

Mãos na massa: Implementando leds_respondidos

Vamos implementar o nosso planejamento e alterar o código! Siga os passos:

1) No **ArduínoIDE**, abra o código do seu programa Genius.

2) Crie uma nova variável `leds_respondidos` (ela deve estar localizada abaixo da variável `rodada`, por exemplo):

```
int leds_respondidos = 0;
```

3) Essa variável será inicializada a cada rodada. Altere a função `preparaNovaRodada` :

```
void preparaNovaRodada() {  
    rodada++;  
    leds_respondidos = 0; //novo  
    tocaLedsRodada();  
}
```

4) Agora, nós vamos trabalhar no processamento da resposta do jogador. Procure a função `loop` e dentro dela o `case` que atende o estado `USUARIO_RESPONDENDO`. Nesse `case` chamaremos uma nova função, a `processaRespostaUsuario` :

```
void loop() {  
    switch (estadoAtual()) {  
        case PRONTO_PARA_PROXIMA_RODADA:  
            Serial.println("Pronto para a proxima rodada");  
            preparaNovaRodada();  
            break;  
        case USUARIO_RESPONDENDO:  
            Serial.println("Usuario respondendo");  
            processaRespostaUsuario(); //novo  
            break;  
        case JOGO_FINALIZADO_SUCESSO:  
            Serial.println("Jogo finalizado com sucesso");  
            break;  
        case JOGO_FINALIZADO_FALHA:  
            Serial.println("Jogo finalizado com falha");  
            break;  
    }  
    delay(MEIO_SEGUNDO);  
}
```

5) Vamos criar essa função `processaRespostaUsuario`. Abaixo da função `loop` adicione o seguinte:

```
void processaRespostaUsuario(){  
    leds_respondidos++;  
}
```

5) Por fim, altere a função `estadoAtual` para verificar se o jogador ainda precisa responder:

```
int estadoAtual() {  
  if (rodada <= TAMANHO_SEQUENCIA) {  
  
    //novo if-else  
    if(leds_respondidos == rodada){  
      return PRONTO_PARA_PROX_RODADA;  
    }else{  
      return USUARIO_RESPONDENDO;  
    }  
  
  } else {  
    return JOGO_FINALIZADO_SUCESSO;  
  }  
}
```

Compile e teste o seu código! Ao rodar, fique de olho no monitor serial.