

Relatório: Projeto final - Sistema de monitoramento e ajuda à saúde mental

Aulo Cezar Ferreira dos Anjos Filho

tic370101098

26 de fevereiro de 2025

Apresentação do Projeto

O projeto *Neurokardy – Monitoramento Vestível e Intervenção em Saúde Mental* propõe o desenvolvimento de um dispositivo inovador, acoplado ao braço, que integra monitoramento fisiológico contínuo e feedback em tempo real para auxiliar na gestão da ansiedade, do estresse e de outros desafios relacionados à saúde mental. Por meio de sensores de alta performance que captam sinais vitais – como tensão muscular, frequência cardíaca e qualidade do ar – e algoritmos de análise, o sistema converte esses dados em alertas, incentivando a prática de técnicas de relaxamento e a conscientização corporal.

Objetivos do Projeto

Desenvolver um dispositivo vestível que capture, de forma contínua e não invasiva, sinais fisiológicos relevantes associados ao estado emocional do usuário.

Fornecer feedback multimodal – visual, sonoro e interativo – incentivando o usuário a realizar intervenções, como técnicas de respiração e relaxamento, quando forem detectadas alterações significativas nos parâmetros fisiológicos.

Personalizar a calibração dos sensores, ajustando os limiares de alerta conforme as características individuais de cada usuário para garantir maior precisão e eficácia no monitoramento.

Incorporar elementos lúdicos e interativos que promovam o engajamento do usuário, tornando o acompanhamento da saúde mental uma experiência intuitiva e motivadora.

Descrição do Funcionamento

O dispositivo emprega um conjunto de sensores estratégicos para a captação de dados fisiológicos:

- Sensor de eletromiografia (EMG): registra a tensão muscular e identifica padrões de rigidez associados ao estresse.
- Sensor de frequência cardíaca: monitora os batimentos do coração com precisão, fornecendo dados essenciais para a análise do estado emocional.
- Sensor de qualidade do ar: avalia as condições ambientais e contribui para a compreensão do impacto do ambiente na saúde mental.

Esses sensores são integrados a um microcontrolador que processa os sinais em alta resolução, realizando uma calibração inicial baseada em uma amostra robusta de dados para estabelecer limiares personalizados. Ao identificar desvios significativos que indiquem um estado instável, o sistema aciona alertas visuais – por meio de animações em um display OLED e mudanças na cor de uma matriz LED – e sonoros, utilizando buzzer com frequências ajustadas de acordo com a gravidade das alterações. Adicionalmente, o dispositivo oferece modos interativos, como jogos simples, que atuam como ferramentas de distração e promoção do relaxamento, contribuindo para a neuroplasticidade adaptativa.

Justificativa

Diante do crescente impacto dos transtornos relacionados à ansiedade e ao estresse – que comprometem a qualidade de vida e a produtividade de milhões de pessoas – há uma demanda urgente por soluções acessíveis e eficazes. Métodos convencionais de monitoramento muitas vezes se mostram invasivos ou economicamente inviáveis. O *Neurokardy* destaca-se ao oferecer uma abordagem inovadora e não invasiva, combinando a precisão do monitoramento fisiológico com intervenções em tempo real. Essa solução não só possibilita a detecção precoce de estados de desregulação emocional, mas também promove a autogestão da saúde mental, permitindo uma resposta imediata e personalizada diante de crises.

Originalidade

Projetos Correlatos e Referências

Embora diversas iniciativas se dediquem ao monitoramento fisiológico e ao biofeedback para o manejo da ansiedade, o *Neurokardy* diferencia-se pela integração de sensores que captam parâmetros essenciais – como tensão muscular, frequência cardíaca e qualidade do ar – com a oferta de feedback em tempo real e intervenções adaptativas.

Micro:Calm (do your :bit Challenge, 2021)

Descrição: Desenvolvido por Alicia e Esmée no Canadá, o *Micro:Calm* utiliza a plataforma micro:bit para criar um dispositivo multifuncional que propicia exercícios de respiração, checagem emocional e jogos interativos, oferecendo estratégias práticas para o gerenciamento da ansiedade e promovendo a saúde e o bem-estar.

Diferencial do Neurokardy: Enquanto o *Micro:Calm* opera com atividades pré-definidas, o *Neurokardy* integra sensores que monitoram continuamente sinais fisiológicos, permitindo uma análise dinâmica do estado emocional e a personalização das intervenções.

Biofeedback baseado em microcontrolador para gestão de ansiedade (Rus Bordas, 2021)

Referência: Projeto acadêmico que utiliza Arduino e sensores dedicados para monitorar a frequência cardíaca, a tensão muscular e a sudorese, fornecendo biofeedback para o controle da ansiedade.

Diferencial do Neurokardy: O *Neurokardy* emprega sensores de alta performance – como o sensor EMG para a captação da tensão muscular, o sensor de frequência cardíaca para monitoramento dos batimentos e o sensor de qualidade do ar para avaliação ambiental – integrados em um sistema coeso que permite uma análise abrangente e em tempo real do estado fisiológico do usuário, possibilitando intervenções imediatas.

Anxiety Detection using IoT (Navya et al., 2022)

Referência: Projeto que detecta episódios de ansiedade em pacientes com TDAH através do monitoramento de sinais fisiológicos, utilizando sensores de pulso e temperatura conectados à nuvem.

Diferencial do Neurokardy: Além do monitoramento passivo, o *Neurokardy* incorpora mecanismos de feedback adaptativo que estimulam o usuário a intervir ativamente – por meio de técnicas de respiração e relaxamento –, promovendo uma resposta imediata e eficaz diante de alterações fisiológicas.

Estudos sobre Biofeedback e Neuroplasticidade (NCBI, 2023)

Referência: Pesquisas demonstram que o feedback imediato dos parâmetros fisiológicos favorece a reestruturação neural e a formação de hábitos saudáveis.

