

# Relatório do Projeto

## Apresentação

Este projeto visa criar um assistente virtual interativo usando a placa BitDogLab RP2040, controlando uma matriz de LEDs 5x5 para representar as emoções do assistente. O assistente recebe mensagens do usuário através dos botões e as emoções podem ser alteradas com o (ou o não) pressionamento, e o assistente também emite sons via buzzer para alertar ou responder ao usuário, e mostra mensagens através do Display OLED.

## Assistente Virtual Interativo

### Objetivos do projeto

O objetivo deste projeto é desenvolver um sistema embarcado inovador que represente emoções de um assistente virtual através de uma matriz de LEDs 5x5. Esse sistema deve ser capaz de responder dinamicamente às interações do usuário por meio de botões físicos e fornecer feedback sonoro utilizando um buzzer. A ideia principal é criar uma experiência interativa onde o assistente virtual demonstre reações emocionais semelhantes às humanas, tornando a interação mais natural e envolvente. Além de servir como um experimento prático em sistemas embarcados, este projeto também pode ser aplicado em contextos educacionais, de acessibilidade e até mesmo em interfaces homem-máquina mais intuitivas.

O projeto explora conceitos essenciais de hardware e software embarcados, como controle de GPIOs, tratamento de interrupções, debounce de botões, manipulação de matrizes de LEDs, e geração de sinais sonoros com o buzzer. Assim, ele também serve como um estudo aprofundado sobre sistemas de resposta rápida e interatividade, promovendo aprendizado prático em eletrônica e programação embarcada.

## Descrição do funcionamento

O sistema embarcado é baseado em uma matriz de LEDs 5x5 que exibe diferentes expressões faciais, refletindo o estado emocional do assistente virtual. Esse estado pode ser modificado pelo usuário utilizando três botões físicos, cada um responsável por uma ação específica:

O usuário "faz perguntas" pressionando botões, ou reinicia as emoções do Assistente.

- Botão 1 = Pergunta "Como você está?".
- Botão 2 = Pergunta "Me dê um conselho"
- Botão 3 = Reinicia o estado emocional para "feliz"

A qualquer momento, o usuário pode redefinir o estado emocional do assistente pressionando o botão de reinicialização, restaurando a expressão "feliz" como padrão.

O assistente responde no display e com LEDs.

- Se estiver "feliz", exibe um rosto sorridente na matriz de LEDs 5x5 e a mensagem "Estou ótimo! Vamos codar!" no display OLED.
- Se estiver "triste" (muito tempo sem interagir), exibe um rosto triste na matriz e "Estou meio solitário..." no display OLED.
- Se for pressionado repetidamente, pode "se irritar" e responder de forma engraçada.

Além das expressões faciais, o assistente reage quando recebe um estímulo prolongado no botão associado à "cócegas". Se o usuário pressionar repetidamente o botão, o sistema interpreta isso como uma ação de cócegas, ativando o buzzer para emitir um som específico e mudando momentaneamente a expressão facial para uma reação de desconforto. O som e a exibição da expressão duram um curto intervalo de tempo antes de retornar ao estado anterior.

O projeto opera de forma contínua, monitorando a entrada dos botões e atualizando as expressões faciais conforme necessário. A estrutura do código segue um modelo baseado em estados, onde cada estado representa uma emoção e as transições entre estados ocorrem conforme os eventos de entrada dos botões.

Cada botão é conectado a um pino digital e utiliza interrupções para garantir resposta rápida e eficiente. Para evitar leituras falsas ou múltiplas detecções indesejadas, um sistema de debounce é implementado.

