A close-up photograph of an Arduino Uno R3 microcontroller board. The board is blue with various components visible, including the ATmega328P microcontroller, a USB Type-C to UART adapter connected to the USB port, and several push buttons. The text "ARDUINO UNO" and "MADE IN ITALY" are printed on the board. A black plastic component, likely a breadboard or a connector, is attached to the bottom of the board.

# Arduino – aula 03

# Tópicos a serem abordados

- ◇ Criando variáveis no início do sketch
- ◇ `digitalWrite(pino, 0-255);` em portas digitais
- ◇ `analogWrite(pino, 0-255);` em portas analógicas
- ◇ Semáforo
- ◇ Fading
- ◇ Botão
- ◇ Efeito Bounce

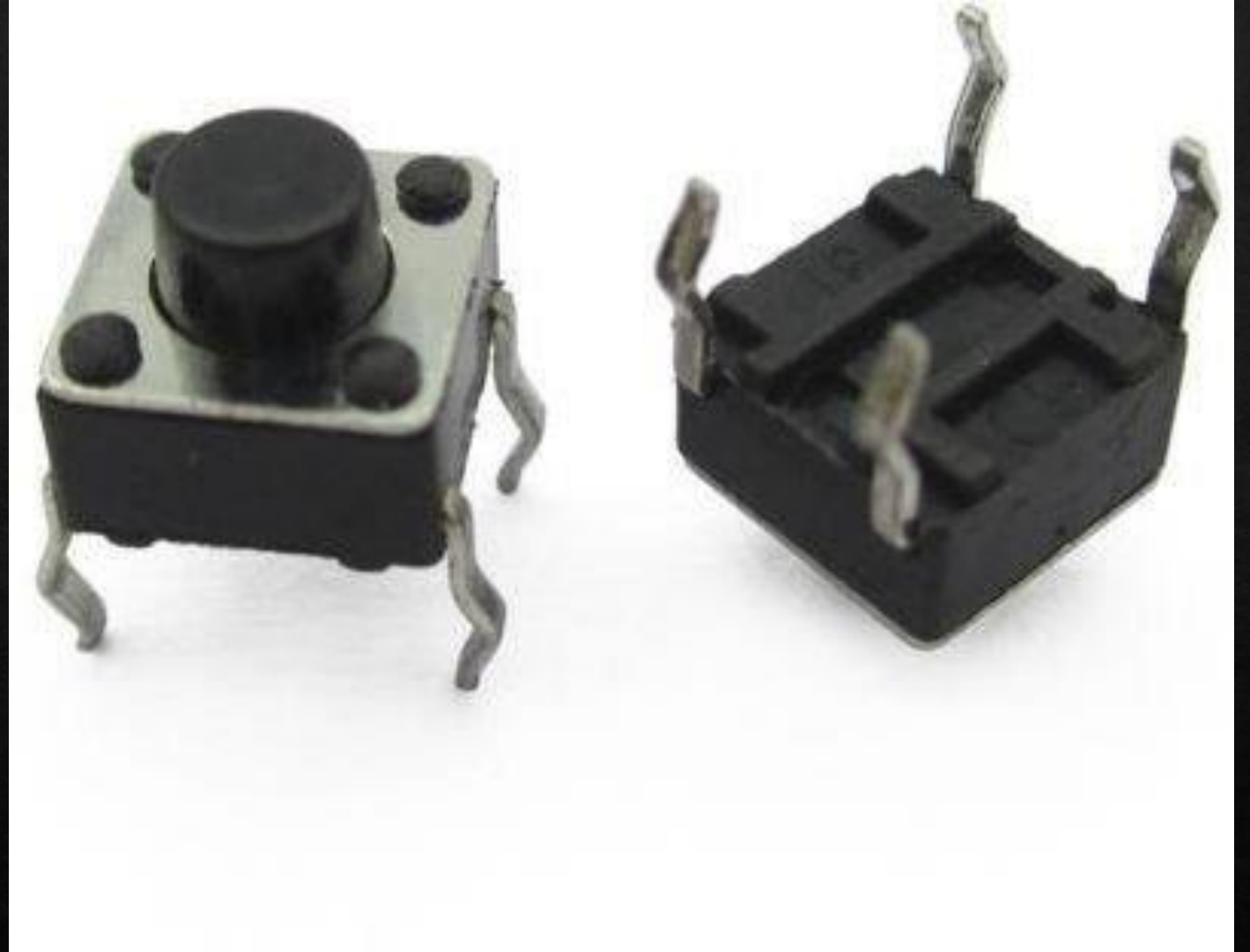
# Comandos Digitais e Analógicos

- ◇ porta digital >>> digitalWrite(); e digitalWrite();
- ◇ porta digital PWM >>> digitalWrite(); digitalWrite(); e analogWrite();
- ◇ porta analógica >>> digitalWrite(); digitalWrite(); analogWrite(); e analogRead();

