



PROGRAMA DE RESIDÊNCIA TECNOLÓGICA EM SISTEMAS EMBARCADOS
CURSO DE FORMAÇÃO BÁSICA EM SOFTWARE EMBARCADO

EDNA RODRIGUES DE SOUZA

SISTEMA DE ILUMINAÇÃO E IRRIGAÇÃO PARA CULTIVO INTERIOR DE
PLANTAS

JUAZEIRO - BA

2025

PROGRAMA DE RESIDÊNCIA TECNOLÓGICA EM SISTEMAS EMBARCADOS
CURSO DE FORMAÇÃO BÁSICA EM SOFTWARE EMBARCADO

EDNA RODRIGUES DE SOUZA

SISTEMA DE ILUMINAÇÃO E IRRIGAÇÃO PARA CULTIVO INTERIOR DE PLANTAS

Projeto de Pesquisa apresentado à Residência Tecnológica Em Sistemas Embarcados, EmbarcaTech, Polo Juazeiro, como requisito para obtenção de nota parcial no curso de Formação Básica em Software Embarcado.

JUAZEIRO - BA

2025

RESUMO

Este projeto apresenta o desenvolvimento de um sistema automatizado de irrigação e iluminação para cultivo de plantas, utilizando um Raspberry Pi Pico W como unidade central de controle. O sistema monitora a umidade do solo e a intensidade luminosa do ambiente por meio de sensores, acionando automaticamente uma bomba d'água e uma matriz de LEDs RGB conforme a necessidade da planta.

A automação permite a otimização do uso da água e da luz, garantindo que a irrigação ocorra apenas quando necessário e que a iluminação artificial se ajuste dinamicamente de acordo com as condições ambientais. Além disso, o sistema conta com uma interface interativa, composta por um display OLED para exibição de dados em tempo real e push-buttons para ajuste manual dos parâmetros de irrigação. Um buzzer passivo gera alertas para indicar baixo nível de água no reservatório.

Os principais diferenciais deste projeto incluem a integração multissensorial, o controle inteligente da iluminação e a arquitetura compacta baseada no Raspberry Pi Pico W, resultando em uma solução sustentável e eficiente. Com essa abordagem, busca-se reduzir desperdícios de recursos, facilitar o cuidado com plantas em ambientes internos e externos e incentivar práticas de cultivo mais ecológicas e acessíveis.

Palavras-chave: Iluminação Artificial; Irrigação Automática; Sistema Embarcado Inteligente.

SUMÁRIO

1 INTRODUÇÃO.....	5
1.1 OBJETIVOS DO PROJETO.....	5
1.1.1 Objetivos Gerais.....	5
1.1.2 Objetivos Específicos.....	6
1.2 DESCRIÇÃO DO FUNCIONAMENTO.....	6
1.4 JUSTIFICATIVA.....	7
1.5 ORIGINALIDADE.....	8
2 ESPECIFICAÇÃO DO HARDWARE.....	9
2.1 DIAGRAMA EM BLOCO.....	9
2.2 FUNÇÃO E CONFIGURAÇÃO DE CADA BLOCO.....	10
2.2.1 Microcontrolador (RP2040).....	10
2.2.2 Sensor de Luminosidade (LDR).....	10
2.2.3 Sensor de Umidade do Solo (Higrômetro).....	10
2.2.4 Módulo Relé de 1 Canal.....	10
2.2.5 Buzzer Passivo (MLT8530).....	10
2.2.6 Display OLED 128x64 (SSD1306).....	11
2.2.7 Botões (Push Buttons - Esquerdo, Direito e de Confirmação).....	11
2.2.8 Matriz de LEDs RGB (WS2812 5x5).....	11
2.3.1 Portas de Uso Geral de Entrada e Saída (Periférico GPIO).....	12
2.3.2 Comunicação via Protocolo de Tempo Preciso (Periférico PIO).....	12
2.3.2 Comunicação via Protocolo I2C (Periférico I2C).....	12
2.3.3 Geração de Sinal com Frequências Específicas (Periférico PWM).....	13
2.3.4 Leitura de Informações por Conversão Analógica Digital (Periférico ADC).....	13
2.4 DESCRIÇÃO DA PINAGEM USADA.....	13
2.5 CIRCUITO COMPLETO DO HARDWARE.....	14
2.5.1 Circuito Equivalente no Wokwi.....	14
2.5.2 Componentes Equivalentes.....	14
3 ESPECIFICAÇÃO DO FIRMWARE.....	16
3.1 BLOCOS FUNCIONAIS.....	16
3.2 DESCRIÇÃO DAS FUNCIONALIDADES.....	16
Aqui está uma explicação detalhada de todas as funções do seu código:.....	16
3.2.1 Funções de Inicialização: void setup().....	16
3.3 DEFINIÇÃO DAS VARIÁVEIS.....	16
3.3.1 Variáveis de Configuração de Hardware.....	16
3.3.2 Variáveis de Comunicação I2C.....	17
3.3.3 Variáveis de Controle para Exibição no Display.....	17
3.3.4 Variáveis de Controle dos Botões.....	17
3.3.5 Variáveis de Controle do Alerta.....	18
3.3.6 Variáveis de Controle da Conversão ADC.....	18
3.3.7 Variáveis Globais para a Matriz de LEDs.....	18
3.3.8 Variáveis para Debounce dos Botões.....	18

3.4 FLUXOGRAMA.....	20
4 METODOLOGIA.....	21
5 TESTES DE VALIDAÇÃO.....	23
6 DISCUSSÃO DOS RESULTADOS.....	23
7 CONCLUSÃO.....	23
REFERÊNCIAS.....	24
ANEXOS.....	25
Repositório no GitHub.....	25
Vídeo de Demonstração.....	25

1 INTRODUÇÃO

Cuidar de múltiplas plantas em uma casa de médio a grande porte é uma tarefa que demanda atenção, organização e, muitas vezes, conhecimentos específicos. As plantas podem estar espalhadas por diversos ambientes, como salas, varandas, quintais ou jardins verticais, e cada uma possui necessidades únicas de água, luz e temperatura. Enquanto algumas espécies requerem rega frequente, outras são mais sensíveis ao excesso de umidade, que pode causar apodrecimento das raízes e comprometimento do seu desenvolvimento. Por outro lado, a falta de água pode resultar no ressecamento, ou queda, das folhas e até mesmo na morte da planta. O desafio torna-se ainda maior quando o proprietário precisa lidar com uma rotina agitada ou períodos de ausência, como viagens ou longos dias de trabalho. Nesse contexto, o cuidado com as plantas é frequentemente negligenciado ou deixado nas mãos de terceiros que podem não ter o mesmo nível de comprometimento ou conhecimento necessário. Soluções improvisadas, como regar todas as plantas de uma vez ou confiar apenas na intuição, frequentemente resultam em um desequilíbrio, causando problemas como desperdício de água ou negligência de plantas específicas.