## Justificativa

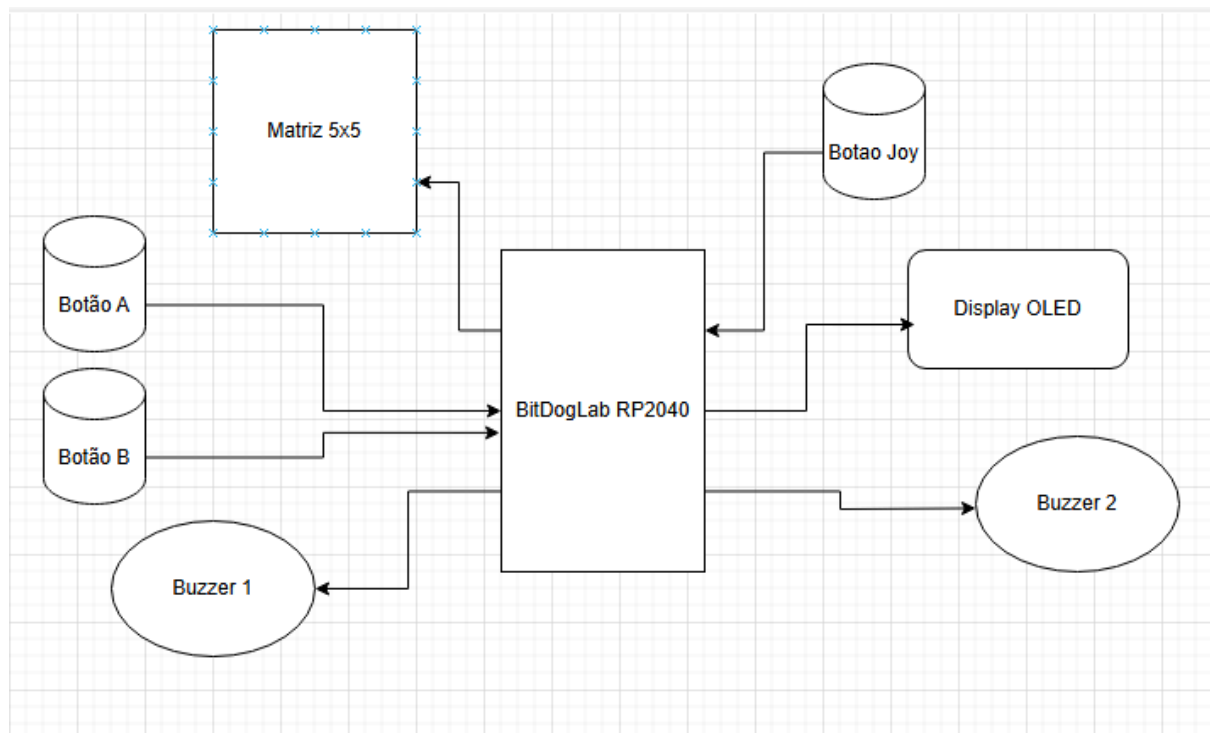
A execução deste projeto justifica-se pela necessidade de criar interfaces de interação mais ricas e dinâmicas em sistemas embarcados. Utilizando apenas a BitDogLab RP2040, o projeto mostra como é possível criar interações emocionais simples em dispositivos de baixo custo e tamanho reduzido. Pode ser usado em diversos contextos, desde educação até entretenimento, e serve como uma excelente ferramenta para aprendizado em sistemas embarcados.

Embora existam projetos correlatos de assistentes virtuais e interações com microcontroladores, a abordagem deste projeto, envolvendo uma simulação de emoções digitais com feedbacks sonoros e visuais integrados, é inovadora. Não foi encontrada nenhuma solução que combine exatamente com esses elementos usando a BitDogLab RP2040.

## Diagrama em bloco

O diagrama em bloco do projeto mostra a interconexão dos principais componentes:

- BitDogLab RP2040 (controlador principal).
- Matriz de LEDs 5x5 (representação das emoções).
- Botões (para mudar emoções e reiniciar o estado).
- Display OLED: Exibição de mensagens
- Buzzers (para alertas de cócegas e mudanças de emoção).



## Função de cada bloco

BitDogLab RP2040 (Microcontrolador Principal):

Atua como o cérebro do sistema, gerenciando todos os periféricos e executando o firmware do assistente.

Processa as entradas dos botões e altera as emoções do assistente conforme as interações.

Controla a matriz de LEDs 5x5 para exibir as expressões faciais.

Gera os sinais necessários para ativar o buzzer e exibir mensagens no display OLED.

Utiliza interrupções para garantir respostas rápidas aos eventos dos botões.

Matriz de LEDs 5x5:

Representa o rosto do assistente, exibindo diferentes expressões faciais com base no estado emocional.

Conectada ao GPIO 7, sendo controlada diretamente pela RP2040.

Cada LED pode ser ligado ou desligado de acordo com padrões predefinidos para formar expressões como "feliz", "triste" ou "raiva".

A matriz responde instantaneamente às mudanças de estado emocional, oferecendo uma interface visual ao usuário.

Botões :

- Botão 1 ou Botão A: Se pressionado 3 vezes em menos de 2 segundos, reinicia o assistente para o estado "feliz".
- Botão 2 ou Botão B: Alterna entre os diferentes estados emocionais do assistente.
- Botão 3 ou Botão do Joystick: Reinicia as emoções para "feliz" de forma imediata.

Funcionam através de interrupções para evitar atrasos e garantir que os comandos sejam reconhecidos sem precisar de polling contínuo.

Buzzers:

Emite diferentes sons dependendo do estado emocional do assistente.

Quando o assistente "sente cócegas", ele ativa um alerta sonoro para indicar desconforto.

Controlado por PWM (Modulação por Largura de Pulso), permitindo variação da frequência do som gerado.

Display OLED (SSD1306, comunicação I2C):

Exibe mensagens textuais relacionadas ao estado emocional do assistente.

Pode mostrar frases como "Estou feliz!", "Estou triste...", ou "Você me faz cócegas pare".

Atualizado em tempo real conforme as interações do usuário.

Usa comunicação I2C para reduzir o número de pinos necessários na RP2040.

## Configuração de cada bloco

Botões:

Configurados como entradas digitais na RP2040.

Utilizam resistores pull-up internos para evitar leitura de valores indefinidos.

Associados a interrupções, garantindo resposta rápida ao pressionamento.

Matriz de LEDs 5x5:

Conectada ao GPIO 7 e operada via tabela de bits no firmware.

O software define padrões de acendimento dos LEDs para representar cada emoção.

A matriz é atualizada sempre que o estado emocional muda.

Buzzers:

Conectado a um pino PWM da RP2040, permitindo controle preciso da frequência do som.

Emite tons diferentes dependendo da interação do usuário.

