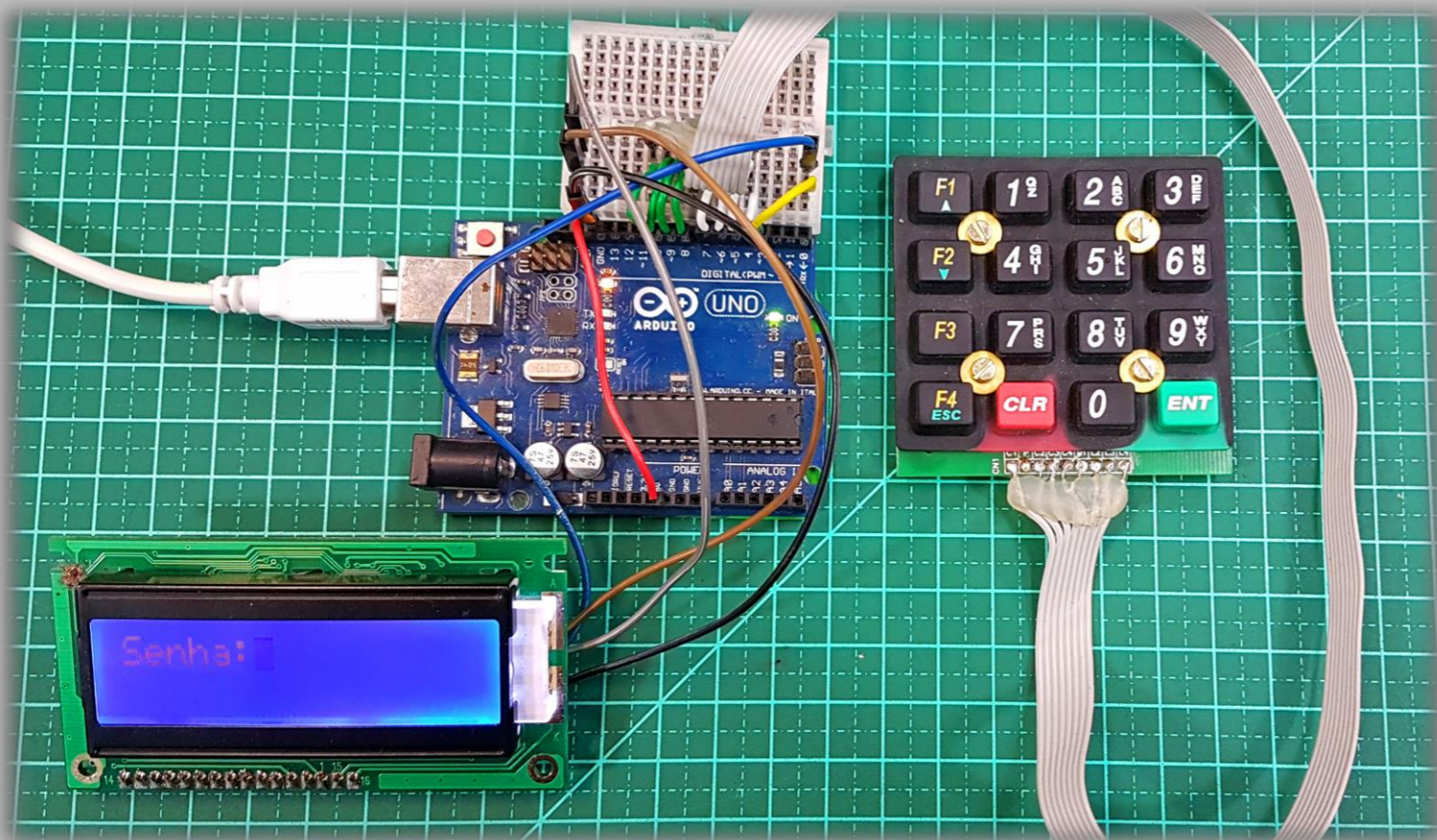


Teclado Matricial e Display LCD Serial

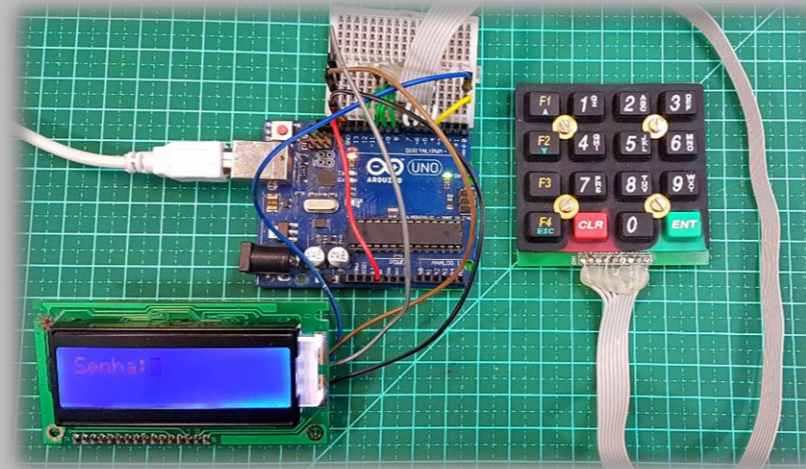


Por Fernando Koyanagi

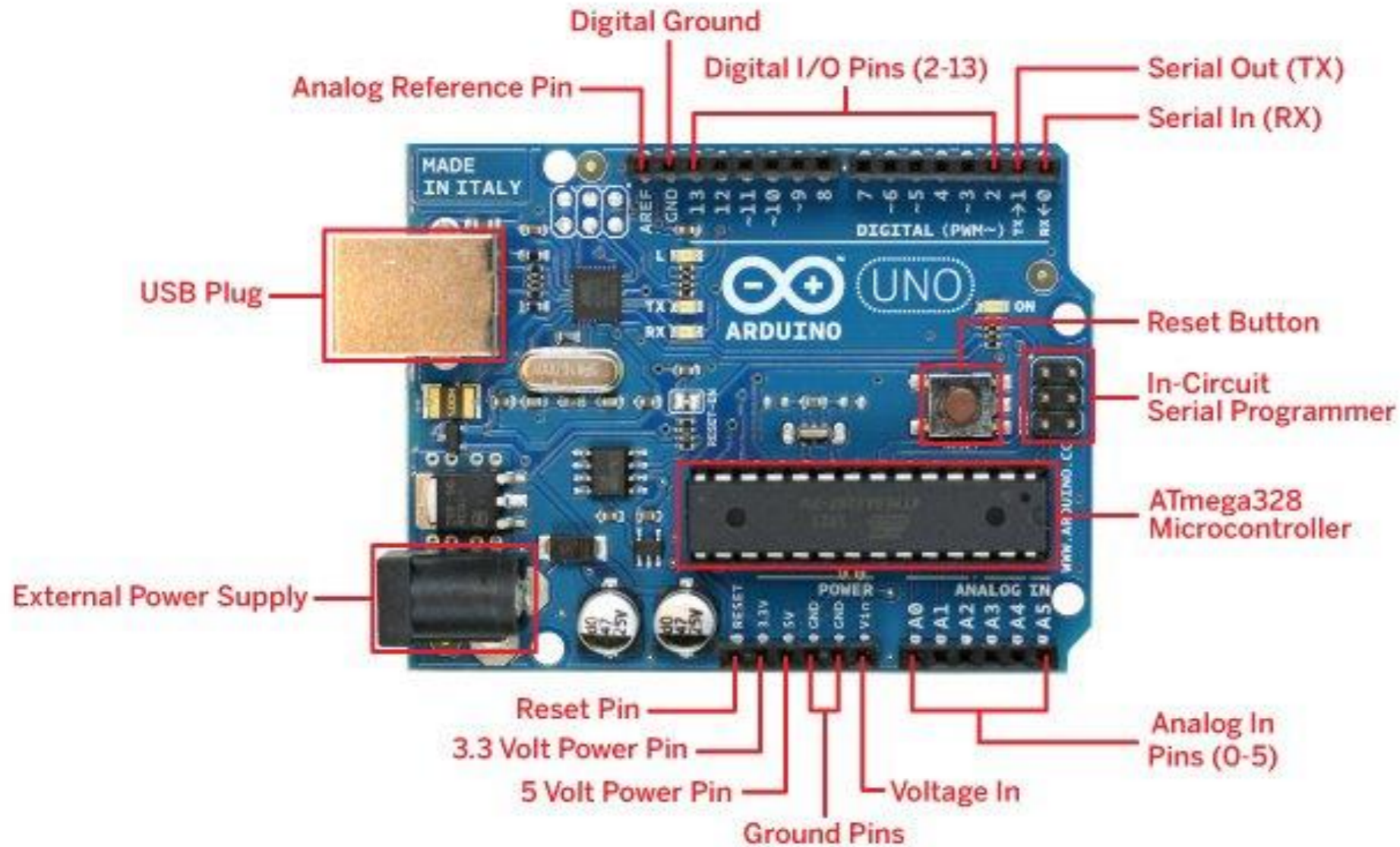


Objetivo

Nosso objetivo será criar um programa que recebe uma senha digitada pelo usuário através do teclado matricial e compara com a senha correta para exibir uma mensagem de correto ou incorreto, a senha e as mensagens serão mostradas no display LCD 2x16.