Importância dos Sensores para Controle da Ansiedade

Cada sensor utilizado no *Neurokardy* desempenha um papel crítico na detecção e gestão da ansiedade:

Frequência Cardíaca (Eixo X do Joystick):

Relevância: A variabilidade da frequência cardíaca (VFC) está diretamente ligada ao estresse. Valores elevados indicam a ativação do sistema nervoso simpático (“luta ou fuga”), comum em crises de ansiedade.

Base Científica: Estudos da American Psychological Association (APA) demonstram que a VFC é um marcador confiável de estresse agudo.

Qualidade do Ar (Eixo Y do Joystick):

Relevância: Poluentes, como CO₂ e amônia, afetam a oxigenação sanguínea, podendo desencadear sintomas de ansiedade (ex.: falta de ar, taquicardia).

Dados: Segundo a Organização Mundial da Saúde (OMS), 90% da população global respira ar poluído, agravando transtornos mentais em áreas urbanas.

Tensão Muscular (Microfone como EMG):

Relevância: A contração muscular crônica é um sintoma físico comum na ansiedade generalizada. O sensor EMG detecta rigidez muscular, alertando para a necessidade de relaxamento.

Estudo: Pesquisa publicada no *Journal of Clinical Psychology* associa a tensão muscular a 70% dos casos de ansiedade clinicamente diagnosticados.

1 Especificação do hardware

1.1 Diagrama em bloco do projeto

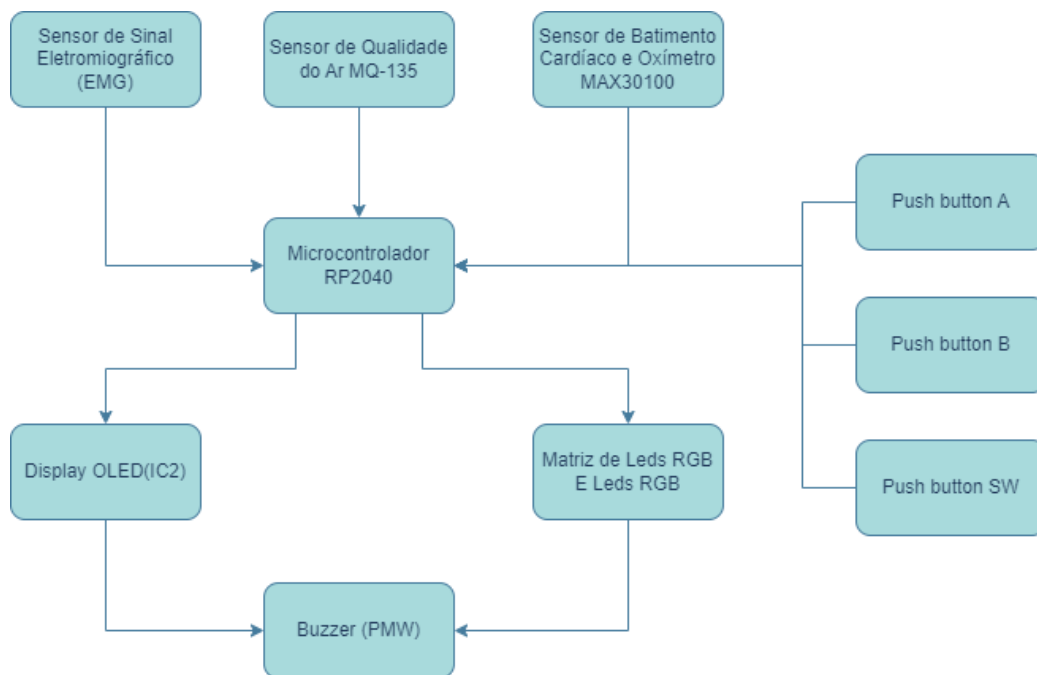


Figura 1: Diagrama em bloco do projeto. Fonte: Autoria própria.

1.2 Função de Cada Bloco

Sensor EMG (Microfone): Captura sinais elétricos musculares simulados através do microfone, permitindo a análise de tensão muscular, detecção de fadiga e monitoramento de esforço físico em tempo real.

MAX30100 (Simulado pelo Eixo X do Joystick): Monitora a frequência cardíaca e a saturação de oxigênio no sangue (SpO), auxiliando na detecção de arritmias e hipóxia. No protótipo atual, este sensor é simulado utilizando o eixo X do joystick.

MQ-135 (Simulado pelo Eixo Y do Joystick): Detecta poluentes ambientais (amônia, benzeno, CO) para monitoramento da qualidade do ar. No protótipo atual, este sensor é simulado utilizando o eixo Y do joystick.

RP2040: Microcontrolador central que processa os dados dos sensores, coordena a comunicação entre componentes e executa o algoritmo de análise para determinação do estado do sistema (estável vs. instável).

Botões:

- **Botão A:** Alterna entre o modo normal de monitoramento e o modo de jogo da cobra.
- **Botão B:** Inicia o processo de calibração dos sensores.
- **Botão Sw:** Reinicia o jogo da cobra quando no modo de jogo.

OLED 128×64: Interface visual que apresenta as leituras dos sensores, estado do sistema e interface do jogo da cobra. Exibe diferentes animações dependendo do estado do sistema.

Matriz LED 5050: Fornece feedback visual através de diferentes padrões e cores (verde para estado estável, vermelho para estado instável) com intensidade variável baseada no desvio dos sensores.

Buzzer: Emite alertas sonoros quando os valores dos sensores excedem os limiares calibrados.