Ativado apenas quando necessário, economizando energia.

Display OLED (SSD1306, I2C):

Usa dois pinos GPIO para comunicação via barramento I2C.

Atualizado sempre que há uma mudança de estado emocional.

Exibe mensagens curtas para complementar a matriz de LEDs.

## Descrição da pinagem usada

A matriz de LEDs é controlada com um único pino que é a GPIO 7.

Os botões são conectados a outros pinos, são eles:

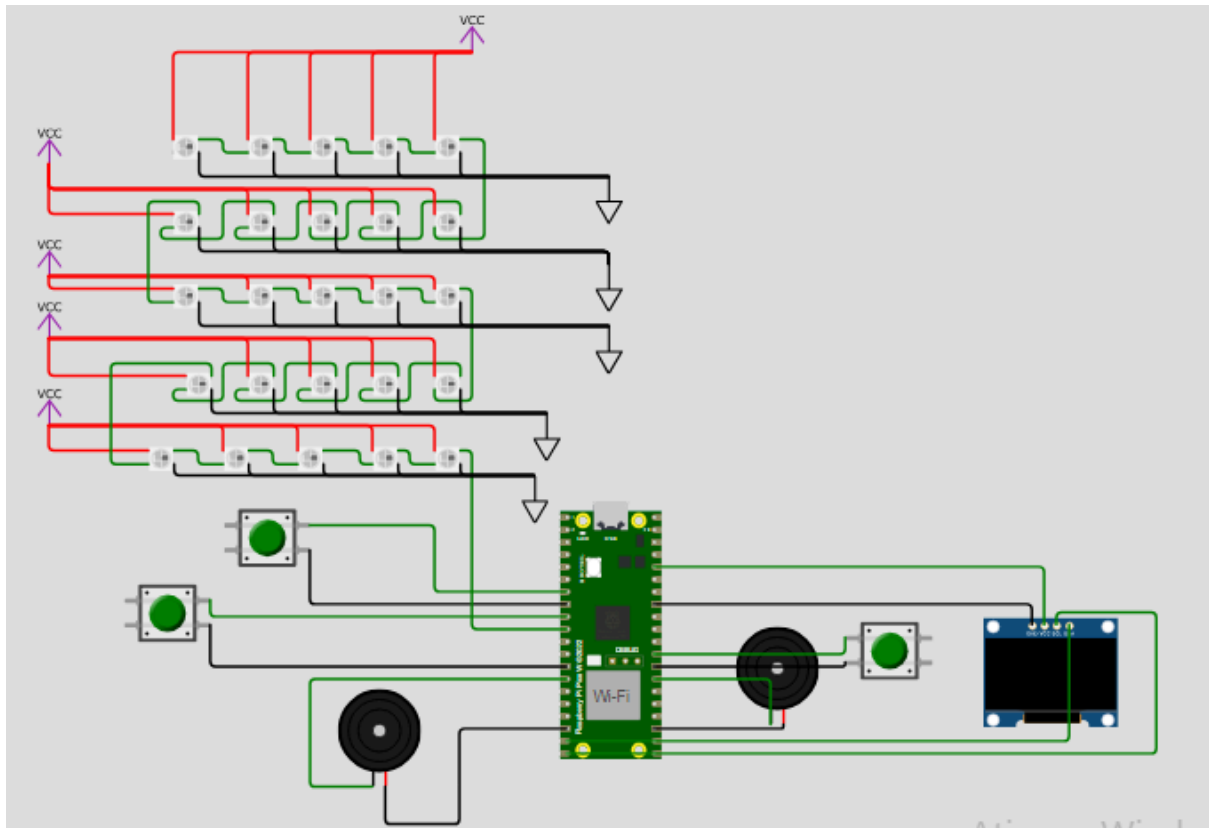
GPIO 5 (Push button A), GPIO 6 (Push button B), GPIO 22 (Joystick Push button).

O buzzer A e o buzzer B conectados, respectivamente, as GPIO 21 e GPIO 10.

Display OLED conectado na GPIO 14 (SDA) e GPIO 15 (SCL).

Componente	Pino GPIO
Matriz de LEDs 5x5	7
Botão 1 ou botão A	5
Botão 2 ou botão B	6
Botão 3 ou botão do Joystick	22
Buzzer 1	21
Buzzer 2	10
Display OLED SDA	14
Display OLED SCL	15

# Diagrama com Circuito completo do hardware:



Foi utilizado a ferramenta Wokwi para a realização desse diagrama;  
Link: <https://wokwi.com/> .

## Blocos funcionais

### CONFIGURAÇÃO INICIAL

Descrição: Este bloco é responsável por configurar o sistema inicial, preparando os periféricos e os recursos necessários para o funcionamento do projeto, como os pinos GPIO, comunicação I2C para o display OLED, PWM para controle de LEDs e outros componentes, e qualquer inicialização necessária para o funcionamento dos sensores e botões.

Entradas: Nenhuma entrada externa. O bloco é iniciado automaticamente no início do processo.

Saídas: Configuração do hardware (GPIOs, I2C, PWM) para uso subsequente.

Objetivo: Garantir que todos os periféricos e recursos estejam prontos para uso antes de iniciar o loop principal do sistema.

