



Instituto Federal de Educação, Ciência e Tecnologia da Bahia  
Campus de Vitória da Conquista  
EmbarcaTech - CEPEDI

## **Projeto Final**

HelpEase: Sistema de Comunicação Assistiva para Pedidos de Ajuda e  
Emergência com Interface Intuitiva

**Curso:** Formação Básica em Software Embarcado

**Instrutores:** Wilton Lacerda Silva e Ricardo Menezes Prates

**Discente:**

João Gabriel Furtado Machado / 370101222

26 de Fevereiro de 2025  
Vitória da Conquista - BA

# Conteúdo

<b>1</b>	<b>Escopo do Projeto</b>	<b>4</b>
1.1	Apresentação do Projeto . . . . .	4
1.2	Título do Projeto . . . . .	4
1.3	Objetivos do Projeto . . . . .	4
1.4	Descrição do Funcionamento . . . . .	4
1.5	Justificativa . . . . .	5
1.6	Originalidade . . . . .	5
<b>2</b>	<b>Especificação do Hardware</b>	<b>6</b>
2.1	Diagrama em Bloco . . . . .	6
2.2	Função de Cada Bloco . . . . .	6
2.3	Configuração de Cada Bloco . . . . .	7
2.4	Comandos e Registros Utilizados . . . . .	8
2.5	Descrição da Pinagem Usada . . . . .	8
2.6	Circuito Completo do Hardware . . . . .	9
<b>3</b>	<b>Especificação do Firmware</b>	<b>10</b>
3.1	Blocos Funcionais . . . . .	10
3.2	Descrição das Funcionalidades . . . . .	10
3.3	Definição das Variáveis . . . . .	11
3.4	Fluxograma . . . . .	12
3.5	Inicialização . . . . .	13
3.6	Configurações dos Registros . . . . .	13
3.7	Estrutura e Formato dos Dados . . . . .	14
3.8	Protocolo de Comunicação . . . . .	14
3.9	Formato do Pacote de Dados . . . . .	14
<b>4</b>	<b>Execução do Projeto</b>	<b>15</b>
4.1	Metodologia . . . . .	15
4.2	Testes de Validação . . . . .	16
4.3	Discussão dos Resultados . . . . .	17
<b>5</b>	<b>Conclusão</b>	<b>17</b>
	<b>Referências</b>	<b>18</b>

**Lista de Figuras**

1	Diagrama em bloco do sistema HelpEase. . . . .	6
2	Circuito completo do hardware do projeto HelpEase. . . . .	9
3	Diagrama de camadas do firmware. . . . .	10
4	Fluxograma do software. . . . .	12

**Lista de Tabelas**

1	Pinagem do Raspberry Pi Pico W no projeto HelpEase. . . . .	9
---	---	---

# 1 Escopo do Projeto

## 1.1 Apresentação do Projeto

O projeto **HelpEase** é uma solução tecnológica desenvolvida para auxiliar pessoas em situações de vulnerabilidade, como idosos, pacientes em recuperação ou pessoas com dificuldades de locomoção, a se comunicarem de forma eficiente com seus cuidadores. O sistema permite que o usuário solicite ajuda de maneira simples e intuitiva, utilizando um display OLED para exibir opções de pedidos, um joystick para navegação e uma matriz de LEDs para indicar o nível de urgência. Além disso, o projeto inclui funcionalidades de emergência, como alertas sonoros e visuais, para situações críticas. A ideia é que o cuidador tenha um sistema embarcado que receba as informações via Bluetooth, Wi-Fi, IoT ou um aplicativo de celular, permitindo a comunicação em tempo real.

## 1.2 Título do Projeto

**HelpEase: Sistema de Comunicação Assistiva para Pedidos de Ajuda e Emergência com Interface Intuitiva**

## 1.3 Objetivos do Projeto

- **Objetivo Geral:** Desenvolver um sistema de comunicação assistiva que facilite a solicitação de ajuda e emergências para pessoas com dificuldades de locomoção ou comunicação.
- **Objetivos Específicos:**
  - Implementar um menu interativo no display OLED para seleção de pedidos de ajuda.
  - Utilizar um joystick para navegação no menu e seleção de opções.
  - Integrar uma matriz de LEDs 5x5 para indicar o nível de urgência do pedido.
  - Adicionar funcionalidades de emergência, como alertas sonoros e visuais.
  - Facilitar a comunicação entre o usuário e o cuidador por meio de notificações visuais e sonoras.
  - Desenvolver um sistema embarcado para o cuidador, que receba as informações via Bluetooth, Wi-Fi, IoT ou um aplicativo de celular.

## 1.4 Descrição do Funcionamento

O projeto **HelpEase** funciona da seguinte forma:

- **Menu de Opções:** Um display OLED exibe um menu rolável com opções de pedidos de ajuda, como "Água", "Comida", "Banheiro", "Remédio", entre outros. O usuário navega pelo menu utilizando um joystick.
- **Seleção de Opções:** Ao selecionar uma opção com o joystick, o sistema envia uma notificação visual (LED verde) e registra a escolha no monitor serial.

