

Nome: Isabella Mendes Garcia

Matrícula: 2020007791

## Programação Embarcada

### Relatório do projeto

#### Fechadura Eletrônica com senha

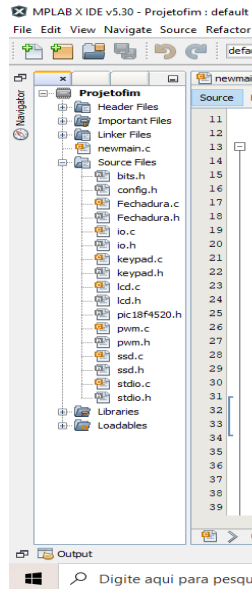
Com o fechamento da disciplina de ECOP04 e ECOP14 ministrada pelo professor Otávio de Souza Martins Gomes no formato RTE, foi solicitado a realização um projeto final com os conceitos aprendidos ao longo do semestre como LEDs, display LCD, displays de 7-seg, teclas, conversores analógicos definir algumas diretrizes. Para que o projeto ficasse semelhante a uma fechadura eletrônica, com a utilização do teclado matricial da placa *PICgenius*, em conjunto com os displays de sete segmentos para mostrar a senha digitada, e o display lcd, com as informações para direcionar o usuário.

Com isso, foi feito o desenvolvimento de um sistema de fechadura eletrônica utilizando a placa *PICGenius*. Na qual o usuário poderá definir uma senha para destravar a fechadura que ficará gravada. O display LDC irá mostrar quantas vezes a fechadura foi destravada. Caso o usuário erre a senha mais de 5 vezes o sistema não permite tentar novamente.

#### Sobre o código

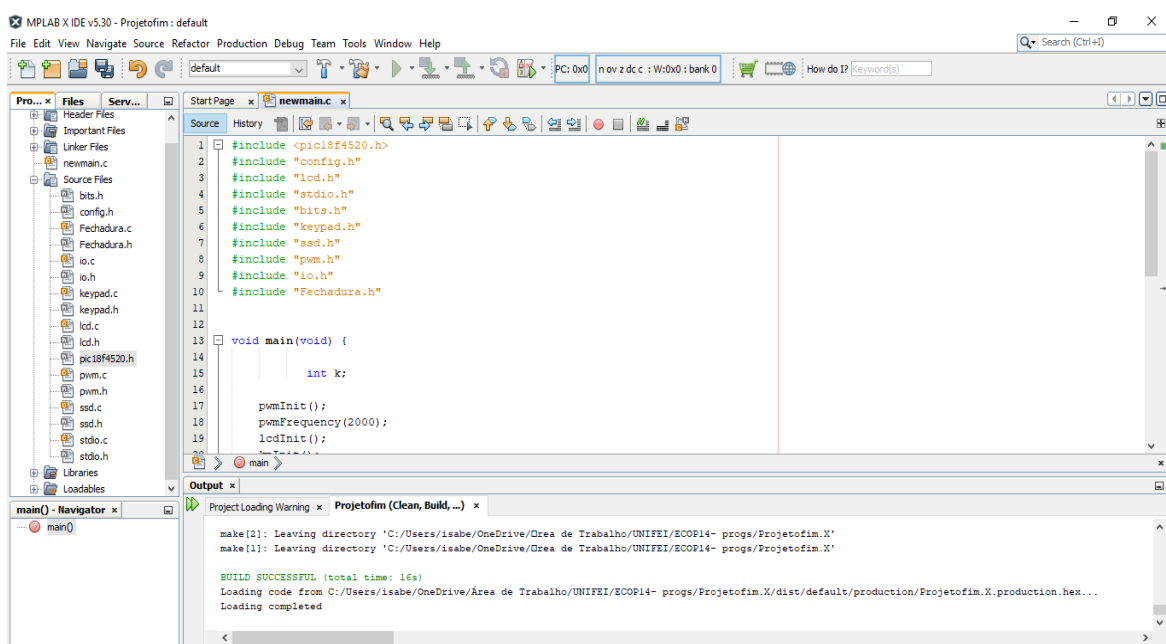
Com o intuito de que o código do projeto ficasse o mais organizado possível, optei por criar um arquivo principal, com o nome de *newmain.c*, onde está somente a função main do meu projeto, com as inicializações e chamadas de funções iniciais.

