
UNIVERSIDADE FEDERAL FLUMINENSE – UFF
ESCOLA DE ENGENHARIA – TCE
CURSO DE ENGENHARIA DE TELECOMUNICAÇÕES – TGT

PROGRAMA DE EDUCAÇÃO TUTORIAL – PET
GRUPO PET-TELE

Tutoriais PET-Tele

Tutorial sobre sistema de controle de acesso RFID

(Versão: A2015M01D21)

Autor: Roberto Brauer di Renna
Thiago Elias Bitencourt Cunha
Thiago Chequer Coelho

Tutor: Alexandre Santos de la Vega

Niterói – RJ
Agosto / 2014

Sumário

1	Introdução	4
2	Um pouco sobre a tecnologia RFID	5
3	Componentes necessários para confecção do projeto	6
4	Circuito de controle	7
4.1	Como tudo funciona	7
4.1.1	Linhas de alimentação	7
4.1.2	Indicador de circuito ligado	7
4.1.3	<i>Reset e reset prog</i>	7
4.1.4	O microcontrolador	8
4.1.5	Circuito relé para acionamento da fechadura	8
4.2	Diagrama do módulo relé	8
4.3	Diagrama da central de controle	9
4.4	A central de controle	9
5	Circuito LCD 16X2 I2C	10
5.1	Um pouco sobre o protocolo I2C	10
5.2	Ó módulo <i>Display</i> LCD 16x2 I2C	10
5.3	Diagrama do circuito LCD 16X2 I2C	12
6	Circuito para acionamento da fechadura através de botão interno	13
6.1	Diagrama do circuito botão	13
7	Confecção artesanal dos circuitos utilizados	13
8	O módulo RFID	14
9	Código de leitura do ID da carteirinha	15
10	Código de controle utilizando os IDs obtidos	16
11	Programando a central através de um conversor USB/serial	20

Lista de Figuras

1	Esquema do módulo relé	8
2	Esquema da central de controle	9
3	A central de controle	9
4	LCD 16X2	11
5	Circuito integrado PCF8574AP	12
6	Esquema do shield LCD I2C	12
7	Esquema do módulo botão	13
8	Módulo RFID	14
9	Conversor USB para Serial	20

Lista de Tabelas

1	Materiais necessários para confecção do projeto	6
2	Conexões do LCD 16X2 ao PCF8574AP	11
3	Conexões do Módulo RFID	14
4	Conexões do conversor USB/Serial na Central de controle	20

1 Introdução

O grupo PET-Tele trabalha com a plataforma Arduino desenvolvendo materiais didáticos, projetos e cursos práticos adicionando conhecimento a formação dos alunos de Engenharia de Telecomunicações da Universidade Federal Fluminense.

Buscando desenvolver um projeto que exemplificasse o uso do Arduino no dia-a-dia e modernizando o acesso a sala do grupo trazendo a tecnologia RFID, já presente na carteirinha estudantil, integrantes do grupo propuseram a construção do Controle de Acesso RFID.

Sua construção se deu em algumas etapas. Primeiramente os integrantes Roberto Brauer di Renna e Thiago Elias Bitencourt Cunha fizeram uma pesquisa sobre a viabilidade do projeto, custo e materiais necessários. Após esta pesquisa e compra dos materiais a dupla deu início a confecção do controle de acesso. Primeiramente todo o projeto foi desenvolvido através de *protoboards* e posteriormente o mesmo foi confeccionado em placas de circuito impresso.

A confecção das placas foi realizada através de uma técnica também estudada pelo grupo e descrita no Tutorial sobre Confecção Artesanal de Circuitos Impressos que possui download gratuito diretamente pela página online do grupo PET-Tele. <http://www.telecom.ufl.br/pet>

São descritas nesse tutorial todas as etapas da construção do projeto assim como os materiais utilizados e os scripts para a programação da central de controle desenvolvida.

2 Um pouco sobre a tecnologia RFID

Utilizando radiofrequência concomitantemente com variações de campo eletromagnético busca-se um simples e único objetivo: A comunicação entre componentes distintos. Se esse foi atingido, o RFID permite uma maior eficiência no rastreamento, localização e registro de objetos (sejam esses de várias naturezas), por exemplo.

Enraizada na Segunda Guerra Mundial, essa tecnologia advém dos radares que eram usados para verificar a aproximação de aviões com uma certa antecedência.

Porém, a história do RFID começa veementemente em 1973 quando foi patenteado um sistema de memória regravável e outro sistema para destravar uma porta utilizando uma etiqueta. Esse era o momento em que a solução para muitos problemas nascia.