Arduino UNO

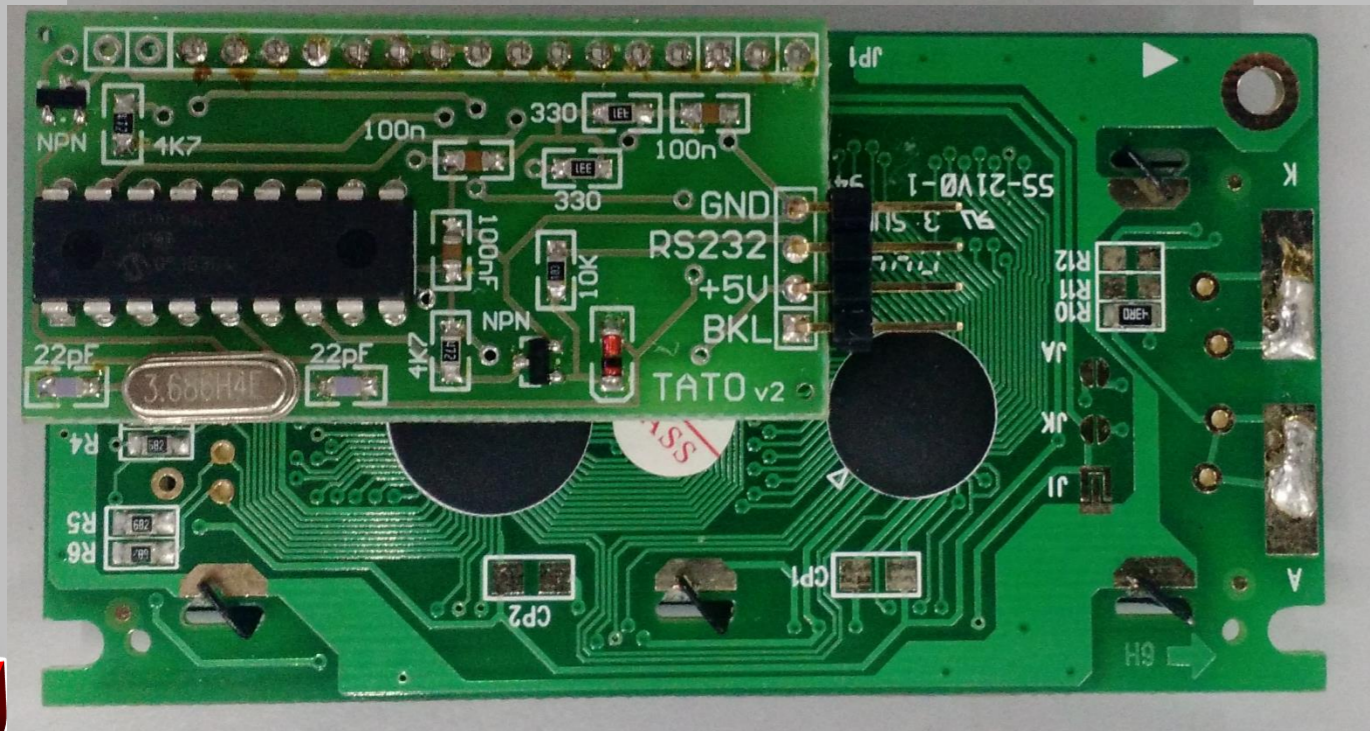


Em www.fernandok.com

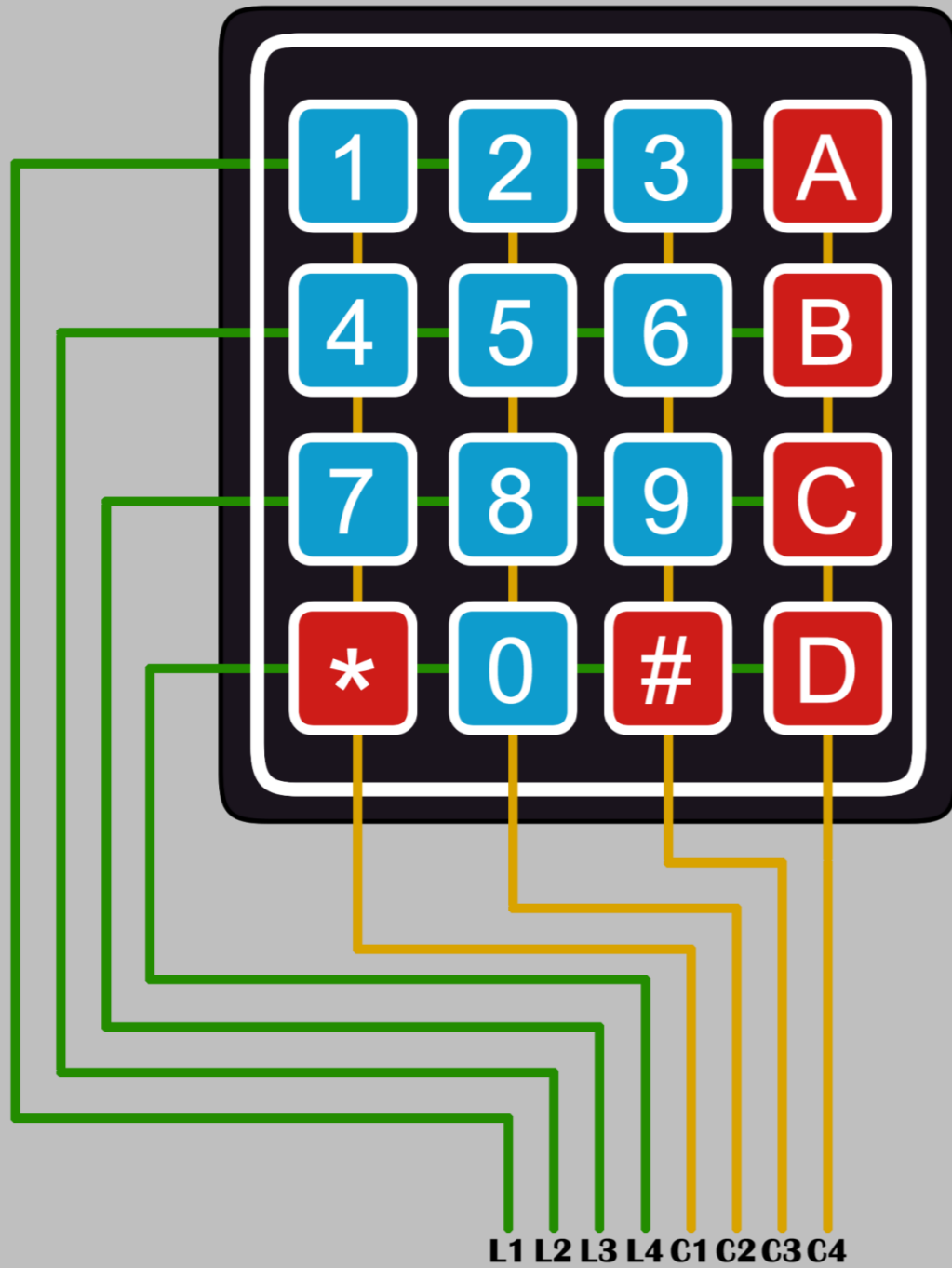
Download arquivo **PDF** dos diagramas
Download arquivo **INO** do código fonte



