

Universidade Católica de Petrópolis
Semana Científica do CEC 2022

Mini curso Arduino

Estação Meteorológica

Dia 2

Vanessa Wendling
Felipe Baldner
Ana Carolina Carius

Mini curso Arduino Estação Meteorológica - Dia 2



Repositório GitHub com o material do curso:

<https://github.com/VanessaWendling/Curso-Arduino-SC-CEC-2022.git>

Outros links e ferramentas úteis:



[Tinkercad](https://www.tinkercad.com)



[Arduino](https://www.arduino.cc)



[KiCad](https://www.kicad.org)

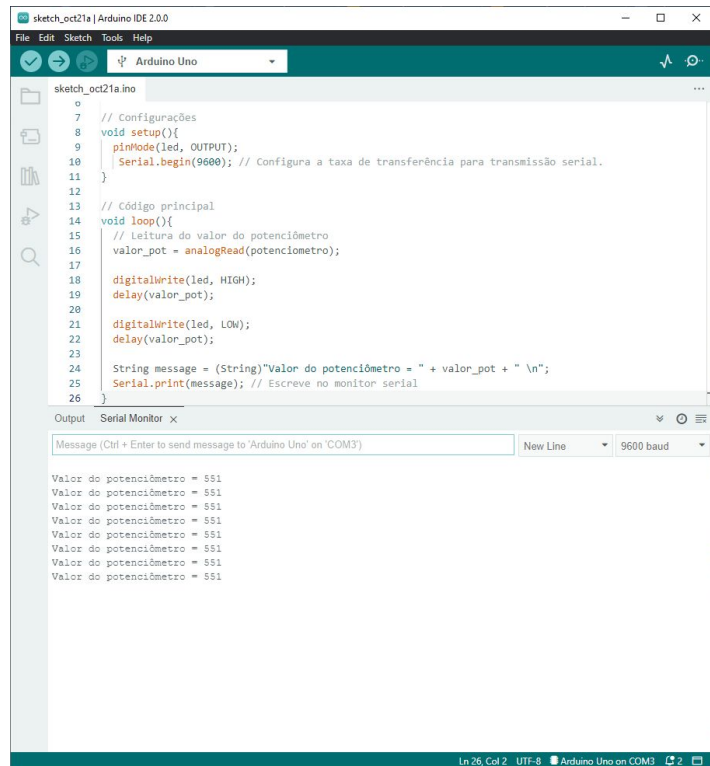
Mini curso Arduino Estação Meteorológica - Dia 2

Programação do dia:

1. Utilizando o monitor serial do Arduino
2. **Programa 1: Programa Oi Mundo no monitor serial**
3. **Programa 2: Programa Pisca-LED + Potenciômetro no monitor serial**
4. Funcionamento do shield display LCD keypad
5. Composição da placa display LCD keypad
6. Diagrama esquemático do shield display LCD keypad
7. **Programa 3: Programa Oi Mundo no display LCD**
8. **Programa 4: Programa display LCD com botões**
9. **Programa 5: Programa Pisca-LED + Potenciômetro + display LCD**

Mini curso Arduino Estação Meteorológica - Dia 2

1. Utilizando o monitor serial do Arduino



The screenshot shows the Arduino IDE 2.0.0 interface. The sketch editor displays a program named 'sketch_oct21a.ino'. The code includes a setup function for a pin and a serial port, and a loop function that reads an analog value from a potentiometer, toggles an LED, and prints the value to the serial monitor. The serial monitor at the bottom shows the output: 'Valor do potenciômetro = 551' repeated seven times. The status bar at the bottom indicates 'Ln 26, Col 2 UTF-8 Arduino Uno on COM3'.

```
sketch_oct21a.ino
0
1 // Configurações
2
3 void setup(){
4   pinMode(led, OUTPUT);
5   Serial.begin(9600); // Configura a taxa de transferência para transmissão serial.
6 }
7
8 // Código principal
9 void loop(){
10  // Leitura do valor do potenciômetro
11  valor_pot = analogRead(potenciometro);
12
13  digitalWrite(led, HIGH);
14  delay(valor_pot);
15
16  digitalWrite(led, LOW);
17  delay(valor_pot);
18
19  String message = (String)"Valor do potenciômetro = " + valor_pot + " \n";
20  Serial.print(message); // Escreve no monitor serial
21
22 }
```

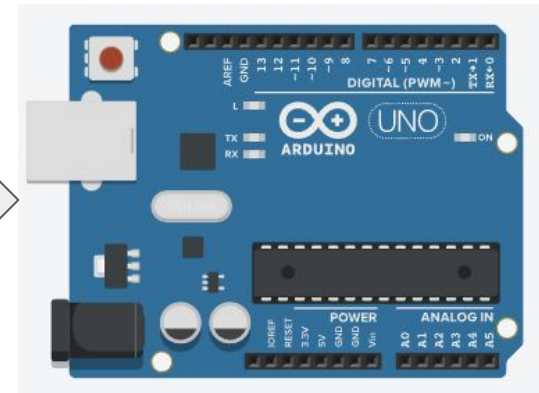
Output Serial Monitor x

Message (Ctrl + Enter to send message to 'Arduino Uno' on 'COM3') New Line 9600 baud

Valor do potenciômetro = 551
Valor do potenciômetro = 551
Valor do potenciômetro = 551
Valor do potenciômetro = 551
Valor do potenciômetro = 551
Valor do potenciômetro = 551
Valor do potenciômetro = 551

