**UNIVERSIDADE DO ESTADO DO PARÁ – UEPA**

**Sistema de Tranca Eletrônica com Senha**

**Disciplina: Sistemas Embarcados  
Docente: Renato Carr**

*Melissa Lunara Santos Ferreira  
 Wenderson Souza dos Santos*

**Resumo**

Este projeto apresenta o desenvolvimento de uma tranca eletrônica controlada por senha, utilizando um microcontrolador Arduino UNO em conjunto com um teclado matricial 4x4, um microservo SG90 e LEDs indicadores de estado. O sistema foi implementado integralmente em ambiente de simulação na plataforma Wokwi, o que permitiu a validação do protótipo sem a necessidade de hardware físico. A lógica de funcionamento contempla uma senha fixa de quatro dígitos, com possibilidade de correção de dígitos e confirmação da entrada, além de um mecanismo de bloqueio automático por vinte segundos após tentativas incorretas consecutivas. O acionamento do servo simula a abertura e o fechamento da fechadura, enquanto os LEDs e o Monitor Serial fornecem feedback visual e informativo ao usuário. Foram realizados testes em diferentes cenários, confirmando o desempenho esperado do sistema. O resultado obtido demonstra a viabilidade de um protótipo funcional, simples e de baixo custo, adequado tanto para fins acadêmicos quanto como base para soluções práticas em automação de controle de acesso.

**Introdução**

A busca por soluções acessíveis e eficientes na área de segurança residencial e comercial tem impulsionado o desenvolvimento de sistemas de controle de acesso baseados em tecnologias embarcadas. Entre essas soluções, as trancas eletrônicas se destacam por aliarem praticidade e confiabilidade, possibilitando a substituição ou reforço de métodos tradicionais, como chaves físicas. Com a popularização das plataformas de prototipagem eletrônica, como o Arduino, tornou-se viável implementar projetos funcionais de baixo custo, que podem ser aplicados tanto em contextos acadêmicos quanto em protótipos de sistemas reais.

Neste trabalho, desenvolveu-se uma tranca eletrônica com senha fixa de quatro dígitos, programada em um microcontrolador Arduino UNO e implementada integralmente em ambiente de simulação utilizando o Wokwi. O sistema utiliza um teclado matricial 4x4 como interface de entrada, um microservo SG90 para simular o acionamento da fechadura e LEDs como indicadores visuais de estado, complementados pelo monitoramento em tempo real por meio do Serial Monitor. Além disso, foram incorporados recursos de segurança adicionais, como a contagem de tentativas incorretas e o bloqueio temporário do sistema, o que reforça a confiabilidade do protótipo.

A escolha pelo uso de um simulador online em detrimento do hardware físico fundamenta-se na praticidade, rapidez de implementação e redução de custos, características que tornam o Wokwi uma ferramenta adequada para fins didáticos. Dessa forma, este projeto não apenas demonstra a aplicabilidade de sistemas embarcados em soluções de automação e segurança, como também reforça o potencial do uso de simulações como recurso pedagógico no processo de ensino-aprendizagem.

**Objetivo Geral**

Desenvolver uma tranca eletrônica com senha fixa, baseada em microcontrolador Arduino UNO, utilizando simulação no ambiente virtual Wokwi, de forma a reproduzir o funcionamento de um sistema de controle de acesso seguro, acessível e funcional.  
  
**Objetivos Específicos**

* Implementar uma senha fixa de quatro dígitos para liberar o acesso.
* Utilizar um teclado matricial 4x4 como interface de entrada de dados.
* Controlar um microservo SG90 para simular a abertura e fechamento da fechadura.
* Inserir LEDs indicadores para fornecer feedback visual sobre o estado do sistema (acesso autorizado ou negado).
* Configurar mensagens no Monitor Serial para acompanhamento em tempo real das operações.
* Adicionar mecanismo de bloqueio temporário após tentativas incorretas consecutivas.
* Realizar todos os testes em simulação no Wokwi, garantindo viabilidade e validação do sistema sem necessidade de hardware físico.