## EXIBIÇÃO DA EMOÇÃO INICIAL

Descrição: Este bloco exibe a emoção inicial do assistente no display OLED e na matriz de LEDs. A emoção é geralmente definida como "feliz" no início do sistema.

Entradas: Nenhuma entrada externa. A emoção inicial é definida como "feliz".

Saídas: Emoção exibida no display OLED e na matriz de LEDs.

Objetivo: Mostrar ao usuário o estado inicial do assistente, que é "feliz".

## LEITURA DOS BOTÕES

Descrição: Esse bloco verifica os botões pressionados pelo usuário e determina qual ação o sistema deve realizar em resposta à interação. Ele detecta as entradas dos botões (Botão 1, Botão 2 e Botão 3) e aciona a ação correspondente.

Entradas: Sinais dos botões (Botão 1, Botão 2, Botão 3).

Saídas: Ação correspondente (exibição de resposta emocional ou reset do estado).

Objetivo: Permitir que o assistente interaja com o usuário por meio de pressões de botão, alterando seu comportamento conforme as interações.

## ALTERAÇÃO DA EMOÇÃO

Descrição: Este bloco altera a emoção do assistente com base nas interações do usuário ou no tempo decorrido sem interações. As emoções podem ser "feliz", "triste" ou "raiva", dependendo das condições definidas (como falta de interação ou pressionamento dos botões).

Entradas: Estado dos botões, tempo de inatividade do usuário.

Saídas: Alteração da emoção (feliz, triste, raiva).

Objetivo: Modificar o estado emocional do assistente para refletir as interações ou a falta delas.

## RESPOSTAS INTERATIVAS

Descrição: Esse bloco exibe respostas interativas baseadas na emoção atual do assistente ou nas ações do usuário. Por exemplo, ao pressionar o Botão 2, o assistente pode dar um conselho diferente dependendo de sua emoção (feliz, triste ou raiva).

Entradas: Estado emocional do assistente e ações do usuário (pressionamento de botões).

Saídas: Resposta de texto exibida no display OLED e reação emocional na matriz de LEDs.

Objetivo: Fornecer uma interação mais rica e adaptada com o usuário, dependendo do estado emocional do assistente.

## DETECÇÃO DE CÓCEGAS

Descrição: Este bloco detecta quando os botões 1 e 2 são pressionados simultaneamente em um intervalo de tempo curto (2 segundos). Caso isso aconteça, o assistente muda sua emoção para "raiva", aciona o buzzer e exibe a mensagem "Você me faz cócegas, pare".

Entradas: Estado dos botões 1 e 2, temporizador.

Saídas: Emoção alterada para "raiva", acionamento do buzzer, exibição da mensagem "Você me faz cócegas, pare kkk".

Objetivo: Criar uma interação engraçada e inesperada para o usuário, simulando uma reação de "cócegas" quando os botões são pressionados rapidamente.

## ATUALIZAÇÃO DO DISPLAY E LEDS

Descrição: Esse bloco é responsável por atualizar o display OLED e a matriz de LEDs para refletir a emoção atual do assistente e outras mensagens importantes, garantindo que o usuário veja as reações emocionais do assistente de forma visual.

Entradas: Emoção atual do assistente e outras informações a serem exibidas (mensagens ou estados).

Saídas: Exibição de texto e emoções no display OLED e na matriz de LEDs.

Objetivo: Manter a interface visual atualizada com o estado emocional e interações do assistente.

## MONITORAMENTO DE TEMPO DE INATIVIDADE

Descrição: Esse bloco monitora o tempo de inatividade do usuário e altera a emoção do assistente para "triste" caso não haja interação por um período definido (X segundos).

Entradas: Tempo sem interação do usuário.

Saídas: Mudança de emoção para "triste".

Objetivo: Simular uma reação emocional do assistente quando não há interações, criando uma sensação de engajamento constante.

## Descrição das funcionalidades:

O código gerencia a matriz de LEDs para exibir diferentes expressões faciais.

O controle de botões altera o estado emocional do assistente.

Os buzzers emitem sons conforme o comando.

Gerenciamento de estado emocional: Monitora as interações do usuário e ajusta o estado emocional do assistente, alternando entre "feliz", "triste" e "irritado". A mudança ocorre de acordo com o tempo sem interação ou pelo número de vezes que os botões são pressionados.

Entrada do usuário: Detecta quando um botão é pressionado e interpreta qual ação deve ser tomada, seja para perguntar algo ao assistente ou para resetar seu estado emocional.

Saída visual: Controla a matriz de LEDs 5x5, exibindo expressões faciais correspondentes ao estado emocional do assistente. Além disso, utiliza o display OLED para mostrar mensagens complementares às emoções expressas.

Saída sonora: Ativa o buzzer para fornecer feedback sonoro ao usuário, reforçando os estados emocionais e emitindo alertas em determinadas condições, como quando o assistente "sente cócegas".

## Definição das variáveis:

### Variáveis Globais

As variáveis globais armazenam o estado atual do assistente e controlam a detecção de eventos dos botões.