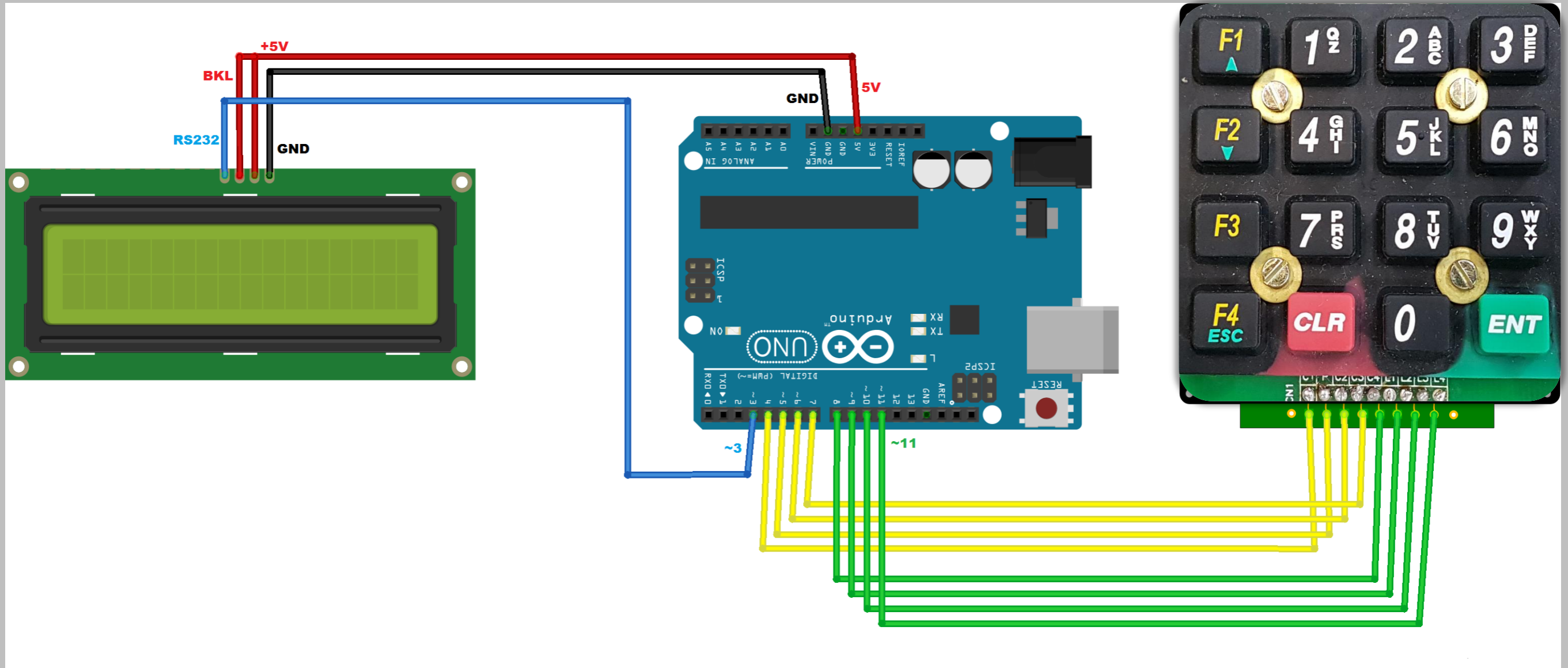
Display LCD 2x16 Serial



Teclado Matricial 4x4



Montagem



Vamos ao código

Começaremos definindo as bibliotecas e constantes que utilizaremos durante nosso código.

```
//biblioteca responsável pela comunicação com o display LCD
#include <SoftwareSerial.h>
//biblioteca responsável por capturar a tecla que foi pressionada no teclado
#include <Keypad.h>

// pino inválido apenas usado no construtor do SoftwareSerial
#define RX_PIN 255
// pino TX da nossa ligação Arduino x Display LCD
#define TX_PIN 3
// inverte a lógica dos pinos Rx e Tx, tratando LOW como HIGH e vice-versa
//alguns drivers para amplificar o sinal usam a entrada com lógica invertida
#define INVERTED 1
```



Para nos comunicar com o display, temos algumas coisas importantes a pontuar antes.

- **Baud rate com velocidade de 2400**
- **Para qualquer comando que seja instrução ao display (não dados), devemos preceder com o prefixo 254 . (Manual pode ser encontrado [Aqui](#))**

Comandos do Display TS-216

Comando/Ação	Código
Limpa a tela	1
Desloca um caracter para esquerda	24
Desloca um caracter para direita	28
Home (desloca o cursor para primeira posição)	2
Move o cursor para esquerda	16
Move o cursor para direita	20
Liga cursor	14
Liga cursor piscando	13
Desliga cursor	12
Apaga o display (sem limpar)	8
Mostra o display	12
Posiciona cursor	128+pos
Posiciona no gerador de caracter	64+pos
Desliga o Backlight	254,255
Liga o Backlight	254,254
Ajusta o contraste	254,253,valor (0-15)

0 é o valor para o maior contraste e 15 para o menor