**Metodologia**

**1. Abordagem metodológica**

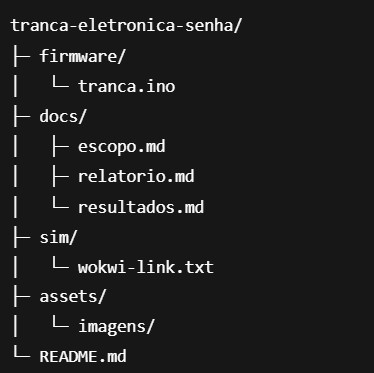
Adotou-se uma abordagem incremental e orientada a módulos, de modo a viabilizar a construção e validação progressiva do sistema. As funcionalidades foram organizadas em etapas lógicas (estrutura do repositório, simulação, esqueleto do firmware, lógica da senha, integração com atuadores e indicadores, e por fim testes), permitindo rastreabilidade entre o planejamento e a implementação. Cada incremento foi acompanhado de verificação funcional no simulador, registro das evidências (prints do circuito e do Monitor Serial) e controle de versão no GitHub.

**2. Ambiente e ferramentas**

* Plataforma de simulação: Wokwi (execução 100% em navegador, sem necessidade de hardware físico).
* Plataforma de desenvolvimento: Arduino (linguagem C/C++ para microcontroladores).
* Bibliotecas utilizadas: Keypad.h (leitura do teclado matricial 4x4) e Servo.h (acionamento do microservo SG90).
* Controle de versão e armazenamento em nuvem: Git e GitHub (repositório remoto, histórico de commits por módulo, organização de documentação e evidências).
* Sistema de arquivos do projeto: estrutura de diretórios padronizada, com separação entre firmware, documentação, simulação e recursos visuais.

**3. Estrutura do repositório e versionamento (nuvem/GitHub)**

O repositório foi estruturado para facilitar manutenção, auditoria e reprodutibilidade:



* **Padrão de commits:** cada módulo concluído originou um commit descritivo (ex.: “Módulo 06 – Lógica da senha completa com bloqueio e feedback Serial”), garantindo clareza sobre o escopo da alteração.
* **Rastreamento de evidências:** capturas do Wokwi (circuito) e do Serial Monitor foram salvas em assets/imagens/, permitindo referenciar diretamente no relatório.
* **Disponibilização do link da simulação:** o endereço do projeto Wokwi foi armazenado em sim/wokwi-link.txt, viabilizando reprodução imediata.
* **Armazenamento do relatório final:** além do envio em Word/PDF, recomenda-se arquivar a versão final do relatório em /docs/ para consolidação do projeto no repositório.  
    
  **4. Configuração da simulação no Wokwi**

A simulação foi configurada no Wokwi com os seguintes elementos e ligações:

**Componentes simulados:**

* Arduino UNO (controlador principal).
* Keypad 4x4 (interface de entrada da senha).
* Microservo SG90 (atuador da tranca).
* LEDs virtuais (verde e vermelho) como indicadores de estado.

**Mapeamento de conexões:**

* Teclado 4x4:
* Linhas → pinos digitais 2, 3, 4, 5
* Colunas → pinos digitais 6, 7, 8, 9
* Servo SG90:
* Sinal → D10
* VCC → 5V
* GND → GND do Arduino
* LEDs (virtuais): conectados aos pinos digitais definidos no firmware, com retorno comum ao GND. No Wokwi, os LEDs virtuais já contemplam o comportamento elétrico, simplificando a montagem.

**Boas práticas elétricas na simulação:** todos os componentes compartilham referência comum de GND; alimentação do servo pelo 5V do UNO; organização das conexões para leitura clara no diagrama.

**5. Desenvolvimento do firmware**

O firmware foi desenvolvido em camadas de responsabilidade, com foco em legibilidade e validação incremental:

**5.1. Definições e constantes de projeto:**

Senha fixa de quatro dígitos (5555), tamanho do buffer, número máximo de tentativas, tempo de bloqueio (20 s), ângulos do servo (fechado em 0°, aberto em 90°), tempo de abertura (10 s), e pinos dos periféricos.

**5.2. Variáveis de estado:**

Buffer de entrada, índice de posição digitada, contador de tentativas inválidas, flag/estado de bloqueio e temporizações associadas.

**5.3. Inicialização (setup):**

Configuração da Serial (para feedback e depuração), anexação do servo ao pino D10, configuração dos pinos dos LEDs e exibição de mensagem/banner inicial no Monitor Serial indicando o início da simulação.