Além da rega, outros fatores críticos precisam ser monitorados, como a exposição à luz e as condições de temperatura. Em ambientes internos, a luz natural pode variar significativamente ao longo do dia, e nem sempre todas as plantas recebem a iluminação necessária. Já em ambientes externos, condições climáticas imprevisíveis, como chuva intensa ou sol em excesso, podem impactar negativamente o equilíbrio hídrico e térmico das plantas.

Portanto, sem um sistema automatizado ou ferramentas que auxiliem no cuidado eficiente, o proprietário enfrenta um processo trabalhoso, muitas vezes sujeito a erros. Uma solução que integre tecnologia ao cuidado diário das plantas pode otimizar esse processo, garantindo o desenvolvimento saudável das espécies e reduzindo desperdícios de recursos.

1.1 OBJETIVOS DO PROJETO

1.1.1 Objetivos Gerais

- Automatizar a irrigação de plantas, otimizando o uso da água;
- Regular a iluminação artificial, permitindo níveis ideais de luz durante o dia;
- Fornecer alertas inteligentes, para manutenção do sistema de irrigação;

- Facilitar a interação do sistema com o usuário, tornando o cultivo mais eficiente e sustentável.

1.1.2 Objetivos Específicos

- Monitorar a umidade do solo com um sensor e acionar uma bomba d'água apenas quando necessário, evitando desperdícios e garantindo umidade ideal;
- Reduzir o consumo excessivo de água por meio de um controle eficiente da irrigação, contribuindo para a sustentabilidade;
- Controlar a intensidade da matriz de LEDs RGB com base na leitura do sensor LDR, garantindo iluminação adequada independentemente da estação ou acesso a iluminação externa pela planta;
- Utilizar um buzzer passivo para notificar o usuário sobre baixo nível de água no reservatório;
- Exibir informações em tempo real no display OLED SSD1306, como iluminação, umidade e parâmetros de irrigação, permitindo ajustes manuais via push-buttons;
- Automatizar o manejo de água e luz para reduzir desperdícios e otimizar o crescimento das plantas, promovendo práticas de cultivo mais ecológicas e acessíveis.

1.2 DESCRIÇÃO DO FUNCIONAMENTO

Este projeto propõe um sistema automatizado de irrigação e iluminação para plantas, utilizando um Raspberry Pi Pico W (RP2040) como unidade central de processamento. O sistema monitora as condições ambientais e realiza ajustes automáticos para garantir um ambiente adequado para o desenvolvimento das plantas. Tem como principais funcionalidades:

- **Monitoramento e Acionamento de Iluminação:** Um sensor capta a intensidade luminosa do ambiente, seu valor lido é convertido pelo conversor analógico-digital (ADC) do RP2040 e processado para controle de uma matriz de LEDs RGB. A matriz fornecerá iluminação artificial adequada, garantindo um ambiente controlado independente da estação do ano ou acesso a iluminação externa. A intensidade da iluminação é ajustada com base na leitura do sensor.

- **Monitoramento da Umidade do Solo:** Um sensor mede a umidade do substrato do solo, seu valor analógico é lido e convertido pelo RP2040, permitindo o acionamento da bomba de água conforme a necessidade da planta. A bomba será acionada por relé, que entra em funcionamento sempre que a umidade do solo cai abaixo do nível configurado. O sistema interrompe a irrigação após um intervalo de tempo definido pelo tamanho do vaso de planta.
- **Configuração de Parâmetros de Irrigação:** Push-buttons permitirão que o usuário ajuste os níveis de umidade desejados para a planta, assim como a duração de cada rega ao detectar a necessidade do acionamento, as configurações serão exibidas no display OLED SSD1306.
- **Alertas Sonoros:** Um buzzer passivo gera alertas para o baixo nível de água no reservatório (caso a bomba funcione, mas a umidade não aumente).
- **Interface com o Usuário via Display OLED:** Através do display é exibido em tempo real: a iluminação do ambiente (%), a umidade do solo (%) e os parâmetros de irrigação que podem ser alterados pelo usuário.

1.4 JUSTIFICATIVA

O projeto propõe um sistema automatizado de irrigação e iluminação para plantas, utilizando um Raspberry Pi Pico W (RP2040). Ele monitora a umidade do solo e a iluminação ambiente por meio de sensores, acionando uma bomba d'água e uma matriz de LEDs RGB 5x5 WS2812 conforme a necessidade da planta.

A otimização do uso da água é um dos principais benefícios, pois o sensor de umidade do solo aciona a irrigação apenas quando necessário, evitando desperdícios e garantindo umidade adequada. Um buzzer passivo alerta sobre a necessidade de reabastecimento do reservatório ou falhas no sensor, garantindo funcionamento contínuo.

O sistema também regula a iluminação artificial, ajustando automaticamente a intensidade da luz de acordo com a leitura do sensor LDR. Isso permite que as plantas recebam a luz ideal independentemente da estação do ano ou localização.

A interface do usuário inclui um display OLED SSD1306, que exibe dados de iluminação, umidade e parâmetros de irrigação, além de push-buttons para configuração personalizada.

Esse sistema reduz a necessidade de monitoramento manual, promove sustentabilidade no uso da água e energia e é aplicável tanto para cultivos domésticos quanto para estufas agrícolas. A

tecnologia empregada facilita o cultivo de plantas e incentiva práticas mais eficientes e ecológicas.

1.5 ORIGINALIDADE

O projeto proposto apresenta uma abordagem inovadora ao integrar diferentes tecnologias para otimização do cultivo de plantas, unindo controle inteligente de irrigação, iluminação e monitoramento ambiental. A partir da análise das fontes citadas, os principais diferenciais do projeto são evidenciados nos seguintes aspectos:

- **Integração Multissensorial e Automação Completa:** Diferente de projetos convencionais que focam em um único parâmetro, este sistema combina sensores de umidade do solo e iluminação (LDR), permitindo uma automação precisa da irrigação e do controle luminoso. O uso do higrômetro descrito pela MakerHero e MasterWalkerShop possibilita a irrigação apenas quando necessário, evitando desperdícios.
- **Controle de Iluminação Inteligente e Adaptável:** A utilização da matriz de LEDs RGB WS2812B 5x5 possibilita um ajuste dinâmico da iluminação artificial, garantindo que as plantas recebam luz adequada independentemente da estação ou ambiente. Diferente de sistemas fixos de iluminação, este projeto permite a personalização do espectro luminoso, algo essencial para diferentes fases do crescimento vegetal.
- **Eficiência no Uso de Recursos Hídricos e Energéticos:** O projeto utiliza uma mini bomba d'água 12V RS385, com controle via relé, otimizando o consumo hídrico. A implementação se baseia em projetos como o descrito na UsinaInfo, mas aprimora a eficiência ao integrar sensores e um buzzer passivo para alertas de nível de água.
- **Arquitetura Compacta e Uso do Raspberry Pi Pico W:** Diferente de projetos que utilizam microcontroladores maiores, como ESP32 (exemplo da UsinaInfo), a escolha do Raspberry Pi Pico W possibilita baixo consumo energético e maior flexibilidade na programação em C. Além disso, permite futuras integrações com conectividade sem fio, ampliando suas aplicações.
- **Interface Interativa e Configuração Personalizável:** O uso do display OLED SSD1306 proporciona feedback visual em tempo real, permitindo ao usuário

monitorar e ajustar parâmetros facilmente. A presença de push-buttons para configuração direta destaca a interatividade do sistema em comparação a projetos de irrigação convencionais.