- **Nível de Urgência:** O usuário pode definir o nível de urgência do pedido (1 a 9) utilizando o joystick (eixos X e Y) e visualizar o valor na matriz de LEDs 5x5.
- **Modo de Emergência:** Ao pressionar o botão B, o sistema entra em modo de emergência, ativando uma sirene e piscando os LEDs de forma intensa para alertar o cuidador.
- **Matriz de LEDs:** A matriz de LEDs exibe o nível de urgência selecionado ou é desligada conforme a preferência do usuário (botão A).
- **Comunicação com o Cuidador:** O sistema pode ser expandido para enviar as informações ao cuidador via Bluetooth, Wi-Fi, IoT ou um aplicativo de celular, permitindo que ele receba notificações em tempo real sobre os pedidos de ajuda e emergências.

## 1.5 Justificativa

O projeto se justifica pela necessidade de proporcionar uma solução acessível e eficiente para pessoas que enfrentam dificuldades de comunicação ou locomoção. Muitas vezes, essas pessoas dependem de cuidadores para atender suas necessidades básicas, mas nem sempre conseguem se expressar de forma clara ou rápida. O **HelpEase** oferece uma interface simples e intuitiva, permitindo que o usuário solicite ajuda de maneira rápida e eficaz, melhorando sua qualidade de vida e reduzindo a carga de trabalho dos cuidadores. A possibilidade de integração com sistemas embarcados ou aplicativos de celular amplia a eficácia do sistema, garantindo que o cuidador seja notificado imediatamente em caso de necessidade.

## 1.6 Originalidade

Existem projetos correlatos que abordam a comunicação assistiva e a solicitação de ajuda, mas o **HelpEase** se destaca pela combinação de funcionalidades e pela simplicidade de uso. Abaixo estão alguns exemplos de projetos similares:

- **Sistemas de Comunicação Alternativa (AAC):**
  - Dispositivos como o **Tobii Dynavox** e **Proloquo2Go** são usados para ajudar pessoas com dificuldades de comunicação. No entanto, esses sistemas são complexos e caros, enquanto o **HelpEase** é uma solução acessível e de baixo custo.
- **Sistemas de Alerta para Idosos:**
  - Projetos como o **Life Alert** oferecem botões de emergência para idosos, mas não possuem funcionalidades de menu interativo ou indicadores de urgência.
- **Dispositivos de Controle por Joystick:**
  - Alguns projetos utilizam joysticks para controle de interfaces, mas não são voltados especificamente para a comunicação assistiva.
- **Matriz de LEDs para Indicadores Visuais:**

- A matriz de LEDs é comumente usada em projetos de sinalização, mas sua aplicação em conjunto com um sistema de comunicação assistiva é inovadora.

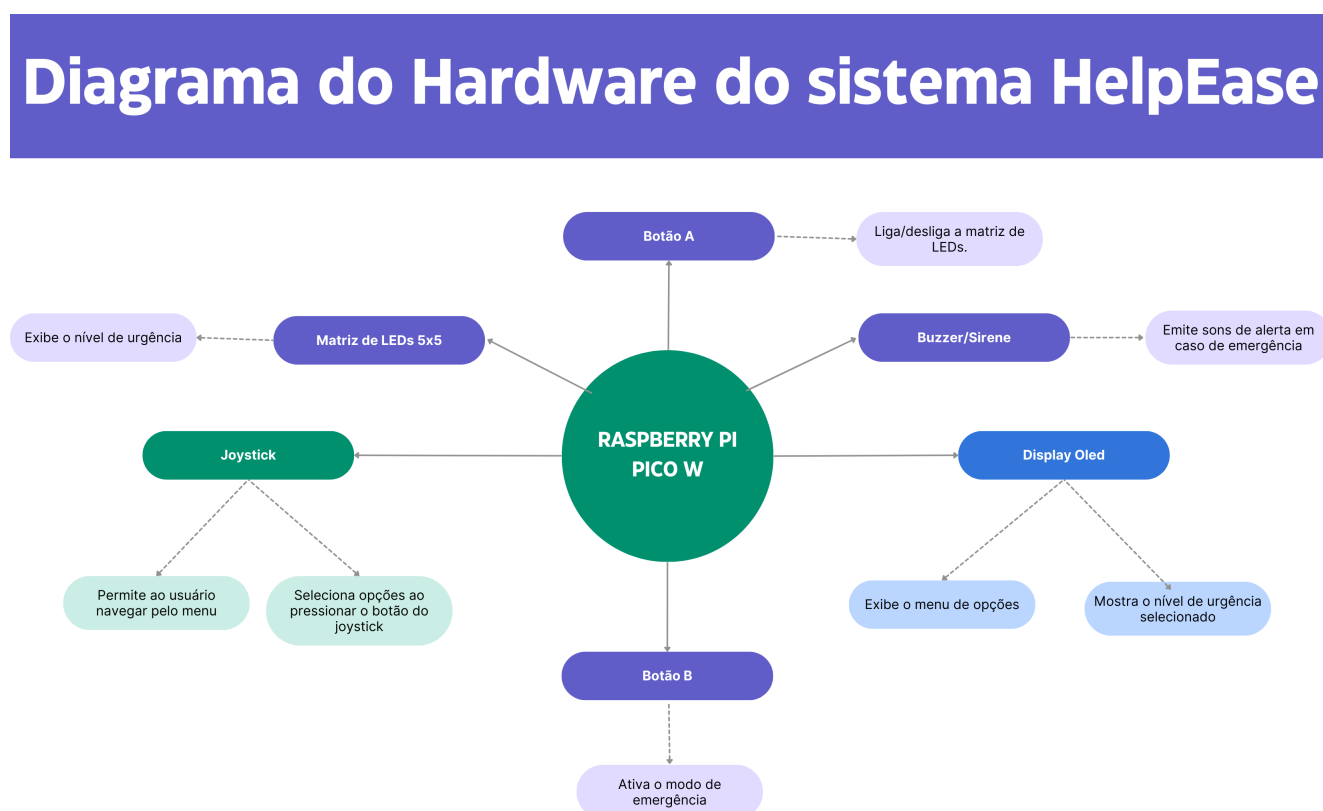
O **HelpEase** se diferencia por integrar todas essas funcionalidades em um único dispositivo, com foco na simplicidade, acessibilidade e eficiência. Além disso, a possibilidade de integração com sistemas embarcados ou aplicativos de celular para notificação do cuidador é um diferencial inovador.