Ln 26, Col 2 UTF-8 Arduino Uno on COM3

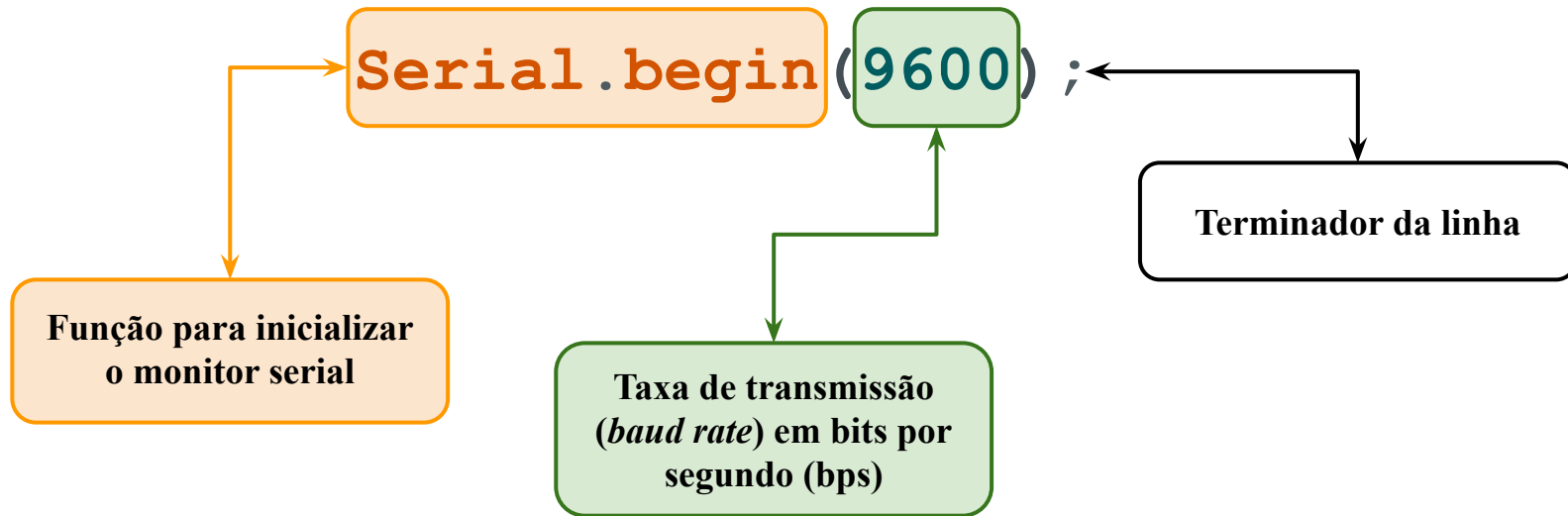
Comunicação serial



Mini curso Arduino Estação Meteorológica - Dia 2

1. Utilizando o monitor serial do Arduino

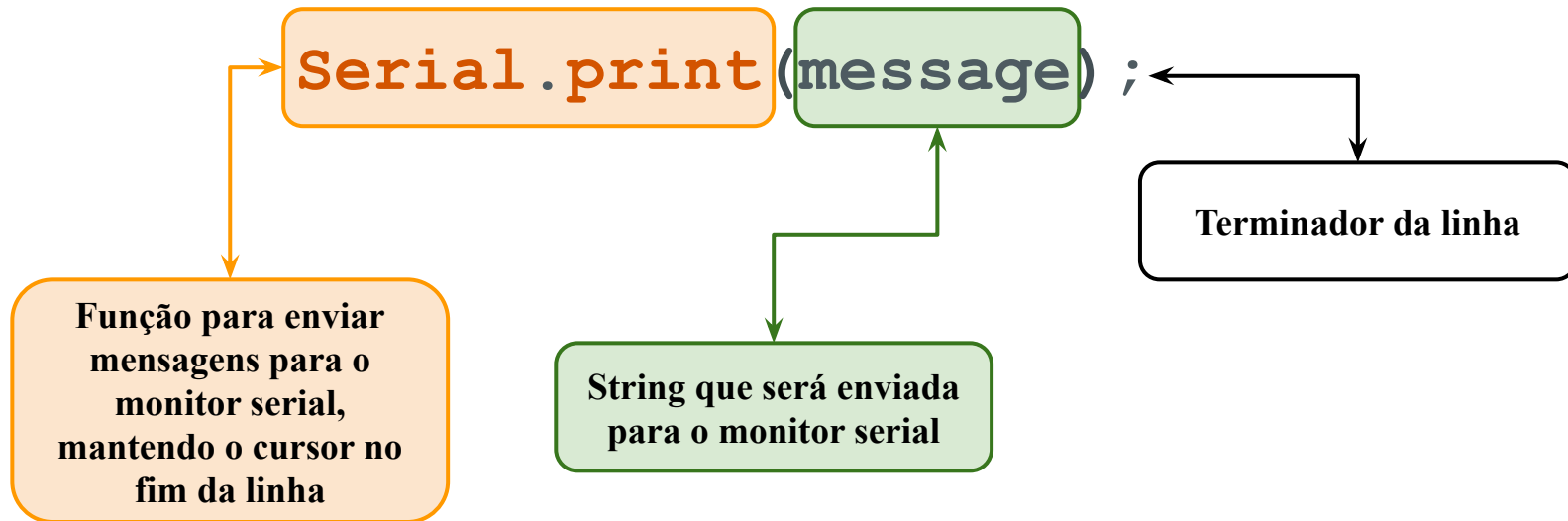
Dentro da estrutura de configuração:



Mini curso Arduino Estação Meteorológica - Dia 2

1. Utilizando o monitor serial do Arduino

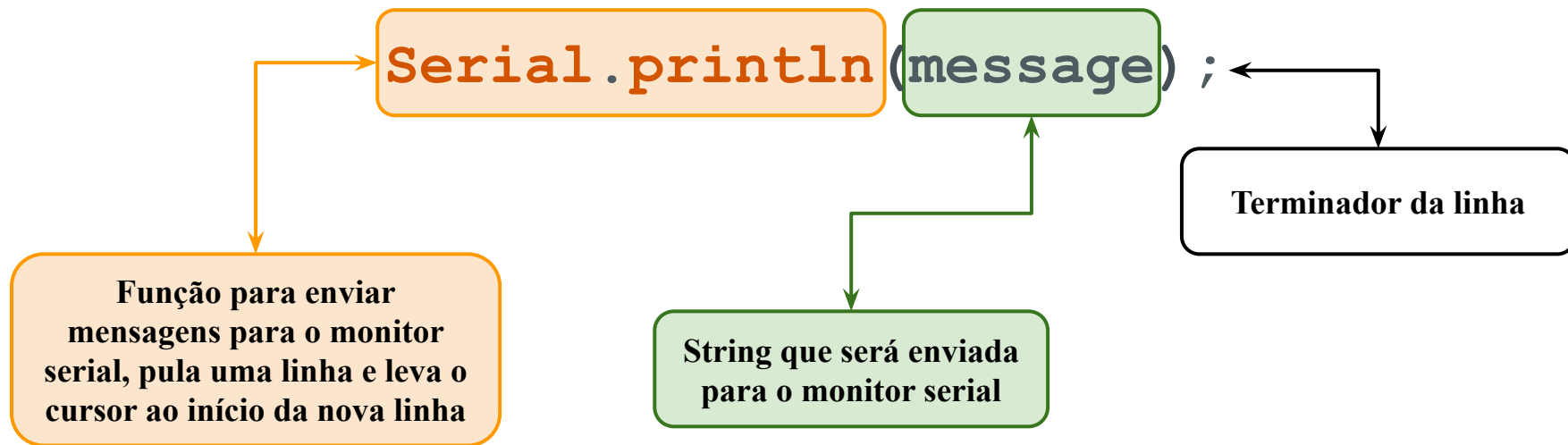
Dentro do programa principal:



Mini curso Arduino Estação Meteorológica - Dia 2

1. Utilizando o monitor serial do Arduino

Dentro do programa principal:



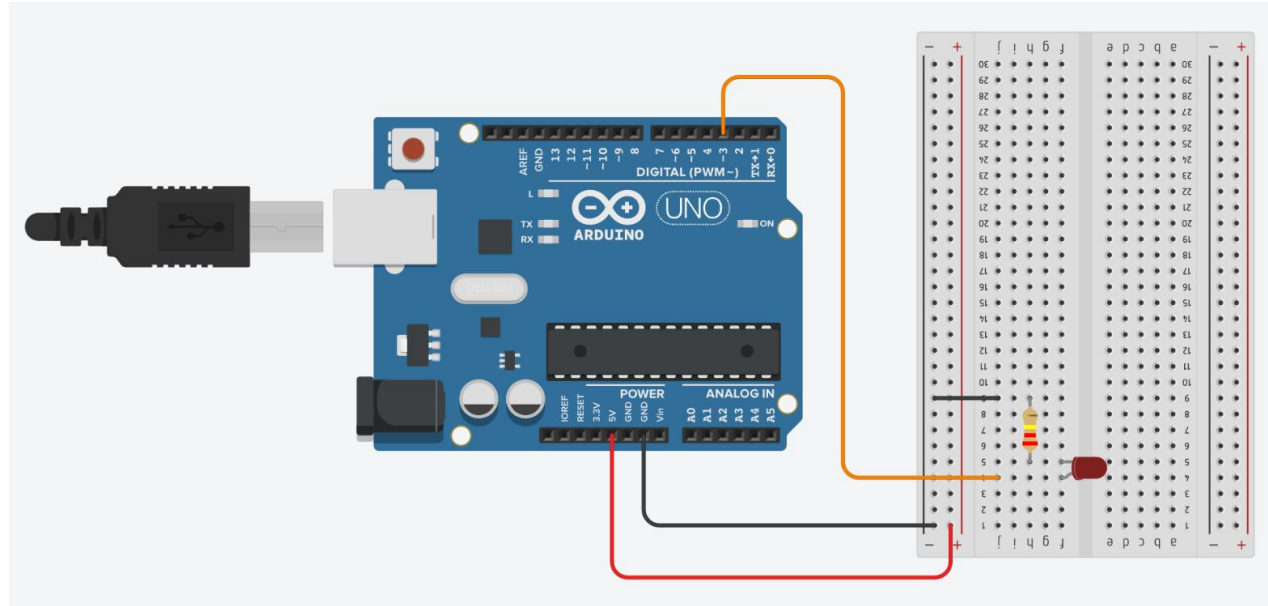
Mini curso Arduino Estação Meteorológica - Dia 2

2. Programa 1: Programa Oi Mundo no monitor serial

Materiais:

- Arduino Uno

Ligação dos componentes:



Mini curso Arduino Estação Meteorológica - Dia 2

2. Programa 1: Programa Oi Mundo no monitor serial

Código fonte:

```
// Declarando terminais, constantes e variáveis
const int portaSerial = 9600; // Inicia dados para 9600 bps

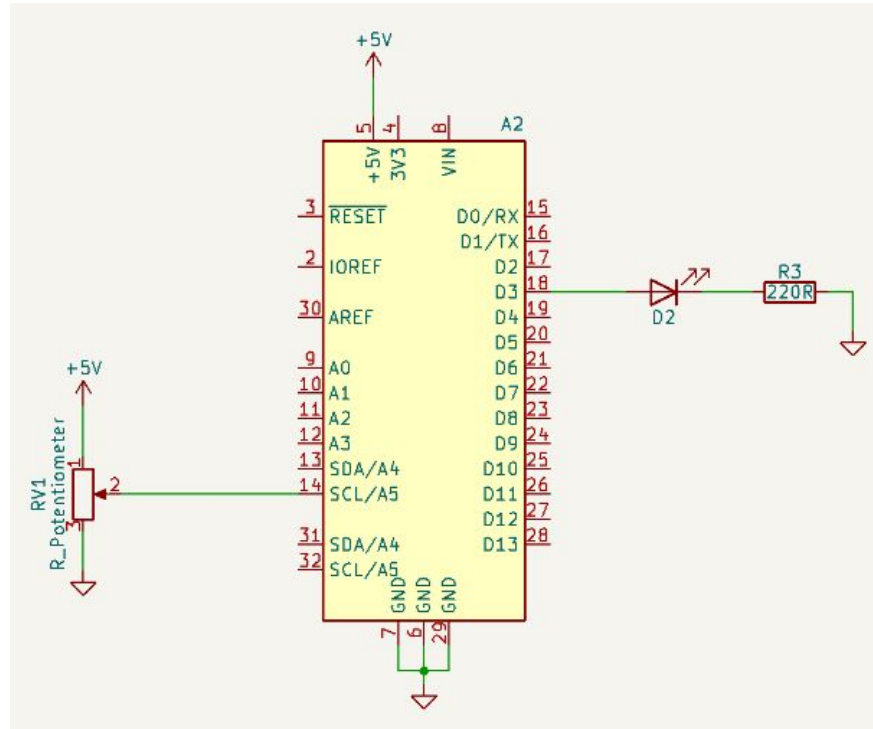
// Configurações
void setup() {
    Serial.begin(portaSerial); // Configura a taxa de transferência
    para transmissão serial.
}

// Código principal
void loop() {
    Serial.println("Oi Mundo"); // Escreve no monitor serial
}
```


Mini curso Arduino Estação Meteorológica - Dia 2

3. Programa 2: Programa Pisca-LED + Potenciômetro no monitor serial

Diagrama esquemático:



Mini curso Arduino Estação Meteorológica - Dia 2

3. Programa 2: Programa Pisca-LED + Potenciômetro no monitor serial

Código fonte:

```
// Declarando terminais, constantes e variáveis
int valor_pot = 0;
const int led = 3; // LED ligado ao pino D8
const int potenciometro = A5; // Potenciômetro
ligado ao pino analógico A5
const int portaSerial = 9600; // Inicia dados para
9600 bps
```

```
// Configurações
void setup(){
    pinMode(led, OUTPUT);
    Serial.begin(9600); // Configura a taxa de
transferência para transmissão serial.
}
```

```
// Código principal
void loop(){
    // Leitura do valor do potenciômetro
    valor_pot = analogRead(potenciometro);

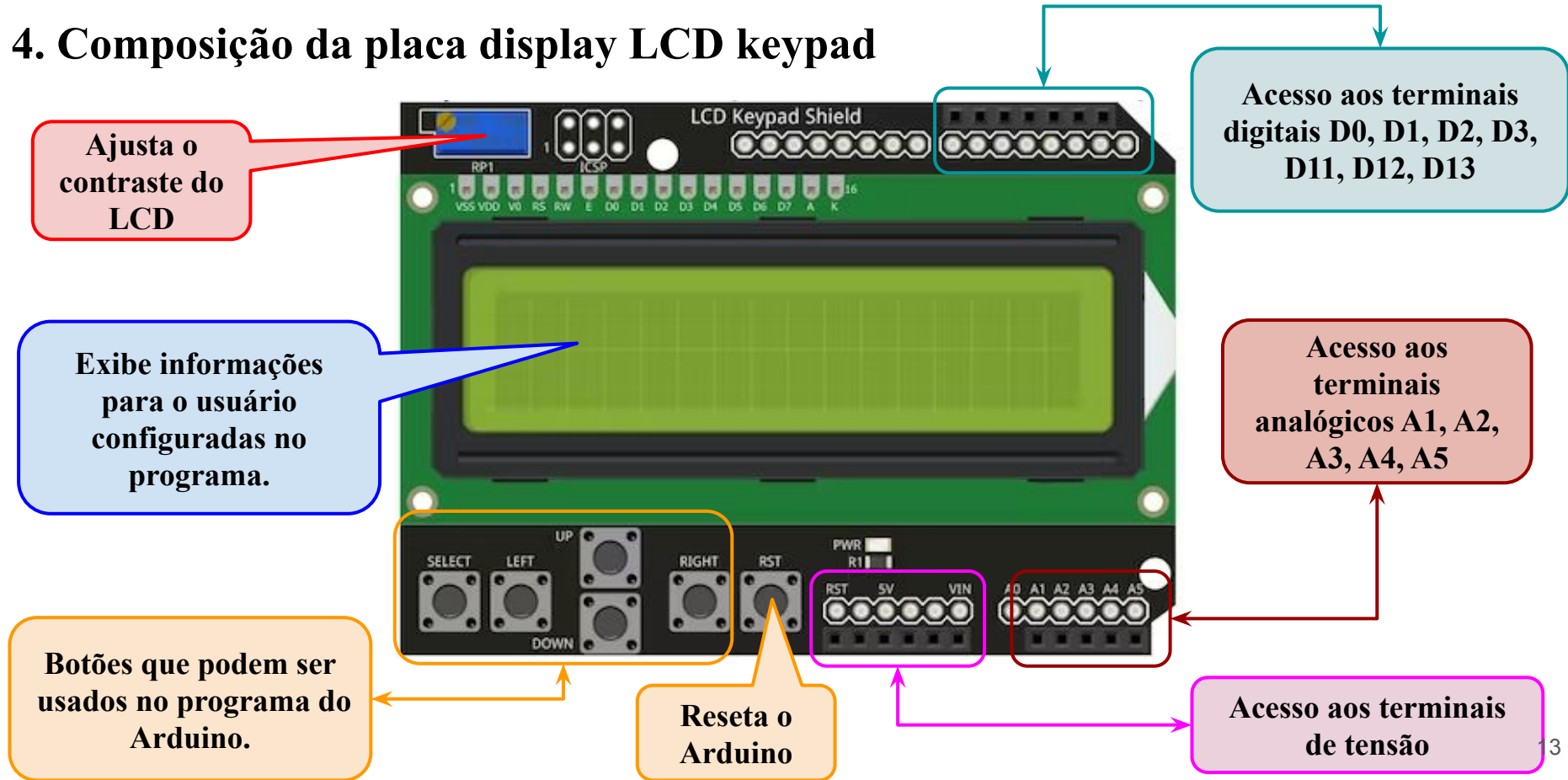
    digitalWrite(led, HIGH);
    delay(valor_pot);

    digitalWrite(led, LOW);
    delay(valor_pot);

    String message = (String)"Valor do potenciômetro =
" + valor_pot + " \n";
    Serial.print(message); // Escreve no monitor serial
}
```

Mini curso Arduino Estação Meteorológica - Dia 2

4. Composição da placa display LCD keypad



Mini curso Arduino Estação Meteorológica - Dia 2

4. Composição da placa display LCD keypad

Os botões são acessados pela porta analógica A0.

Botão	Valor A0
SELECT	0 - 60
LEFT	60 - 200
UP	200 - 400
DOWN	400 - 600
RIGHT	600 - 800



O Arduino controla o LCD utilizando os terminais:

Arduino	LCD
D8	RS
D9	Enable
D7	D7
D6	D6
D5	D5
D4	D4
D10	Backlight

Mini curso Arduino Estação Meteorológica - Dia 2

5. Funcionamento do display LCD

Na parte de declaração de constantes do programa:

Biblioteca necessária para utilização das funções do LCD

```
#include <LiquidCrystal.h>
```

```
const int backLight = 10;  
const int rs = 8, en = 9, d4 = 4, d5 = 5, d6 = 6, d7 = 7;
```

Nomeando os terminais do LCD

O backlight é um controle digital.

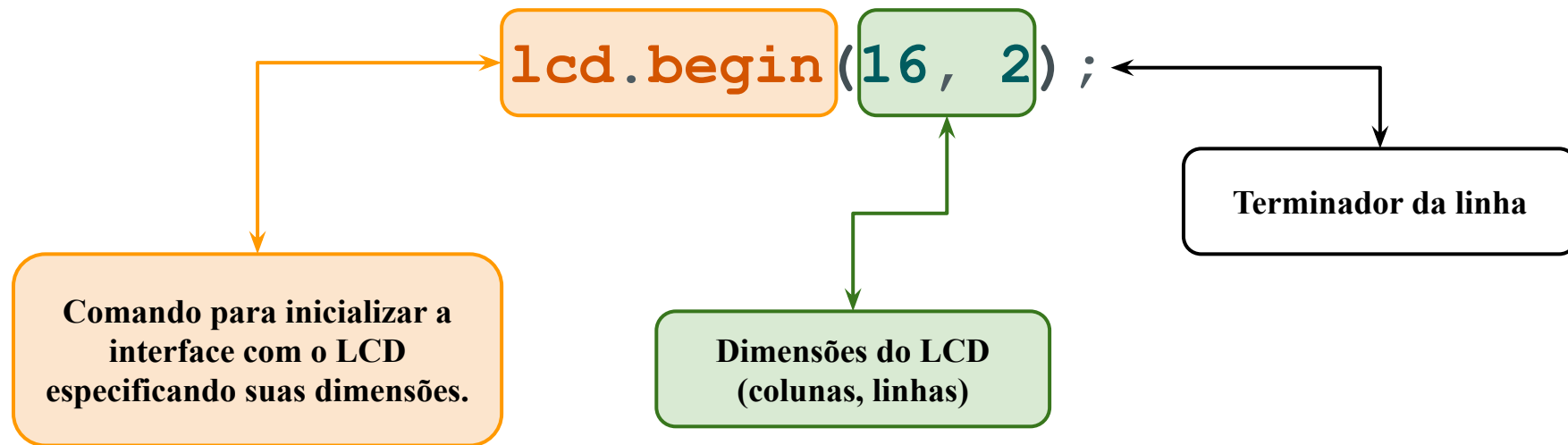
```
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);
```

Comando para criar uma instância do display LCD que se comunicará com o Arduino. Os argumentos obrigatórios são:
rs, en, d4, d5, d6 e d7.

