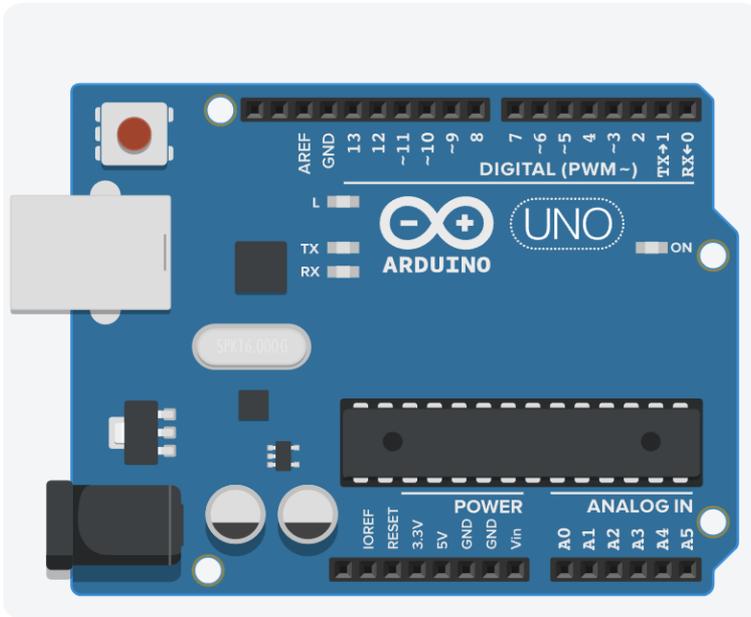


# Introdução ao Arduino

Prof. Dr. Sérgio Carlos Portari  
Júnior

# O que é o Arduino?



- Plataforma de prototipagem eletrônica baseada em hardware e software livres.
- Arduino Uno é a placa mais popular.
- Possui pinos digitais, analógicos, alimentação e comunicação USB.

# Estrutura básica de um programa

```
1 // C++ code
2 //
3 void setup()
4 {
5   pinMode(LED_BUILTIN, OUTPUT);
6 }
7
8 void loop()
9 {
10  digitalWrite(LED_BUILTIN, HIGH);
11  delay(1000); // Wait for 1000 millisecond(s)
12  digitalWrite(LED_BUILTIN, LOW);
13  delay(1000); // Wait for 1000 millisecond(s)
14 }
```

- Todo programa Arduino (sketch) tem duas funções principais:
- `setup()` → executado uma vez ao iniciar.
- `loop()` → executado continuamente (como um `while(1)` em C).

# Comandos principais

- Funções básicas do Arduino:
- `pinMode(pino, modo)` → configura pino como entrada ou saída.
- `digitalWrite(pino, valor)` → escreve HIGH (1) ou LOW (0).
- `digitalRead(pino)` → lê valor de entrada digital.
- `delay(ms)` → pausa em milissegundos.

```
1 // C++ code
2 //
3 void setup()
4 {
5   pinMode(LED_BUILTIN, OUTPUT);
6 }
7
8 void loop()
9 {
10  digitalWrite(LED_BUILTIN, HIGH);
11  delay(1000); // Wait for 1000 millisecond(s)
12  digitalWrite(LED_BUILTIN, LOW);
13  delay(1000); // Wait for 1000 millisecond(s)
14 }
```

# Exemplo prático – LED piscando

- Primeiro programa com Arduino:
- LED no pino 13 acendendo e apagando a cada 1 segundo.
- Demonstração do uso de `digitalWrite()` e `delay()`.

```
1 // C++ code
2 //
3 void setup()
4 {
5     pinMode(LED_BUILTIN, OUTPUT);
6 }
7
8 void loop()
9 {
10    digitalWrite(LED_BUILTIN, HIGH);
11    delay(1000); // Wait for 1000 millisecond(s)
12    digitalWrite(LED_BUILTIN, LOW);
13    delay(1000); // Wait for 1000 millisecond(s)
14 }
```