1.3 Configuração de Cada Bloco

Sensor EMG (Microfone):

- Pino de conexão: GPIO28 (ADC2)
- Modo de operação: Entrada analógica (ADC)
- Taxa de amostragem: Aproximadamente 8 kHz (intervalo de amostragem de 125 s)
- Fator de desvio para detecção de instabilidade: $1.0 \times$ o desvio padrão calibrado
- Método de calibração: Média e desvio padrão calculados a partir de 100.000 amostras

MAX30100 (Simulado pelo Eixo X do Joystick):

- Pino de conexão: GPIO26 (ADC0)
- Modo de operação: Entrada analógica (ADC)
- Taxa de amostragem: Compartilhada com outros sensores (aprox. 8 kHz)
- Fator de desvio para detecção de instabilidade: $3.0 \times$ o desvio padrão calibrado

- Método de calibração: Média e desvio padrão calculados a partir de 100.000 amostras

MQ-135 (Simulado pelo Eixo Y do Joystick):

- Pino de conexão: GPIO27 (ADC1)
- Modo de operação: Entrada analógica (ADC)
- Taxa de amostragem: Compartilhada com outros sensores (aprox. 8 kHz)
- Fator de desvio para detecção de instabilidade: $2.0 \times$ o desvio padrão calibrado
- Método de calibração: Média e desvio padrão calculados a partir de 100.000 amostras

RP2040:

- Velocidade de clock: Padrão (125MHz)
- Modo de operação ADC: 12 bits, 0-4095 faixa de valores
- Intervalo de atualização de animação: 400 ms
- Debounce dos botões: 200 ms
- Algoritmo de filtro: Confirmação necessária de 3 leituras consecutivas para mudança de estado
- Memória utilizada: Flash interna para armazenamento de bitmaps e código

Botões:

- **Botão A:** GPIO5, configurado como entrada com pull-up interno
- **Botão B:** GPIO6, configurado como entrada com pull-up interno
- **Botão Sw:** GPIO22, configurado como entrada com pull-up interno
- Modo de interrupção: IRQ em borda de descida (EDGE_FALL)
- Tempo de debounce: 200 ms

OLED 128×64:

- Interface: I2C (i2c1)
- Pinos de conexão: SDA = GPIO14, SCL = GPIO15
- Endereço I2C: 0x3C
- Velocidade I2C: 400 kHz
- Resolução: 128×64 pixels
- Modo de operação: Monocromático
- Taxa de atualização: 400 ms para animações

Matriz LED 5050:

- Pino de dados: GPIO7
- Protocolo: NeoPixel/WS2812
- Arranjo: Matriz 5×5
- Intensidade base: 0.05 (5% do brilho máximo)
- Modos de cor: Verde para estado estável, Vermelho para estado instável

Buzzers:

- Pino de conexão: 10 e 21
- Frequência do buzzer 1: 1000 Hz
- Frequência do buzzer 2: 1500 Hz
- Modo de operação: PWM
- Ativação: Quando os valores dos sensores excedem o limiar calibrado

LEDs RGB (GPIO13, GPIO11):

- **GPIO11:** Cor verde para indicar que o sistema está operando normalmente.
- **GPIO13:** Cor vermelha para quando os valores dos sensores excedem o limiar calibrado.

1.4 Comandos e Registros Utilizados

Nesta subseção são listados os comandos e registros empregados na configuração dos diversos periféricos e recursos avançados do sistema, abrangendo ADC, GPIO, I2C, temporizadores e a configuração avançada da matriz de LEDs RGB via PIO.

ADC

- `adc_init()` — Inicializa o ADC.
- `adc_gpio_init()` — Configura os pinos GPIO como entradas ADC.
- `adc_select_input()` — Seleciona o canal ADC (0, 1 ou 2).
- `adc_read()` — Lê o valor atual do canal ADC selecionado.

GPIO

- `gpio_init()` — Inicializa os pinos GPIO.
- `gpio_set_dir()` — Define a direção do pino (entrada ou saída).
- `gpio_pull_up()` — Ativa o resistor pull-up interno.
- `gpio_set_function()` — Define a função alternativa do pino (por exemplo, para I2C, PWM ou PIO).
- `gpio_set_irq_enabled_with_callback()` — Configura a interrupção e associa uma função callback.

I2C

- `i2c_init()` — Inicializa a interface I2C com a velocidade especificada.
- `i2c_write_blocking()` — Envia dados via I2C (utilizado, por exemplo, na comunicação com o display OLED).

Temporizadores

- `to_us_since_boot()` — Obtém o tempo em microssegundos desde o boot do sistema.
- `get_absolute_time()` — Obtém o tempo absoluto do sistema.
- `sleep_ms()` — Introduz um atraso em milissegundos.
- `sleep_us()` — Introduz um atraso em microssegundos.

Configuração da Matriz de LEDs RGB via PIO

- `pio_add_program()` — Adiciona o programa `ws2818b_program` ao PIO, essencial para o controle dos LEDs RGB.
- `pio_claim_unused_sm()` — Aloca uma máquina de estado (SM) no PIO para gerenciar o programa.
- `pio_gpio_init()` — Inicializa o pino GPIO utilizado pelo PIO.
- `pio_sm_set_consecutive_pindirs()` — Configura os pinos consecutivos como saída para a máquina de estado do PIO.
- `ws2818b_program_init()` — Função que encapsula a configuração completa do programa PIO para os LEDs WS2818B, incluindo:
 - `sm_config_set_sideset_pins()` — Define o pino utilizado para *side-set*.
 - `sm_config_set_out_shift()` — Configura o deslocamento (shift) da saída, com 8 bits.
 - `sm_config_set_fifo_join()` — Configura o FIFO para uso exclusivo de transmissão (TX).
 - `sm_config_set_clkdiv()` — Ajusta o divisor de clock (prescaler) para definir a frequência do PWM, utilizando a função `clock_get_hz(clk_sys)`.
 - `pio_sm_init()` — Inicializa a máquina de estado do PIO com a configuração definida.
 - `pio_sm_set_enabled()` — Habilita a máquina de estado do PIO.