Mini curso Arduino Estação Meteorológica - Dia 2

5. Funcionamento do display LCD

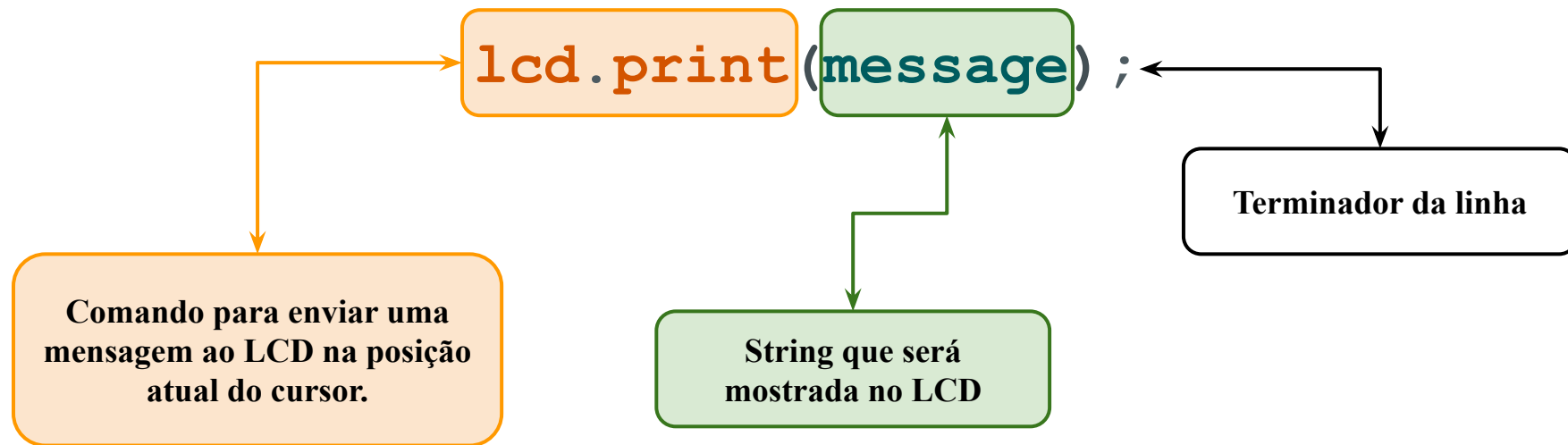
Dentro da estrutura de configuração:



Mini curso Arduino Estação Meteorológica - Dia 2

5. Funcionamento do display LCD

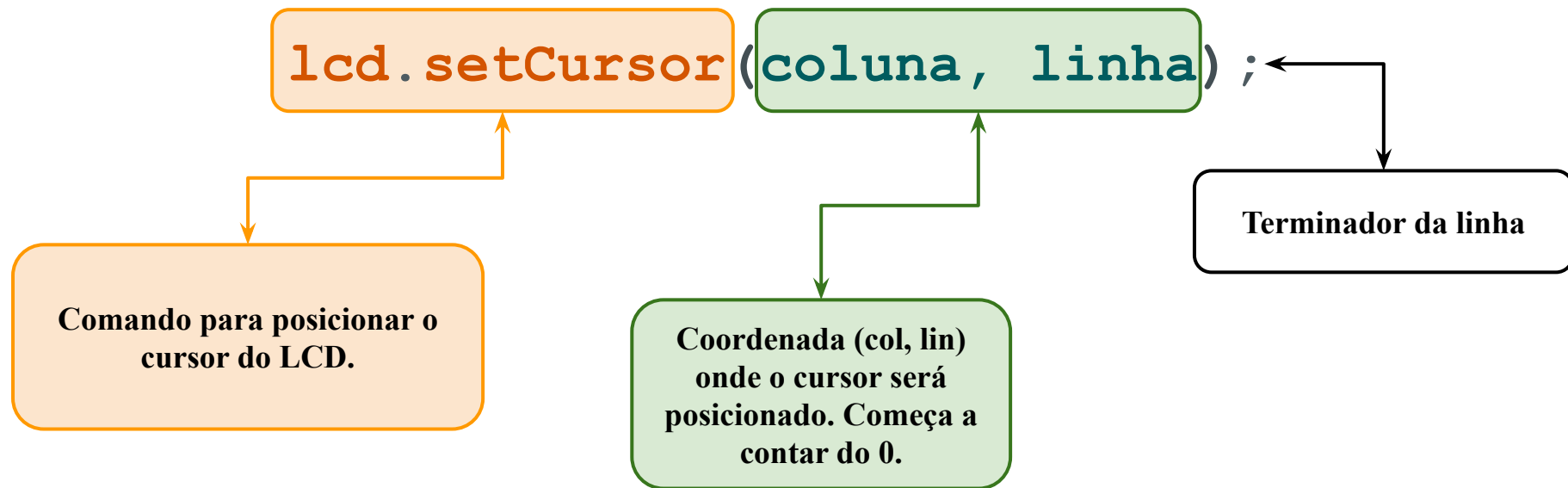
Dentro do programa principal:



Mini curso Arduino Estação Meteorológica - Dia 2

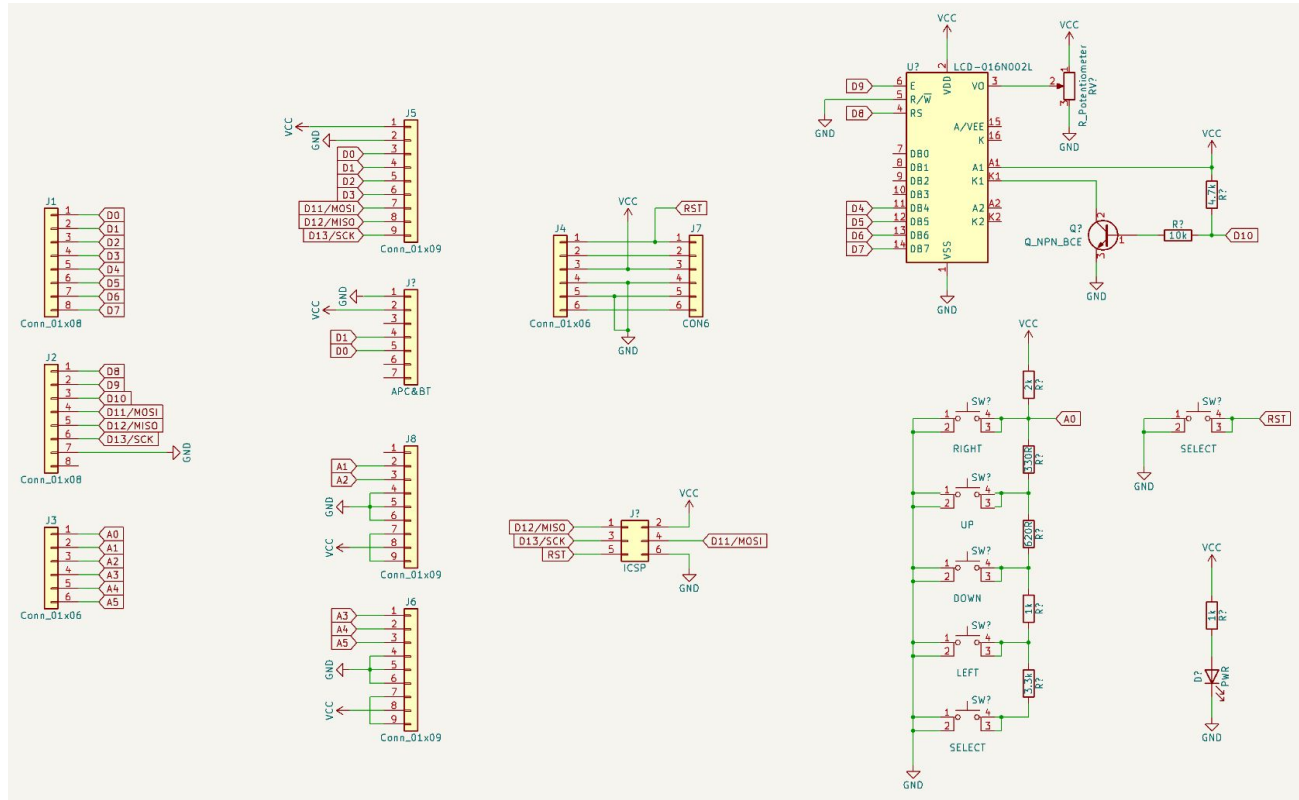
5. Funcionamento do display LCD

Dentro do programa principal:



Mini curso Arduino Estação Meteorológica - Dia 2

6. Diagrama esquemático do shield display LCD keypad



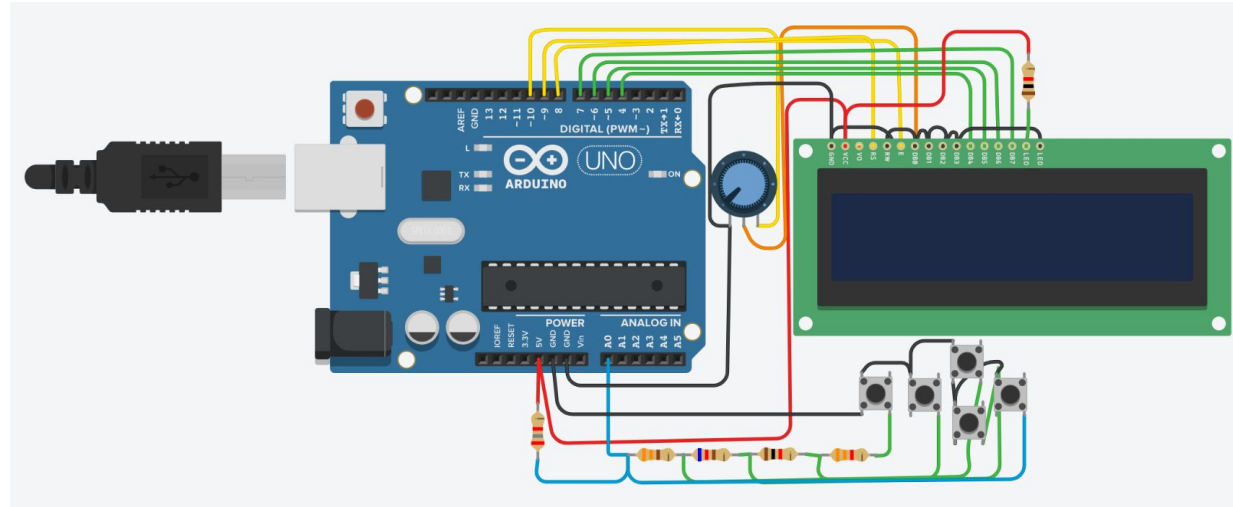
Mini curso Arduino Estação Meteorológica - Dia 2

7. Programa 3: Programa Oi Mundo no display LCD

Materiais:

- Arduino Uno
- LCD keypad shield

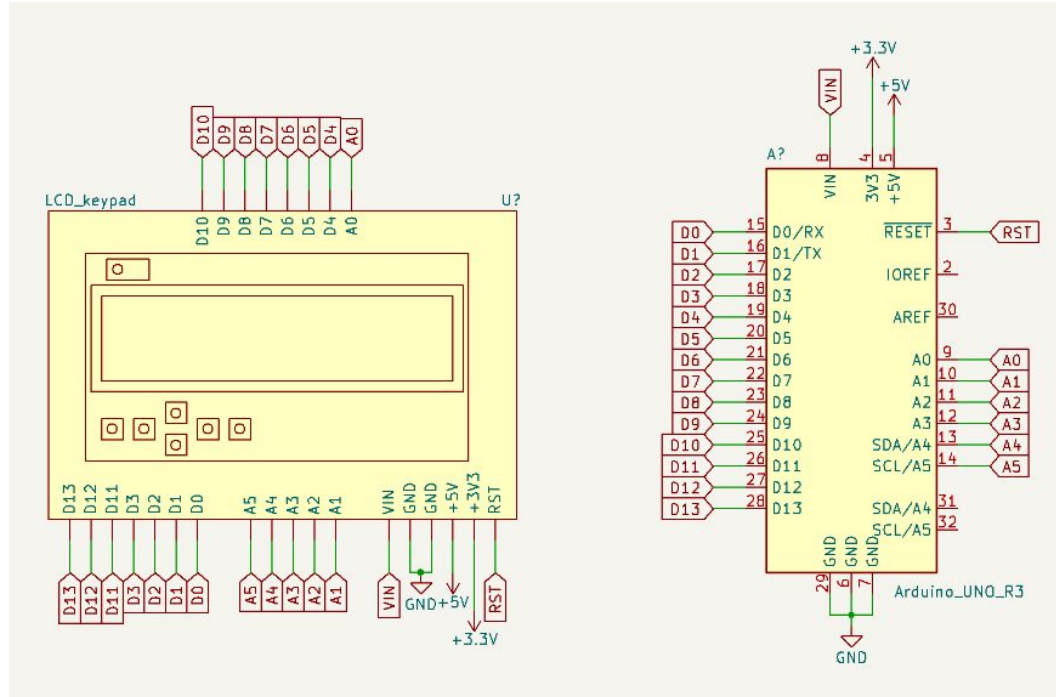
Ligação dos componentes:



Mini curso Arduino Estação Meteorológica - Dia 2

7. Programa 3: Programa Oi Mundo no display LCD

Diagrama esquemático:



Mini curso Arduino Estação Meteorológica - Dia 2

7. Programa 3: Programa Oi Mundo no display LCD

Código fonte:

```
#include <LiquidCrystal.h>

const int backLight = 10;
const int rs = 8, en = 9, d4 = 4, d5 = 5, d6 = 6, d7 = 7;
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);

// Configurações
void setup() {
    // Inicia o display de dimensão 16X2
    lcd.begin(16, 2);
}

// Código principal
void loop() {
    lcd.setCursor(2,0);
    lcd.print("Oi,");

    lcd.setCursor(5,1);
    lcd.print("Mundo :)");
}
```