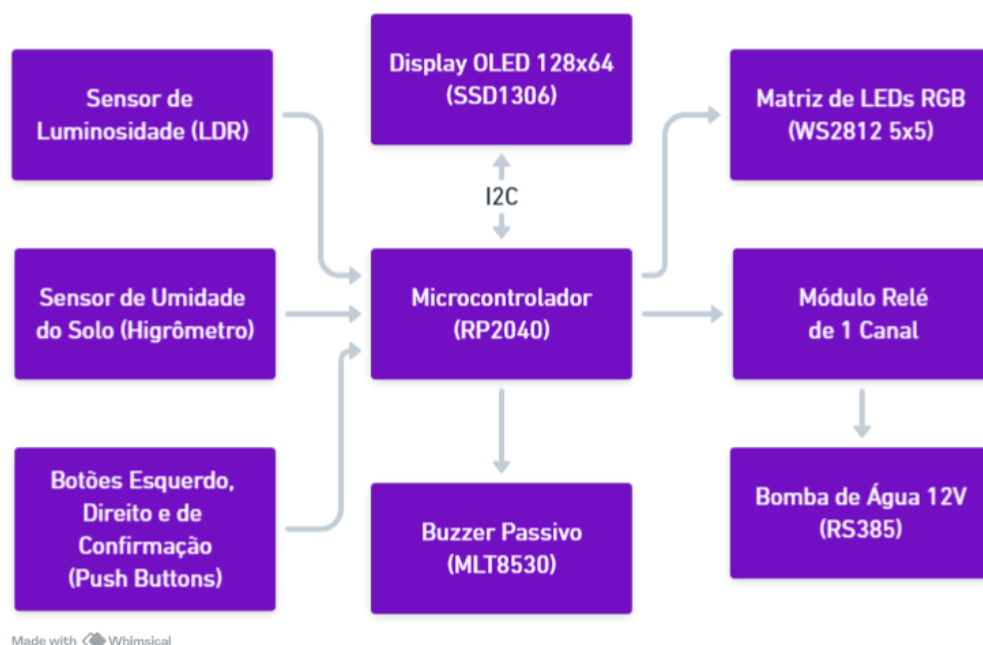
Portanto, esse projeto se diferencia ao combinar tecnologias de sensoriamento ambiental, automação inteligente e interação intuitiva, garantindo cultivo otimizado com menor desperdício de água e energia. A implementação baseada no Raspberry Pi Pico W, aliada ao controle adaptativo de iluminação e irrigação, torna o sistema altamente eficiente, sustentável e inovador no contexto da automação agrícola.

2 ESPECIFICAÇÃO DO HARDWARE

2.1 DIAGRAMA EM BLOCO

O diagrama de blocos apresentado na fig. 1 representa a estrutura de hardware de um sistema baseado no microcontrolador RP2040, que gerencia sensores, atuadores e uma interface de usuário para um projeto automatizado.

Figura 1: Diagrama em blocos do projeto.



2.2 FUNÇÃO E CONFIGURAÇÃO DE CADA BLOCO

2.2.1 Microcontrolador (RP2040)

O RP2040 é o núcleo do sistema, responsável por processar dados dos sensores e controlar os atuadores. Ele recebe informações dos sensores e, com base nelas, toma decisões como ativar a bomba de água ou modificar a iluminação. Além disso, faz o gerenciamento de interações com o usuário, manipulando as informações exibidas no display.

2.2.2 Sensor de Luminosidade (LDR)

Mede a intensidade luminosa do ambiente e a converte em tensão. Essa informação é utilizada para medir os níveis de luz do local e consequentemente controlar o acionamento de atuadores de iluminação.

2.2.3 Sensor de Umidade do Solo (Higrômetro)

Mede a umidade do solo e a converte em diferença de potencial. A informação é utilizada pelo microcontrolador para decidir processos de irrigação.

2.2.4 Módulo Relé de 1 Canal

Controla a Bomba de Água de 12V (RS385), permitindo que o RP2040 ative ou desative o bombeamento de água conforme necessário para a irrigação, respeitando a tensão de alimentação da bomba, incompatível com a tensão de saída dos pinos do microcontrolador.

2.2.5 Buzzer Passivo (MLT8530)

Gera sinais sonoros para alertas e notificações do sistema, sendo controlado por modulação por largura de pulso (PWM) e acionado quando o sistema necessita de atenção especial.

2.2.6 Display OLED 128x64 (SSD1306)

Exibe informações do sistema, permitindo a interação com o usuário, sendo atualizado com frequência imperceptível ao olho humano. A comunicação com o RP2040 ocorre via I2C.

2.2.7 Botões (Push Buttons - Esquerdo, Direito e de Confirmação)

Permitem a interação do usuário com o sistema, podendo navegar por opções no display e definir parâmetros específicos.

2.2.8 Matriz de LEDs RGB (WS2812 5x5)

Fornece iluminação configurável e é usada como atuador de iluminação do sistema.

2.3 COMANDOS E REGISTROS UTILIZADOS

O microcontrolador RP2040 gerencia diversos periféricos para a automação do sistema, utilizando suas portas GPIO, PWM, ADC, I2C e PIO. Cada periférico requer comandos e registros específicos para operação adequada. A tabela a seguir resume os principais registros utilizados para controle dos dispositivos conectados ao RP2040.

Tabela 1: Periféricos e Registros.