Programa WS2818B (Assembly para PIO)

- **Diretivas do Programa:**
 - `.program ws2818b` — Inicia a definição do programa para controle dos LEDs RGB.
 - `.side_set 1` — Habilita o uso de *side-set* para controle simultâneo dos pinos durante a execução.
 - `wrap_target` e `wrap` — Delimitam o trecho do código que será repetido ciclicamente.
- **Instruções Específicas:**

- Instruções como `out x, 1`, `jmp !x, 3`, `jmp 0` e `nop` com tempos de espera (ex.: `side 0 [2]`, `side 1 [1]`) são utilizadas para sincronizar precisamente o sinal de saída para os LEDs.
- `clock_get_hz(clk_sys)` — Recupera a frequência do clock do sistema para o cálculo do divisor de clock no PWM.

1.5 Descrição da Pinagem Utilizada

Pino	Função	Tipo	Configuração
GPIO5	Botão A (Modo Jogo)	Entrada Digital	Pull-up, IRQ em borda de descida
GPIO6	Botão B (Calibração)	Entrada Digital	Pull-up, IRQ em borda de descida
GPIO7	Matriz LED (WS2812)	Saída Digital (PIO)	Frequência específica do PIO
GPIO10	Buzzer 1	Saída Digital (PWM)	1000 Hz, PWM
GPIO21	Buzzer 2	Saída Digital (PWM)	1500 Hz, PWM
GPIO11	LED RGB - Status (Verde)	Saída Digital (PWM)	Indicador de operação normal
GPIO13	LED RGB - Status (Vermelho)	Saída Digital (PWM)	Indicador de sensor fora do equilíbrio
GPIO14	I2C1 SDA (OLED)	I2C	Pull-up, 400 kHz
GPIO15	I2C1 SCL (OLED)	I2C	Pull-up, 400 kHz
GPIO22	Botão Sw (Reset Jogo)	Entrada Digital	Pull-up, IRQ em borda de descida
GPIO26	Eixo X Joystick (MAX30100)	Entrada Analógica (ADC0)	—
GPIO27	Eixo Y Joystick (MQ-135)	Entrada Analógica (ADC1)	—
GPIO28	Microfone (EMG)	Entrada Analógica (ADC2)	—

Tabela 1: Mapeamento dos pinos e suas funções

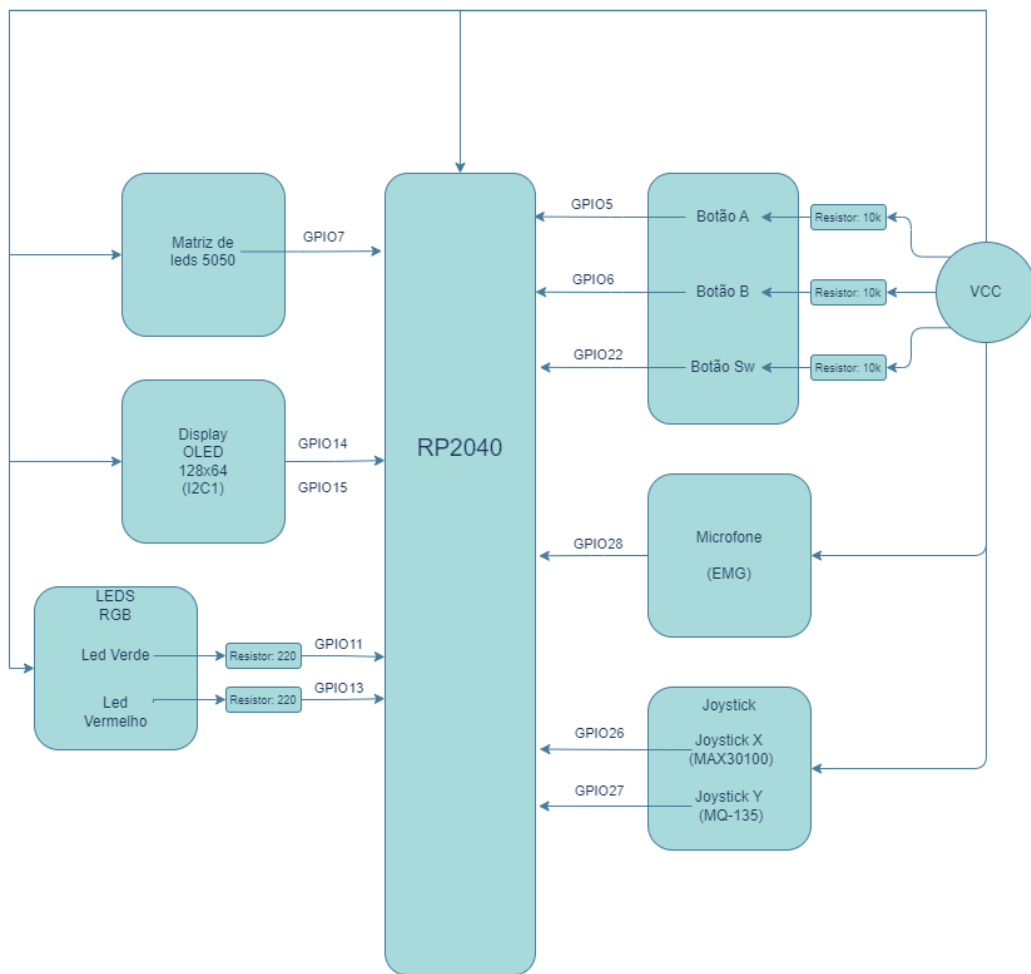


Figura 2: Esquema do circuito do projeto. Fonte: Autoria própria.

