silva, D. G.; Paes, H. M. F.; Sousa, E. F. “*Irrigação por gotejamento*”. Manual Técnico, 32. ISSN 1983 - 5671. Niterói - RJ: Programa Rio Rural, 2012. 18 p.
- [7] Curto Circuito (2018). “*Conhecendo o ESP32*”. Acedido em 20 de maio de 2020, em: <https://www.curtocircuito.com.br/blog/conhecendo-esp32>
- [8] Halfacree, G. “*The Official Raspberry Pi Beginner's Guide – How to use your new computer*”. 1ª ed. Cambridge: Raspberry Pi Press, 2018. 241 p.
- [9] Yuan, M. “*Conhecendo o MQTT*”. IBM Developer. [S. I.]. 2017. Acedido em 01 de abril de 2020, em: <https://www.ibm.com/developerworks/br/library/iot-mqtt-why-good-for-iot/index.html>
- [10] Basílio, S. “*O que é Node-RED? Conhecendo e instalando*”. Blog MasterWalker Shop. [S. I.]. [2018?]. Acedido em 11 de abril de 2020, em: <https://blogmasterwalkershop.com.br/outros/o-que-e-node-red-conhecendo-e-instalando/>
- [11] Agrojet (2018). “*Gotejador regulável GA-2*”. Acedido em 30 de abril de 2020, em: <https://www.agrojet.com.br/produto/gotejador-regulavel-ga-2/>
- [12] Arduino (2018). “*Language Reference – map()*”. Acedido em 20 de maio de 2020, em: <https://www.arduino.cc/reference/en/language/functions/math/map/>