

# *Cofre digital.*

*Um dispositivo capaz de guardar objetos com segurança e permitir acesso restrito ao uso de senha numérica*

Luiz Henrique Rocha Marinho

15/0041527

Engenharia eletrônica.

Universidade de Brasília

Brasília-DF, Brasil.

[luizhenriquemarinhoFGA@gmail.com](mailto:luizhenriquemarinhoFGA@gmail.com)

Pedro Henrique Brito Checchia

15/0044488

Engenharia eletrônica.

Universidade de Brasília

Brasília-DF, Brasil

[pedrobcbr@hotmail.com](mailto:pedrobcbr@hotmail.com)

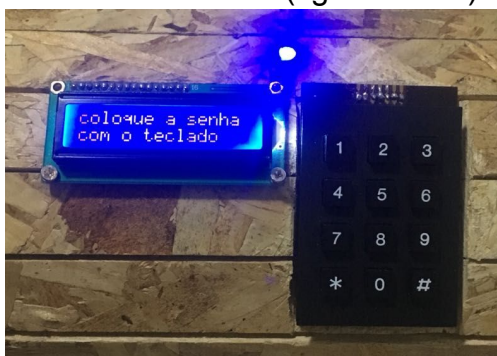
## **Resumo —.**

Este documento contém informações básicas sobre o projeto da disciplina de Microcontroladores e Microprocessadores. Este projeto consiste em um cofre com desbloqueio com senha numérica de 4 dígitos que tem como objetivo oferecer segurança de alta qualidade para que o usuário armazene objetos.

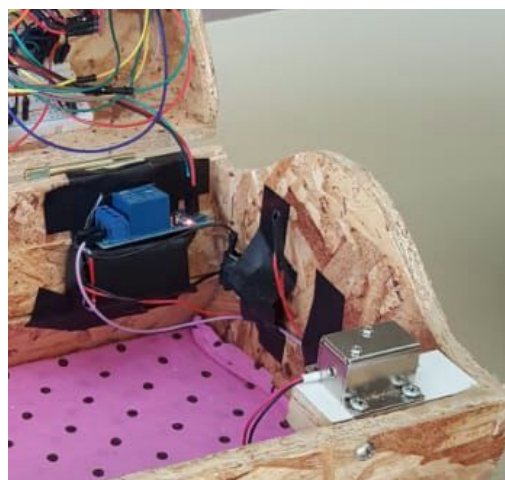
**Palavras-chave — cofre; senha numérica, trava automática;**

## **I. REFERENCIAL TEÓRICO**

O projeto se baseia em inserir uma senha inicial que será a chave para os próximos comandos (alterar senha ou liberar o cofre) (figura 1). Caso a senha inserida seja a senha cadastrada a trava solenóide é liberada automaticamente (figura 2).



**figura 1:** configuração inicial do projeto, senha inicial.

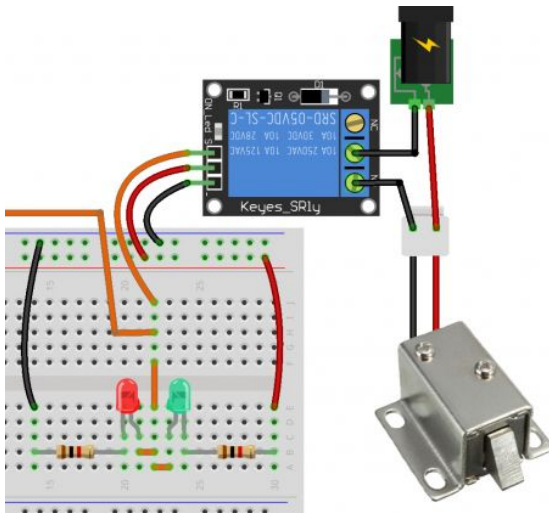


**Figura 2-** trava elétrica solenóide, quando é sujeita a uma tensão de 12 V a trava é liberada.

A interface do projeto se dá a partir de um teclado numérico de matriz 4x3 e um display lcd que foram os principais componentes do projeto.

A trava elétrica precisa de 12 V para funcionar, porém a MSP430 fornece uma saída de aproximadamente 3 V, portanto para o acionamento da trava será utilizada uma bateria de 12 V (foram acionadas duas baterias de 9V em série e alguns resistores para que a tensão de saída fosse próxima de 12 V) (figura 4) e um relé que será

acionado a partir da MSP430 como mostra o esquema da figura 3.