2 Especificação do firmware

2.1 Diagrama em bloco

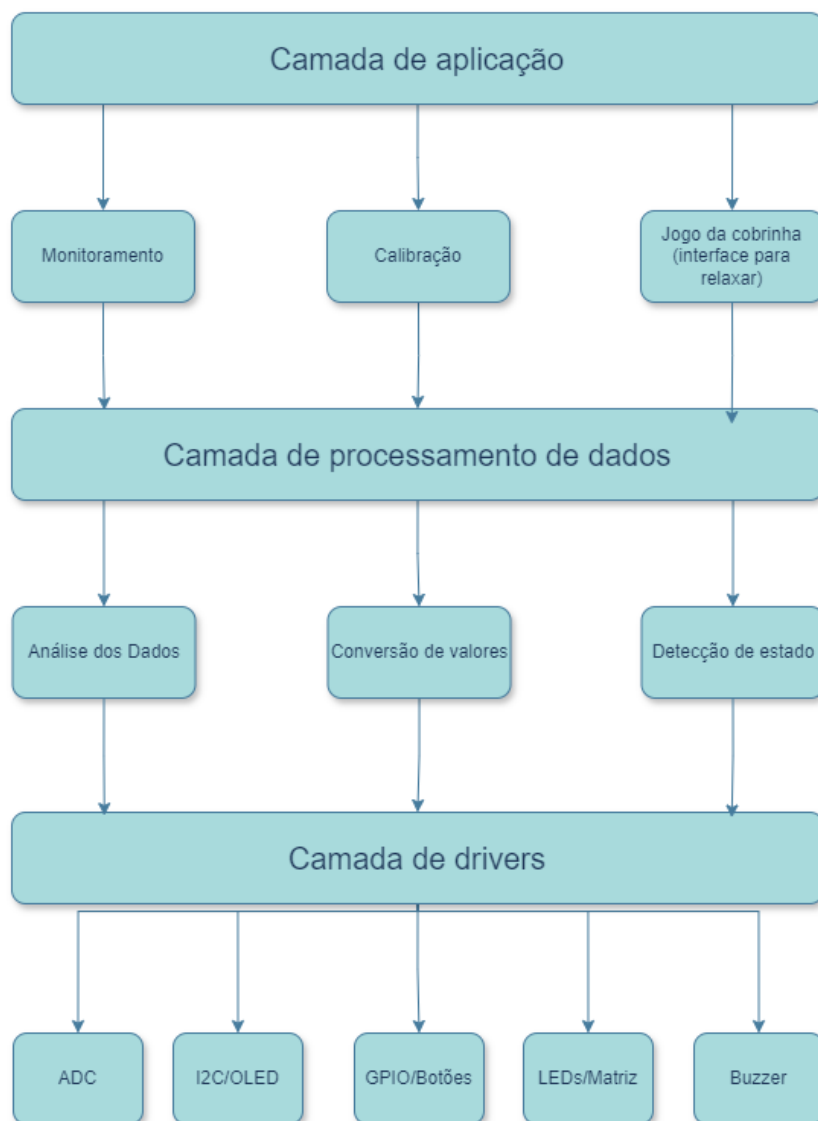


Figura 3: Esquema do circuito do projeto. Fonte: Autoria própria.

2.2 Descrição das Funcionalidades

2.2.1 Camada de Aplicação

Monitoramento: Responsável pela leitura contínua dos sensores, análise dos dados e exibição do estado do usuário (estável ou instável). Inclui a visualização de métricas de EMG (tensão muscular), batimentos cardíacos e oximetria (MAX30100) e qualidade do ar (MQ-135).

Calibração: Gerencia o processo de calibração dos sensores, coletando 100.000 amostras para calcular a média e o desvio padrão de cada sensor. Estes valores são utilizados para determinar os limiares de estado estável/instável.

Jogo da Cobra: Implementa o jogo Snake como uma funcionalidade de diversão/relaxamento, usando o joystick para controle e a matriz LED 5x5 para exibição.

2.2.2 Camada de Processamento de Dados

Análise de dados: Processa os sinais brutos dos sensores, aplicando filtragem e cálculos estatísticos para determinar se os valores estão dentro dos parâmetros calibrados.

Conversão de Valores: Converte as leituras ADC em unidades significativas (percentual de tensão muscular, frequência respiratória, índice de qualidade do ar).

Deteção de Estado: Implementa o algoritmo que determina se o sistema está em estado estável ou instável, exigindo um número mínimo de leituras consecutivas (`CONFIRMACAO_NECESSARIA`) para confirmar uma transição de estado.

2.2.3 Camada de Drivers

ADC: Gerencia as entradas analógicas para leitura dos sensores (microfone para EMG, joystick para simulação do MAX30100 e MQ-135).

I2C/OLED: Controla a comunicação I2C com o display OLED para exibição de informações, status e animações.

GPIO/Botões: Gerencia as interrupções e leitura dos botões, implementando debounce e tratamento de eventos.

LEDs/Matriz: Controla a matriz LED 5x5 e os LEDs RGB para feedback visual do estado do sistema.

Buzzer: Gerencia a emissão de alertas sonoros quando os valores dos sensores excedem os limiares calibrados.

2.3 Definição das Variáveis

2.3.1 Variáveis de Configuração

- VRY_PIN, VRX_PIN, SW_PIN: Pinos do joystick (27, 26, 22)
- BUTTON_A, BUTTON_B: Pinos dos botões (5, 6)
- microfone: Pino do microfone (28)
- I2C_PORT, I2C_SDA, I2C_SCL, endereco: Configurações do I2C (i2c1, 14, 15, 0x3C)

2.3.2 Variáveis de Estado

- calibracao_loading: Flag que indica se o processo de calibração está em andamento
- calibracao_realizada: Flag que indica se a calibração foi concluída
- estado_atual: Estado atual do sistema (0=indefinido, 1=estável, 2=instável)
- contador_confirmacao: Contador para confirmar transição de estado
- jogo_da_cobra: Flag que indica se o jogo Snake está ativo

2.3.3 Variáveis de Calibração

- Eletromiografia_calibrado: Valor médio calibrado do microfone (EMG)
- Sensor_de_Respiracao_calibrado: Valor médio calibrado do joystick eixo X (MAX30100)
- Sensor_de_Qualidade_do_Ar_calibrado: Valor médio calibrado do joystick eixo Y (MQ-135)
- Eletromiografia_calibrado_desvio: Desvio padrão calibrado do microfone
- Sensor_de_Respiracao_calibrado_desvio: Desvio padrão calibrado do joystick eixo X
- Sensor_de_Qualidade_do_Ar_calibrado_desvio: Desvio padrão calibrado do joystick eixo Y