Radio Frequency Identification é uma tecnologia focada em rastreamento, identificação e gerência de produtos das mais variadas naturezas. O diferencial dela está exatamente de não ser necessário campo visual direto com o que está sendo monitorado. Isso possibilita essa tecnologia ser adotada como solução para diversos problemas de logística. A estratégia é simples.

Usar RF em uma frequência específica, variação de campo eletromagnético, além de um conjunto de antenas e transmissores corretamente estruturados para capturar dados. Aparentemente é algo não muito complexo, mas quando bem trabalhado possibilita soluções inovadoras.

O RFID é composto por três vertentes principais: um identificador, um leitor e um banco de dados anexo.

Cada componente tem suas próprias características e subdivisões que quando organizadas de uma certa maneira gera um método de utilização distinto do RFID. E esse visa atuar como solução para um determinado tipo de problema.

3 Componentes necessários para confecção do projeto

Quantidade	Material
1	ATmega 328p
1	Cristal oscilador de 16 Mhz
1	LED verde 5mm
1	LED vermelho 5mm
1	1 Relé 5 V
1	Transistor BC337
1	Regulador de tensão 3.3 V LM33V
1	Regulador de tensão 5 V 7805
1	Regulador de tensão 12 V 7812
1	Capacitor eletrolítico 100uF
4	Capacitor cerâmico 100nF
1	LCD 16X2
1	Circuito integrado PCF8574AP
1	Fechadura elétrica 12 V
1	Módulo RFID 13.4 Mhz
1	<i>Display</i> LCD 16X2
1	CI PCF8574AP
2	Resistor 10k Ω
2	Resistor 1k Ω
2	Resistor 470 Ω
1	Trimpot 10k Ω
1	Fonte de alimentação entre 12 V e 30 V com conector P4
1	Conector P4 para placa de circuito impresso
2	Placas de fenolite 10 cm x 10 cm
2	<i>Pushbutton</i>
	<i>Jumpers</i> e pinos para ligação

Tabela 1: Materiais necessários para confecção do projeto

4 Circuito de controle

4.1 Como tudo funciona

O microcontrolador ATmega 328p é responsável pela interpretação dos dados lidos pelo módulo RFID e por um botão interno a sala e pelo acionamento da fechadura quando necessário. Ele possui em sua memória interna o script do projeto. Este microcontrolador é o mesmo que encontramos em placas Arduino como a Arduino UNO. Toda a prototipagem do projeto foi realizada inicialmente utilizando a plataforma de desenvolvimento Arduino.

Este componente realiza a leitura do sensor RFID e do botão de abertura interno constantemente. O script presente no microcontrolador interpreta os dados lidos e caso o botão seja pressionado ou algum ID cadastrado for lido o circuito relé é acionado e a porta é aberta.

Procurando uma maior interação com o usuário acrescentou-se ao projeto um display LCD. Quando uma carteirinha se aproxima ao sensor RFID o circuito realiza imediatamente sua leitura. Caso o ID da mesma não estiver cadastrado no script presente no microcontrolador uma mensagem de "Acesso negado!" aparece no display, caso contrário, se o ID estiver cadastrado, a mensagem "Acesso autorizado!" aparece.

A central de controle foi construída de forma a viabilizar conexão de todas as saídas/entradas digitais ou analógicas presentes no microcontrolador através de pinos para conexão de jumpers. Pinos com saídas de tensão 3.3 V, 5 V e GND também estão presentes. Dessa forma, caso seja necessário a conexão de mais algum dispositivo externo não haverá a necessidade da construção de uma nova placa.

4.1.1 Linhas de alimentação

Foram construídas três linhas de alimentação. Uma de tensão de 3.3v para alimentação do sensor RFID, uma de 5v para alimentação do microcontrolador e uma de 12v para acionamento da fechadura elétrica. Para alimentação do circuito pode-se utilizar uma fonte de alimentação de tensão entre 12v e 30v conectada a entrada P4 do circuito.

Capacitores filtram essa tensão e logo após, as linhas de alimentação são divididas e suas tensões são reguladas através dos reguladores de tensão. O regulador 7805 regula a tensão de entrada em 5 V, o 7812 em 12 V e o LM33V em 3.3 V.

Podemos visualizar essas linhas de alimentação no diagrama do circuito da Figura 2.

4.1.2 Indicador de circuito ligado

Afim de sinalizar que o circuito está devidamente alimentado um led indicador verde foi adicionado na linha de alimentação de 5 V. Visualizamos este led na Figura 2.

