

Desafio Arduino - INPUT e OUTPUT

Mapeamento e Configuração

Este exercício inicial é o ponto de partida para todos os projetos. Ele se concentra em preparar o ambiente de programação, mapeando os pinos do microcontrolador com os componentes do circuito no protoboard.

Objetivo: Mapear os pinos dos LEDs e botões e configurar seus modos de operação corretamente.

Lógica de Programação:

1. Comece declarando constantes para os pinos que serão usados. Utilize a diretiva `#define`.
2. No bloco `setup()`, use a função `pinMode()` para configurar cada pino:
 - Para os LEDs, que são saídas, a configuração deve ser `OUTPUT`.
 - Para os botões, que são entradas e, segundo sua descrição, estão configurados com **INPUT_PULLUP** para detectar borda de subida (transição de **LOW** para **HIGH**), a configuração deve ser `INPUT_PULLUP`.
3. Inicie a comunicação serial com `Serial.begin()` para poder imprimir mensagens de feedback para o monitor serial.

Exercício 1: Contador de Acionamentos

Objetivo: Contar quantas vezes o primeiro botão (o mais à esquerda) é pressionado. A ação ocorre somente na **borda de subida** do sinal.

Lógica de Programação:

1. Use o primeiro LED (o mais à esquerda) como indicador.
2. Inicie um contador (uma variável inteira) com o valor zero.
3. Quando o primeiro botão for pressionado, a borda de subida do sinal irá ocorrer. Use a transição de estado de `LOW` para `HIGH` para incrementar o contador em 1.
4. Após o primeiro acionamento, faça o primeiro LED piscar uma vez. Após a segunda, duas vezes, e assim por diante.
5. Quando o contador chegar a 5, acenda o LED de forma contínua e imprima a mensagem "Meta atingida!".
6. O restante dos LEDs e botões não será usado neste exercício.

Exercício 2: Placar de Pontos Simples

Objetivo: Fazer uma competição de quem pressiona o botão mais rápido. A ação ocorre somente na **borda de subida** do sinal.

Lógica de Programação:

1. Use os dois primeiros botões e os dois primeiros LEDs. O primeiro botão e LED serão para o **Jogador 1** e o segundo par para o **Jogador 2**.
 2. Inicie duas variáveis de placar, placar_1 e placar_2, ambas com zero.
 3. A cada vez que o botão do Jogador 1 for pressionado (na borda de subida), incremente placar_1.
 4. A cada vez que o botão do Jogador 2 for pressionado (na borda de subida), incremente placar_2.
 5. O LED do jogador cuja pontuação for maior deve piscar mais rapidamente. Por exemplo, se placar_1 for 5 e placar_2 for 2, o LED do Jogador 1 pisca mais rápido que o LED do Jogador 2.
 6. Quando qualquer um dos jogadores atingir 10 pontos, o LED dele deve permanecer aceso e o outro deve se apagar, indicando o vencedor. Imprima "Vencedor: Jogador X!".
 7. O terceiro botão pode ser configurado como **botão de reset**. Ao ser pressionado (na borda de subida), os placares de ambos os jogadores devem voltar a zero.
-

Exercício 3: Placar com Comparação e Reset

Objetivo: Criar um placar para dois jogadores com um botão para comparação de pontuação e outro para reset. A ação ocorre somente na **borda de subida** do sinal.

Lógica de Programação:

1. Use os três primeiros botões e os três primeiros LEDs. O primeiro par é para o **Jogador 1**, o segundo par para o **Jogador 2** e o terceiro botão é o **botão de reset**. O terceiro LED será o **LED de comparação**.
2. Inicie as variáveis placar_1 e placar_2 com zero.
3. Quando o botão do Jogador 1 for pressionado (na borda de subida), incremente placar_1. Quando o do Jogador 2 for pressionado (na borda de subida), incremente placar_2.
4. O LED do Jogador 1 deve piscar uma vez para cada ponto que ele tiver, e o mesmo para o LED do Jogador 2. Por exemplo, se o placar_1 é 3, o LED pisca 3 vezes.
5. Quando o quarto botão for pressionado (na borda de subida), compare os placares:
 - Se placar_1 > placar_2, acenda o LED do Jogador 1.
 - Se placar_2 > placar_1, acenda o LED do Jogador 2.
 - Se placar_1 == placar_2, faça ambos os LEDs piscarem simultaneamente.
6. Quando o botão de reset (o terceiro botão) for pressionado (na borda de subida), zere os placares e apague todos os LEDs.

Exercício 4: Sequência de LEDs e Botão de Reset Mestre

Objetivo: Fazer com que cada botão avance a "sequência" de LEDs acendendo-os um por um. A ação ocorre somente na **borda de subida** do sinal.

Lógica de Programação:

1. Use todos os quatro botões e todos os quatro LEDs.
2. Mantenha uma variável de estado que rastreia qual LED deve ser aceso (estado_led, com valor de 0 a 4).
3. Inicialmente, todos os LEDs devem estar apagados.
4. Quando o primeiro botão for pressionado (na borda de subida), acenda o primeiro LED.
5. Quando o segundo botão for pressionado (na borda de subida), apague o primeiro LED e acenda o segundo.
6. Quando o terceiro botão for pressionado (na borda de subida), apague o segundo LED e acenda o terceiro.
7. Quando o quarto botão for pressionado (na borda de subida), apague o terceiro LED e acenda o quarto.
8. Quando o quarto LED estiver aceso, pressione o primeiro botão (na borda de subida) para voltar ao início e acender o primeiro LED.
9. O último botão (o quarto) também pode ser configurado como um **botão de reset mestre**: quando pressionado (na borda de subida), ele apaga todos os LEDs, não importa qual está aceso.