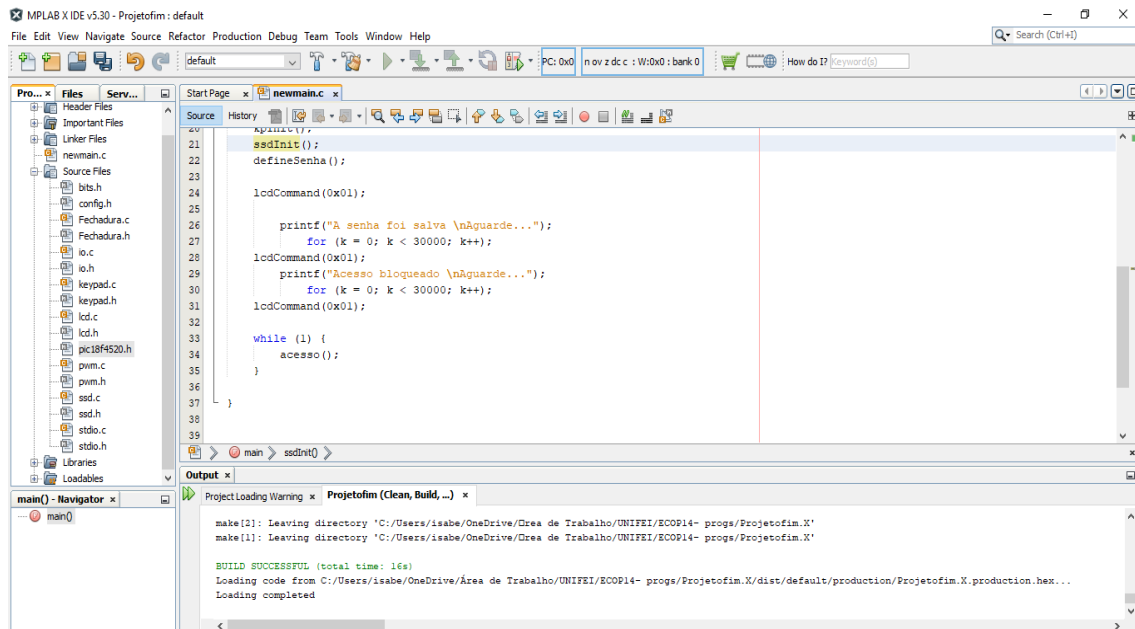
Com as bibliotecas *pic18f4520.h*, *"config.h"*, *"lcd.h"*, *"stdio.h"*, *"bits.h"*, *"keypad.h"*, *"ssd.h"*, *"pwm.h"*, *"io.h"*, *"Fechadura.h"* incluídas.



Com a função *pwmInit()* para iniciar as saídas pwm. A função *pwmFrequency(2000)* para a definição da frequência. O *lcdInit()* para a inicialização do LCD. O *kplnit()* para iniciar o teclado. O *ssdInit()* para iniciar o ssd. A função *defineSenha()* para chamar a função para definir a senha. Após essas funções, foi adicionado o *lcdCommand(0x01)* para a limpeza da tela do LCD. Além da utilização do *printf* para imprimir a mensagem que a senha foi salva e que o acesso está travado. Como também foi usado o laço *for* para o *time*. Com o *while (1)* para o loop infinito e o *acesso()* chama a função *acesso*.

Segue as imagens da tela do MPLAB X IDE :