4.1.3 *Reset e reset prog*

Pare resetar o microcontrolador foi adicionado ao circuito um botão *Pushbutton* com um dos pinos conectado ao pino *reset* do microcontrolador e o outro ao GND. Assim o microcontrolador reseta quando o botão é pressionado. Um resistor de 1k Ω é conectado ao +5 V e ao reset para manter o pino de reset do microcontrolador em nível alto evitando oscilações e possível reset acidental. Visualizamos este circuito na Figura 2.

O *reset prog* é um pino específico para programação ele possui um capacitor conectado ao pino de reset do microcontrolador e é utilizado quando se programa a central de controle utilizando um conversor USB para Serial.

4.1.4 O microcontrolador

O microcontrolador utilizado é o ATmega 328p presente na plataforma Arduino. O mesmo foi escolhido pela facilidade de programação pois pode-se utilizar a IDE do Arduino para a programação da central e também pelo objetivo principal do projeto que seria exemplificar o uso do Arduino em um projeto prático possível de ser utilizado no dia-a-dia

4.1.5 Circuito relé para acionamento da fechadura

O circuito relé possui uma entrada de alimentação 5 V, uma entrada de controle e uma entrada GND. Podemos visualizar todo o circuito pelo diagrama da Figura 1.

Quando uma tensão de 5 V é enviada pelo microcontrolador ao pino de controle do circuito tem-se então uma ddp de 5 V na base do transistor fazendo com que o mesmo se polarize e permitindo a passagem de corrente entre o coletor e o emissor. Um LED vermelho está conectado em paralelo a base do transistor sinalizando o envio da tensão de acionamento. Quando esse processo acontece uma corrente flui pela bobina do relé fazendo com que o mesmo seja acionado. Como o relé está funcionando como chave para acionamento da fechadura teremos por consequência a abertura da porta.