**Figura 3** - Ilustração da alimentação da trava(fonte:<https://www.filipeflop.com/blog/acionando-trava-eletrica-com-rfid/>)



**Figura 4** - As duas baterias ligadas em série, estão cobertas por fita isolante.

O cofre é feito de madeira de alta qualidade que garante a proteção dos bens que estarem do lado de dentro.

## II. JUSTIFICATIVA

- O motivo da realização deste projeto é oferecer a máxima segurança para armazenar objetos dos mais variados tipos, o sistema de desbloqueio por senha garante uma boa segurança, visto que uma vez cadastrada, o cofre só poderá ser desbloqueado com o uso da

mesma, e para mudá-la o usuário terá que entrar com a senha antiga. Para executar comandos como cadastrar, remover, entre outros, existe no produto um teclado numérico e um display LCD que exibe as informações:



**Figura 5** - funcionamento simultâneo do display e do teclado em fase de testes..

Foi utilizado o Software Code Composer Studio 7.4.0 para executar o código que se encontra nos anexos no final do documento. Basicamente foram criadas funções para printar no display a informação a partir do botão que foi pressionado, desenvolvendo assim a interface do projeto:



**Figura 7** - mensagem de orientação 1.

Após inserir os 4 dígitos:



**Figura 8** - mensagem de orientação 2 com uma senha aleatória.

Esse é o único momento que a senha é mostrada no display, para os outros comandos serão printados asteriscos no lugar dos números.



**Figura 9** - mensagem de orientação 3

o usuário tem a opção de alterar a senha ou abrir o cofre, ambos precisam da senha cadastrada anteriormente.



**Figura 10** - mensagem de orientação 4

III. Problemas encontrados no decorrer do projeto:

Quando este projeto foi idealizado, esperava-se utilizar o desbloqueio por leitor biométrico, porém não foi possível realizar a sua implementação à tempo devido à complexidade de realizar a comunicação serial na msp430. Tentou-se alterar para o desbloqueio com o ibutton, porém foram encontradas as mesmas dificuldades do leitor biométrico. Diante disso, o melhor recurso encontrado foi utilizar o teclado para o desbloqueio do cofre.



**Figura 10** - comunicação serial fonte:

<http://www.eletrica.ufpr.br/>

Tentou-se fazer a comunicação serial a partir do RXBUF, que é o comando utilizado para ler(recebe) a partir do pino P1\_1 e armazenar bit por bit em uma comunicação serial UART, os bits que estão no ibutton seriam armazenados na variável char hs\_i [ ] de 8 posições.

```

34  while(1)
35  {
36  for(i=0;i<8;i++)
37  {
38      while((IFG2 & UCA0RXIFG)==0);
39      hs_i[i] = UCA0RXBUF;
40  }

```

**Figura 12** - armazenamento dos bits em hexa na variável hs\_i [ ].

Porém na prática não se teve os resultados esperados, ajustou-se o baud rate na velocidade requerida, porém os bits não estavam sendo armazenados na quantidade esperada (12 dígitos em hexadecimal que seriam 48 bits), o mesmo problema aconteceu com o leitor biométrico, os datasheets dos componentes são muito confusos a respeito dessas especificações, mesmo utilizando o comando breakpoint não foi possível ver os valores que estavam sendo armazenados no vetor hs\_i [ ].

#### IV. OBJETIVOS

- Construir um cofre simples de se utilizar, com segurança, praticidade e confiabilidade para o usuário.

#### V. BENEFÍCIOS

- Alta resistência a choques mecânicos
- Utilização do desbloqueio por senha numérica fácil de configurar.
- Interação com o usuário através de um display LCD
- 

#### VI. REQUISITOS

##### Hardware:

- O projeto deverá contar com uma MSP430 G2553;
- teclado mecânico 3x4, display lcd, trava elétrica, baú de madeira, relé e alimentação de 12 V.