Periférico	Comunicação	Registros
GPIO (Botões, Relé, LEDs)	Digital	SIO_GPIO_IN (leitura de estado), SIO_GPIO_OUT (escrita de estado), GPIO_OE_SET (habilita saída)
PIO (Matriz de LEDs WS2812 RGB)	Protocolo de tempo	PIO_TXF0 (buffer de transmissão), PIO_SM0_CLKDIV (configuração de clock), PIO_SM0_EXECCTRL (controle de execução)

I2C (Display OLED SSD1306)	I2C	I2C_IC_CON (configuração do protocolo), I2C_IC_DATA_CMD (envio de comandos/dados), I2C_IC_STATUS (status da comunicação)
PWM (Buzzer Passivo MLT8530)	PWM	PWM_CHx_CSR (configuração do canal), PWM_CHx_DIV (definição de frequência), PWM_CHx_CC (controle do duty cycle)
ADC (Sensor LDR e Higrômetro)	ADC	ADC_CS (controle do ADC), ADC_RESULT (leitura do valor convertido), ADC_FCS (gerenciamento do FIFO do ADC)

2.3.1 Portas de Uso Geral de Entrada e Saída (Periférico GPIO)

O GPIO (General Purpose Input/Output) do RP2040 é utilizado para interagir com dispositivos digitais. Nesse sistema, gerencia a entrada de sinal digital dos botões, a serem tratados por interrupções de firmware. Assim como o acionamento do relé, para controle da bomba de água.

As entradas de sinais são declaradas no código e é feito o acionamento dos resistores de pull-up em todos os botões, garantindo a leitura estável do sinal.

2.3.2 Comunicação via Protocolo de Tempo Preciso (Periférico PIO)

A matriz de LEDs WS2812 utiliza um protocolo de tempo preciso, muito delicado, que necessita de contagens em nível de microsegundos. Através do PIO (Programmable I/O) do RP2040, é permitido gerar sinais com tempos exatos e sem sobrecarregar a CPU.

Ademais, cada LED RGB da matriz recebe 24 bits (8 bits para cada canal R, G, B).

2.3.2 Comunicação via Protocolo I2C (Periférico I2C)

O display SSD1306 se comunica via I2C, que requer dois pinos: SDA (dados) e SCL (clock), através da biblioteca *hardware/adc.h*, e possui um conjunto de comandos para configuração e escrita de dados contidos na biblioteca criada pelo programador *include/ssd1306.h*.

2.3.3 Geração de Sinal com Frequências Específicas (Periférico PWM)

O buzzer passivo MLT8530 precisa de um sinal PWM para emitir sons em diferentes frequências, de modo que o periférico permite definir a frequência exata do som, através da biblioteca *hardware/pwm.h* através da configuração de *WRAP*, do divisor de frequência e do ciclo de trabalho.

2.3.4 Leitura de Informações por Conversão Analógica Digital (Periférico ADC)

Os sensores LDR (luminosidade) e higrômetro (umidade do solo) são sensores analógicos e necessitam de Conversão Analógico-Digital (ADC) para que assim sejam tratados seus valores através da biblioteca *hardware/adc.h*.

2.4 DESCRIÇÃO DA PINAGEM USADA

A correta alocação dos pinos do Raspberry Pi Pico W (RP2040) é essencial para garantir o funcionamento adequado dos sensores e atuadores utilizados no projeto. A Tabela 2 apresenta a distribuição da pinagem empregada, especificando os pinos do hardware e suas respectivas funções.

Tabela 2 - Distribuição da Pinagem no RP2040

Componente	Pino RP2040	Função
Sensor LDR (Iluminação ambiente)	GP26 (ADC0)	Entrada analógica para medição da luminosidade ambiente.
Sensor de Umidade do Solo	GP27 (ADC1)	Entrada analógica para detecção da umidade do substrato.
Push-buttons (Configuração)	GP5, GP6, GP22	Entradas digitais para ajuste dos parâmetros de irrigação.
Display OLED SSD1306 (I2C - Comunicação)	GP14 (SDA), GP15 (SCL)	Comunicação via protocolo I2C para exibição de dados.
Matriz de LEDs RGB 5x5 WS2812	GP7	Saída digital para controle da iluminação artificial.

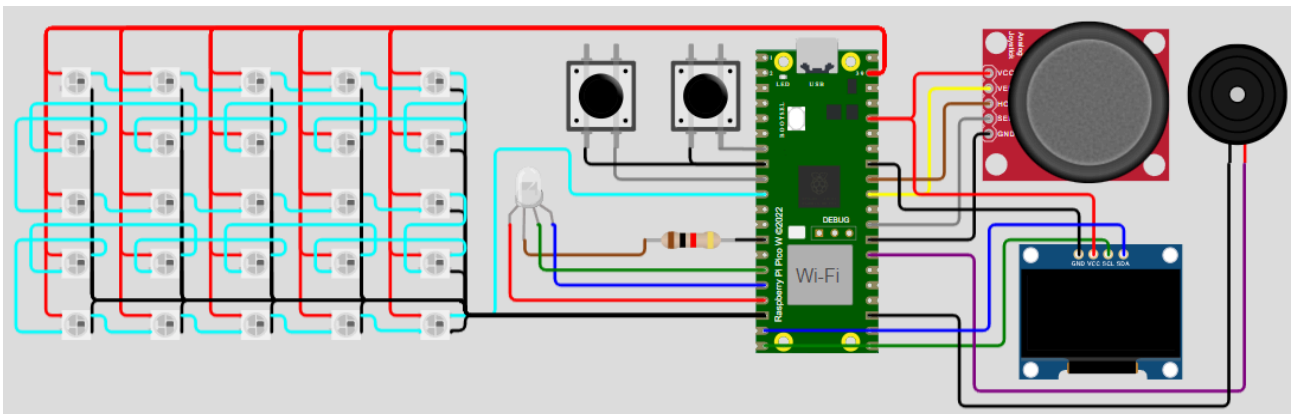
Bomba d'água (Acionada por relé)	GP12	Saída digital para ativação do relé da bomba.
Buzzer Passivo	GP21	Saída digital para emissão de alertas sonoros.

A escolha dos pinos levou em consideração a compatibilidade com os requisitos elétricos dos sensores e atuadores, bem como a otimização do uso das interfaces de comunicação do RP2040, como ADC (Conversor Analógico-Digital) e I2C (Inter-Integrated Circuit). Dessa forma, a configuração estabelecida permite a integração eficiente dos componentes, garantindo um controle preciso da iluminação, irrigação e notificações sonoras no sistema proposto. Além disso, é possível a simulação do sistema do projeto na plataforma de desenvolvimento BitDogLab.

2.5 CIRCUITO COMPLETO DO HARDWARE

2.5.1 Circuito Equivalente no Wokwi

Figura 2: Esquema de Componentes no Wokwi Integrado ao VS Code.



2.5.2 Componentes Equivalentes

Considerando as dificuldades para implementação inicial do sistema com os componentes reais, será utilizado a plataforma de desenvolvimento BitDogLab contendo os componentes

da Fig. 2. A seguir será apresentado os componentes equivalentes (que são diferentes da imagem):

- **LED RGB:** Representa o conjunto de Relé (Fig. 3.a) conectado à bomba de água RS385 (Fig. 3.b) como mostrado.