4.2 Diagrama do módulo relé

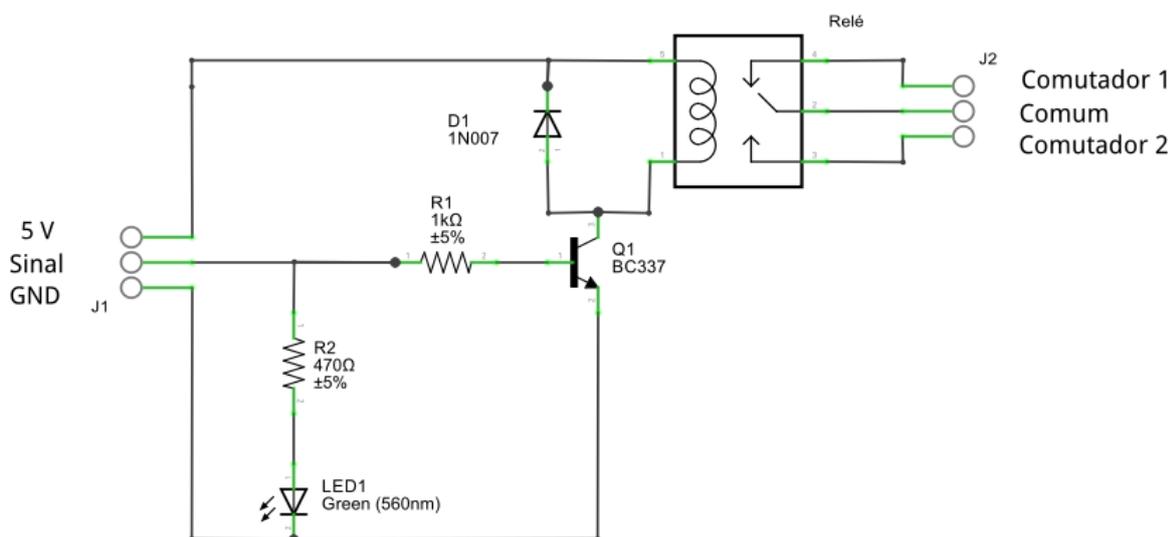


Figura 1: Esquema do módulo relé

4.3 Diagrama da central de controle

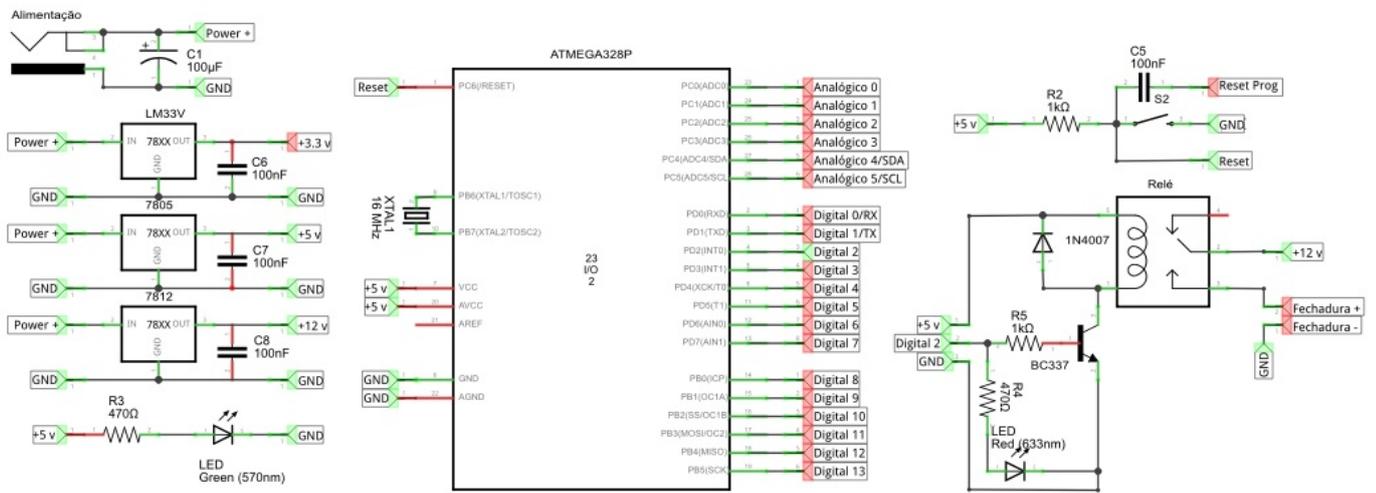


Figura 2: Esquema da central de controle

4.4 A central de controle

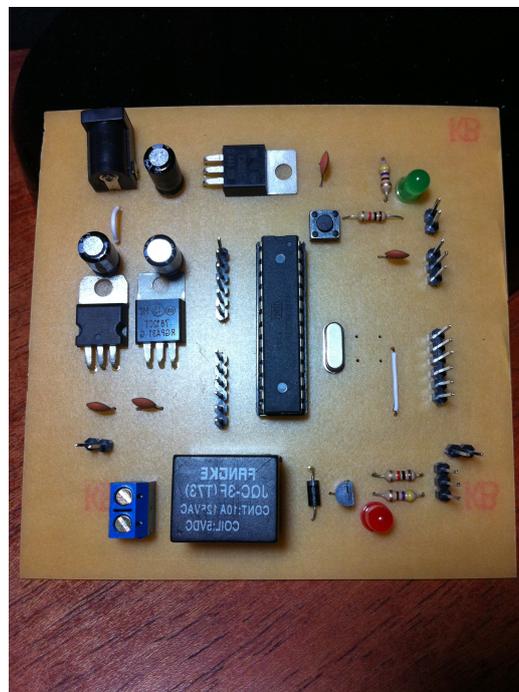


Figura 3: A central de controle

5 Circuito LCD 16X2 I2C

5.1 Um pouco sobre o protocolo I2C

Inter-integrated-circuit é um sistema de comunicação digital e de administração de micro controladores que busca a conexão entre periféricos distintos.

Esses que normalmente apresentam um caráter de baixa velocidade e que serão associados a uma placa mãe ou então a um sistema embutido.

O I2C utiliza um barramento de duas linhas direcionais de comunicação que consiste na associação de uma serial data (SDA) e um serial clock (SLC). O primeiro é responsável pela transmissão de dados digitais; já o segundo é incumbido de controlar o clock do barramento.

Utilizar linhas de comunicação desse tipo faz com que seja possível trabalhar numa tensão de até 5V. Nas próximas páginas será explicada a importância da possibilidade de operação sob essa tensão.

Esse protocolo apresenta dois tipos básicos de dispositivos a serem associados: o master (que é a unidade de controle de todos os periféricos) e o slave (que é cada periférico associado.) Anexado corretamente o sistema pode alcançar taxas de transmissão de até 400 kbit/s.

Para projetos homemade de pequeno porte a utilização do I2C é fundamental. Visto que ele possibilita uma certa modularização dos dispositivos ministrados, isso faz com que haja redução de custos e uma melhor manutenção do sistema separadamente.

Introdutoriamente falando, o sistema utiliza como central o ATmega328p dessa forma o I2C se encaixará de forma ideal para utilização de um display LCD, por exemplo. Além de permitir que os dispositivos funcionem tanto como transmissores ou então receptores.

5.2 O módulo *Display LCD 16x2 I2C*

O LCD 16X2 é um dispositivo que junto ao Arduino nos permite graficar informações diversas. Dentre elas podemos citar desde valores de sensores a algumas palavras.