- `estado_emocional`: Variável do tipo `int` que representa o estado emocional do assistente. Pode assumir os seguintes valores:

`0` → Feliz

`1` → Triste

`2` → Irritado

- `button_1_pressed`, `button_2_pressed`, `button_3_pressed`: Variáveis do tipo `bool`, utilizadas como flags para indicar quando os botões forem pressionados.

## Matrizes para Expressões Faciais

As expressões faciais são armazenadas em matrizes 5x5, onde cada posição representa um LED ligado (1) ou desligado (0).

- `feliz`: Representa a expressão facial quando o assistente está feliz.
- `triste`: Define o padrão de LEDs para quando o assistente está triste.
- `raiva`: Exibe a expressão de raiva na matriz de LEDs.

## Definição de Pinos

Os pinos GPIO são atribuídos a cada componente do sistema, garantindo a comunicação e controle adequados.

Botões:

- `BUTTON_1_PIN` (GPIO 5): Captura a interação "Como você está?".
- `BUTTON_2_PIN` (GPIO 6): Alterna entre estados emocionais.
- `BUTTON_3_PIN` (GPIO 22): Reinicia as emoções para "feliz".

Buzzer:

- `BUZZER_1_PIN` (GPIO 21): Emite sons correspondentes ao estado emocional.
- `BUZZER_2_PIN` (GPIO 10): Auxiliar para variação de tons sonoros.

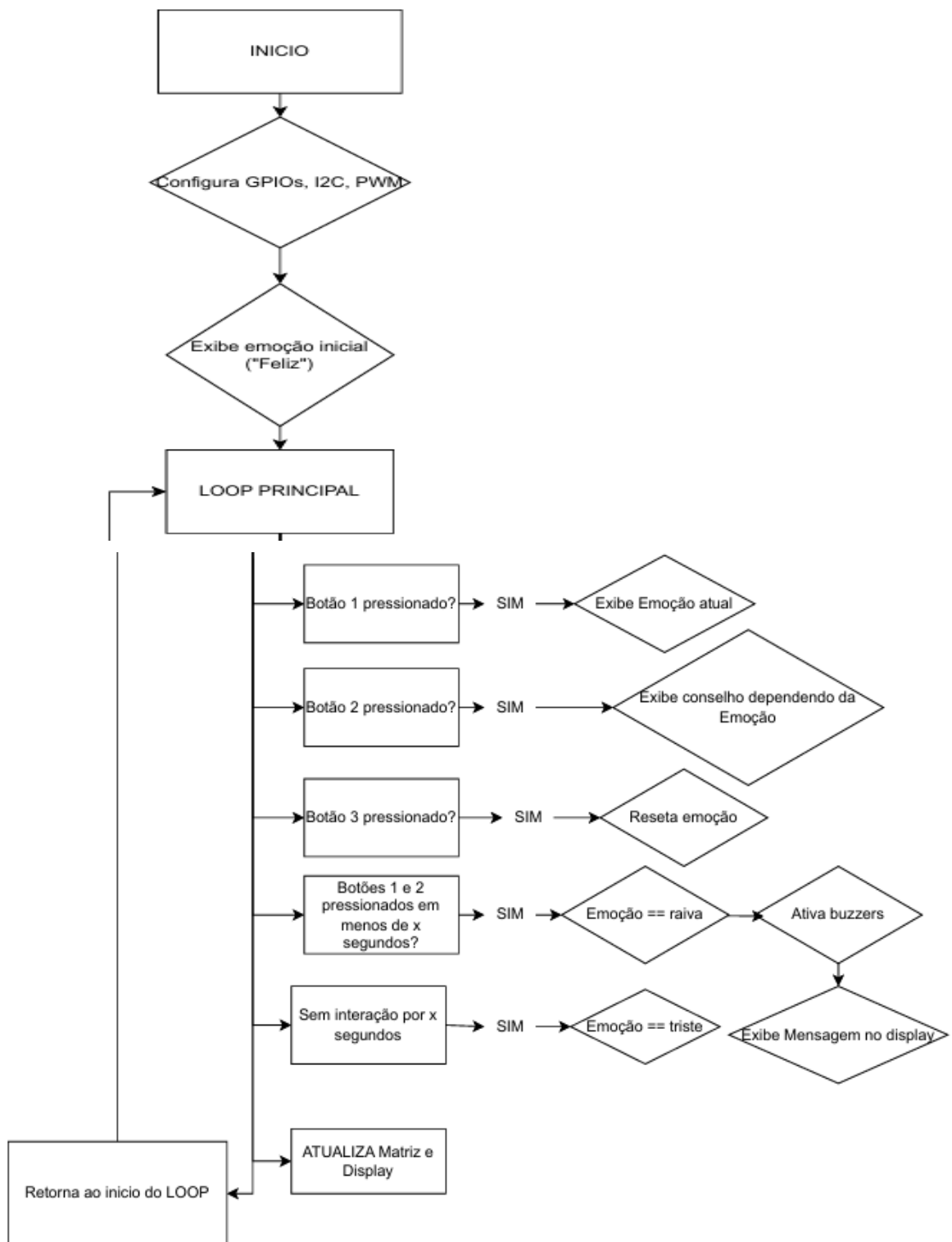
Matriz de LEDs:

- `LED_MATRIX_PIN` (GPIO 7): Controla os LEDs que representam o rosto do assistente.