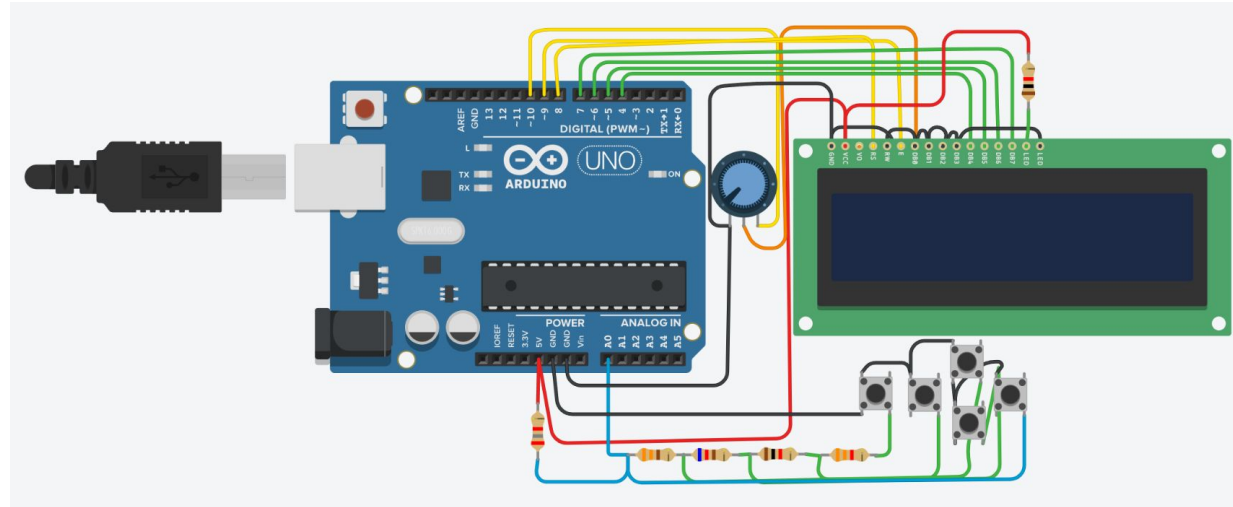
Mini curso Arduino Estação Meteorológica - Dia 2

8. Programa 4: Programa display LCD com botões

Materiais:

- Arduino Uno
- LCD keypad shield

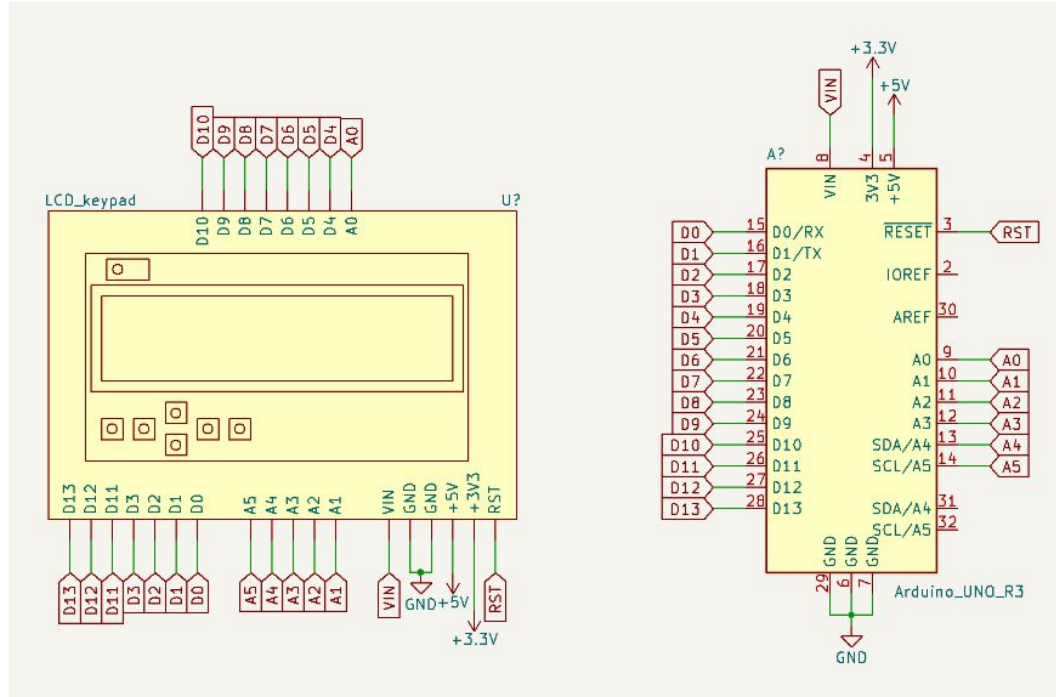
Ligação dos componentes:



Mini curso Arduino Estação Meteorológica - Dia 2

8. Programa 4: Programa display LCD com botões

Diagrama esquemático:



Mini curso Arduino Estação Meteorológica - Dia 2

8. Programa 4: Programa display LCD com botões

Código fonte:

```
// Bibliotecas
#include <LiquidCrystal.h>

// Define os terminais e cria o objeto para acesso
const int rs = 8, en = 9, d4 = 4, d5 = 5, d6 = 6, d7 = 7;
const int backLight = 10;
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);

// Limites para detecção das teclas, em ordem crescente
struct {
    int limite;
    char *nome;
} teclas[] = {
    { 50, "Direita " },
    { 150, "Cima    " },
    { 300, "Baixo   " },
    { 500, "Esquerda" },
    { 750, "Select  " },
    { 1024, "        " } // nenhuma tecla apertada
};
```

```
// Configurações
void setup() {
    // Programa o pino de backlight como saída
    pinMode(backLight, OUTPUT);
    // Inicia o display e coloca uma mensagem
    lcd.begin(16, 2);
    lcd.print("Arduino CEC 2022");
    // Acende o backlight
    digitalWrite(backLight, HIGH);
}
```

Mini curso Arduino Estação Meteorológica - Dia 2

8. Programa 4: Programa display LCD com botões

Código fonte:

```
// Código principal
void loop() {
    static int teclaAnt = -1;    // última tecla detectada
    // Lê a tensão no pino A0
    int leitura = analogRead(A0);
    // Identifica a tecla apertada pela tensão lida
    int teclaNova;
    for (teclaNova = 0; ; teclaNova++) {
        if (leitura < teclas[teclaNova].limite) {
            break;
        }
    }
    // Atualiza a tela se pressionou uma nova tecla
    if (teclaNova != teclaAnt) {
        lcd.setCursor(0,1);
        lcd.print(teclas[teclaNova].nome);
        teclaAnt = teclaNova;
    }
    // Espera até a próxima leitura
    delay (100);
}
```