Usuários avançados: Estes são os códigos enviados para o display. Quando a interface recebe o prefixo de comando (254) ele coloca a linha RS do display em 0. O caracter seguinte ao prefixo será enviado para o display com o pino RS em 0. Após enviar o byte o pino RS volta para nível 1(modos de dados). Isto significa que você pode enviar qualquer comando válido para o display. Como a interface inicializa o display quando é ligada, você NÃO pode enviar nenhum comando de inicialização. Se você não entendeu o que foi explicado acima, não se preocupe. Você não precisa entender isto para usar o display TS-216.



Abaixo vemos alguns comandos para o display que utilizaremos durante o código. Respeitando sempre o prefixo 254 antes de cada instrução.

```
//comando para limpar toda a tela do display
const char limparTela[ ] = {
254,1};

//comandos de configuração inicial
/*
254,254 --> acende o backlight
254,1 --> limpa a tela
254,253,1 --> configura o contraste em nível alto
254,13 --> liga o cursor para ficar piscando
*/
const char configInicial[ ] = {
254,254, 254,1, 254,253,1, 254,13};
```



Dados referentes ao teclado matricial

```
const byte LINHAS = 4; //número de linhas do teclado
const byte COLUNAS = 4; //número de colunas do teclado

//define uma matriz com os símbolos que deseja ser lido do teclado
char SIMBOLOS[LINHAS][COLUNAS] = {
    {'A', '1', '2', '3'},
    {'B', '4', '5', '6'},
    {'C', '7', '8', '9'},
    {'D', 'c', '0', 'e'} // 'c' = CLEAR    'e' = ENTER
};

byte PINOS_LINHA[LINHAS] = {8, 9, 10, 11}; //pinos que indicam as linhas do
teclado
byte PINOS_COLUNA[COLUNAS] = {4, 5, 6, 7}; //pinos que indicam as colunas do
teclado

//instancia de Keypad, responsável por capturar a tecla pressionada
Keypad customKeypad = Keypad( makeKeymap(SIMBOLOS), PINOS_LINHA, PINOS_COLUNA,
LINHAS, COLUNAS);
```