**5.4. Leitura do teclado (loop):**

Dígitos numéricos (0–9): inseridos sequencialmente no buffer até atingir quatro posições.

Correção (\*): remoção do último dígito inserido.

Confirmação (#): dispara a validação da senha quando o buffer contém quatro dígitos.

**5.5. Validação e tratamento de tentativas:**

Senha correta: chama a rotina de abertura da tranca (movimento do servo para 90°), aciona LED verde, registra evento no Serial e mantém o estado por ~10 s. Em seguida, retorna ao estado fechado (servo em 0°) e reativa o LED vermelho.

Senha incorreta: incrementa contador de erros, informa no Serial e verifica limite de tentativas. Ao atingir o limite, ativa-se o bloqueio de 20 s, durante o qual novas entradas são ignoradas e o LED vermelho permanece ativo.

**5.6. Feedback ao usuário:**

Toda a interação é registrada no Monitor Serial (teclas pressionadas, correções, confirmações, sucesso/erro, início e término de bloqueios), conferindo a observação do sistema.

**6. Procedimento de testes (planejamento e execução)**

Os testes foram conduzidos exclusivamente no Wokwi, com evidências coletadas para cada caso:

**Preparação:**

**(i)** abrir o projeto no Wokwi;

**(ii)** clicar em Play para iniciar a simulação;

**(iii)** abrir o Serial Monitor;

**(iv)** verificar mensagem inicial do sistema.

**Casos de teste (Caso 1 – Caso 4):**

Caso 1 — Senha correta (5555):

Inserir 5-5-5-5 e confirmar. Esperado: servo 0→90°, LED verde aceso, log de sucesso no Serial.

Caso 2 — Uma tentativa errada seguida de correta:

Inserir senha incorreta (ex.: 1-2-3-4), observar erro; em seguida inserir 5-5-5-5 e confirmar. Esperado: registro de erro e posterior sucesso no Serial.

Caso 3 — Exceder limite de tentativas:

Inserir senhas erradas até atingir o limite. Esperado: ativação de bloqueio por 20 s, LED vermelho ativo, entradas ignoradas.

Caso 4 — Pressionar teclas durante bloqueio:

Acionar teclas no período de bloqueio. Esperado: nenhuma alteração de estado; logs informam bloqueio vigente.

**Critérios de aprovação:**

A funcionalidade é considerada aprovada se o comportamento observado (movimento do servo, estados dos LEDs e mensagens no Serial) corresponder ao resultado esperado do caso de teste.

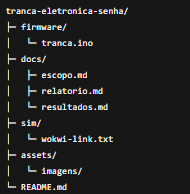
**Registro de evidências:**

Para cada caso, foram capturadas imagens do circuito e prints do Monitor Serial. A consolidação dos resultados foi registrada na tabela de docs/resultados.md e reproduzida na seção de Resultados do relatório.

**Desenvolvimento**

**4.1 Estrutura do Projeto**

O projeto foi estruturado em diretórios bem definidos, visando organização, rastreabilidade e fácil acesso às diferentes partes do sistema. A árvore de diretórios do repositório é a seguinte:

****

**Descrição das pastas:**

* firmware/: contém o código-fonte principal (tranca.ino), incluindo a lógica de controle da senha, acionamento do servo e LEDs, e feedback via Serial Monitor.
* docs/: reúne a documentação do projeto, como escopo, relatório final e tabela de resultados dos testes.
* sim/: armazena o link da simulação no Wokwi, permitindo reprodução imediata do projeto em qualquer navegador.
* assets/: contém imagens do circuito simulado e prints do Monitor Serial, utilizados no relatório e para fins de registro.
* README.md: fornece instruções gerais sobre o projeto, sua estrutura e como executar a simulação.