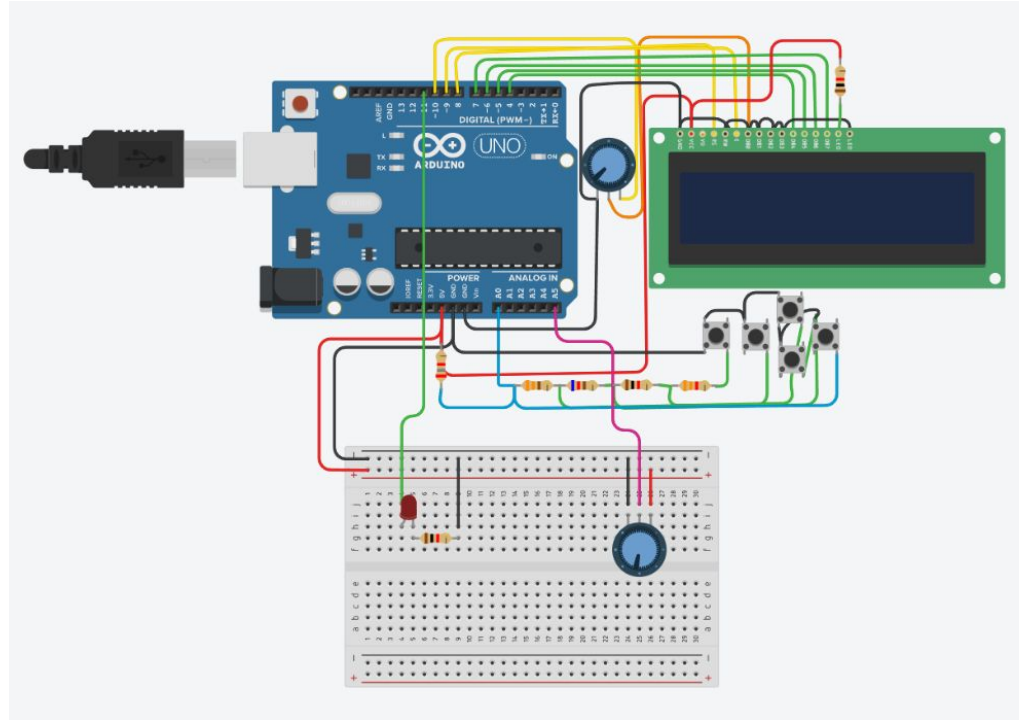
Mini curso Arduino Estação Meteorológica - Dia 2

9. Programa 5: Programa Pisca-LED + Potenciômetro + display LCD

Materiais:

- Arduino Uno
- LCD keypad shield
- Protoboard
- 1x Resistor 220 Ω
- 1x Potenciômetro 10k Ω
- 1x LED

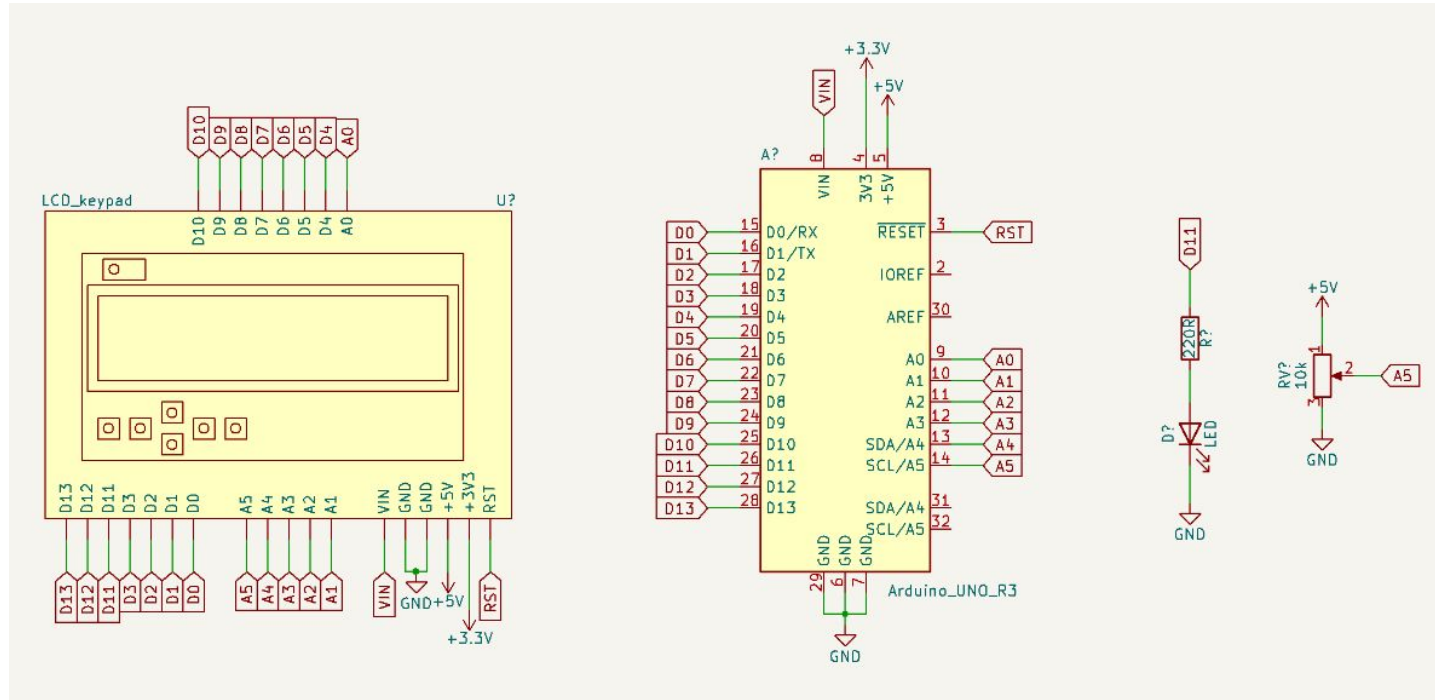
Ligação dos componentes:



Mini curso Arduino Estação Meteorológica - Dia 2

9. Programa 5: Programa Pisca-LED + Potenciômetro + display LCD

Diagrama esquemático:



Mini curso Arduino Estação Meteorológica - Dia 2

9. Programa 5: Programa Pisca-LED + Potenciômetro + display LCD

Código fonte:

```
// Bibliotecas
#include <LiquidCrystal.h>

// Define os terminais e cria o objeto para acesso
const int rs = 8, en = 9, d4 = 4, d5 = 5, d6 = 6, d7 = 7;
const int backLight = 10;
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);

// Limites para detecção das teclas, em ordem crescente
struct {
    int limite;
    char *nome;
} teclas[] = {
    { 50, "Direita " },
    { 150, "Cima    " },
    { 300, "Baixo   " },
    { 500, "Esquerda" },
    { 750, "Select  " },
    { 1024, "        " } // nenhuma tecla apertada
};
```

```
// Configurações
void setup() {
    // Programa o pino de backlight como saída
    pinMode(backLight, OUTPUT);
    // Inicia o display e coloca uma mensagem
    lcd.begin(16, 2);
    lcd.print("Arduino CEC 2022");
    // Acende o backlight
    digitalWrite(backLight, HIGH);
}
```