A seguir veremos as variáveis de controle para a senha digitada e a correta para comparação. Além do construtor do objeto (displaySerial) que fará a comunicação com o display.

```
//variáveis responsáveis por armazenar as senhas
const String SENHA_ESPERADA = "1234ABCD";
String SENHA_DIGITADA = "";

//instancia de SoftwareSerial para nos comunicar com o Display via serial
SoftwareSerial displaySerial = SoftwareSerial(RX_PIN, TX_PIN,
INVERTED);
```



Setup

Na função *setup()*, vamos configurar o baud rate do objeto serial e chamar os primeiros comandos para inicializar o display.

```
void setup() {  
    //inicializando a serial de comunicação com o display  
    //importante o baud rate ser de 2400  
    displaySerial.begin(2400);  
    //tempo de espera pela inicialização do display  
    delay(700);  
    //seta a configuração inicial do display  
    displaySerial.print(configInicial);  
    delay(10);  
    inicio();  
}  
  
//função responsável por imprimir na tela a mensagem para digitar a senha.  
//é chamada toda vez q a senha foi digitada e comparada, também quando  
//a tecla limpar display foi pressionada.  
void inicio(){  
    displaySerial.print("Senha:");  
}
```



Loop

Na função *loop()*, teremos um objeto verificando o teclado a todo momento para verificar se alguma tecla foi pressionada. A partir daí, verificaremos a tecla através do comando “switch” e manipularemos os dados de acordo com nosso objetivo.

```
//captura a tecla pressionada do teclado
char customKey = customKeypad.getKey();

//caso alguma tecla foi pressionada
if (customKey){

    switch(customKey)
    {
        //.../
    }
}
```



Loop

Por dentro do comando *switch* : parte 1

```
switch(customKey)
{
    //caso alguma das teclas imprimíveis foi pressionada
    case 'A':
    case 'B':
    case 'C':
    case 'D':
    case '0':
    case '1':
    case '2':
    case '3':
    case '4':
    case '5':
    case '6':
    case '7':
    case '8':
    case '9':
        //concatena o novo símbolo a senha que estamos digitando
        SENHA_DIGITADA+=customKey;
        //imprime na tela o símbolo pressionado
        displaySerial.print(customKey);
        break;
```



Loop

Por dentro do comando *switch* : parte 2

```
//caso a tecla CLEAR tenha sido pressionada
case 'c':
    //limpa a variável que guarda a senha que está sendo digitada
    SENHA_DIGITADA = "";
    //chama o comando para limpar a tela
    displaySerial.print(limparTela);
    //configura a mensagem para digitar a senha
    inicio();
    break;
```



Loop

Por dentro do comando *switch* : parte 3

```
//caso a tecla ENTER seja pressionada, devemos comparar as senhas
case 'e':
    //limpa a tela
    displaySerial.print(limparTela);
    //se a senha digitada foi igual a ESPERADA
    if(SENHA_ESPERADA==SENHA_DIGITADA)
    { //imprime mensagem de senha correta
        displaySerial.print("Senha Correta!!!");
    }
    //caso senha esteja errada
    else{//imprime mensagem de senha incorreta
        displaySerial.print("Senha Incorreta!");
    }
    //aguarda 2 segundos para limpar a tela novamente e esperar uma nova senha
    ser digitada
    delay(2000);
    displaySerial.print(limparTela);
    SENHA_DIGITADA = "";
    inicio();
    break;
```



Em www.fernandok.com

Download arquivo **PDF** dos diagramas
Download arquivo **INO** do código fonte