O nome deste módulo é *LCD 16X2 I2C*, pois se trata de um display LCD de 16 caracteres por coluna com duas colunas. Ou seja, podemos graficar até 32 caracteres no nosso LCD 16X2. O I2C encontra-se presente no nome pois utilizaremos o protocolo de comunicação I2C para envio de dados entre o Arduino e o dispositivo.

Primeiramente é preciso reconhecer as conexões do nosso *Display LCD* com o *CI PCF8574AP*. Para isso teremos a Figura 4 e a Tabela 2.

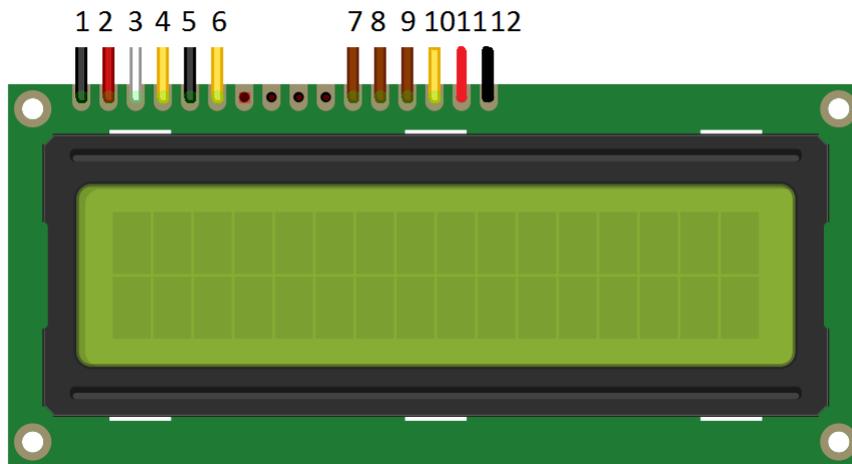


Figura 4: LCD 16X2

Pino	Ligamos a
1	Ground
2	+5V
3	Trimpot 10k Ω
4	Pino 11 PCF8574AP
5	GND
6	Pino 9 PCF8574AP
7	Pino 4 PCF8574AP
8	Pino 5 PCF8574AP
9	Pino 6 PCF8574AP
10	Pino 7 PCF8574AP
11	LED backlight +5 V
12	LED backlight GND

Tabela 2: Conexões do LCD 16X2 ao PCF8574AP

Podemos ver a pinagem no Circuito integrado na Figura 5.

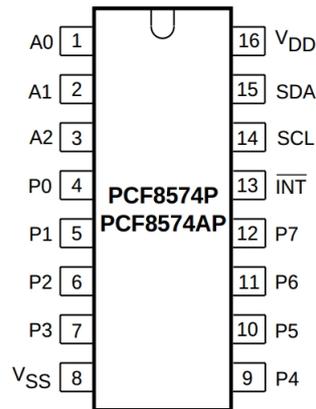


Figura 5: Circuito integrado PCF8574AP

Os pinos A0,A1 e A2 são pinos de endereçamento. No caso como teremos um único PCF8574AP no barramento ligaremos os pinos A0, A1 e A2 na trilha GND gerando um endereço 000. Caso tivéssemos outro *CI* bastava ligar um dos pinos de endereçamento a trilha +5 V gerando um novo endereço. Por exemplo, se ligássemos A2 a trilha +5 V teríamos o endereço 001. O processo resulta no circuito da Figura 6.