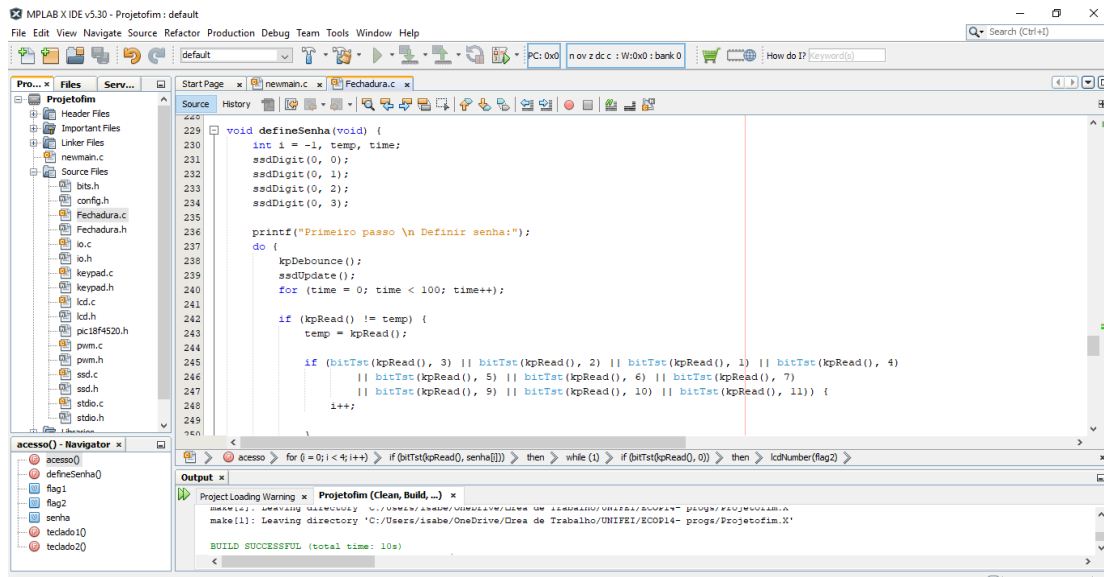
## FUNÇÕES:

### Função *defineSenha()*:

A primeira função que é chamada é a `defineSenha()` para definir a senha digitada pelo usuário, e ela está em um arquivo secundário chamado `fechadura.c`, junto com as outras funções, como `acesso()` e `teclado()`. A `defineSenha()` é a responsável por ler o teclado matricial e salvar as teclas pressionadas na variável `senha[5]`. Essa função só é chamada uma única vez durante a execução do código.

O código abaixo pertence a função `defineSenha()`, e foi feita para realizar a varredura do teclado, salvando as teclas pressionadas. O código se estende da mesma forma para todas as outras teclas, até encerrar o loop, quando "i" for maior que 3, o que significa que as quatro posições do vetor `senha` já foram preenchidas.

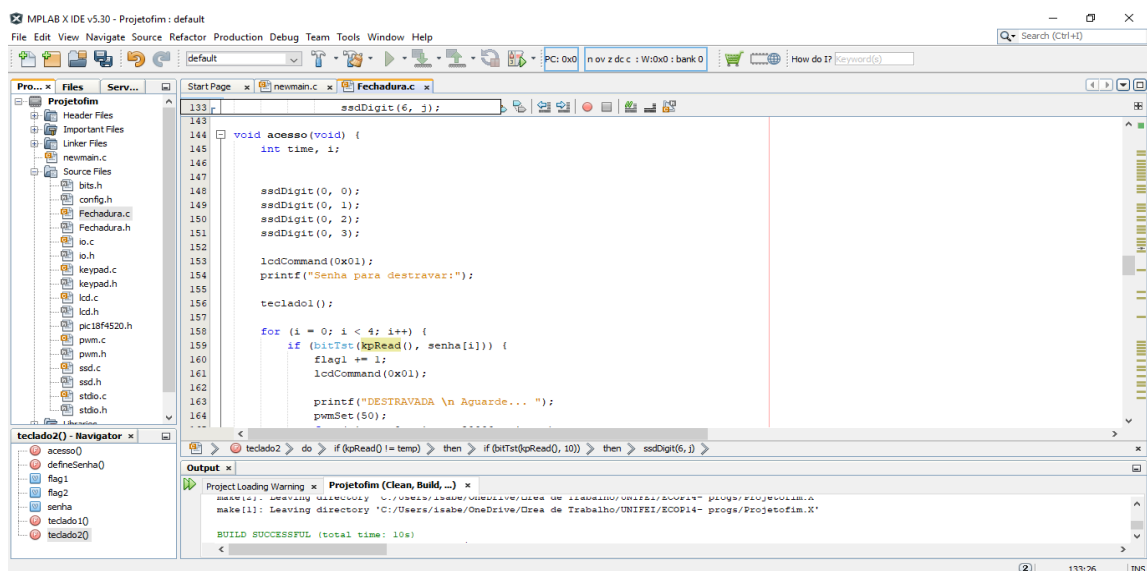
Segue a imagem da tela do MPLAB X IDE:

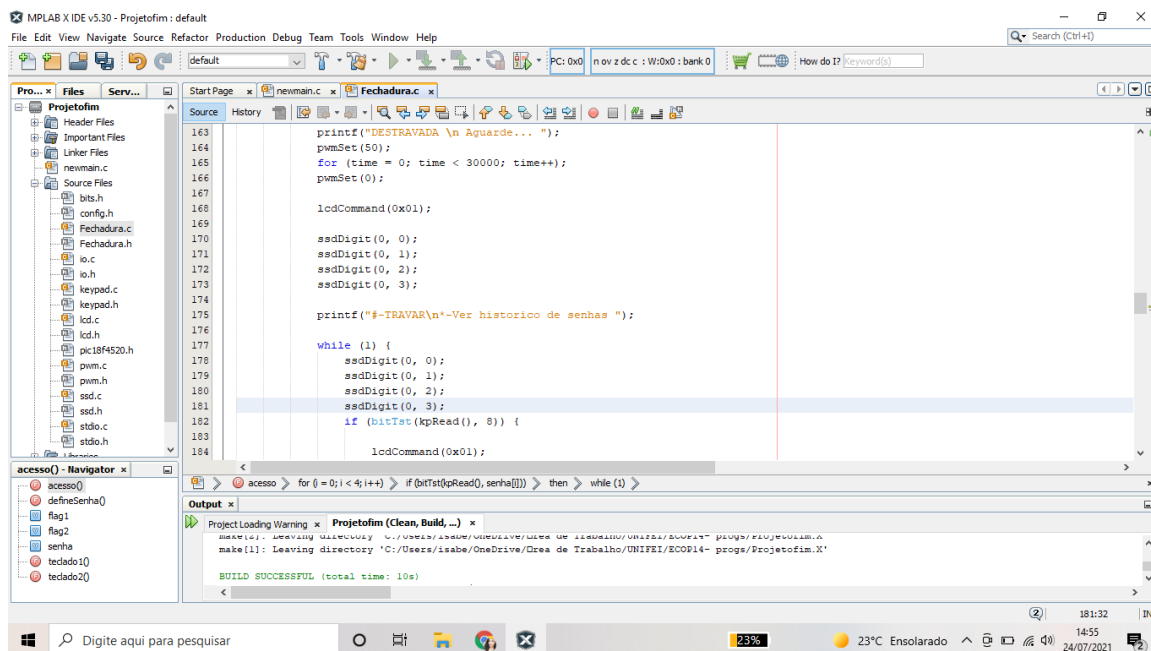


## Função acesso()

Na função `acesso()` é onde é feita a parte mais importante. Ela exibe as mensagens para o usuário travar, destravar, ou consultar o histórico. Além disso, a função também testa se as senhas digitadas estão corretas ou incorretas, e também se as outras demais opções foram escolhidas.

Segue as imagens da tela do MPLAB X IDE:



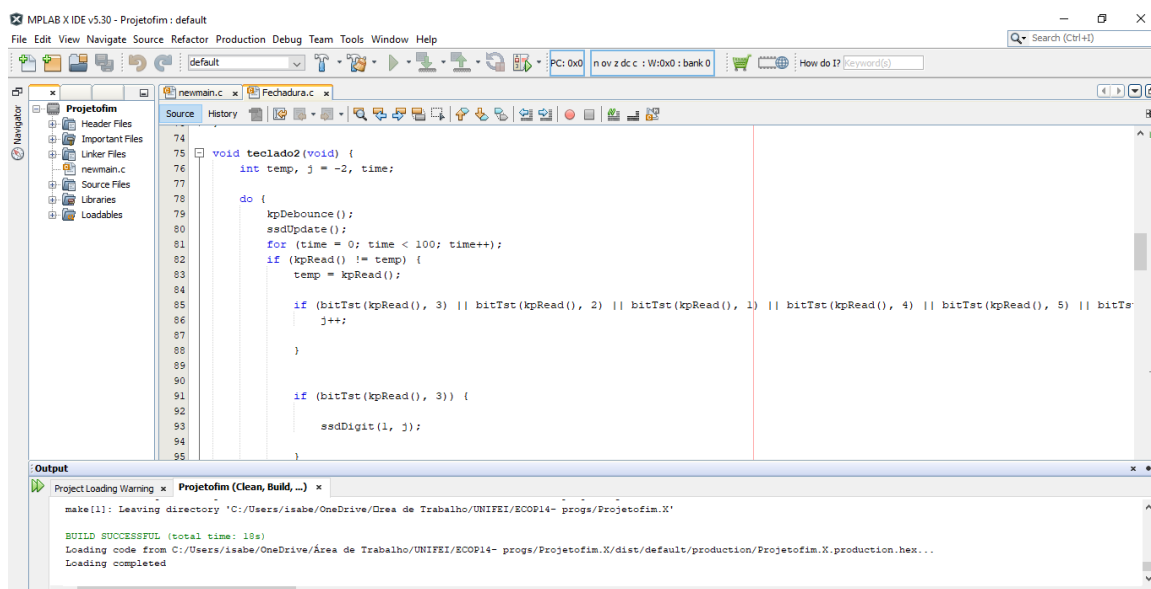
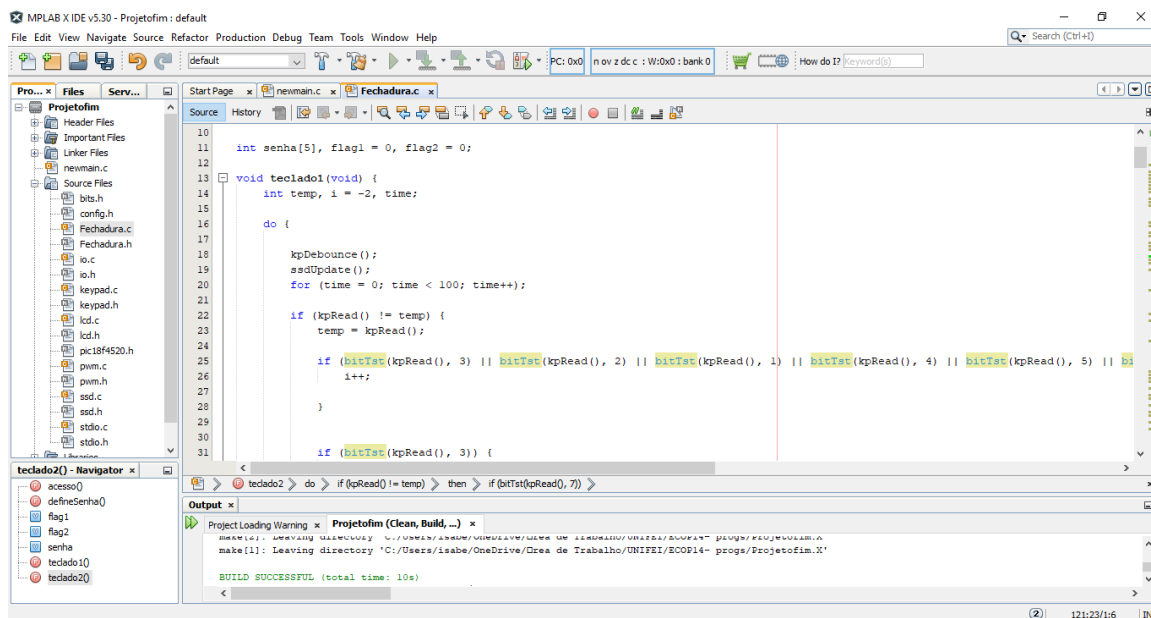


O código acima testa se as senhas foram digitadas corretamente, e trava, destrava e exibe as opções disponíveis. Ele está dentro da função `acesso()`.

## Funções Teclado()

Foram elaboradas duas funções teclado, a `teclado1()` e a `teclado2()`. Ambas funções são semelhantes a função `defineSenha()`. Elas realizam a varredura das teclas pressionadas. Entretanto, diferente da função `defineSenha()`, elas não salvam nada ou guardam valor. Como também, foram criadas separadamente para não interferirem uma na outra. Com as funções `kpDebounce()` e `ssdUpdate()`. Que permite a execução de uma função apenas se um determinado tempo se passou.

Segue as imagens da tela do MPLAB X IDE:



## Dificuldades encontradas e Considerações finais

A disciplina de programação embarcada foi bastante interessante para aprender a manusear a placa PicGenuis, mesmo em ensino remoto foi possível aprender diversos conceitos.

Uma das maiores dificuldades foi desenvolver o tema para que fosse simples e pudesse agregar a maioria dos conceitos que foram ensinados ao longo do curso. Além da dificuldade em conciliar todos os conceitos para que o código ficasse o mais claro possível e com fácil entendimento. Fazer com que o teclado e o display sincronizasse foi uma das dificuldades encontradas também.

Agradeço ao professor por proporcionar o aprendizado necessário, e também aos meus colegas, que ofereceram ajuda e apoio durante o semestre.