Display OLED:

- `OLED_SDA_PIN` (GPIO 14): Linha de dados da comunicação I2C.
- `OLED_SCL_PIN` (GPIO 15): Linha de clock da comunicação I2C.

O fluxograma foi dividido em duas imagens.



## ## Inicialização

### CONFIGURAÇÃO DE PARÂMETROS INICIAIS

No início da execução do programa, o software configura as variáveis e os parâmetros necessários. As variáveis de controle do sistema, como a emoção inicial e o tempo de inatividade, são definidas logo no início para garantir que o sistema comece com um estado definido e previsível.

**Emoção Inicial:** O assistente começa com a emoção "feliz", que será exibida no display OLED e na matriz de LEDs 5x5.

**Tempo de Inatividade:** O software configura um temporizador para monitorar o tempo de inatividade, o que permitirá ao assistente alterar sua emoção automaticamente caso o usuário não interaja por um determinado período de tempo.

### CONFIGURAÇÃO DOS PINOS GPIO

A próxima etapa consiste na configuração dos pinos GPIO do microcontrolador. Os pinos são atribuídos às suas respectivas funções, de modo a garantir a comunicação correta entre o microcontrolador e os periféricos conectados, como botões e LEDs.

**Botões (Entrada):** O software configura os pinos conectados aos botões como entradas, permitindo que o sistema detecte quando o usuário interage com o assistente.

**LEDs e Atuadores (Saída):** Os pinos que controlam os LEDs e outros atuadores (como o buzzer) são configurados como saídas, de modo que o software possa controlar o comportamento visual e auditivo do assistente.

### INICIALIZAÇÃO DA COMUNICAÇÃO I2C

O protocolo I2C é utilizado para a comunicação entre o microcontrolador e o display OLED. O software inicializa o barramento I2C para garantir que o microcontrolador possa enviar comandos ao display e exibir as informações sobre a emoção do assistente.

### CONFIGURAÇÃO DO CONTROLE PWM

A modulação por largura de pulso (PWM) é configurada para controlar a intensidade dos LEDs RGB, permitindo que o software ajuste dinamicamente o brilho e as cores dos LEDs conforme a emoção do assistente.

### INICIALIZAÇÃO DO OLED

Após a configuração dos periféricos e dos recursos, o display OLED é inicializado. O software limpa o display e exibe a emoção inicial do assistente, garantindo que o usuário tenha uma interface visual clara para interagir.

### CONFIGURAÇÃO DE TEMPORIZADORES E INTERRUPÇÕES

Para garantir que o assistente reaja à inatividade do usuário ou a eventos específicos, o software configura temporizadores e interrupções. O temporizador monitora o tempo de inatividade, e as interrupções permitem que o sistema reaja rapidamente ao pressionamento dos botões.

## ALOCAÇÃO DE MEMÓRIA

Durante a inicialização, o software aloca a memória necessária para armazenar as variáveis e buffers usados para controlar o comportamento do sistema. Isso inclui a alocação de memória para a emoção atual do assistente, o tempo de inatividade e outros dados temporários importantes.

## ESTADO INICIAL DE PREPARAÇÃO

Após a configuração de todos os parâmetros e componentes, o sistema entra em um estado de preparação. O software está agora pronto para interagir com o usuário. As emoções do assistente serão alteradas com base na interação do usuário, e o sistema estará pronto para operar de forma contínua e eficiente.

## Configurações dos registros:

Configuração dos GPIOs (Entradas e Saídas)--> Os botões (GPIO 5, GPIO 6 e GPIO 22) são configurados como entradas com pull-up interno para detectar pressões. Isso é feito ativando os registradores de controle do GPIO (SIO).

A matriz de LEDs (GPIO 7) e os buzzers (GPIO 21 e GPIO 10) são configurados como saídas digitais para permitir o controle do assistente emocional.

Configuração do PWM para o Buzzer--> Os pinos GPIO 21 e GPIO 10 utilizam PWM para gerar tons sonoros. Para isso, os registradores PWM\_CHx\_CSR (Control and Status Register) são configurados, ativando o modo PWM e ajustando a frequência do sinal.

Configuração do I2C para o Display OLED --> A comunicação com o display OLED ocorre por meio do barramento I2C, utilizando os pinos GPIO 14 (SDA) e GPIO 15 (SCL). Esses pinos são configurados com a função I2C ajustando os registradores I2C\_CTRL (I2C Control Register), garantindo uma taxa de comunicação eficiente.

Configuração dos Temporizadores Internos --> Para monitorar o tempo sem interação e alterar o estado emocional do assistente, são utilizados temporizadores internos do RP2040. Os registradores correspondentes são ajustados para gerar interrupções em intervalos regulares, garantindo que o estado emocional mude dinamicamente.

## Estrutura e formato dos dados:

No software do assistente virtual, os dados são organizados de maneira eficiente para representar as interações do usuário, as emoções do assistente, e o controle dos dispositivos conectados ao microcontrolador. Abaixo estão descritos os principais tipos de dados utilizados no software, suas estruturas e formatos.