Figura 3.a: Relé de 1 Canal de 3V.



Figura 3.b: Bomba de Água RS385 de 12V.



- **Joystick:** Ambas as coordenadas verticais e horizontais do componente, representam, respectivamente, os sensores de umidade do solo (higrômetro, Fig. 4.a) e de luminosidade (LDR, Fig 4.b).

Figura 4.a: Sensor Higrômetro.

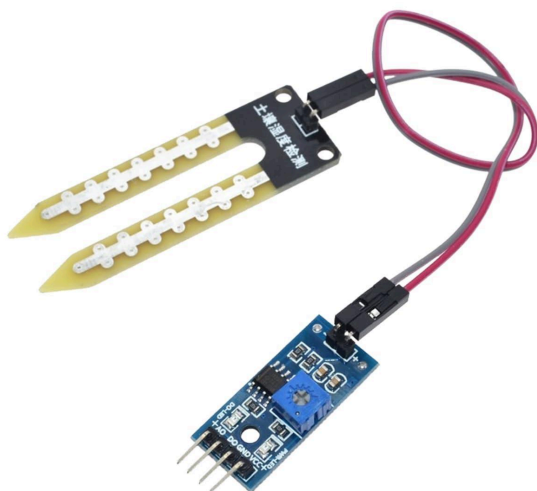
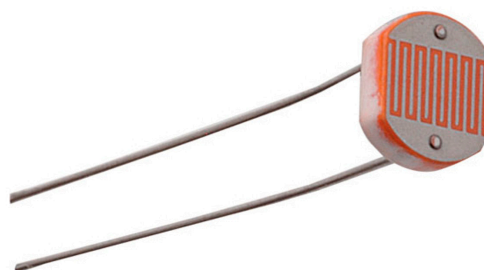


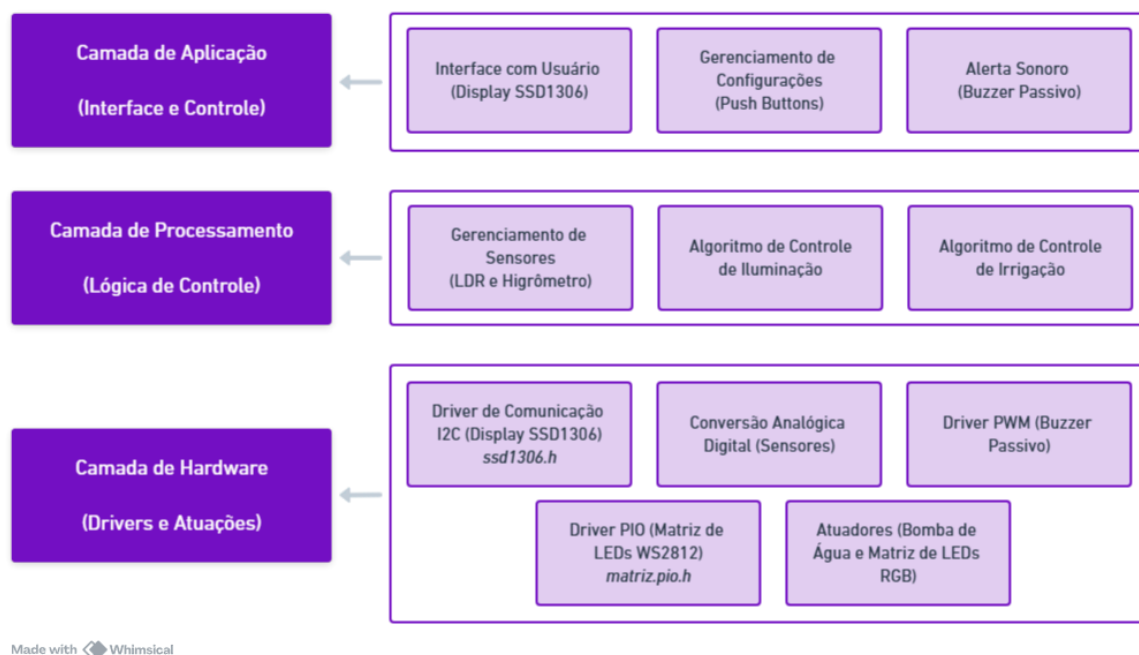
Figura 3.b: Sensor LDR.



3 ESPECIFICAÇÃO DO FIRMWARE

3.1 BLOCOS FUNCIONAIS

Figura 5: Diagrama de Blocos Funcionais em Camadas



3.2 DESCRIÇÃO DAS FUNCIONALIDADES

Aqui está uma explicação das funções presentes no código.

- **matrix_rgb(double r, double g, double b)**
 - Converte valores de intensidade de cores (RGB) para um formato adequado para a matriz de LEDs.
 - Retorna um valor de 32 bits contendo as intensidades de vermelho, verde e azul.
- **desenho()**
 - Define e envia os valores de intensidade da matriz de LEDs de acordo com a luminosidade do ambiente.
 - Usa a função `pio_sm_put_blocking()` para atualizar os LEDs.

- **gpio_irq_handler(uint gpio, uint32_t events)**
 - Trata as interrupções dos botões pressionados.
 - Se Bot_Left for pressionado, diminui o valor ideal de umidade.
 - Se Bot_Right for pressionado, aumenta o valor ideal de umidade.
 - Se Bot_Confirm for pressionado, ajusta o tempo de rega.
- **init_pins()**
 - Inicializa os pinos dos botões e configura interrupções para detectar pressionamentos.
 - Inicializa os pinos do relé e da matriz de LEDs.
 - Configura o PIO para controlar a matriz de LEDs.
- **pwm_setup(uint32_t freq)**
 - Configura o PWM no pino do buzzer para gerar um som com a frequência desejada.
 - Ajusta o clock, o divisor e define um duty cycle de 50%.
- **turn_off_callback(alarm_id_t id, void *user_data)**
 - Callback que desliga o relé após um tempo programado.
 - Define o pino do relé como false, desligando-o.
- **tocar_alerta()**
 - Ativa o buzzer para emitir um alerta caso a umidade esteja abaixo do valor ideal.
 - Alterna entre duas frequências (250 Hz e 400 Hz) para gerar um som intermitente.
 - Configura um alarme para desligar o alerta após um tempo predefinido.
- **verificar_alerta_callback(alarm_id_t id, void *user_data)**
 - Callback que verifica se o alerta ainda precisa ser acionado e, se necessário, toca o alerta novamente.
- **init_process()**
 - Inicializa a comunicação I2C para o display OLED.
 - Inicializa o ADC para os sensores de luz e umidade.
- **main()**
 - Inicializa as configurações do sistema.
 - Entra em um loop infinito para:
 - Ler os sensores de luminosidade e umidade.
 - Ajustar a intensidade da matriz de LEDs.
 - Atualizar o display OLED com os valores dos sensores.
 - Controlar o relé para irrigação com base na umidade do solo.
 - Acionar o alerta sonoro se necessário.