**4.2 Funcionamento da Lógica**

O sistema foi desenvolvido com uma senha fixa de quatro dígitos (5555), armazenada em memória, e um buffer de entrada que aceita os dígitos inseridos pelo usuário. As principais funcionalidades incluem:

* Entrada de dados: os números são digitados pelo teclado matricial 4x4.
* Correção de entrada: a tecla \* permite apagar o último dígito digitado.
* Confirmação: a tecla # valida a senha quando o buffer contém quatro dígitos.
* Validação: a senha inserida é comparada com a senha fixa, incrementando o contador de tentativas em caso de erro.
* Bloqueio temporário: ao atingir o limite de tentativas incorretas, o sistema permanece bloqueado por 20 segundos, ignorando novas entradas.
* Feedback visual e informativo: LEDs e Monitor Serial informam ao usuário o estado do sistema, enquanto o servo simula fisicamente a abertura e fechamento da tranca.

**4.3 Estados do Sistema**

O sistema apresenta quatro estados principais, cada um indicado por LEDs e posição do servo:

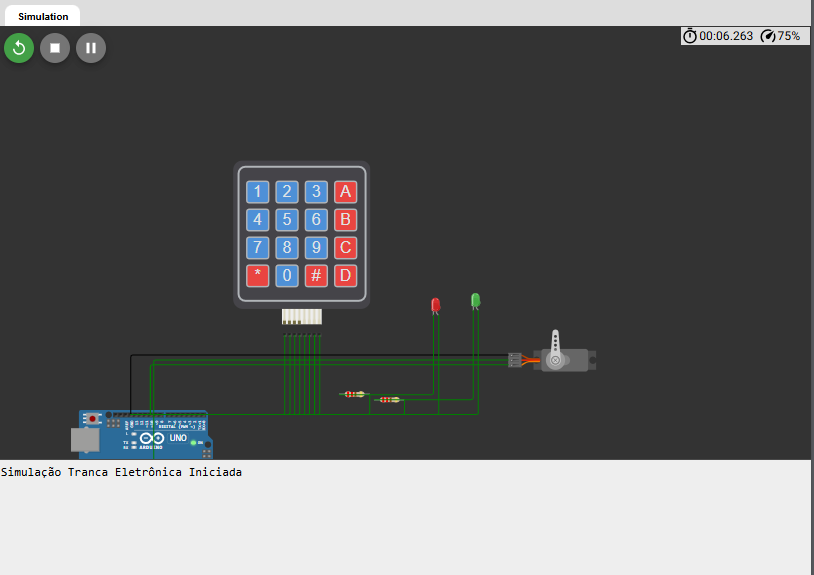
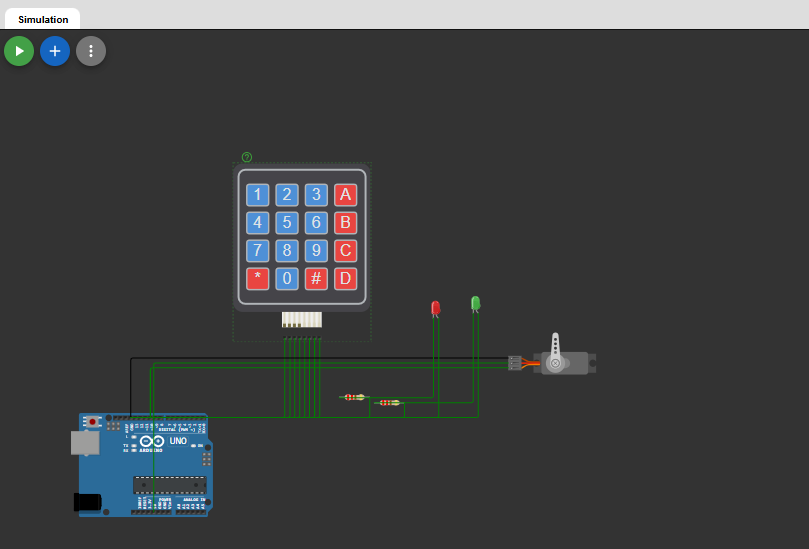
* Fechado: LED vermelho aceso, servo na posição 0°.
* Digitando senha: aguardando entrada de dígitos pelo usuário, LED vermelho permanece aceso.
* Aberto: senha correta → servo na posição 90°, LED verde aceso por 10 segundos.
* Bloqueado: senha incorreta repetida → bloqueio por 20 segundos, entradas ignoradas, LED vermelho aceso.