## EMOÇÃO DO ASSISTENTE

A emoção do assistente é uma das variáveis mais importantes no sistema, pois determina a expressão visual no display OLED e a matriz de LEDs 5x5, além de afetar a resposta do assistente aos comandos do usuário.

Tipo de Dados: Enumeração (enum)

Formato: A emoção é representada por uma enumeração que contém os possíveis estados emocionais do assistente.

Valor Inicial: o assistente começa com a emoção "feliz", que será exibida logo após a inicialização do sistema.

## TEMPO DE INATIVIDADE DO USUÁRIO

O tempo de inatividade é monitorado para detectar quando o usuário não interage com o assistente por um determinado período. Essa informação é crucial para mudar automaticamente a emoção do assistente caso o usuário não interaja por um tempo.

Tipo de Dados: Inteiro (int)

Formato: Armazena o tempo decorrido em milissegundos ou em unidades de tempo.

## ESTADO DOS BOTÕES (Entradas do Usuário)

Os botões são configurados como entradas digitais que indicam as ações do usuário.

Quando pressionados, eles acionam eventos que alteram a emoção ou fazem o assistente responder de forma específica.

Tipo de Dados: Inteiro (int) ou booleano (bool)

Formato: Cada botão tem um valor associado que pode ser lido como um valor lógico (0 ou 1) indicando se o botão foi pressionado ou não.

## CONTAGEM DE EMOÇÕES E RESPOSTAS

Durante a execução do sistema, o software pode precisar contar quantas vezes uma determinada emoção foi ativada ou registrar interações para gerar respostas.

Tipo de Dados: Inteiro (int)

Formato: Armazena a quantidade de interações ou mudanças de emoção.

## CONTROLE DOS LEDs WS2812(PWM)

Os LEDs WS2812 são controlados via PWM, e a intensidade de cada cor é ajustada conforme a emoção do assistente. A estrutura de dados usada para controlar os LEDs é representada por três variáveis que armazenam os valores de intensidade para as cores vermelha, verde e azul.

Tipo de Dados: Inteiro (int)

Formato: Cada cor (vermelho, verde, azul) é representada por um valor de intensidade variando de 0 a 255.

## Protocolo de comunicação e Formato de pacote de dados

O protocolo de comunicação utilizado no sistema é baseado na interface I2C, responsável pela troca de dados entre o microcontrolador e dispositivos periféricos, como o display OLED SSD1306.

A estrutura do pacote de dados segue o formato padrão do I2C, onde o primeiro byte contém o endereço do dispositivo, seguido por bytes de controle e os dados transmitidos.

Para comunicação serial UART, caso utilizada, os pacotes são estruturados com um byte de início, payload contendo a informação necessária e um byte de verificação para integridade dos dados. Esse formato garante uma comunicação eficiente e confiável entre os componentes do sistema.

## Metodologia

O desenvolvimento do assistente emocional foi realizado seguindo um conjunto de etapas organizadas para garantir um funcionamento eficiente e previsível. A abordagem adotada seguiu um fluxo estruturado, conforme descrito abaixo:

### PESQUISA INICIAL

Antes do desenvolvimento, foi realizada uma pesquisa sobre projetos similares que utilizam expressões faciais em matrizes de LEDs e interação com botões. Além disso, foram analisadas as capacidades da placa BitDogLab RP2040, seus periféricos e protocolos suportados, como GPIO, PWM e I2C. Essa etapa foi essencial para entender as limitações e possibilidades do hardware disponível.

### DEFINIÇÃO DAS FUNCIONALIDADES

Com base na pesquisa, foi elaborado um escopo detalhado das funcionalidades do assistente. A lógica de mudança emocional foi planejada, considerando os diferentes estados emocionais e as condições que os ativam. Foram estabelecidos os seguintes pontos principais:

- Quais botões afetariam cada estado emocional.
- Como a matriz de LEDs exibirá as expressões faciais.
- O uso do display OLED para exibir mensagens relacionadas às emoções.
- O acionamento do buzzer em situações específicas, como alertas e reações.

O Wokwi, um simulador de hardware embarcado, foi escolhido para a prototipagem inicial.

Nele, foram adicionados os componentes necessários:

- Matriz de LEDs 5x5 para exibir expressões faciais.
- Botões físicos para permitir a interação do usuário.
- Display OLED via comunicação I2C.
- Buzzer controlado por PWM.
- RP2040 como unidade central de controle.

Essa configuração permitiu testar o comportamento do sistema sem a necessidade imediata de hardware físico.

### IMPLEMENTAÇÃO DO FIRMWARE

O código foi escrito em C, utilizando as bibliotecas adequadas para a BitDogLab RP2040.

Foram desenvolvidos os seguintes módulos principais:

- Gerenciamento de estados emocionais: lógica para mudar entre "feliz", "triste" e "irritado".
- Leitura dos botões com interrupções: para detectar interações do usuário de forma eficiente.
- Controle da matriz de LEDs: exibição de diferentes expressões faciais conforme o estado emocional.

- Exibição de mensagens no display OLED: comunicação via I2C para mostrar respostas do assistente.
- Acionamento do buzzer: geração de sons conforme a emoção do assistente.

