

Fechadura Eletrônica com Leitor de RFID

Rebecca Quintino Do Ó

rebeccaquintino@gmail.com, Engenharia Elétrica
EEL7323 - 08235

1 Introdução

O projeto tem como finalidade desenvolver uma fechadura eletrônica usando um módulo leitor rádio-frequência (RFID) modelo MFRC522. Além disso, terá o uso de um Teclado Matricial de Membrana de 16 Teclas para permitir que o usuário cadastre novas tags através de uma senha. Ademais, temos um botão de reset para a fechadura ser destravada a qualquer momento dentro do ambiente. Todos esse sensores estarão conectados a um microcontrolador ESP32S ESP-WROOM-32 que também estará conectado aos leds de sinalização, nosso atuador. Lembrando que os logs de leituras das tags estarão sendo enviados ao computador que também estará conectado ao microcontrolador.

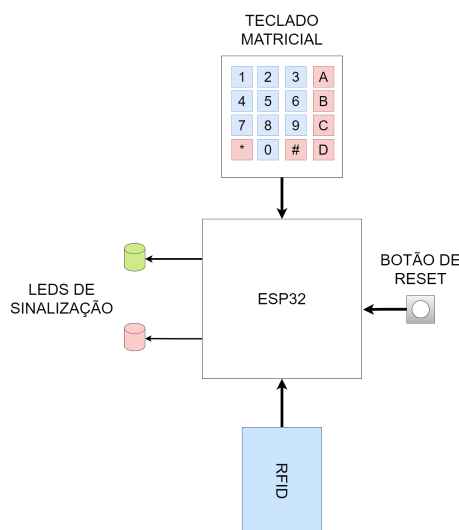


Figure 1: Diagrama simplificado do projeto

2 Funcionamento

Como mencionado anteriormente o sensor de RFID será o responsável por ler e gravar a tag. Assim, para entrar no modo de leitura é necessário que o usuário aproxime a tag no sensor e após verificar se a tag tem seu buffer igual a um caracter específico, a porta se abrirá. Por outro lado, para entrar no modo de

gravação, é necessário que a pessoa aperte uma tecla qualquer para posteriormente digitar uma senha específica. Dessa forma, ela conseguirá gravar algum dado na tag e posteriormente gravar o identificador da tag no sensor RFID. Para que o usuário consiga digitar a senha teremos um Teclado Matricial de 16 teclas. Caso também a pessoa esteja dentro do ambiente e queira abrir a fechadura eletrônica, terá disponível um botão assíncrono que abrirá a porta automaticamente. Temos um resumo do funcionamento descrito neste fluxograma abaixo.

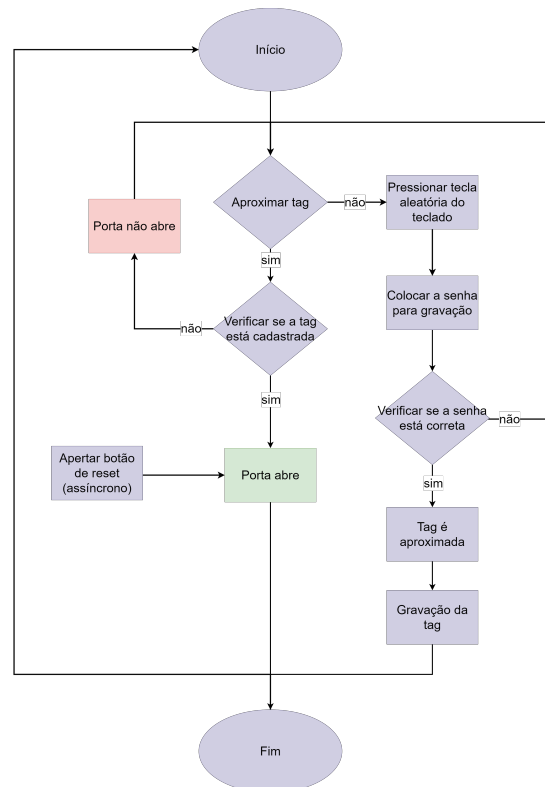


Figure 2: Fluxograma de funcionamento

3 Classes

Implementando todo o conjunto de forma embarcada e utilizando programação C++ com programação orientada à objetos, foi criada a classe *"Peripheral"*, a classe base, que terá como derivadas as classes *"Keyboard"* e *"Buttons"*. Essa classe base terá como funções membro virtuais *init()* e *take_action()* responsáveis pela inicialização do periférico e seu funcionamento básico. As classes derivadas implementarão o Teclado Matricial e botão responsável pela abertura da porta por dentro do ambiente. Ademas, teremos a classe *"RFID"* utilizará a biblioteca externa do RFID para implementar as funções de leitura *read_tag()* e de gravação *write_tag()*. Além disso, utilizando polimorfismo através de um ponteiro do tipo *Peripherals*, esse ponteiro vai acessar os resultados adquiridos