## 2 Especificação do Hardware

### 2.1 Diagrama em Bloco

O diagrama em bloco do projeto **HelpEase** é composto pelos seguintes blocos principais e suas interligações:

Figura 1: Diagrama em bloco do sistema HelpEase.



Fonte: Próprio Autor

### 2.2 Função de Cada Bloco

- **Raspberry Pi Pico W:**
  - Controla todos os componentes do sistema.
  - Processa as entradas do joystick e botões.
  - Gerencia a exibição do menu no display OLED.
  - Controla a matriz de LEDs e o buzzer/sirene.

- **Display OLED:**
  - Exibe o menu de opções ("Água", "Comida", "Banheiro", etc.).
  - Mostra o nível de urgência selecionado.
  - Fornece feedback visual ao usuário.
- **Joystick:**
  - Permite ao usuário navegar pelo menu (eixos X e Y).
  - Seleciona opções ao pressionar o botão do joystick.
- **Matriz de LEDs 5x5:**
  - Exibe o nível de urgência (1 a 9) de forma visual.
  - Pisca em caso de emergência.
- **Botões (A e B):**
  - **Botão A:** Liga/desliga a matriz de LEDs.
  - **Botão B:** Ativa o modo de emergência, acionando a sirene e piscando os LEDs.
- **Buzzer/Sirene:**
  - Emite sons de alerta em caso de emergência.
  - Pode ser configurado para diferentes tons e intensidades.

## 2.3 Configuração de Cada Bloco

- **Raspberry Pi Pico W:**
  - Configurado com o SDK do Raspberry Pi Pico.
  - Utiliza GPIOs para comunicação com os periféricos.
  - Configuração de PWM para controle do buzzer.
- **Display OLED:**
  - Interface I2C para comunicação com o Pico.
  - Resolução de 128x64 pixels.
  - Biblioteca SSD1306 para controle do display.
- **Joystick:**
  - Eixos X e Y conectados a entradas analógicas do Pico (ADC).
  - Botão do joystick conectado a um GPIO com resistor pull-up.
- **Matriz de LEDs 5x5:**
  - Controlada via protocolo WS2812B (NeoPixel).

- Conexão com um único pino de dados do Pico.
- Biblioteca PIO para controle dos LEDs.
- **Botões (A e B):**
  - Conectados a GPIOs com resistor pull-up.
  - Configurados para gerar interrupções ao serem pressionados.
- **Buzzer/Sirene:**
  - Controlado por PWM para gerar diferentes tons.
  - Configurado com um duty cycle variável para controle de volume.

## 2.4 Comandos e Registros Utilizados

- **Raspberry Pi Pico W:**
  - GPIOs: Configuração de entrada/saída.
  - ADC: Leitura dos eixos X e Y do joystick.
  - PWM: Controle do buzzer/sirene.
  - I2C: Comunicação com o display OLED.
  - PIO: Controle da matriz de LEDs WS2812B.
- **Display OLED:**
  - Comandos I2C para inicialização e atualização do display.
  - Biblioteca SSD1306 para desenho de texto e gráficos.
- **Matriz de LEDs:**
  - Comandos PIO para envio de dados no protocolo WS2812B.
  - Registros de controle de cor (RGB) para cada LED.
- **Buzzer/Sirene:**
  - Registros PWM para controle de frequência e duty cycle.