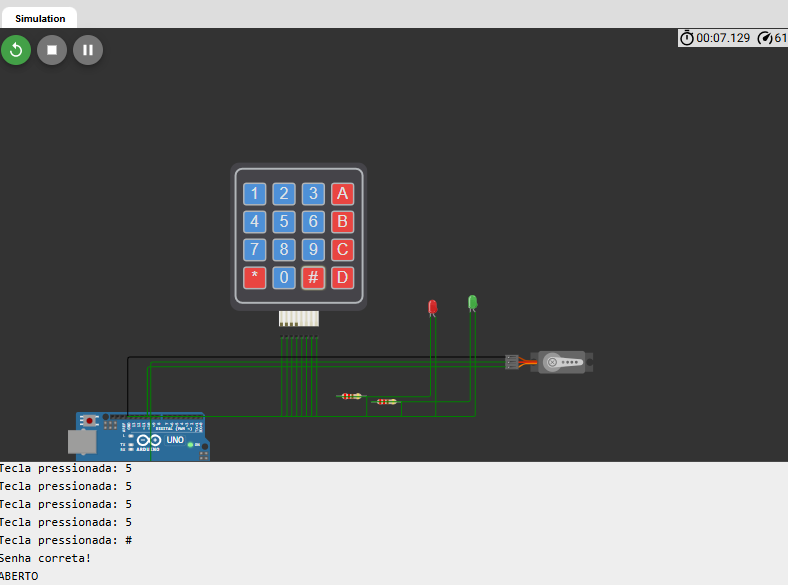
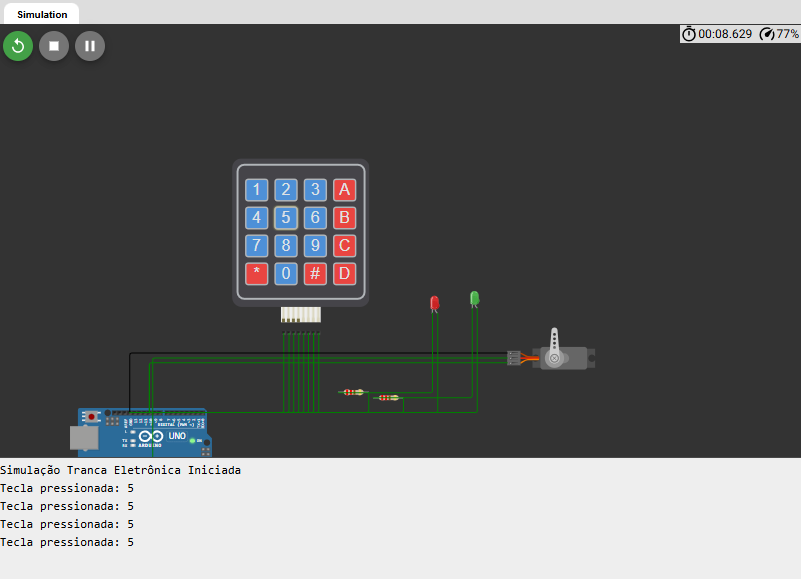
**5. Testes e Resultados**

