

Sistema de Monitoramento com Sensores e Interface Keypad/Joystick

Este documento descreve o funcionamento e a arquitetura do firmware desenvolvido para um sistema de controle e monitoramento. O sistema utiliza um Teclado de membrana matricial (Keypad) 4x4, um Joystick analógico, um Sensor Ultrassônico de distância e um Sensor de Obstáculos Infravermelho para oferecer diferentes modos de operação, controle e feedback ao usuário.

Integrantes: Bruno Norton 202503452041 TA, João Arthur 202307164501 TA, Lucas Alcântara 202501001629 TA, Micael Dali 202501000975 TA, Roger Pires 202501001556 TA, Vinicius Machado 202503006725 TA.

1. Componentes de Hardware e Configuração de Pinos

O sistema construído em torno de um microcontrolador Arduino com a seguinte alocação de pinos:

Componente	Pinos de Controle/Sinal	Tipo de Sinal	Descrição
Joystick (Eixo X)	A0 (pinoJoystickX)	Analógico (Entrada)	Movimento horizontal (esquerda/direita)
Joystick (Eixo Y)	A1 (pinoJoystickY)	Analógico (Entrada)	Movimento vertical (frente/trás)
Joystick (Botão SW)	12 (pinoJoystickSW)	Digital (Entrada PULLUP)	Clique do botão (confirmação)
Ultrassônico (Trig)	11 (pinoTrig)	Digital (Saída)	Pulso de disparo do sensor
Ultrassônico (Echo)	10 (pinoEcho)	Digital (Entrada)	Medição do tempo de retorno do pulso
Sensor IR	13 (pinoSensorIR)	Digital (Entrada)	Deteção de obstáculo (nível LOW = Detectado)
Keypad (Linhas)	9, 8, 7, 6 (pinosLinhas)	Digital (E/S)	Pinos de varredura para linhas do Keypad
Keypad (Colunas)	5, 4, 3, 2 (pinosColunas)	Digital (E/S)	Pinos de varredura para colunas do Keypad

2. Detalhes do Sistema operacional

2.1. Biblioteca Utilizada

A biblioteca Keypad.h foi utilizada, essencial para a interface com o Keypad 4x4, tratando a varredura e o mapeamento das teclas

2.2. Sistema de controle e monitoramento(Enumerações)

O programa opera em três estados principais (EstadoSistema) e quatro modos de operação (ModoOperacao).

Estado do Sistema (EstadoSistema):

1. **AGUARDANDO_MODO:** O sistema está no menu principal, aguardando a seleção do modo via **Joystick**.
2. **MODULO_SELECIONADO:** Um modo foi pré-selecionado pelo Joystick, aguardando a confirmação.
3. **OPERANDO:** O teste escolhido está ativo.

Modo de Operação (ModoOperacao):

1. **NENHUM:** Estado inicial ou após retorno ao menu.
2. **CONTINUO:** Modo de monitoramento constante dos sensores.
3. **SOB_DEMANDA:** Modo de leitura única, ativada sob demanda do usuário
4. **TESTE:** Rotação de testes de funcionalidade dos sensores.

2.3. Variáveis Globais de Controle

- **distanciaCritica (int):** Parâmetro limite (em cm) para acionamento de alerta. Padrão inicial de 20 cm, configurável pelo usuário.
- **centroJoystick (const int):** 512. Valor de referência para a posição de repouso do Joystick.
- **limiarJoystick (const int):** 400. Define a sensibilidade para detecção de movimento do Joystick (movimento detectado se o valor for menor que 112 ou maior que 912).
- **bufferEntradaNumerica (String):** Buffer temporário para armazenar dígitos do Keypad para a configuração da distanciaCritica.

- **previousMillis (unsigned long):** Variável de controle de tempo para operação.

3. Funcionalidades Principais (Fluxo de Execução)

3.1. Seleção e Configuração de Modo (AGUARDANDO_MODO -> MODO_SELECIONADO)

1. **Seleção (Joystick):** O modo é escolhido movendo o Joystick:
 - **Esquerda ($X < 112$):** Modo Contínuo.
 - **Direita ($X > 912$):** Modo Sob Demanda.
 - **Para Cima ($Y > 912$):** Modo Teste.
2. **Configuração (Keypad):** No estado MODO_SELECIONADO, o usuário pode digitar um valor numérico no Keypad para redefinir a distanciaCritica.
3. **Confirmação:** Pressionar '#' no Keypad ou clicar no **botão SW** do Joystick (função verificaCliqueJoystick) inicia o modo de operação e atualiza a distanciaCritica se um valor tiver sido digitado.

4.2. Modo de Operação Contínuo (gerenciaModoContínuo)

- **Monitoramento:** Realiza leituras contínuas dos sensores Ultrassônico e IR.
- **Aviso:** As leituras só são impressas no **Monitor Serial** se houver uma mudança significativa (distância variar mais de 0.5 cm ou o estado do IR mudar).
- **Alerta Crítico:** Emite um **ALERTA** se a distância medida for menor que a **Distância Crítica** OU se o **Sensor IR** detectar um obstáculo (nível LOW).

