



Tópicos Especiais em Sistemas de Informação

Prof. Dr. André Luiz Nasserla Pires

Docentes - Grupo: Discípulos do Jean

Andre Ferreira Santana

Dayan Freitas Alves

Mateus de Souza Lopes

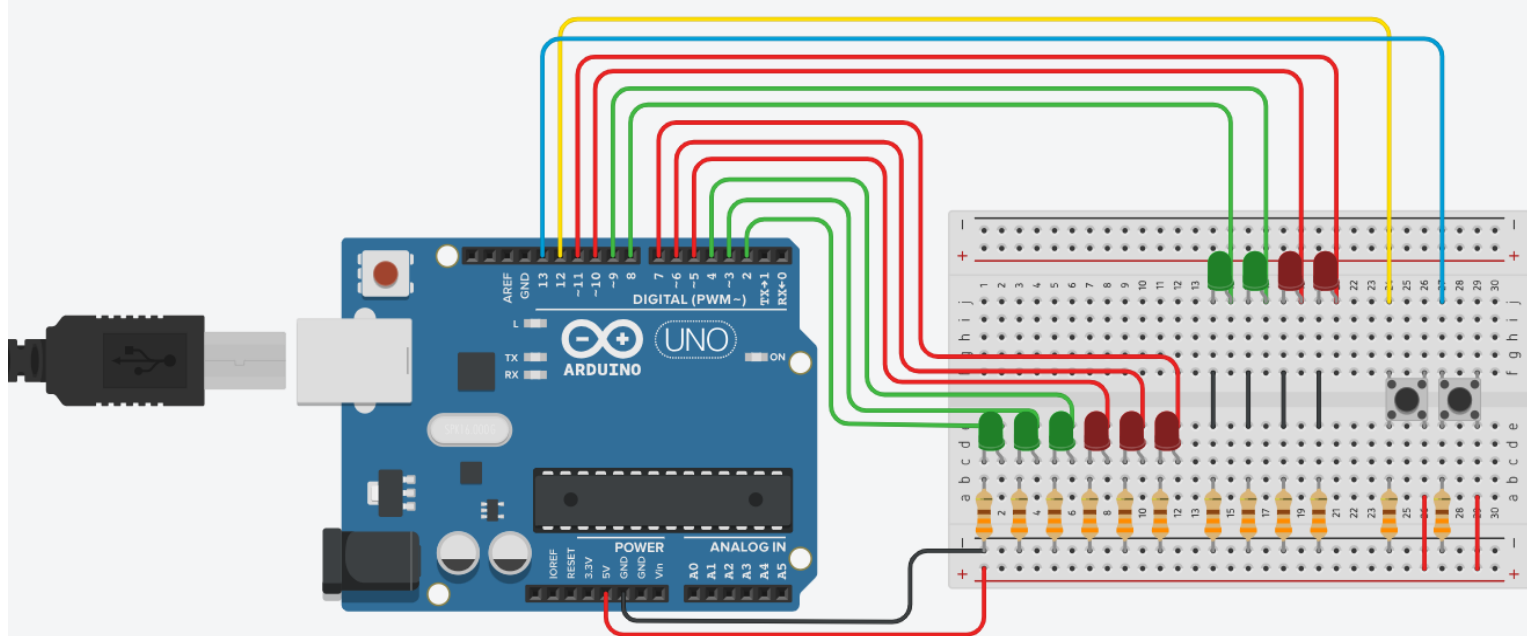
Brendo Lyu Malveira Benício de Melo

Rafael Alves Braga

Problema 4 – Calculadora Binária

Fazer um circuito em Arduino que, dada duas entrada binária de 3 dígitos (6 leds, 3 para cada entrada), efetue 2 operações: adição e subtração. As operações devem ser escolhidas através de 2 botões, um para cada, e o resultado exibido em 4 leds, onde led aceso representa 1 e apagado 0.