## 2.5 Descrição da Pinagem Usada

A tabela 1 descreve a pinagem utilizada no projeto:



Tabela 1: Pinagem do Raspberry Pi Pico W no projeto HelpEase.

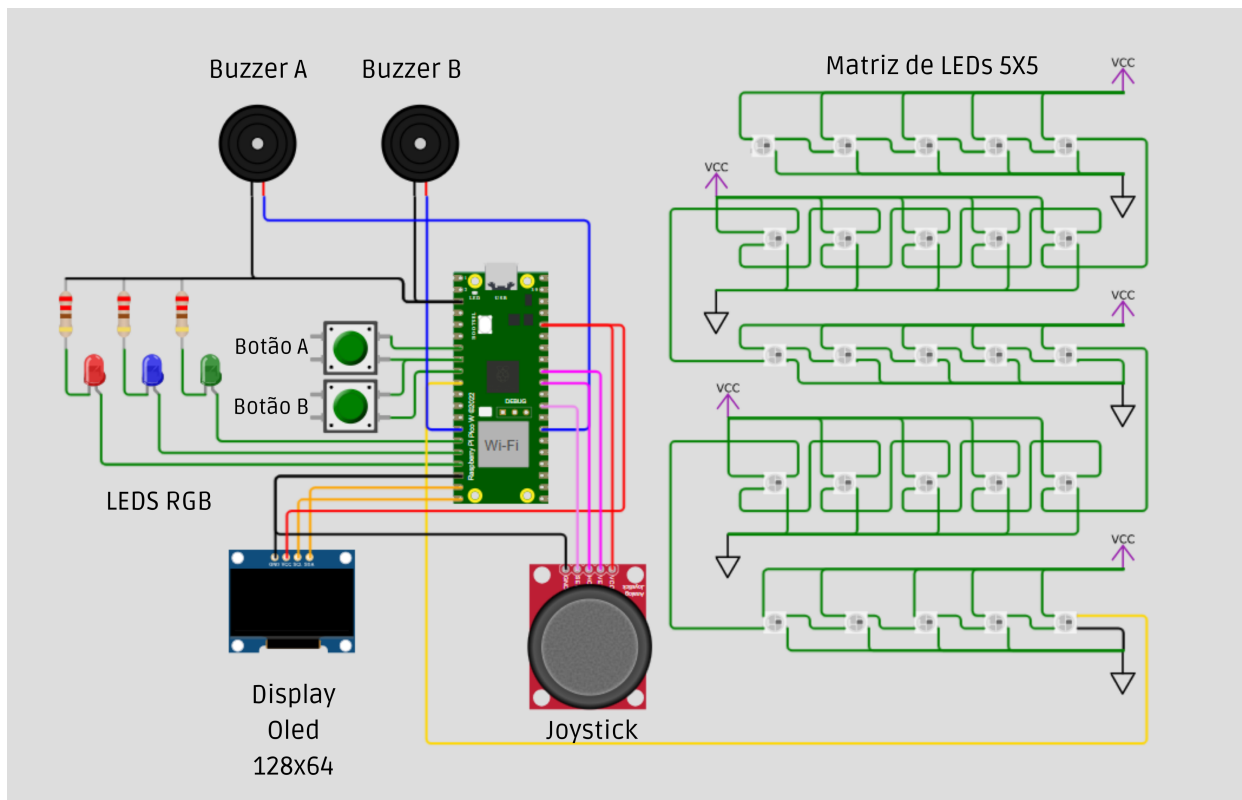
Pino do Pico	Componente	Função
GP0	Display OLED (SDA)	Comunicação I2C (dados)
GP1	Display OLED (SCL)	Comunicação I2C (clock)
GP26	Joystick (Eixo Y)	Leitura analógica (ADC0)
GP27	Joystick (Eixo X)	Leitura analógica (ADC1)
GP22	Joystick (Botão)	Entrada digital (interrupção)
GP5	Botão A	Entrada digital (interrupção)
GP6	Botão B	Entrada digital (interrupção)
GP7	Matriz de LEDs	Saída de dados WS2812B
GP21	Buzzer A	Saída PWM (tom de alerta)
GP10	Buzzer B	Saída PWM (tom de alerta)

**Fonte:** Autoria própria.

## 2.6 Circuito Completo do Hardware

O circuito completo do hardware é mostrado na figura 2 abaixo.

Figura 2: Circuito completo do hardware do projeto HelpEase.