4.3. Modo de Operação Sob Demanda (gerenciaModoSobDemanda)

- **Interação:** Permite ao usuário definir um **limite de distância** no momento da leitura, digitando um valor no Keypad.
- **Medição:** Pressionar a tecla '**A**' aciona uma leitura **única** do sensor Ultrassônico.
- **Resultado:** O sistema reporta a distância atual e verifica se há um obstáculo dentro do limite definido (o limite é o padrão distanciaCritica se nenhum valor for digitado).
- **Buffer Dedicado:** Utiliza o bufferDemanda para isolar as entradas numéricas deste modo, evitando conflito com o buffer de configuração global.

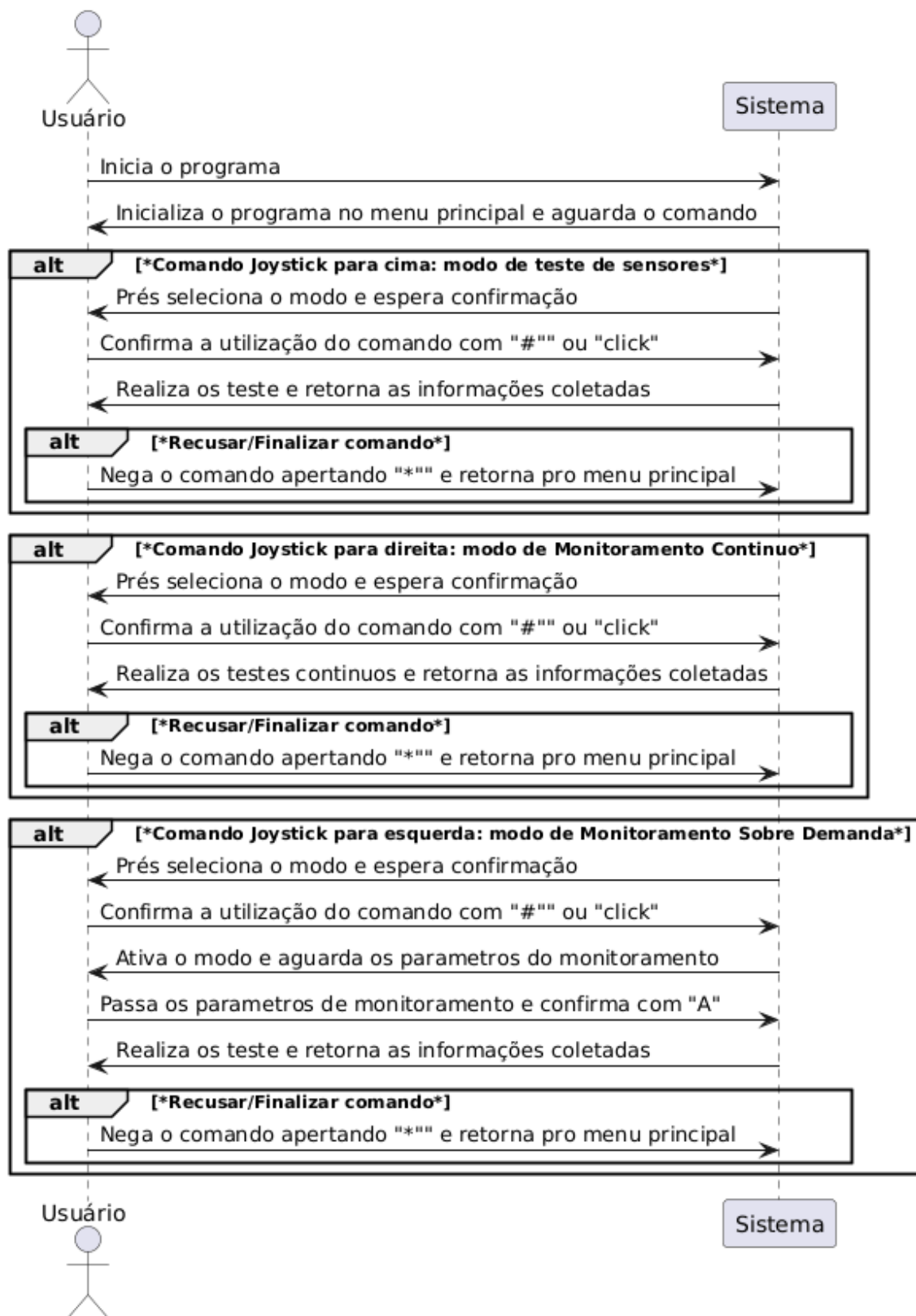
4.4. Modo de Teste de Sensores (gerenciaModoTeste)

- **Teste Cíclico:** Alterna a leitura dos sensores a cada 3000 ms (intervaloTeste), utilizando o método de tempo **não bloqueante** (millis() e previousMillis).
 - **Etapa 0:** Lê e reporta a distância do **Sensor Ultrassônico**.
 - **Etapa 1:** Lê e reporta o estado do **Sensor Infravermelho**.
- **Feedback:** Imprime no Monitor Serial se cada sensor está **OK** com a leitura atual.

4.5. Retorno ao Menu Principal (voltarAoMenu)

- **Keypad:** Pressionar a tecla '*' a qualquer momento (exceto durante a digitação de números de configuração/demanda) faz o sistema retornar ao estado AGUARDANDO_MODO.
- **Reset:** Limpa os buffers de entrada e redefine o estadoAtual para o menu principal.

5. Fluxo de Uso do Sistema



6. Diagrama do sistema em Arduino