5.3 Diagrama do circuito LCD 16X2 I2C

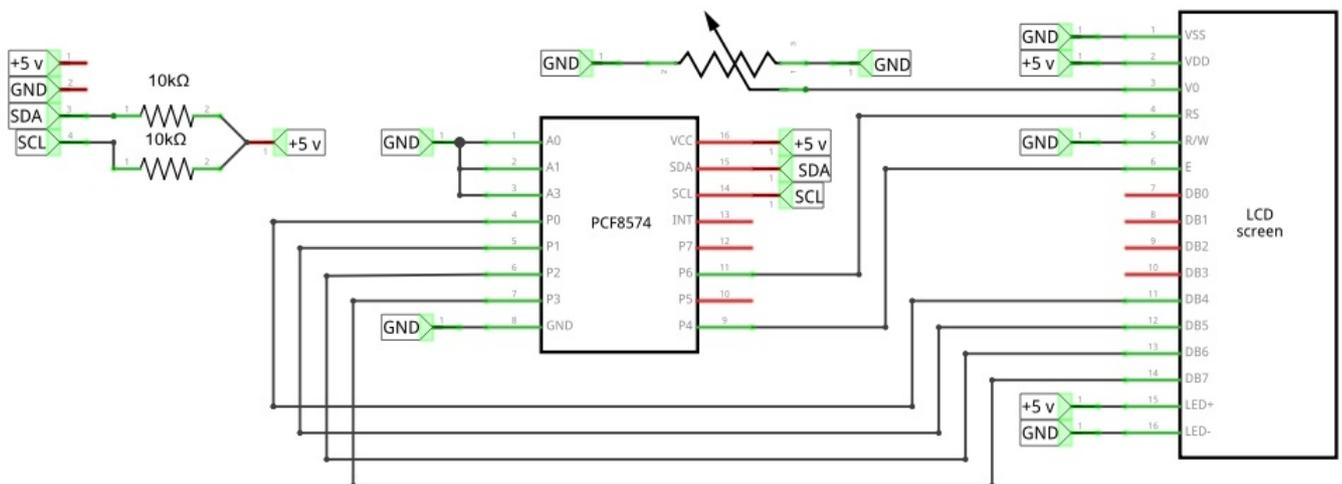


Figura 6: Esquema do shield LCD I2C

6 Circuito para acionamento da fechadura através de botão interno

Pensando em uma maior comodidade para abertura da porta se estando no interior da sala adiciona-se a central de controle um botão para acionamento da fechadura. O circuito é bem simples e pode ser visualizado na Figura 7. Quando o botão é acionado um nível lógico de 5 V é lido pelo circuito de controle e a fechadura é liberada.

6.1 Diagrama do circuito botão

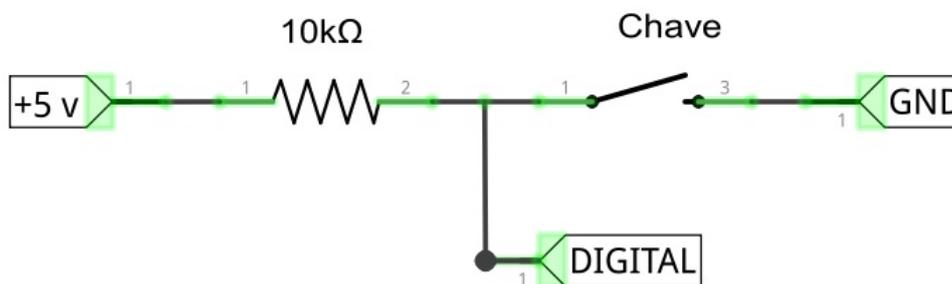


Figura 7: Esquema do módulo botão

7 Confeção artesanal dos circuitos utilizados

A confeção de todos os circuitos utilizados exceto o sensor RFID foi realizada através do tutorial "Confeção artesanal de circuitos impressos" presente para download gratuito no site do grupo www.telecom.ufrj/pet.

8 O módulo RFID



Figura 8: Módulo RFID

Após uma pesquisa sobre o funcionamento da tecnologia RFID e sobre a frequência de comunicação do sensor RFID presente no cartão de identificação dos estudantes descobriu-se que a frequência de comunicação do mesmo é de 13.56 Mhz.

Identificada a frequência de comunicação pode-se adquirir o módulo RFID adequado ao projeto. Para identificar os IDs de cada carteirinha utilizaremos o script "Código de leitura do ID da carteirinha" que será explicado no próximo tópico.

Primeriamente precisamos conectar os pinos de comunicação do módulo a central de controle. Para isso utilizaremos jumpers. Conectaremos os pinos como orientado na Tabela 3

Pino digital da central de controle	Ligamos ao pino do módulo
9	RST
10	SDA
11	MOSI
12	MISO
13	SCK
+3.3 v	+3.3 v
GND	GND

Tabela 3: Conexões do Módulo RFID

9 Código de leitura do ID da carteirinha

Esse código tem por finalidade identificar o ID da carteirinha que for aproximada ao módulo RFID. Uma vez aproximada a carteirinha o módulo realiza a leitura da mesma, envia o ID lido para o Arduino que por sua vez grafica esse dado na ferramenta *Serial Monitor*.