**Fonte:** Próprio Autor

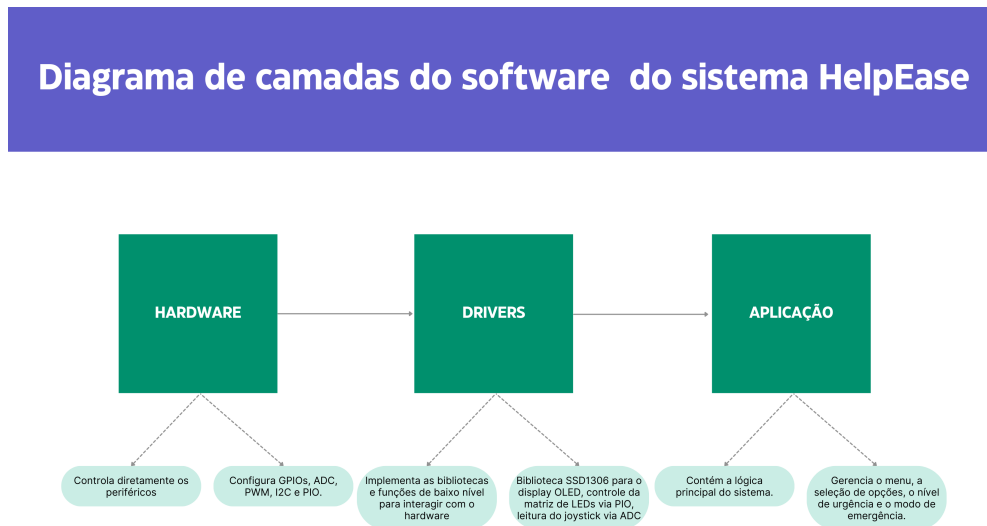
O circuito foi projetado para ser simples e eficiente, utilizando componentes acessíveis e de baixo custo. A pinagem foi escolhida para facilitar a montagem e evitar conflitos entre os periféricos. O uso do Raspberry Pi Pico W permite futuras expansões, como a integração com Bluetooth, Wi-Fi ou IoT.

### 3 Especificação do Firmware

#### 3.1 Blocos Funcionais

O firmware do projeto **HelpEase** é organizado em camadas funcionais, conforme o diagrama abaixo:

Figura 3: Diagrama de camadas do firmware.



- **Camada de Hardware:**

- Controla diretamente os periféricos (display OLED, joystick, matriz de LEDs, botões, buzzer).
- Configura GPIOs, ADC, PWM, I2C e PIO.

- **Camada de Drivers:**

- Implementa as bibliotecas e funções de baixo nível para interagir com o hardware.
- Exemplos: Biblioteca SSD1306 para o display OLED, controle da matriz de LEDs via PIO, leitura do joystick via ADC.

- **Camada de Aplicação:**

- Contém a lógica principal do sistema.
- Gerencia o menu, a seleção de opções, o nível de urgência e o modo de emergência.

#### 3.2 Descrição das Funcionalidades

- **Menu Interativo:**

- Exibe opções de ajuda no display OLED.
- Permite navegação e seleção de opções usando o joystick.

- **Nível de Urgência:**

- Define e exibe o nível de urgência (1 a 9) na matriz de LEDs.
- Atualiza o valor com base no movimento do joystick.
- **Modo de Emergência:**
  - Ativado pelo botão B.
  - Aciona a sirene e pisca os LEDs de forma intensa.
- **Controle de LEDs e Buzzer:**
  - Gerencia a matriz de LEDs e o buzzer/sirene.
  - Utiliza PWM para controle de intensidade e tons.
- **Interrupções:**
  - Gerencia eventos dos botões e joystick.
  - Implementa debouncing para evitar leituras incorretas.