##### Desempenho:

- o microcontrolador receberá instruções a partir do teclado
- caso a senha digitada seja a cadastrado, a trava deve receber 1 na saída (12V) para liberar o cofre. Tudo isso deve acontecer num intervalo de 1 a 5 segundos.[1]

##### Ambiente:

- o teclado e o display devem ficar do lado de fora do cofre para o acesso do usuário, já o microprocessador, a trava elétrica, a alimentação e o relé devem ficar do lado de dentro para evitar violações.

#### VI REFERÊNCIAS

- [1][http://www.cerne-tec.com.br/Artigo\\_07\\_ComunicacaoSerial.pdf](http://www.cerne-tec.com.br/Artigo_07_ComunicacaoSerial.pdf)



[2]<https://www.up.edu.br/blogs/engenharia-da-computacao/wp-content/uploads/sites/6/2015/06/2003.5.pdf>

[3]<http://www.instructables.com/id/16x2-LCD-interfacing-in-4-bit-mode/>

[4]<https://datasheets.maximintegrated.com/en/ds/DS1990A.pdf>

[5]<http://www.eletrica.ufpr.br/~rogerio/MSP430/00%20-%20CD%20DO%20ALUNO%20-%20FRAM/04%20-%20APOSTILAS/APOSTILA%20MSP430%20-%20C%20-%20PARTE%20IV.pdf>

## VII ANEXOS

### 1) CÓDIGO DISPLAY LCD E TECLADO NUMÉRICO INT MAIN(VOID)

```
1623
1624 int main(void){
1625
1626     int atual;
1627     WDTCTL = WDTHOLD|WDTPW;
1628     P1OUT |= ENTR5;
1629     P1REN |= ENTR5;
1630     P1OUT &= ~RELE;
1631     P1OUT &= ~BIT6;
1632     P1OUT |= SAIDAS;
1633     P1DIR |= RELE + SAIDAS;
1634     P1DIR &= ~(ENTRS);
1635     call_lcd_principal();
1636     while(1)
1637     {
1638         P1OUT |= SAIDAS;
1639         P1OUT &= ~SAIDA1;
1640         atual = (P1IN & ENTR5);
```

```

1640     atual = (P1IN & ENTR5);
1641     if(atual!= ENTR5)
1642     {
1643         if (atual == ENTRADA2 + ENTRADA3 + ENTRADA4)
1644         {
1645             pisca(1);
1646             senha_string[j]= '1';
1647             j++;
1648             if(j==4)
1649             {
1650                 digito_1 = senha_string[0];
1651                 digito_1 -= 48;
1652                 digito_2 = senha_string[1];
1653                 digito_2 -= 48;
1654                 digito_3 = senha_string[2];
1655                 digito_3 -= 48;
1656                 digito_4 = senha_string[3];
1657                 digito_4 -= 48;

```

```

1658             call_lcd_senha();
1659         }
1660     }
1661     else if (atual == ENTRADA1 + ENTRADA3 + ENTRADA4)
1662     {
1663         pisca(1);
1664         senha_string[j]= '4';
1665         j++;
1666         if(j==4)
1667         {
1668             digito_1 = senha_string[0];
1669             digito_1 -= 48;
1670             digito_2 = senha_string[1];
1671             digito_2 -= 48;
1672             digito_3 = senha_string[2];
1673             digito_3 -= 48;
1674             digito_4 = senha_string[3];
1675             digito_4 -= 48;

```

```

1675             digito_4 -= 48;
1676             call_lcd_senha();
1677         }
1678     }
1679     else if(atual==(ENTRADA1 + ENTRADA2+ ENTRADA4))
1680     {
1681         pisca(1);
1682         senha_string[j]= '7';
1683         j++;
1684         if(j==4)
1685         {
1686             digito_1 = senha_string[0];
1687             digito_1 -= 48;
1688             digito_2 = senha_string[1];
1689             digito_2 -= 48;
1690             digito_3 = senha_string[2];
1691             digito_3 -= 48;

```

```

1693             digito_4 = senha_string[3];
1694             digito_4 -= 48;
1695             call_lcd_senha();
1696         }
1697     }
1698     else if(atual==(ENTRADA1 + ENTRADA2+ ENTRADA3)) //asterisco
1699     {
1700         call_digite_a_senha_cadastrada();
1701         criar_senha_nova_teclado();
1702     }
1703 }
1704 P1OUT |= SAIDAS;
1705 P1OUT &= ~SAIDA2;
1706 atual = (P1IN & ENTR5);
1707 if(atual!= ENTR5)
1708 {
1709     if (atual == ENTRADA2 + ENTRADA3 + ENTRADA4 )
1710     {