Com os IDs identificados pode-se prosseguir para a programação do microcontrolador com o script final de controle. É importante ressaltar que para que a programação desse código funcione todas as bibliotecas utilizadas no mesmo devem estar presentes na pasta *Libraries* da IDE do Arduino.

```
/*
 * Lembramos que o sensor MFRC522 deve ser de 13.56 MHz
 * Pinagem de ligação:
 * Sinal      Pino           Pino           Pino
 *            Arduino Uno   Arduino Mega   Sensor MFRC522
 * -----
 * Reset      9             5             RST
 * SPI SS     10            53            SDA
 * SPI MOSI   11            51            MOSI
 * SPI MISO   12            50            MISO
 * SPI SCK    13            52            SCK
 */

#include <SPI.h>
#include <MFRC522.h>

#define SS_PIN 10
#define RST_PIN 9
MFRC522 mfr522(SS_PIN, RST_PIN); // Cria uma instância MFRC522.

void setup() {
  Serial.begin(9600); // Inicia a comunicação
  SPI.begin();       // Inicia o barramento SPI
  mfr522.PCD_Init(); // Inicia o sensor MFRC522
  Serial.println("Scan a MIFARE Classic PICC to demonstrate Value Blocks.");
}

void loop() {

  // Procurando novos cartões
  if ( ! mfr522.PICC_IsNewCardPresent() ) {
    return;
  }
  if ( ! mfr522.PICC_ReadCardSerial() ) {
    return;
  }
  // Grafica o ID lido
```

```
Serial.print("Card UID:");
for (byte i = 0; i < mfrc522.uid.size; i++) {
    Serial.println(mfrc522.uid.uidByte[i]);
}
Serial.println();
delay(5000);
}
```

10 Código de controle utilizando os IDs obtidos

Identificados os IDs das carteirinhas podemos programar a central com o código de controle. Os IDs ficam armazenados em vetores cuja variável recebe o nome do dono do cartão como exemplificado no código.

Após configuram-se os periféricos como o botão, a fechadura, o LCD I2C e o módulo. Na função *Void loop()* realizamos a leitura do botão para saber se alguém de dentro da sala deseja abrir a porta e a leitura do módulo RFID para saber se alguém que esteja fora da sala deseja abrir a porta.

Feita a leitura do cartão o ID lido é armazenado em uma variável e posteriormente essa variável é comparada as demais variáveis declaradas anteriormente que armazenam IDs cadastrados. Caso esse cartão lido possua um ID previamente cadastrado, a fechadura é aberta e uma mensagem de "Acesso permitido!" aparece no display, caso contrário a fechadura permanece fechada e a mensagem "Acesso negado!" aparece.

```
/*

* Pinagem
* Sinal      Pino          Pino          Pino
*            Arduino Uno   Arduino Mega  MFRC522 board
* -----
* Reset      9             5             RST
* SPI SS     10            53            SDA
* SPI MOSI   11            51            MOSI
* SPI MISO   12            50            MISO
* SPI SCK    13            52            SCK
*
* 3.3 V E NÃO 5 V!
*/

#include <SPI.h>           // biblioteca
#include <MFRC522.h>       // biblioteca
#include <Wire.h>
#include <LiquidCrystal_I2C.h>

#define SS_PIN 10
#define RST_PIN 9
LiquidCrystal_I2C lcd(0x38,16,2);
```

```
//Variavel para acionamento da fechadura e variavel para
leitura do botão
int fechadura = 2;
int botao = 3;

// cartões programados
int id_roberto[4]={168,198,122,405};
int id_lorraine[4]={184,67,52,333};

int id_maristela[4]={184,176,27,333};
int id_diogo[4]={184,194,217,223};
int id_diogo2[4]={106,251,6,116};
int id_bia[4]={168,105,215,167};
int id_paula_cunha[4]={169,243,123,108};
int id_thiagocunha[4]={164,22,128,108};
int id_thiagochequer[4]={140,85,315,112};
int id_paula_woyames[4]={184,145,42,222};
int id_lucas[4]={184,191,62,232};
int id_tutor[4]={119,219,74,124};
int id_thais[4]={58,43,148,237};
int id_carol[4]={184,210,99,222};
int id_cassio[4]={42,47,2,149};
int id_gustavo[4]={184,174,155,280};;

MFRC522 mfrc522(SS_PIN, RST_PIN); // Cria a instância referente ao MFRC522

void setup() {
    lcd.init();
    lcd.backlight();
    SPI.begin(); // Inicia o barramento SPI
    mfrc522.PCD_Init(); // Inicia o cartão MFRC522
    pinMode(fechadura, OUTPUT);
    pinMode(botao, INPUT);
    lcd.setCursor(0,0);
    lcd.print("PET-Tele!");
    lcd.setCursor(0,1);
    lcd.print("Acesso RFID!");
}

void loop() {
    if(digitalRead(botao)==LOW){
        delay(1000);
        digitalWrite(fechadura, HIGH); // abre a porta
        delay(500);
        digitalWrite(fechadura, LOW); // fecha a porta
    }
}
```

```
    }
// Recebe a procura por algum novo cartão
if ( ! mfr522.PICC_IsNewCardPresent() ) {
    return;
}

// Seleciona o cartão
if ( ! mfr522.PICC_ReadCardSerial() ) {
    return;
}

int verificador = 0;

// Realiza o teste para verificar se o cartão está habilitado
for (byte i = 0; i < 4; i++) {
    if(id_roberto[i] == mfr522.uid.uidByte[i] ||
        id_lorraine[i]== mfr522.uid.uidByte[i]){
        verificador=verificador+1;
    }
    else if(id_maristela[i]== mfr522.uid.uidByte[i] ||
        id_diogo[i]== mfr522.uid.uidByte[i] ||
        id_diogo2[i]== mfr522.uid.uidByte[i]){
        verificador=verificador+1;
    }
    else if(id_thiagocunha[i]== mfr522.uid.uidByte[i] ||
        id_lucas[i]== mfr522.uid.uidByte[i] ||
        id_paula_woyames[i]== mfr522.uid.uidByte[i]){
        verificador=verificador+1;
    }
    else if(id_carol[i]== mfr522.uid.uidByte[i] ||
        id_thiagochequer[i]== mfr522.uid.uidByte[i] ||
        id_tutor[i]== mfr522.uid.uidByte[i]){
        verificador=verificador+1;
    }
    else if(id_cassio[i]== mfr522.uid.uidByte[i] ||
        id_gustavo[i]== mfr522.uid.uidByte[i] ||
        id_thais[i]== mfr522.uid.uidByte[i]){
        verificador=verificador+1;
    }
}
if(verificador == 4){
    verificador=0;
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("Acesso permitido!");
    lcd.setCursor(0,1);
    lcd.print("Bem-vindo! :D");
}
```

```
        digitalWrite(fechadura, HIGH);    // abre a porta
        delay(2000);
        lcd.clear();
        lcd.setCursor(0,0);
        lcd.print("PET-Tele!");
        lcd.setCursor(0,1);
        lcd.print("Acesso RFID!");
    }
    else{
        lcd.clear();
        lcd.setCursor(0,0);
        lcd.print("Acesso negado!");
        lcd.setCursor(0,1);
        lcd.print("Tente novamente!");
        delay(2000);
        lcd.clear();
        lcd.setCursor(0,0);
        lcd.print("PET-Tele!");
        lcd.setCursor(0,1);
        lcd.print("Acesso RFID!");
    }

    digitalWrite(fechadura, LOW);    // fecha a porta
}
```