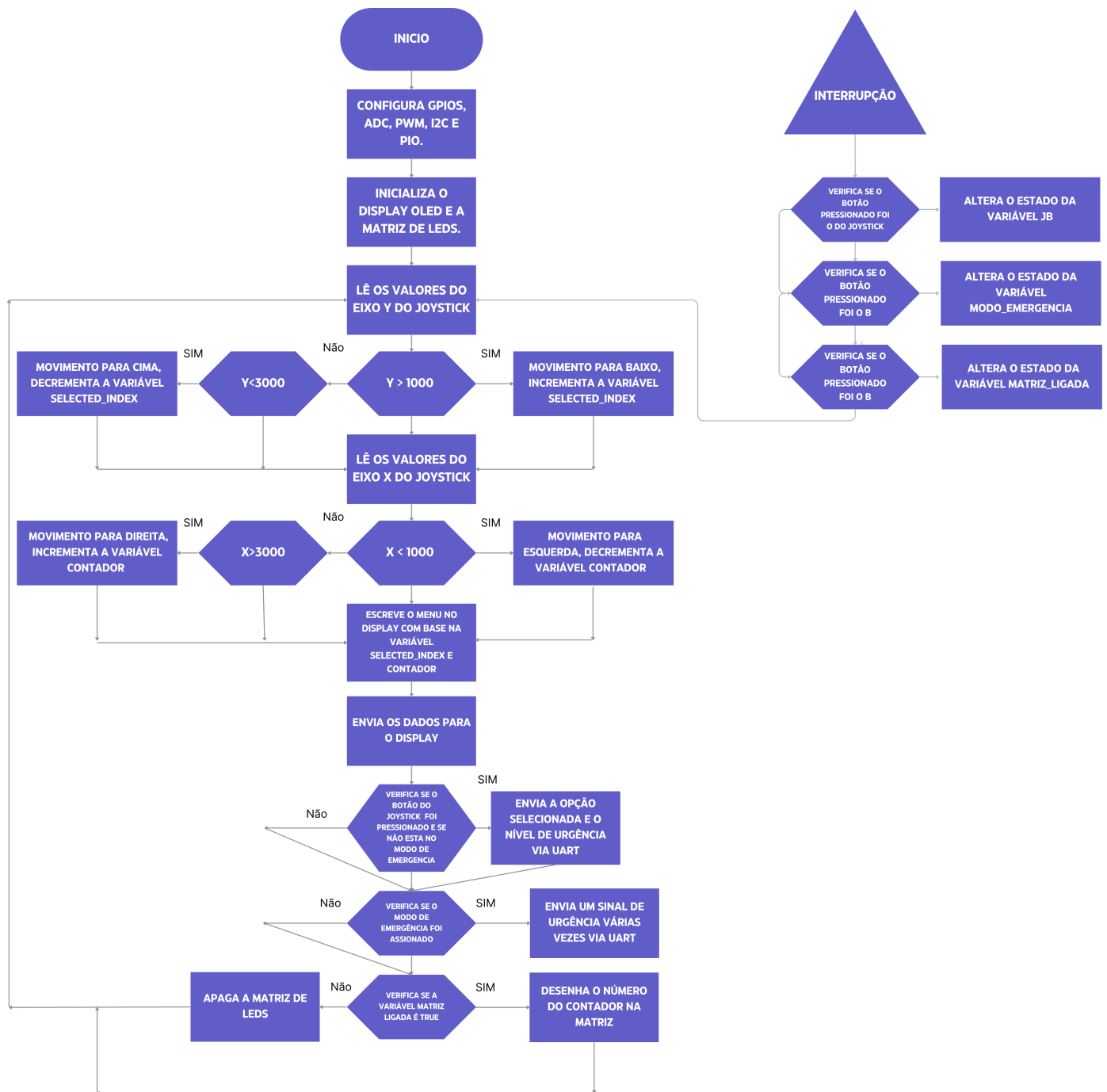
### 3.3 Definição das Variáveis

- **Variáveis Globais:**
  - JB: Estado do botão do joystick (confirmar seleção).
  - modo\_emergencia: Estado do modo de emergência.
  - matriz\_ligada: Estado da matriz de LEDs (ligada/desligada).
  - selected\_index: Índice da opção selecionada no menu.
  - scroll\_offset: Deslocamento do menu para rolagem.
  - Contador: Nível de urgência selecionado (0 a 9).
- **Estruturas de Dados:**
  - pixel\_t: Define a cor de um LED (R, G, B).
  - leds[LED\_COUNT]: Buffer de pixels para a matriz de LEDs.
- **Constantes:**
  - menu\_items[]: Lista de opções do menu.
  - WRAP\_PERIOD: Valor máximo do contador PWM.
  - PWM\_DIVISER: Divisor de clock para o PWM.

### 3.4 Fluxograma

O fluxograma do software pode ser dividido em três partes principais:

Figura 4: Fluxograma do software.



Fonte: Próprio Autor

- **Inicialização:**

- Configura GPIOs, ADC, PWM, I2C e PIO.
- Inicializa o display OLED e a matriz de LEDs.

- **Loop Principal:**

- Lê os valores do joystick (eixos X e Y).

- Atualiza o menu e o nível de urgência.
- Verifica se o modo de emergência foi ativado.
- Controla a matriz de LEDs e o buzzer.
- **Interrupções:**
  - Gerencia eventos dos botões (A, B e joystick).
  - Implementa debouncing para evitar leituras incorretas.

### 3.5 Inicialização

O processo de inicialização do software inclui:

- **Configuração do Hardware:**
  - Inicializa GPIOs, ADC, PWM, I2C e PIO.
  - Configura os pinos dos botões e joystick.
- **Inicialização do Display OLED:**
  - Configura a comunicação I2C.
  - Limpa o display e exibe o menu inicial.
- **Inicialização da Matriz de LEDs:**
  - Configura o protocolo WS2812B via PIO.
  - Limpa a matriz de LEDs.
- **Configuração de Interrupções:**
  - Habilita interrupções para os botões e joystick.
  - Configura a função de callback para tratar os eventos.

### 3.6 Configurações dos Registros

- **GPIO:**
  - Configuração de entrada/saída para botões e LEDs.
  - Uso de resistores pull-up para os botões.
- **ADC:**
  - Configuração dos canais ADC para leitura dos eixos X e Y do joystick.
- **PWM:**
  - Configuração do divisor de clock e valor de wrap para controle do buzzer.
- **I2C:**

- Configuração da comunicação I2C para o display OLED.
- **PIO:**
  - Configuração da máquina de estados para controle da matriz de LEDs.