```

```

1710      }
1711      //call_lcd_2();
1712      pisca(1);
1713      senha_string[j]= '2';
1714      j++;
1715      if(j==4)
1716      {
1717          digito_1 = senha_string[0];
1718          digito_1 -= 48;
1719          digito_2 = senha_string[1];
1720          digito_2 -= 48;
1721          digito_3 = senha_string[2];
1722          digito_3 -= 48;
1723          digito_4 = senha_string[3];
1724          digito_4 -= 48;
1725          call_lcd_senha();
1726      }
1727  }

```

```

1728      else if (atual == ENTRADA1 + ENTRADA3 + ENTRADA4)
1729      {
1730          pisca(1);
1731          senha_string[j]= '5';
1732          j++;
1733          if(j==4)
1734          {
1735              digito_1 = senha_string[0];
1736              digito_1 -= 48;
1737              digito_2 = senha_string[1];
1738              digito_2 -= 48;
1739              digito_3 = senha_string[2];
1740              digito_3 -= 48;
1741              digito_4 = senha_string[3];
1742              digito_4 -= 48;
1743              call_lcd_senha();
1744          }
1745      }

```

```

1746      }
1747      else if(atual==(ENTRADA1 + ENTRADA2 + ENTRADA4))
1748      {
1749          pisca(1);
1750          senha_string[j]= '8';
1751          j++;
1752          if(j==4)
1753          {
1754              digito_1 = senha_string[0];
1755              digito_1 -= 48;
1756              digito_2 = senha_string[1];
1757              digito_2 -= 48;
1758              digito_3 = senha_string[2];
1759              digito_3 -= 48;
1760              digito_4 = senha_string[3];
1761              digito_4 -= 48;
1762              call_lcd_senha();
1763          }

```

```

1764      }
1765      else if(atual==(ENTRADA1 + ENTRADA2 + ENTRADA3))
1766      {
1767          pisca(1);
1768          senha_string[j]= '0';
1769          j++;
1770          if(j==4)
1771          {
1772              digito_1 = senha_string[0];
1773              digito_1 -= 48;
1774              digito_2 = senha_string[1];
1775              digito_2 -= 48;
1776              digito_3 = senha_string[2];
1777              digito_3 -= 48;
1778              digito_4 = senha_string[3];
1779          }

```

```

1780         digito_4 -= 48;
1781         call_lcd_senha();
1782     }
1783 }
1784 }
1785 P1OUT |= SAIDAS;
1786 P1OUT &= ~SAIDA3;
1787 atual = (P1IN & ENTR3);
1788 if(atual!= ENTR3)
1789 {
1790     if (atual == ENTRADA2 + ENTRADA3 + ENTRADA4)
1791     {
1792         pisca(1);
1793         senha_string[j]= '3';
1794         j++;
1795         if(j==4)
1796         {
1797             digito_1 = senha_string[0];

```

```

main.c main.c main.c main.c main.c main.c
1798         digito_1 -= 48;
1799         digito_2 = senha_string[1];
1800         digito_2 -= 48;
1801         digito_3 = senha_string[2];
1802         digito_3 -= 48;
1803         digito_4 = senha_string[3];
1804         digito_4 -= 48;
1805         call_lcd_senha();
1806     }
1807 }
1808 else if (atual == ENTRADA1 + ENTRADA3 + ENTRADA4)
1809 {
1810     pisca(1);
1811     senha_string[j]= '6';
1812     j++;
1813     if(j==4)
1814     {
1815         digito_1 = senha_string[0];

```

```

1816         digito_1 -= 48;
1817         digito_2 = senha_string[1];
1818         digito_2 -= 48;
1819         digito_3 = senha_string[2];
1820         digito_3 -= 48;
1821         digito_4 = senha_string[3];
1822         digito_4 -= 48;
1823         call_lcd_senha();
1824     }
1825 }
1826 else if(atual==(ENTRADA1 + ENTRADA2 + ENTRADA4))
1827 {
1828     pisca(1);
1829     senha_string[j]= '9';
1830     j++;
1831     if(j==4)
1832     {
1833         digito_1 = senha_string[0];

```