Esse código implementa um sistema que monitora luminosidade e umidade, acionar a matriz de LEDs e um relé (bomba de água), e usa um buzzer para alertas.

3.3 DEFINIÇÃO DAS VARIÁVEIS

3.3.1 Variáveis de Configuração de Hardware

Definidas no código fonte, são variáveis fixas para manipulação dos pinos do microcontrolador RP2040.

- Bot_Left (5): Pino do botão esquerdo.
- Bot_Right (6): Pino do botão direito.
- Bot_Confirm (22): Pino do botão de confirmação.
- Sensor_Luz (26): Pino do sensor de luminosidade.
- Sensor_Umidade (27): Pino do sensor de umidade do solo.
- Matriz (7): Pino de controle da matriz de LEDs 5x5.
- Buzzer (21): Pino de controle do buzzer.
- Relay (12): Pino de controle do relé.

3.3.2 Variáveis de Comunicação I2C

Utilizadas para configuração da comunicação I2C, essas funções guardam informações para manter a organização do código e a declaração da ssd permite a utilização das funções da biblioteca em todo o código.

- I2C_PORT (i2c1): Define a porta I2C utilizada.
- I2C_SDA (14): Pino do barramento de dados do I2C (SDA).
- I2C_SCL (15): Pino do barramento de clock do I2C (SCL).
- address (0x3C): Endereço do display OLED SSD1306.
- ssd: Estrutura de controle do display OLED, contido na biblioteca ssd1306.h.

3.3.3 Variáveis de Controle para Exibição no Display

Variáveis do tipo char, que receberão informações a serem exibidas no display e atualizadas na rotina de repetição while (true) da função main.

- char str_luz[7]: Armazena a string formatada da luminosidade lida.
- char str_umid[7]: Armazena a string formatada da umidade do solo lida.
- char str_ideal[5]: Armazena a string do valor de umidade ideal.
- char str_rega[4]: Armazena a string do tempo de rega.

3.3.4 Variáveis de Controle dos Botões

Do tipo int8_t, são valores de até 8 bits, que serão alterados através das rotinas de interrupção dos botões.

- int8_t ideal = 50 → Valor ideal de umidade do solo (inicialmente 50%).
- int8_t tempo_rega = 5 → Tempo de rega em segundos (inicialmente 5s).

3.3.5 Variáveis de Controle do Alerta

Utilizadas para configurar a rotina de acionamento do buzzer passivo por PWM e filtragem de rotina de acionamento.

- Alerta_ms 3000: Tempo de duração de alerta em milissegundos.
- volatile bool evento = false: Indica se deve haver um evento de alerta ativo (inicialmente falso).
- uint slice_num: Número do slice do PWM do buzzer (tipo inteiro).

3.3.6 Variáveis de Controle da Conversão ADC

Variáveis do tipo uint16_t guardarão os valores convertidos pelos sensores e a variável umidade, que será alterada a partir dos valores de adc_value_umidade.

- `uint16_t adc_value_luz`: Armazena o valor lido do sensor de luminosidade.
- `uint16_t adc_value_umidade`: Armazena o valor lido do sensor de umidade do solo.
- `double umidade`: Variável para armazenar a umidade do solo em porcentagem.

3.3.7 Variáveis Globais para a Matriz de LEDs

Para que seja possível o acionamento da matriz de LEDs pelo uso da função `desenho()`, assim como a definição da intensidade dos LEDs, é necessário a utilização das variáveis a seguir. Ademais, a variável `intensidade` será alterada a partir dos valores de `adc_value_luz`.

- `PIO pio`: Estrutura para controle do periférico PIO.
- `uint sm`: Número do state machine do PIO.
- `double intensidade = 0.5`: Intensidade inicial da matriz de LEDs, que irá variar em proporção aos valores convertidos do sensor LDR.

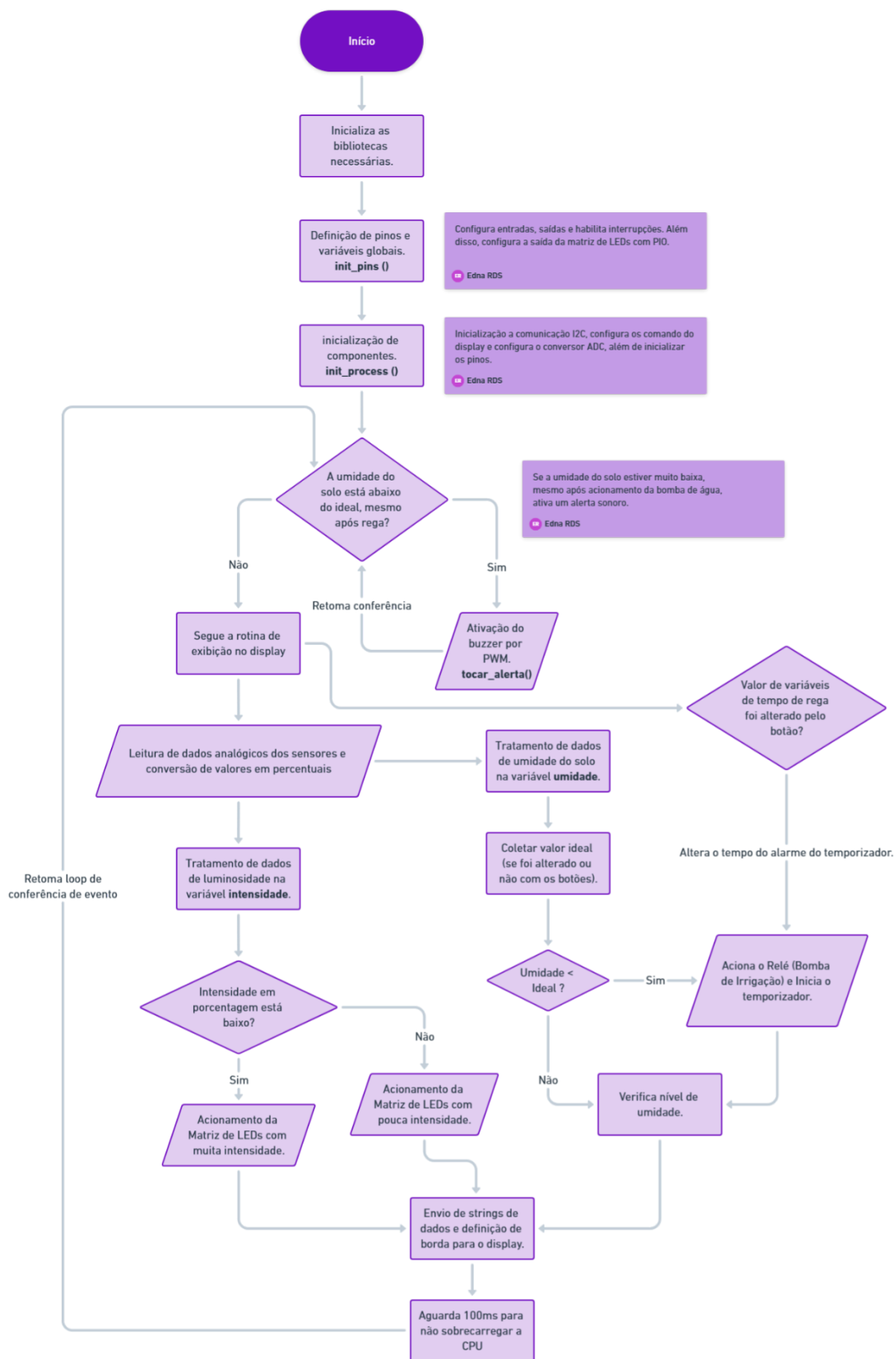
3.3.8 Variáveis para Debounce dos Botões