2.3.4 Variáveis de Temporização

- `last_button_time`: Timestamp da última interação com botão (debounce)
- `last_calibration_time`: Timestamp da última atualização da animação de calibração
- `last_running_time`: Timestamp da última atualização da interface normal

2.3.5 Constantes de Configuração

- `CONFIRMACAO_NECESSARIA`: Número de leituras consecutivas para confirmar mudança de estado (3)
- `EMG_FATOR_DESVIO`, `RESP_FATOR_DESVIO`, `AR_FATOR_DESVIO`: Fatores multiplicadores para desvio padrão (1.0, 3.0, 2.0)
- `DEBOUNCE_DELAY`: Tempo de debounce para botões (200.000 μs)
- `ANIMATION_UPDATE_INTERVAL`: Intervalo de atualização da animação (400.000 μs)

2.4 Fluxograma do projeto

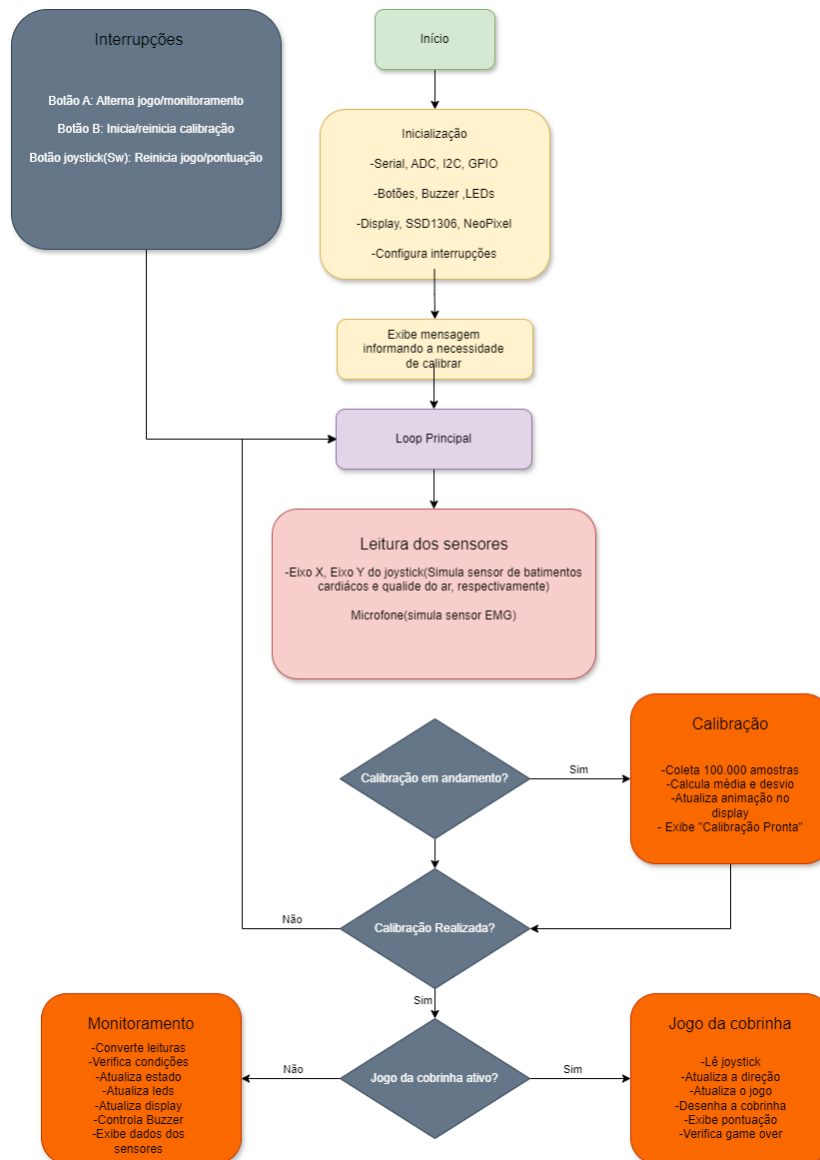


Figura 4: Fluxograma do projeto. Fonte: Autoria própria.

2.5 Inicialização

O processo de inicialização do software segue a seguinte sequência:

- Inicialização da comunicação serial (`stdio_init_all()`) para permitir comunicação com o computador via UART.

- Inicialização do ADC (`adc_init()`) e configuração dos pinos para leitura analógica:
 - Pino 26 (`VRX_PIN`) para o eixo X do joystick.
 - Pino 27 (`VRY_PIN`) para o eixo Y do joystick.
 - Pino 28 (microfone) para o sensor EMG.
- Inicialização do I2C (`i2c_init(I2C_PORT, 400 * 1000)`) na velocidade de 400 kHz.
- Inicialização dos botões:
 - `SW_PIN` (pino 22) para o botão do joystick.
 - `BUTTON_A` (pino 5).
 - `BUTTON_B` (pino 6).
- Inicialização do Buzzer (`buzzer_init()`).
- Configuração dos pinos dos botões como entradas com resistores de pull-up.
- Configuração dos pinos I2C com funções apropriadas e pull-ups.
- Inicialização do display SSD1306 via I2C.
- Inicialização dos LEDs comuns (`led_init()`).
- Inicialização da matriz de LEDs 5x5 (`npInit(7)`) no pino 7.
- Configuração das interrupções para os botões.
- Inicialização do gerador de números aleatórios (`srand(time_us_32())`).
- Exibição da mensagem de calibração pendente no display.

O sistema aguarda 1 segundo após a inicialização antes de entrar no loop principal.

2.6 Configurações dos Registros

2.6.1 Registros ADC

- Configura os canais ADC para pinos específicos usando `adc_gpio_init()` e `adc_select_input()`:
 - **ADC0**: Eixo Y do joystick (`VRY_PIN`, pino 27).
 - **ADC1**: Sensor MAX30100 (`VRX_PIN`, pino 26).
 - **ADC2**: Microfone/EMG (pino 28).