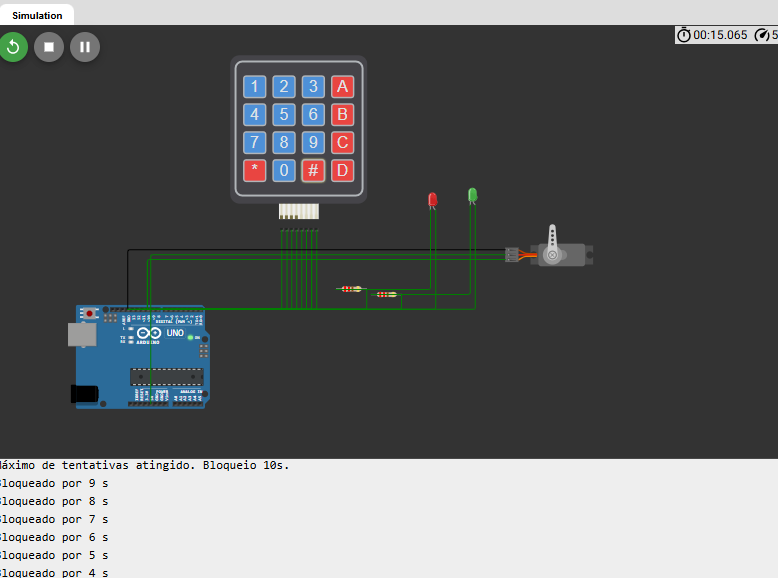
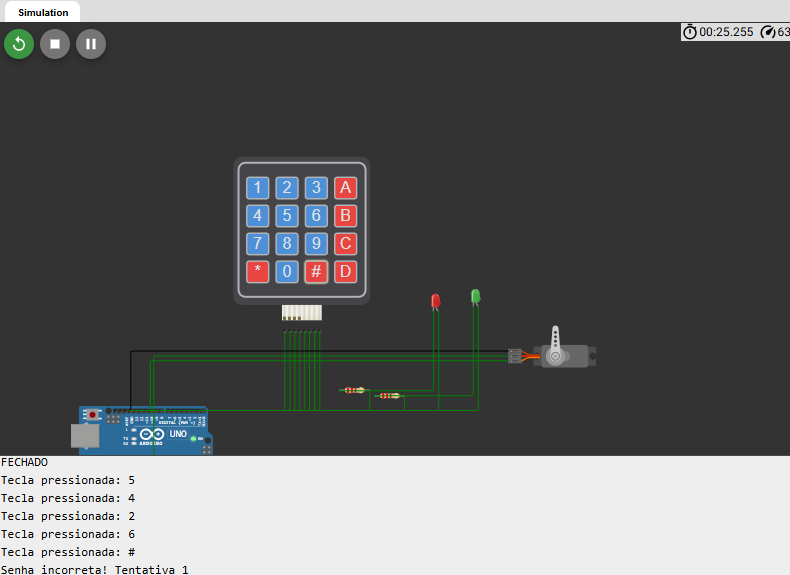
Os testes foram conduzidos exclusivamente na simulação do Wokwi, com registro completo no Monitor Serial e documentação das evidências. Os casos de teste contemplaram diferentes cenários de uso:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Caso | Descrição | Resultado Esperado | Resultado Obtido | Status |
| C1 | Senha Correta | Servo abre (0→90°), LED verde acende | Print Serial + circuito | |  | | --- | |  |  |  | | --- | | Passou | |
| C2 | 1 Tentativa errada + senha correta | Feedback de erro, depois sucesso | |  | | --- | | Print Serial |  |  | | --- | |  | | |  | | --- | |  |  |  | | --- | | Passou | |
| C3 | Exceder limite de tentativas | Bloqueio 20s, LED vermelho | |  | | --- | | Print Serial |  |  | | --- | |  | | |  | | --- | |  |  |  | | --- | | Passou | |
| C4 | Teclas pressionadas no bloqueio | Ignoradas | |  | | --- | | Print Serial |  |  | | --- | |  | | |  | | --- | |  |  |  | | --- | | Passou | |

**Evidências registradas:**

****

****

****

**Conclusão**

O projeto atingiu todos os objetivos definidos, validando com sucesso a lógica de funcionamento de uma tranca eletrônica com senha fixa. A utilização do Wokwi demonstrou-se eficiente e acessível, permitindo simulação completa sem necessidade de hardware físico.

**Principais aprendizados:**

* Importância do versionamento e rastreabilidade via GitHub.
* Validação incremental de funcionalidades em ambiente simulado.
* Registro detalhado de testes e evidências visuais.

**Melhorias futuras:**

* Permitir configuração de senha pelo usuário.
* Armazenamento de senha na EEPROM do Arduino.
* Inclusão de display LCD para interação mais intuitiva.
* Integração com IoT (Wi-Fi, Bluetooth) para monitoramento remoto.

**Referências**

* Arduino Reference. Disponível em: <https://www.arduino.cc/reference>
* Wokwi – Arduino Simulator. Disponível em: <https://wokwi.com>
* Bibliotecas Arduino utilizadas: Keypad.h e Servo.h.
* Repositório GitHub do projeto Tranca Eletrônica: SouzaDev-X/Tranca\_Eletronica. Disponível em: <https://github.com/SouzaDev-X/Tranca_Eletronica> – contém código completo, documentação organizada e links diretos para a simulação no Wokwi.