Utilizada para evitar acionamentos falsos dos botões.

- `static volatile uint32_t last_time = 0`: Armazena o tempo do último acionamento do botão (em microssegundos).

3.4 FLUXOGRAMA

Figura 6: Fluxograma de Firmware.



3.5 PROTOCOLO DE COMUNICAÇÃO

Como já comentado anteriormente, o único protocolo de comunicação utilizado, não personalizado pelo programador, é o I2C (Inter-Integrated Circuit). Com apenas 2 fios de conexão: SDA (Serial Data Line) , com a linha de dados, e SCL (Serial Clock Line), com a linha de clock. A comunicação I2C funciona com um mestre (master) e um ou mais escravos (slaves), onde o mestre controla o clock e inicia as transmissões de dados.

3.6 FORMATO DO PACOTE DE DADOS

A comunicação I²C segue um formato de pacote padronizado, composto pelas seguintes fases:

- **Condição de Início (START):** O mestre inicia a comunicação enviando um bit de START, onde a linha SDA transita de HIGH para LOW enquanto SCL está HIGH.
- **Endereço do Dispositivo:** O mestre envia um byte de endereço (7 bits a 10 bits) correspondente ao periférico escravo desejado.
- **Bit de Leitura/Escrita (R/W):** O oitavo bit indica se o mestre deseja escrever (0) ou ler (1) do escravo.
- **Confirmação (ACK/NACK):** O escravo responde com um bit de reconhecimento (ACK - Acknowledge) puxando a linha SDA para LOW. Se não reconhecer o endereço, responde com NACK (Not Acknowledge).
- **Dados (8 bits):** O mestre ou o escravo envia um byte de dados, seguido de um ACK para confirmar o recebimento.
- **Condição de Parada (STOP):** O mestre finaliza a comunicação enviando um bit de STOP, onde SDA transita de LOW para HIGH enquanto SCL está HIGH.

4 METODOLOGIA

Neste projeto, foi adotada a metodologia descrita por Wayne Wolf (2008) em seu livro *Computers as Components - Principles of Embedded Computing System Design*, que define cinco etapas para o desenvolvimento de sistemas embarcados:

- **Etapa 1 - Definição de Requisitos:** Nesta fase, são estabelecidos os requisitos do sistema, ou seja, suas funcionalidades e, quando aplicável, a forma como essas funções devem ser executadas para atender aos padrões do projeto. Especificamente, identificou-se a necessidade de um componente sonoro para alertar o usuário e um sensor capaz de detectar a abertura ou fechamento de portas ou janelas.
- **Etapa 2 - Especificação:** Os requisitos levantados na etapa anterior são formalmente documentados em uma linguagem clara e sem ambiguidades, garantindo uma interpretação precisa do funcionamento do sistema.
- **Etapa 3 - Arquitetura:** O sistema é representado funcionalmente por meio de diagramas em blocos, que destacam os principais módulos e componentes, além de suas interconexões e funções dentro do projeto.
- **Etapa 4 - Seleção de Componentes:** Nesta fase, são escolhidos os componentes de hardware e software que melhor atendem aos requisitos do sistema. Caso não existam soluções comerciais adequadas, novas alternativas devem ser desenvolvidas. Para este projeto, foram selecionados:
 - **Sensor LDR** – Um sensor de luminosidade que permite medir a intensidade da luz no ambiente, útil para controle de iluminação ou ajuste automático de brilho de telas;
 - **Sensor de Umidade do Solo (Higrômetro)** – Utilizado para medir a umidade do solo, essencial para sistemas de irrigação automatizados;
 - **Botões Esquerdo, Direito e de Confirmação (Push Buttons)** – Implementados para permitir a interação do usuário com o sistema, para alteração de parâmetros de rega;
 - **Display OLED 128x64 (SSD1306)** – Tela utilizada para exibição de informações e interações com o usuário;
 - **Microcontrolador RP2040** – Coração do sistema, responsável pelo processamento e controle de todos os componentes;
 - **Buzzer Passivo (MLT8530)** – Dispositivo utilizado para emitir sinais sonoros, útil para alertas no sistema;
 - **Matriz de LEDs RGB (WS2812 5x5)** – Conjunto de LEDs RGB endereçáveis, utilizado para criar iluminar as plantas;
 - **Módulo Relé de 1 Canal (3V)** – Necessário para controlar cargas elétricas, como a bomba de água, que opera com tensão de 12V;

- **Bomba de Água 12V (RS385)** – Utilizada para irrigação automatizada, acionada conforme a leitura do sensor de umidade do solo.

Além disso, considerando que o projeto será simulado, foram desenvolvidas estratégias para replicar digitalmente o comportamento dos componentes de hardware.

- **Etapa 5 - Integração do Sistema:** Nesta etapa, os componentes de hardware e software são integrados para garantir seu funcionamento conjunto. Esse processo geralmente apresenta diversos desafios, incluindo a ocorrência de bugs, exigindo tempo e dedicação para sua resolução. Como este projeto será simulado, nem todos os problemas reais estarão presentes, mas as dificuldades de execução do software no hardware ainda serão um fator relevante.

5 TESTES DE VALIDAÇÃO

Para assegurar o correto funcionamento de todas as partes do sistema, foram realizados testes unitários, onde cada módulo do código foi avaliado individualmente. Parte desses testes foi simplificada devido às restrições do projeto, que exigiram a simulação de alguns componentes de hardware.