2.6.2 Registros I2C

- Configura o periférico I2C1 (`I2C_PORT`) a 400 kHz.
- Define os pinos para SDA (pino 14) e SCL (pino 15).
- Ativa resistores de pull-up internos.

2.6.3 Registros GPIO

- Configuração para pinos de entrada com `gpio_init()` e `gpio_set_dir()`.
- Ativação de resistores de pull-up com `gpio_pull_up()`.
- Configuração de interrupções com `gpio_set_irq_enabled_with_callback()`.

2.7 Estrutura e Formato dos Dados

O software utiliza várias estruturas de dados para gerenciar o estado do sistema, incluindo:

- Dados do display SSD1306.
- Dados de calibração para sensores.
- Estados do sistema.
- Estruturas específicas para o jogo da cobra e interpretação dos sensores.

2.8 Protocolo de Comunicação

O software utiliza dois protocolos principais:

- **Protocolo I2C:** Utilizado para comunicação com o display OLED SSD1306, configurado a 400 kHz.
- **Protocolo Serial (UART):** Utilizado para depuração, exibindo valores de calibração e estados do sistema.
- **Protocolo para Matriz de LEDs:** Controla os LEDs 5x5 utilizando o PIO do Raspberry Pi Pico.

2.9 Formato do Pacote de Dados

- Pacotes I2C para SSD1306 incluem bytes de comando e dados conforme o protocolo do display.
- Estruturas de dados contêm informações sobre tensão muscular, batimentos cardíacos e qualidade do ar.

2.10 Código e Bibliotecas Utilizadas

O software utiliza bibliotecas padrão do Raspberry Pi Pico, bem como bibliotecas personalizadas para controle dos periféricos. Algumas das principais bibliotecas incluem:

- `pico/stdlib.h` para controle geral do hardware.
- `hardware/adc.h` para controle do ADC.
- `hardware/i2c.h` para comunicação I2C.
- `hardware/gpio.h` para manipulação de GPIOs.
- `stdio.h` para entrada e saída via UART.
- `stdlib.h` para funções padrão da linguagem C.
- `math.h` para operações matemáticas.
- `hardware/pwm.h` para controle PWM.
- `BIBLIOTECAS/ssd1306.h` para controle do display OLED.
- `BIBLIOTECAS/buzzer.h` para controle do buzzer.

- BIBLIOTECAS/desenho.h para armazenamento de desenhos.
- BIBLIOTECAS/leds.h para controle de LEDs.
- BIBLIOTECAS/matrizRGB.h para controle da matriz de LEDs RGB.
- BIBLIOTECAS/menu.h para estruturação do menu do sistema.
- BIBLIOTECAS/bitmaps.h para armazenamento de bitmaps.
- BIBLIOTECAS/conversor.h para conversão de dados.
- BIBLIOTECAS/snake.h para gestão do jogo da cobra.

Metodologia

Execução das Etapas do Projeto

A construção do *Neurokardy* foi estruturada em diversas fases, desde pesquisas e definições iniciais até a implementação, testes e validação.

Pesquisas Realizadas

O desenvolvimento do projeto baseou-se em uma ampla pesquisa envolvendo artigos científicos, relatórios acadêmicos e fontes online confiáveis. As referências citadas anteriormente foram fundamentais para a fundamentação teórica do sistema. Além disso, estudou-se o conceito de **sistema de feedback**, um mecanismo fundamental para o funcionamento do *Neurokardy*.

O sistema de feedback opera com base na ideia de que, ao receber um sinal informando uma alteração em seu estado, o indivíduo adapta seu comportamento para corrigir ou otimizar determinada condição. Esse conceito é amplamente utilizado em neurociência, psicologia e engenharia de sistemas. No contexto do *Neurokardy*, sempre que os sensores identificam uma variação fora do padrão esperado, o dispositivo emite alertas visuais e sonoros. Esses alertas servem como estímulos que auxiliam o usuário a ajustar sua respiração, postura ou outras variáveis fisiológicas, promovendo um retorno ao equilíbrio corporal e mental.

Escolha do Hardware

A seleção dos componentes teve como objetivo principal desenvolver um dispositivo capaz de monitorar a saúde mental e fornecer feedback em tempo real. O projeto foi concebido para reduzir a ansiedade e o estresse, além

de fornecer informações sobre diferentes parâmetros do corpo humano, como qualidade do ar e sinais fisiológicos.

Os sensores escolhidos incluem:

- **Sensor EMG:** Responsável pela captação da tensão muscular e detecção de rigidez excessiva.
- **Sensor de Frequência Cardíaca (MAX30100):** Mede os batimentos cardíacos e a saturação de oxigênio no sangue.
- **Sensor de Qualidade do Ar (MQ-135):** Analisa os níveis de poluentes e fornece alertas sobre a qualidade do ambiente respiratório.
- **Microcontrolador RP2040:** Processa os dados em tempo real e gerencia a comunicação entre os sensores e o sistema de feedback.
- **Display OLED, Matriz LED e leds Rgb:** Exibe os valores medidos, animações e indicadores do estado fisiológico do usuário.
- **Buzzer:** Utilizado para alertas sonoros quando os níveis ultrapassam os limites definidos.