11 Programando a central através de um conversor USB/serial

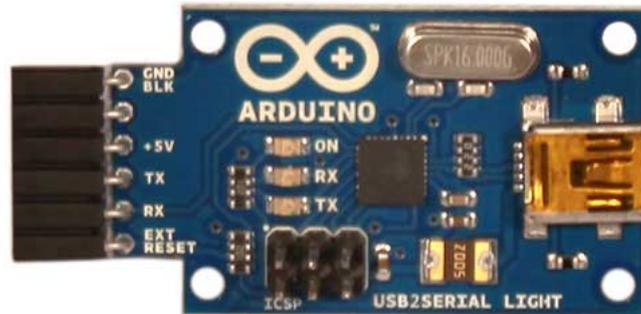


Figura 9: Conversor USB para Serial

A fim de programar a central de controle sem que seja necessário a retirada do microcontrolador da placa e o uso de uma placa Arduino utiliza-se um conversor USB para Serial. A programação se torna mais prática dessa forma. Basta conectarmos o conversor a um computador que tenha a IDE do Arduino instalada e conectarmos os pinos de comunicação.

Desligamos a central de controle e conectamos o pino +5v do conversor ao +5v da central, o GND no GND da central, o pino TX no RX da central e o pino RX no TX da central. Ilustra-se essas conexões na Tabela 4.

Conversor USB/Serial	Central de controle
+5 V	+5V
GND	GND
Tx	Rx
Rx	Tx

Tabela 4: Conexões do conversor USB/Serial na Central de controle

Na IDE selecionamos a plataforma arduino cujo o microcontrolador da central pertencia, ou seja, o bootloader que foi programado no mesmo antes que ele fosse encaixado na central. Por exemplo, o microcontrolador que usamos para esse tutorial está com o bootloader do Arduino Uno.

Lembre-se também de instalar o driver do seu conversor e selecionar na IDE a porta de comunicação do mesmo. Feito todos esses passos e com o script aberto na IDE basta clicar no botão *Upload* e pronto. A central de controle estará funcionando.