### 3.7 Estrutura e Formato dos Dados

- **Menu:**
  - As opções do menu são armazenadas em um array de strings (`menu_items[]`).
  - O índice da opção selecionada é armazenado em `selected_index`.
- **Matriz de LEDs:**
  - Cada LED é representado por uma estrutura `pixel_t` (R, G, B).
  - O buffer `leds[LED_COUNT]` armazena o estado de todos os LEDs.
- **Nível de Urgência:**
  - O valor do nível de urgência é armazenado em `Contador` (0 a 9).
  - O valor é exibido na matriz de LEDs usando padrões pré-definidos.

### 3.8 Protocolo de Comunicação

- **I2C:**
  - Usado para comunicação com o display OLED.
  - Configuração: 400 kHz, pinos GP0 (SDA) e GP1 (SCL).
- **PIO (WS2812B):**
  - Usado para controle da matriz de LEDs.
  - Envia dados no formato GRB (Green, Red, Blue) para cada LED.

### 3.9 Formato do Pacote de Dados

- **Display OLED:**
  - Os dados são enviados no formato de comandos e dados I2C.
  - Comandos para inicialização, limpeza e atualização do display.
- **Matriz de LEDs:**
  - Cada LED recebe 3 bytes (G, R, B) no protocolo WS2812B.
  - Os dados são enviados sequencialmente para todos os LEDs.

O firmware do projeto **HelpEase** foi projetado para ser modular e de fácil manutenção. A divisão em camadas (hardware, drivers e aplicação) permite a reutilização de código e a expansão futura do sistema. A inicialização e configuração dos registros garantem o funcionamento correto dos periféricos, enquanto a lógica principal do sistema é implementada de forma eficiente no loop principal e nas interrupções.

## 4 Execução do Projeto

### 4.1 Metodologia

A execução do projeto **HelpEase** seguiu uma metodologia estruturada, dividida em etapas claras para garantir o sucesso do desenvolvimento. Abaixo estão as etapas realizadas:

- **Pesquisas Realizadas:** Foram realizadas pesquisas sobre sistemas de comunicação assistiva, interfaces intuitivas e tecnologias de hardware e software adequadas para o projeto. A escolha dos componentes foi baseada nos disponíveis na placa didática **BitDogLab**, como ADC, botões, joystick, buzzer, display OLED e matriz de LEDs.
- **Escolha do Hardware:** O hardware foi selecionado com base nos componentes disponíveis na placa **BitDogLab** e nas ferramentas aprendidas durante o curso, como GPIO, PIO, PWM, UART, I2C e ADC. Componentes utilizados:
  - **Raspberry Pi Pico W:** Microcontrolador principal.
  - **Display OLED:** Para exibição do menu.
  - **Joystick:** Para navegação no menu.
  - **Matriz de LEDs 5x5:** Para indicar o nível de urgência.
  - **Botões (A e B):** Para controle adicional.
  - **Buzzer/Sirene:** Para alertas sonoros.
- **Definição das Funcionalidades do Software:**
  - O software foi projetado para ser modular, com funcionalidades claras:
    - \* Menu interativo no display OLED.
    - \* Navegação e seleção de opções com joystick.
    - \* Controle da matriz de LEDs para indicar urgência.
    - \* Modo de emergência com sirene e LEDs piscando.
  - A lógica foi implementada usando as ferramentas aprendidas, como interrupções, PWM, PIO e I2C.
- **Inicialização da IDE:** O ambiente de desenvolvimento utilizado foi o **Visual Studio Code** com a extensão **Pico SDK**. O projeto foi configurado para compilar e carregar o firmware no Raspberry Pi Pico W.
- **Programação na IDE:**

- O código foi desenvolvido em C, utilizando as bibliotecas do SDK do Raspberry Pi Pico.
- Foram implementadas funcionalidades como:
  - \* Leitura do joystick via ADC.
  - \* Controle do display OLED via I2C.
  - \* Gerenciamento da matriz de LEDs via PIO.
  - \* Controle do buzzer via PWM.
- O código foi organizado em funções modulares para facilitar a manutenção e expansão.
- **Depuração:** A depuração foi realizada usando o monitor serial para verificar o funcionamento das funcionalidades. Foram corrigidos problemas como debouncing dos botões, ajuste da sensibilidade do joystick e sincronização da matriz de LEDs.