# Código exemplo – LED piscando

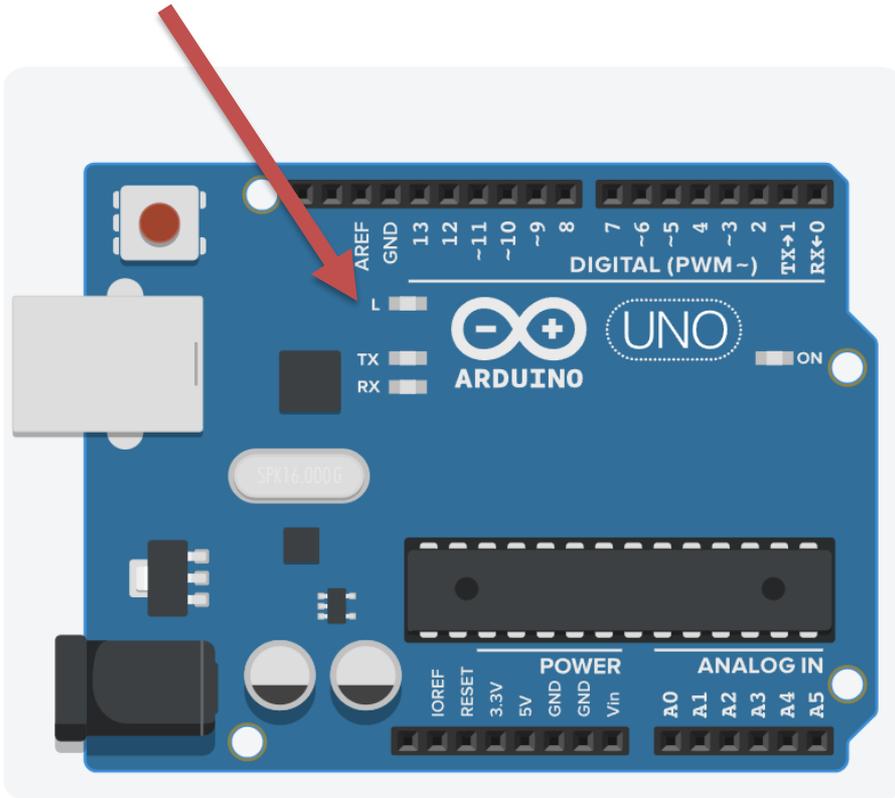
```
1 // C++ code
2 //
3 int ledPin = 13;
4
5 void setup() {
6   pinMode(ledPin, OUTPUT);
7 }
8
9 void loop() {
10  digitalWrite(ledPin, HIGH);
11  delay(1000);
12  digitalWrite(ledPin, LOW);
13  delay(1000);
14 }
```

```
int ledPin = 13;
```

```
void setup() {
  pinMode(ledPin, OUTPUT);
}
```