dos sensores, apontando para o periférico específico. Isso tudo será usado na função *handle_events* para abrigar os inputs e fazer a lógica de funcionamento do sistema.

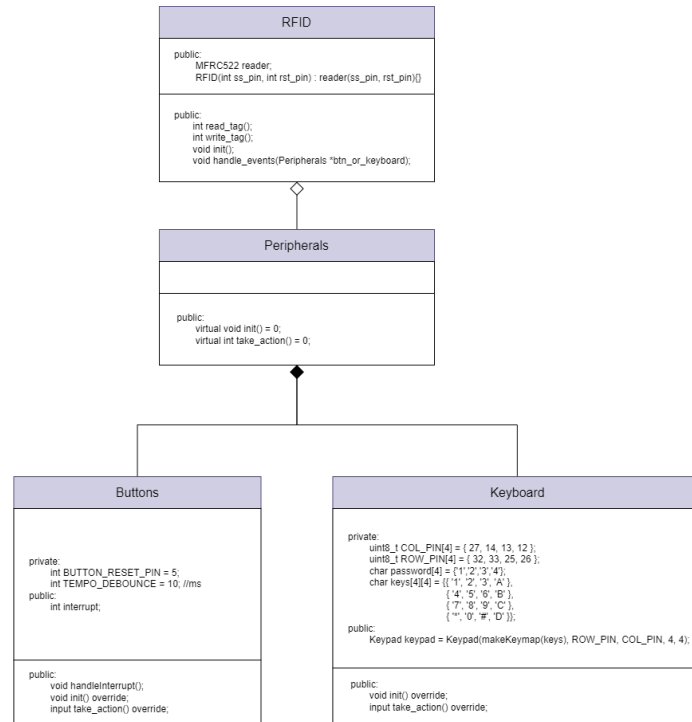


Figure 3: Diagrama de classes RFID, Peripherals, Buttons e Keyboard

Além disso, teremos a classe *ClockCalendar* e *Queue* que será responsável pelo log de data e hora de acesso das tags e isso será mostrado via UART no computador. A classe *ClockCalendar* utiliza *Friends*. O struct *Node* vai conter um valor com uma variável do tipo *Template*, como também, vai ser *Agregada* da classe *Queue*, usando também *Sobrecarga de operadores* para printar os logs. O *tratamento de exceções* vai ser usado dentro da lógica da senha, caso a senha para cadastramento esteja errada, o sistema sairá do modo de gravação e o usuário terá que inserir novamente a senha no teclado.

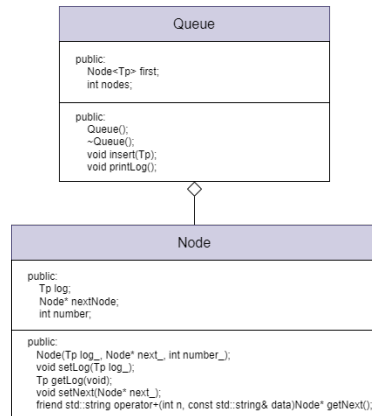


Figure 4: Diagrama de classes Queue e Node

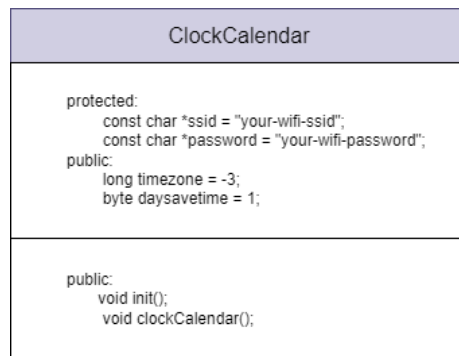


Figure 5: Diagrama de classes do ClockCalendar