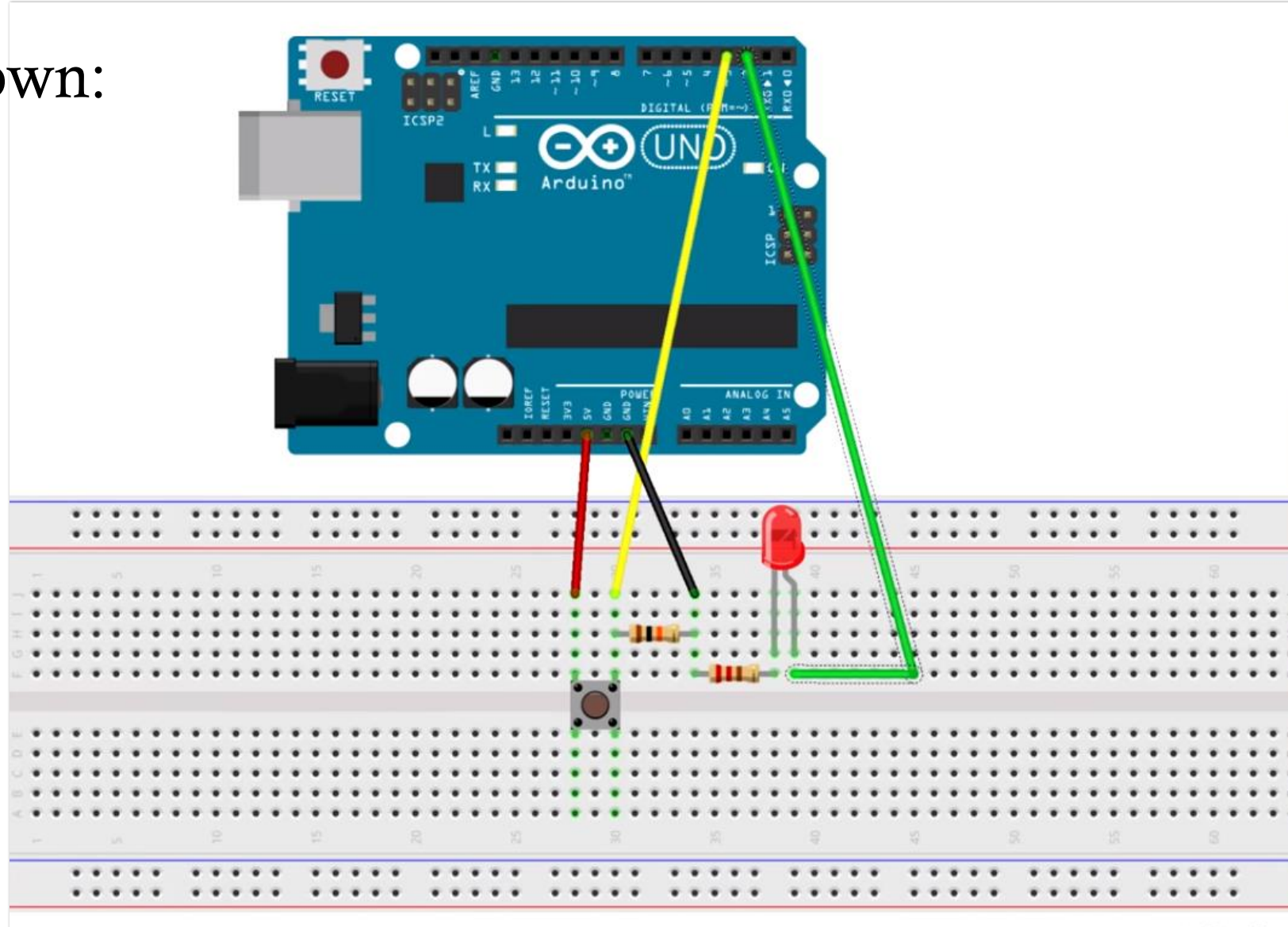
Mais informações sobre  
variáveis de programação  
se encontra na página 50  
tópico 4.5