## TESTES E AJUSTES

Após a implementação, foram realizados testes para validar a resposta do sistema. Os principais testes incluíram:

- Pressionamento dos botões para verificar se as emoções mudam corretamente.
- Tempo sem interação para testar se o assistente muda para o estado "triste" automaticamente.
- Exibição correta das expressões faciais na matriz de LEDs.
- Comunicação I2C funcional para exibir mensagens no display OLED.
- Teste do buzzer para garantir que os sons indicam corretamente as emoções e alertas.
- Com base nos testes, pequenos ajustes foram feitos para otimizar a responsividade dos botões e melhorar a suavidade da transição entre estados emocionais.

## Testes de validação

Os testes foram conduzidos para garantir que cada componente do sistema funcionasse conforme esperado e que o assistente emocional interativo respondesse corretamente às interações do usuário. As validações foram realizadas individualmente para cada módulo e, posteriormente, de forma integrada.

### VERIFICAÇÃO DA MATRIZ DE LEDs 5x5

Objetivo: Confirmar que as expressões faciais mudam corretamente conforme o estado emocional do assistente.

Procedimento: Teste inicial com a matriz de LEDs exibindo o rosto "feliz" ao ligar o sistema. Pressionamento dos botões para alterar o estado emocional e observar as mudanças visuais na matriz.

Simulação de falta de interação para verificar a transição automática para o estado "triste".

Teste de múltiplos pressionamentos rápidos para simular o estado de "irritação".

Resultado esperado: A matriz de LEDs exibe expressões correspondentes aos estados emocionais do assistente.

### RESPOSTAS AOS BOTÕES

Objetivo: Garantir que os botões alterem o estado emocional conforme projetado.

Procedimento: Pressionar o botão 1 e verificar se a resposta do assistente corresponde ao estado emocional esperado.

Pressionar o botão 2 para testar a exibição da resposta no display OLED e a mudança no rosto do assistente.

Pressionar repetidamente os botões para observar mudanças de estado e garantir que o assistente não entre em loops inesperados.

Testar o botão 3 (do joystick) para resetar o estado emocional para "feliz".

Resultado esperado: O assistente responde corretamente a cada botão, mudando expressões e mensagens de forma apropriada.

## INTERAÇÃO COM O BUZZER

Objetivo: Validar que o buzzer é acionado corretamente conforme as interações do usuário.

Procedimento: Pressionar um botão e verificar se há um alerta sonoro quando necessário.

Simular "cócegas" repetidas para garantir que o buzzer emite um alerta de sobrecarga.

Monitorar se há variações na frequência dos sons gerados conforme o estado emocional muda.

Resultado esperado: O buzzer responde de maneira apropriada às interações, emitindo alertas sonoros quando necessário.

## MONITORAMENTO DE TEMPO SEM INTERAÇÃO

Objetivo: Testar se o assistente altera seu estado emocional de forma autônoma quando não há interação por um período prolongado.

Procedimento: Inicializar o sistema e aguardar sem pressionar nenhum botão.

Observar se, após um tempo pré-definido, o assistente muda para um estado "triste".

Verificar se, ao interagir novamente, o assistente retorna ao estado "feliz".

Resultado esperado: O assistente se torna "triste" após um período de inatividade e retorna ao normal ao receber interações.

## Discussão dos Resultados:

Os testes mostraram que o sistema responde bem às interações do usuário. O assistente consegue alterar seu estado emocional dinamicamente e oferece um retorno visual e sonoro eficaz.

Pequenos ajustes foram feitos para melhorar a responsividade dos botões e a suavidade da mudança de emoções.

## Referências

RASPBERRY PI. RP2040 Datasheet. Disponível em:

<https://datasheets.raspberrypi.com/rp2040/rp2040-datasheet.pdf>. Acesso em: 18 fev. 2025.

BITDOGLAB. Documentação técnica da placa BitDogLab RP2040. Disponível em:

<https://github.com/BitDogLab/BitDogLab>. Acesso em: 18 fev. 2025.

ADAFRUIT INDUSTRIES. Adafruit SSD1306 Library. Disponível em:

[https://github.com/adafruit/Adafruit\\_SSD1306](https://github.com/adafruit/Adafruit_SSD1306). Acesso em: 20 fev. 2025.

WOKWI. Simulador Online para RP2040. Disponível em: <https://wokwi.com>. Acesso em: 20 fev. 2025.

ARTIGOS SOBRE EXPRESSÕES FACIAIS EM MATRIZES DE LED: Estudos indicam que expressões simplificadas podem ser eficazmente representadas com pequenos conjuntos de LEDs, melhorando a interação humano-máquina.

SANTIAGO, Hemir da Cunha. "Reconhecimento de Expressões Faciais." Tese de Doutorado, Universidade Federal de Pernambuco, 2017. Disponível em:



<https://repositorio.ufpe.br/bitstream/123456789/24892/1/TESE%20Hemir%20da%20Cunha%20Santiago.pdf>. Acesso em 18 fev. 2025.

PROJETOS SIMILARES: Análise de dispositivos interativos utilizando LEDs para exibição de estados emocionais em ambientes educacionais e robóticos. Pesquisa feita em 17 fev. 2025.