## 4.2 Testes de Validação

Para validar o funcionamento do projeto, foram realizados os seguintes testes:

- **Teste do Menu:**
  - Verificação da exibição correta das opções no display OLED.
  - Teste de navegação no menu usando o joystick.
  - Confirmação de seleção de opções com o botão do joystick.
- **Teste do Nível de Urgência:**
  - Verificação da exibição do nível de urgência na matriz de LEDs.
  - Teste de alteração do nível de urgência usando o joystick.
- **Teste do Modo de Emergência:**
  - Ativação do modo de emergência com o botão B.
  - Verificação do funcionamento da sirene e dos LEDs piscando.
- **Teste de Interrupções:**
  - Verificação do tratamento correto das interrupções dos botões e joystick.
  - Teste de debouncing para evitar leituras incorretas.
- **Teste de Comunicação:**
  - Verificação da comunicação I2C com o display OLED.
  - Teste do envio de dados para a matriz de LEDs via PIO.



### 4.3 Discussão dos Resultados

Os testes realizados demonstraram que o projeto **HelpEase** atende aos requisitos propostos, com resultados positivos em todas as funcionalidades principais. O menu interativo no display OLED funciona corretamente, permitindo a navegação e seleção de opções de forma intuitiva. O nível de urgência é exibido com precisão na matriz de LEDs, atualizando-se em tempo real conforme o movimento do joystick. Além disso, o modo de emergência foi implementado com sucesso, acionando a sirene e fazendo os LEDs piscarem de forma intensa para alertar o cuidador.

Em termos de confiabilidade, o sistema mostrou-se estável durante os testes, com respostas rápidas e precisas às entradas do usuário. A implementação de debouncing nas interrupções garantiu que os eventos dos botões e joystick fossem tratados de maneira adequada, evitando leituras incorretas e garantindo a estabilidade do sistema.

No entanto, algumas limitações foram identificadas. A ausência de comunicação sem fio (Bluetooth ou Wi-Fi) limita a notificação do cuidador em tempo real, o que poderia ser resolvido com a integração de módulos de comunicação sem fio em versões futuras. Além disso, a interface do menu, embora funcional, pode ser aprimorada com a inclusão de ícones ou gráficos para facilitar a compreensão, especialmente para usuários com dificuldades visuais ou cognitivas.

## 5 Conclusão

O projeto **HelpEase** demonstrou ser uma solução tecnológica viável e eficaz para auxiliar pessoas em situações de vulnerabilidade, como idosos, pacientes em recuperação ou indivíduos com dificuldades de locomoção. Através da integração de componentes acessíveis e ferramentas aprendidas durante o curso, foi possível desenvolver um sistema que facilita a comunicação de necessidades básicas e emergenciais de forma simples e intuitiva.

A implementação do menu interativo no display OLED, combinada com a navegação por joystick, mostrou-se uma interface amigável e de fácil utilização. A matriz de LEDs, utilizada para indicar o nível de urgência, e o modo de emergência, com sirene e LEDs piscantes, agregaram funcionalidades essenciais para situações críticas. A escolha dos componentes disponíveis na placa didática **BitDogLab** e o uso de ferramentas como GPIO, PIO, PWM, I2C e ADC permitiram a criação de um sistema robusto e de baixo custo.

Os testes realizados validaram o funcionamento do sistema, comprovando sua confiabilidade e eficiência. No entanto, o projeto também evidenciou oportunidades de melhoria, como a inclusão de comunicação sem fio (Bluetooth ou Wi-Fi) para notificação em tempo real do cuidador e a adição de ícones ou gráficos no menu para melhorar a usabilidade.

Em síntese, o **HelpEase** representa um avanço significativo no desenvolvimento de soluções assistivas, destacando-se pela simplicidade, acessibilidade e potencial de expansão. Futuras versões podem incorporar novas funcionalidades, como integração com aplicativos móveis e sensores adicionais, ampliando ainda mais seu impacto e utilidade. O projeto não apenas atende às necessidades imediatas de seus usuários, mas também abre caminho para inovações tecnológicas que promovam a inclusão e a qualidade de vida de pessoas em situações de vulnerabilidade.

## Referências

RASPBERRY PI FOUNDATION. **Raspberry Pi Pico Datasheet**. Disponível em: <<https://datasheets.raspberrypi.com/pico/pico-datasheet.pdf>>. Acesso em: 24 de fevereiro de 2025.

ARDUINO. **Documentação Oficial do Arduino**. Disponível em: <<https://www.arduino.cc/reference/en/>>. Acesso em: 24 de fevereiro de 2025.

MICROCHIP. **WS2812B Datasheet**. Disponível em: <<https://cdn-shop.adafruit.com/datasheets/WS2812B.pdf>>. Acesso em: 24 de fevereiro de 2025.

TEXAS INSTRUMENTS. **Understanding I2C**. Disponível em: <<https://www.ti.com/lit/an/slva704/slva704.pdf>>. Acesso em: 24 de fevereiro de 2025.

ADAFRUIT. **SSD1306 OLED Display Guide**. Disponível em: <<https://learn.adafruit.com/monochrome-oled-breakouts>>. Acesso em: 24 de fevereiro de 2025.

BITDOGLAB. **Documentação da Placa Didática BitDogLab**. Disponível em: <<https://bitdoglab.com.br/documentacao>>. Acesso em: 24 de fevereiro de 2025.