Inicialmente, os testes foram conduzidos na plataforma Wokwi e, posteriormente, validados de forma prática no BitDogLab.

6 DISCUSSÃO DOS RESULTADOS

Os testes de validação permitiram avaliar o desempenho do sistema automatizado de irrigação e iluminação para plantas. A partir das simulações conduzidas no Wokwi e validações práticas no BitDogLab, foi possível verificar a funcionalidade de cada módulo e a integração dos componentes de hardware e firmware.

O sensor de umidade do solo demonstrou eficiência ao acionar a bomba d'água somente quando necessário, evitando desperdícios e garantindo níveis adequados de irrigação. Durante os testes, observou-se que a resposta do sistema foi coerente com as leituras do sensor, interrompendo a irrigação assim que a umidade atingia o valor configurado. Além disso, o

buzzer passivo notificou corretamente a necessidade de reabastecimento do reservatório de água, assegurando o funcionamento contínuo do sistema.

O sensor LDR, responsável pelo monitoramento da iluminação, também apresentou resultados satisfatórios. A matriz de LEDs RGB WS2812 5x5 respondeu de maneira eficaz ao controle do microcontrolador RP2040, ajustando a intensidade luminosa de acordo com a necessidade do ambiente. Durante a simulação, foi possível observar que o sistema manteve um nível adequado de iluminação para as plantas, independentemente da variação da luz natural.

A interface com o usuário, composta por um display OLED SSD1306 e push-buttons, foi avaliada em relação à sua usabilidade. Os testes demonstraram que o sistema conseguiu exibir as informações de umidade, iluminação e estado da irrigação de forma clara e acessível. A interação por meio dos botões permite ajustes manuais dos parâmetros de irrigação, tornando o sistema flexível e adaptável a diferentes necessidades de cultivo.

No entanto, algumas limitações foram identificadas. A simulação do projeto, apesar de fornecer uma visão clara do comportamento esperado, não reproduziu integralmente todas as variáveis físicas envolvidas, como variações reais de umidade e luminosidade ao longo do tempo. A implementação prática do sistema pode exigir calibrações adicionais dos sensores para garantir a precisão das leituras em um ambiente real. Além disso, a utilização do Raspberry Pi Pico W abre a possibilidade de futuras expansões, como a integração com conectividade sem fio para monitoramento remoto, o que poderia aumentar a autonomia e a eficiência do sistema.

Os resultados obtidos indicam que o sistema proposto atende aos requisitos iniciais do projeto, apresentando uma solução eficiente para o cultivo automatizado de plantas. A abordagem adotada mostrou-se eficaz na otimização do uso de recursos, contribuindo para a sustentabilidade e facilitando a manutenção das plantas em ambientes internos e externos.

7 CONCLUSÃO

O presente projeto demonstrou a viabilidade e eficácia da implementação de um sistema automatizado de irrigação e iluminação para plantas utilizando o Raspberry Pi Pico W como unidade central de controle. A abordagem adotada permitiu integrar sensores de umidade do solo e iluminação, acionando automaticamente a bomba d'água e a matriz de LEDs RGB, garantindo um ambiente adequado para o crescimento das plantas.

Os testes de validação confirmaram que o sistema atendeu aos requisitos estabelecidos, proporcionando um uso eficiente da água e da luz, minimizando desperdícios e otimizando os recursos disponíveis. A interface do usuário, composta por um display OLED e push-buttons, possibilitou a personalização dos parâmetros de irrigação e iluminação, tornando o sistema adaptável às necessidades específicas de diferentes espécies de plantas.

Apesar das limitações inerentes à simulação, os resultados indicam que a proposta pode ser aplicada em cultivos reais com ajustes mínimos. A possibilidade de expandir o projeto com conectividade sem fio abre novas perspectivas para monitoramento remoto e automação avançada, tornando o sistema ainda mais eficiente e versátil.

Dessa forma, este trabalho contribui para o desenvolvimento de soluções sustentáveis e acessíveis para a automação do cuidado com plantas, incentivando práticas agrícolas mais ecológicas e facilitando o manejo de vegetações em diferentes ambientes.

REFERÊNCIAS

BELLAHIRICH, Siwar; MEZGHANI, Dhafer; MAMI, Abdelkader. Design and implementation of an intelligent ANFIS controller on a Raspberry Pi nano-computer for photovoltaic pumping intended for drip irrigation. *Energies*, v. 14, n. 17, p. 5217, 2021.

CIRCUIT BASICS. Setting up a 5V relay on the Arduino. *Circuit Basics*, [2025]. Disponível em: https://www.circuitbasics.com/setting-up-a-5v-relay-on-the-arduino/#google_vignette. Acesso em: 26 fev. 2025.

MAKERHERO. Sensor de umidade do solo higrômetro. *MakerHero*, [2025]. Disponível em: https://www.makerhero.com/produto/sensor-de-umidade-do-solo-higrometro/?srsltid=AfmBOooQsLJHFS0Ho3D6Yf09R3OG2rals2N92of7z_qR3lhTy-fdurDo. Acesso em: 26 fev. 2025.

MASTERWALKERSHOP. Sensor medidor de umidade do solo higrômetro. *Master Walker Shop*, [2025]. Disponível em: https://www.masterwalkershop.com.br/sensor-medidor-de-umidade-do-solo-higrometro?srsltid=AfmBOoq9B3tPR3M1O6I1yNAMuiyKxmo2VRmOnwQIJ_1lirC8WsXfPf-1. Acesso em: 26 fev. 2025.

NAYLAMP MECHATRONICS. Matriz LED RGB WS2812B 5x5. *Naylamp Mechatronics*, [2025]. Disponível em: <https://naylampmechatronics.com/luces-iluminacion/957-matriz-led-rgb-ws2812b-5x5.html>. Acesso em: 26 fev. 2025.

TECNOTRONICS LOJA DE ARDUINO. Funcionamento da bomba e informações técnicas. *YouTube*, [2018]. Disponível em: <https://www.youtube.com/watch?v=SzotZ5Pgi8w>. Acesso em: 26 fev. 2025.

USINAINFO. ESP32: Projeto controlando um relé. *Usina Info*, [2019]. Disponível em: <https://www.usinainfo.com.br/blog/esp32-projeto-controlando-um-rele/?srsltid=AfmBOoqpD GgBDbfEl1ASbJDWd5ZZfeW91vm9Wc5QxuuXa7foxcAGNwMb>. Acesso em: 26 fev. 2025.

ANEXOS

Repositório no GitHub

<https://github.com/EdnaRodrigues/Garden.git>

Vídeo de Demonstração

<https://youtu.be/s0fULUZS2p4>