Tabela 2: Tabela de Custos dos Componentes

Componente	Preço (R\$)	Fornecedor/Link
Raspberry Pi Pico W	60,00	Robocore
Sensor de Sinal Eletromiográfico EMG	170,00	Casa da Robótica
Sensor de Batimento Cardíaco MAX30100	25,00	Smart Kits
Sensor de Qualidade do Ar MQ-135	20,00	Eletrogate
Display OLED 0.96" I2C	30,42	Baú da Eletrônica
Matriz de LEDs 5x5 (WS2812)	20,00	Cinestec
Buzzer (2 unidades)	5,00	Baú da Eletrônica
Botões Pushbutton (3 unidades)	3,00	Baú da Eletrônica
LEDs RGB (2 unidades)	2,00	Baú da Eletrônica
Placa PCB	15,00	JLPCB
Caixa/Case de proteção	25,00	Mercado Livre
Bateria LiPo 3.7V 1200mAh	35,00	Robocore
Fios e conectores	10,00	Diversos
Custo Total	420,42	

Definição das Funcionalidades do Software

O software do *Neurokardy* gerencia a leitura contínua dos sensores, calibração automática, processamento dos sinais e feedback multimodal ao usuário. Utiliza algoritmos para calcular médias e desvios padrão, definindo zonas seguras para evitar alertas falsos. O sistema muda dinamicamente entre estados **estável** e **instável**, ativando respostas visuais e sonoras conforme necessário. Além do monitoramento, há um **modo lúdico** com um jogo interativo para auxiliar no relaxamento. Todas as interações ocorrem em tempo real, otimizadas por interrupções para garantir resposta rápida e eficiente. .

Inicialização e Programação na IDE

O desenvolvimento do software foi realizado no **VSCode**, utilizando o **Pico SDK** e a extensão para o **Raspberry Pi Pico**. O primeiro passo consistiu em importar bibliotecas essenciais previamente criadas em outros projetos, organizando-as no `CMakeLists.txt`. Com isso, foi possível estruturar o ambiente de programação com maior eficiência.

A implementação do código exigiu uma estrutura modular, onde o `main.c` ultrapassou 500 linhas, enquanto o projeto completo, incluindo bibliotecas e animações para o display OLED, superou 1000 linhas de código.

Depuração e Ajustes

Dado o tamanho e a complexidade do código, a depuração foi um dos maiores desafios. Durante o desenvolvimento, diversas estratégias foram utilizadas:

- **Uso de mensagens de depuração:** Impressão de logs para identificar falhas e otimizar o funcionamento dos sensores.
- **Testes incrementais:** Testes frequentes durante a escrita do código para evitar acúmulo de erros.
- **Simulação de falhas:** Execução de cenários extremos para verificar a robustez do software e garantir que o sistema não entrasse em estados inválidos.

Testes de Validação

Para garantir a confiabilidade do *Neurokardy*, foram realizados testes rigorosos com os sensores e o sistema de feedback. O código foi executado repetidamente, simulando diferentes condições para avaliar a resposta do dispositivo.

O método principal consistiu na tentativa de **quebrar o sistema**, simulando cenários onde os sensores apresentariam valores atípicos. Esses testes permitiram verificar se os mecanismos de calibração estavam funcionando corretamente e se o feedback visual e sonoro era acionado apenas quando necessário.

Os principais critérios de validação incluíram:

- **Precisão das medições:** Verificação se os sensores estavam fornecendo leituras coerentes dentro dos limites esperados.
- **Tempo de resposta:** Avaliação do tempo necessário para o sistema processar os dados e gerar feedbacks visuais e sonoros.
- **Estabilidade do software:** Testes prolongados para garantir que o código não apresentasse falhas ou travamentos inesperados.

Discussão dos Resultados

Os testes demonstraram que o *Neurokardy* é um sistema confiável, com medições precisas e feedbacks bem ajustados. O projeto atendeu ao objetivo de fornecer monitoramento fisiológico em tempo real, contribuindo para a redução do estresse e da ansiedade.

No entanto, há diversas melhorias que podem ser implementadas para ampliar a funcionalidade do dispositivo:

- **Adição de novos sensores:** O sistema poderia incluir sensores adicionais para fornecer um diagnóstico mais completo da condição do usuário.
- **Integração com Wi-Fi:** Permitir o envio dos dados para a nuvem, facilitando o armazenamento e análise remota das informações.
- **Módulo GPS:** Implementação de um sistema de navegação acoplado ao display ou matriz LED para auxiliar o usuário na locomoção.
- **Reconhecimento de Voz:** Um assistente inteligente poderia interpretar comandos e oferecer sugestões de técnicas de relaxamento conforme a condição do usuário.
- **Integração com Inteligência Artificial:** Um sistema baseado em IA poderia unificar todos os dados e fornecer um suporte mais avançado, configurando alertas para níveis críticos e até acionando automaticamente uma ambulância ou contato de emergência em situações graves.

Essas melhorias tornariam o *Neurokardy* um dispositivo ainda mais robusto e eficaz para o monitoramento e gestão da saúde mental em tempo real.

Referências

1. **Micro:Calm (do your :bit Challenge, 2021)**. Desenvolvido por Alicia e Esmèe no Canadá, este projeto utiliza a plataforma micro:bit para criar um dispositivo multifuncional de gerenciamento da ansiedade.
2. **Rus Bordas, L.** (2021). *Biofeedback baseado em microcontrolador para gestão de ansiedade*. Projeto acadêmico da UAB.
3. **Navya, Y. et al.** (2022). *Anxiety Detection using IoT*. Projeto que detecta episódios de ansiedade através do monitoramento de sinais fisiológicos com sensores conectados à nuvem.
4. **NCBI** (2023). *Estudos sobre Biofeedback e Neuroplasticidade*. Disponível em: <https://www.ncbi.nlm.nih.gov/>.
5. **American Psychological Association (APA)**. Estudos demonstram que a variabilidade da frequência cardíaca é um marcador confiável de estresse agudo.
6. **Organização Mundial da Saúde (OMS)**. Dados indicam que 90% da população global respira ar poluído, agravando transtornos mentais.
7. **Journal of Clinical Psychology**. Pesquisa que associa a tensão muscular a 70% dos casos de ansiedade clinicamente diagnosticados.

Documentação do Projeto

A documentação do projeto, incluindo os links para o GitHub e o vídeo, deve ser postada na plataforma do Moodle no formato PDF.

- Repositório no GitHub: https://github.com/Aulo007/Projeto_final
- Vídeo demonstrativo: https://youtu.be/5Vmaa3E_u4Y