Resposta



Código

```
// LEDs para entrada
const int ledsNum1[3] = {2, 3, 4};
const int ledsNum2[3] = {5, 6, 7};

// LEDs para o resultado
const int ledsResultado[4] = {8, 9, 10, 11};

const int botao1 = 12;
const int botao2 = 13;

// Estados
enum Estado { ENTRADA1, OPERACAO, ENTRADA2, RESULTADO };
Estado estadoAtual = ENTRADA1;

int num1 = 0, num2 = 0, resultado = 0;
bool operacaoSoma = true;

// Tempo para piscar LEDs
unsigned long intervaloPiscar = 500, ultimoPiscar = 0;

// Controle de botões
bool botao1Pressionado = false, botao2Pressionado = false;

void setup() {
  for (int i = 0; i < 3; i++) {
    pinMode(ledsNum1[i], OUTPUT);
    pinMode(ledsNum2[i], OUTPUT);
  }
  for (int i = 0; i < 4; i++) {
    pinMode(ledsResultado[i], OUTPUT);
  }

  // Configuração de botões como ENTRADA (necessário resistor pull-down externo de 10kΩ)
  pinMode(botao1, INPUT);
  pinMode(botao2, INPUT);

  Serial.begin(9600);
}

void loop() {
  bool estadoBotao1 = digitalRead(botao1);
  bool estadoBotao2 = digitalRead(botao2);

  if (estadoBotao1 == HIGH && !botao1Pressionado) {
    delay(50);
    if (digitalRead(botao1) == HIGH) {
      botao1Pressionado = true;
      tratarBotao1();
    }
  }
}
```

```

    }
    } else if (estadoBotao1 == LOW) {
        botao1Pressionado = false;
    }

    if (estadoBotao2 == HIGH && !botao2Pressionado) {
        delay(50);
        if (digitalRead(botao2) == HIGH) {
            botao2Pressionado = true;
            tratarBotao2();
        }
    } else if (estadoBotao2 == LOW) {
        botao2Pressionado = false;
    }
}

piscarLEDs();
}

void tratarBotao1() {
    switch (estadoAtual) {
        case ENTRADA1:
            num1 = (num1 + 1) % 8;
            atualizarLEDs(num1, ledsNum1);
            Serial.print("Número 1: ");
            Serial.println(num1);
            break;

        case OPERACAO:
            operacaoSoma = false;
            estadoAtual = ENTRADA2;
            Serial.println("Operação: Subtração");
            break;

        case ENTRADA2:
            num2 = (num2 + 1) % 8;
            atualizarLEDs(num2, ledsNum2);
            Serial.print("Número 2: ");
            Serial.println(num2);
            break;

        case RESULTADO:
            resetarCalculadora();
            break;
    }
}

void tratarBotao2() {
    switch (estadoAtual) {
        case ENTRADA1:
            estadoAtual = OPERACAO;
            Serial.println("Número 1 confirmado");
            break;

        case OPERACAO:

```

```

        operacaoSoma = true;
        estadoAtual = ENTRADA2;
        Serial.println("Operação: Soma");
        break;

    case ENTRADA2:
        calcularResultado();
        estadoAtual = RESULTADO;
        Serial.println("Número 2 confirmado");
        break;

    case RESULTADO:
        resetarCalculadora();
        break;
}
}

void piscarLEDs() {
    if (millis() - ultimoPiscar >= intervaloPiscar) {
        ultimoPiscar = millis();
        switch (estadoAtual) {
            case ENTRADA1:
                digitalWrite(ledsResultado[0], !digitalRead(ledsResultado[0]));
                break;
            case OPERACAO:
                digitalWrite(ledsResultado[1], !digitalRead(ledsResultado[1]));
                digitalWrite(ledsResultado[2], !digitalRead(ledsResultado[2]));
                break;
            case ENTRADA2:
                digitalWrite(ledsResultado[3], !digitalRead(ledsResultado[3]));
                break;
            case RESULTADO:
                break;
        }
    }
}

void atualizarLEDs(int numero, const int leds[3]) {
    for (int i = 0; i < 3; i++) {
        digitalWrite(leds[2 - i], (numero >> i) & 0x01);
    }
}

void exibirResultado(int res) {
    for (int i = 0; i < 4; i++) {
        digitalWrite(ledsResultado[3 - i], (res >> i) & 0x01);
    }
}

void calcularResultado() {
    resultado = operacaoSoma ? (num1 + num2) : (num1 - num2);
    Serial.print("Resultado real: ");
    Serial.println(resultado);
    resultado = abs(resultado);
}

```

```
        resultado = constrain(resultado, 0, 15);
        exibirResultado(resultado);
        Serial.print("Resultado exibido: ");
        Serial.println(resultado);
    }

    void resetarCalculadora() {
        num1 = 0;
        num2 = 0;
        resultado = 0;
        estadoAtual = ENTRADA1;

        // Desliga todos os LEDs
        for (int i = 0; i < 3; i++) {
            digitalWrite(ledsNum1[i], LOW);
            digitalWrite(ledsNum2[i], LOW);
        }
        for (int i = 0; i < 4; i++) {
            digitalWrite(ledsResultado[i], LOW);
        }

        Serial.println("Calculadora reiniciada");
    }
```