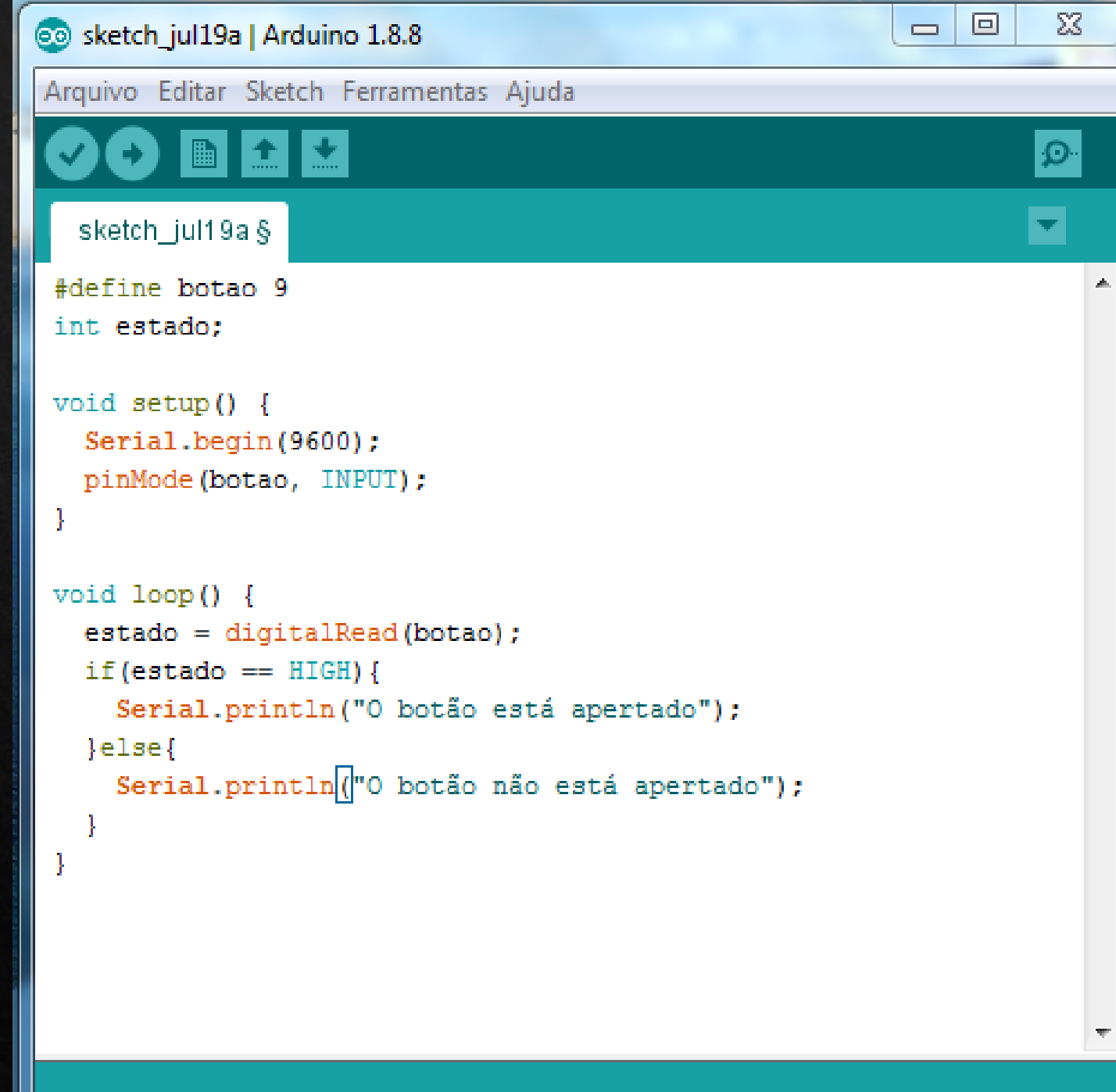
# Push Button



Pull down:



Sketch:

A screenshot of the Arduino IDE interface. The title bar at the top reads "sketch\_jul19a | Arduino 1.8.8". Below the title bar is a menu bar with "Arquivo", "Editar", "Sketch", "Ferramentas", and "Ajuda". Underneath the menu bar is a toolbar with icons for checking, running, saving, and other functions. A tab labeled "sketch\_jul19a" is open. The main text area contains the following C++ code:

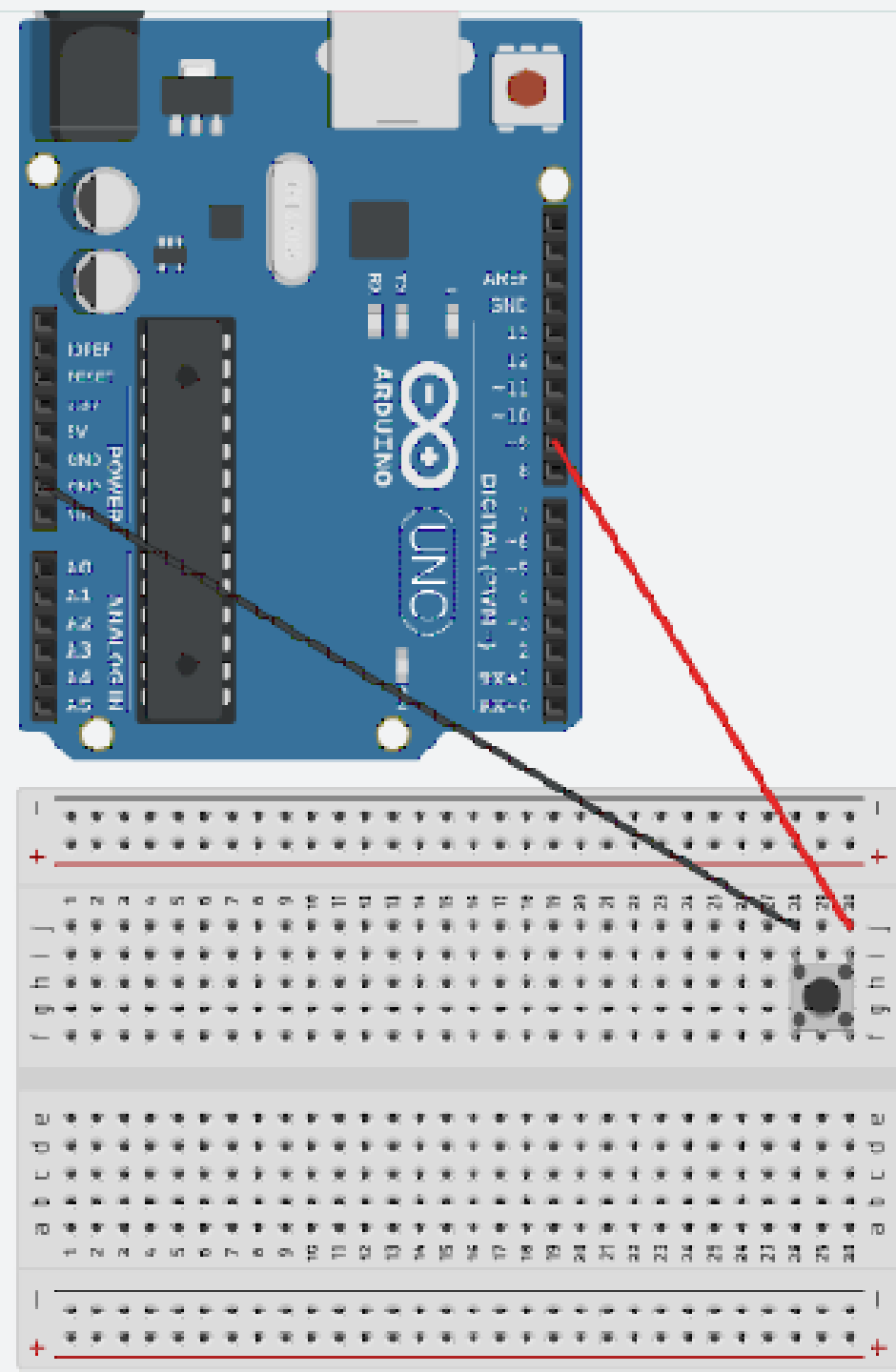
```
#define botao 9
int estado;

void setup() {
  Serial.begin(9600);
  pinMode(botao, INPUT);
}

void loop() {
  estado = digitalRead(botao);
  if(estado == HIGH){
    Serial.println("O botão está apertado");
  }else{
    Serial.println("O botão não está apertado");
  }
}
```

# Comando INPUT\_PULLUP

Se usarmos INPUT\_PULLUP, não é necessário resistor. O circuito ficaria como o da figura ao lado.





sketch\_oct22a\$

```
#define botao 9
int estado;

void setup() {
  Serial.begin(9600);
  pinMode(botao, INPUT_PULLUP);
}

void loop() {
  estado = digitalRead(botao);
  if(estado == LOW){
    Serial.println("O botão está apertado");
  }else{
    Serial.println("O botão não está apertado");
  }
}
```



# Resumo

- ◆ `Serial.begin(9600);` >>> inicializa a comunicação com o monitor serial
- ◆ `if() {}` >>> condicional simples
- ◆ `Serial.print();` >>> mostra uma mensagem no monitor serial
- ◆ `Serial.println();` >>> mostra uma mensagem no monitor serial e pula linha
- ◆ `#define` >>> cria uma definição