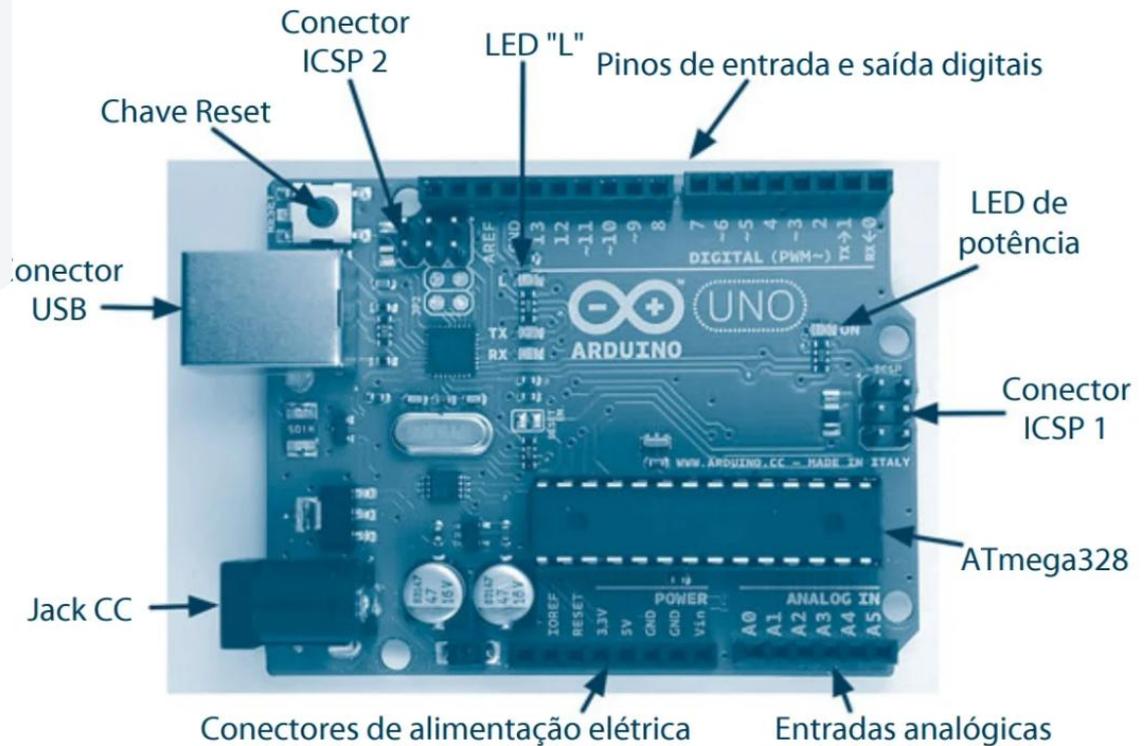
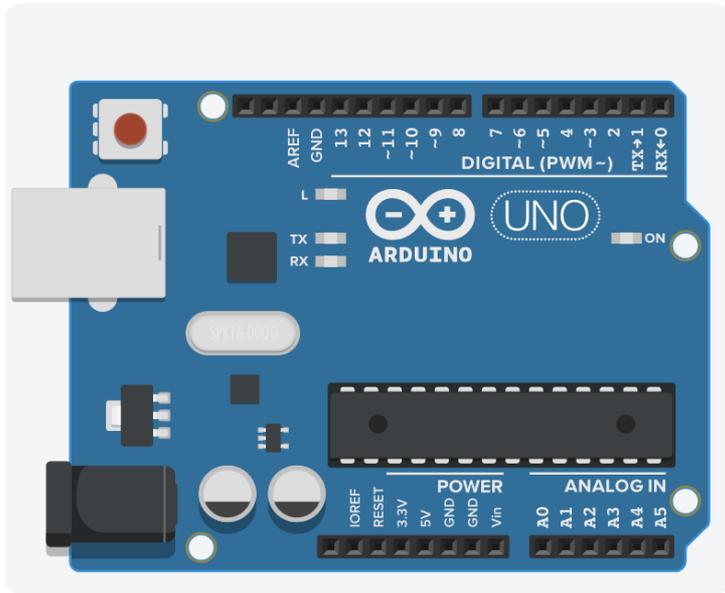
```
void loop() {
  digitalWrite(ledPin, HIGH);
  delay(1000);
  digitalWrite(ledPin, LOW);
  delay(1000);
}
```

# Curiosidade

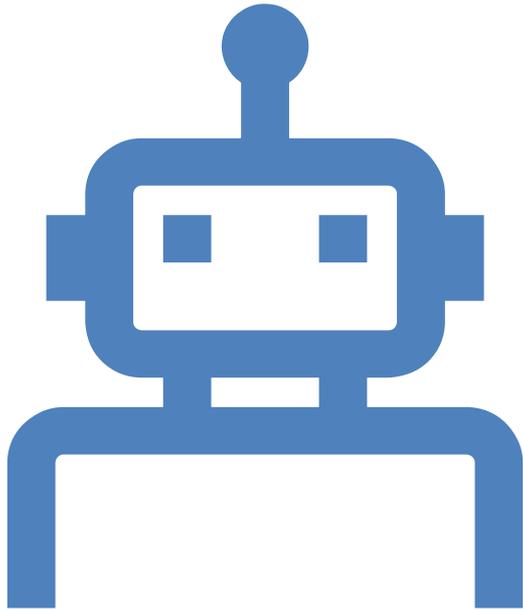


- O diodo emissor de luz (LED), assinalado com a letra L na placa, está ligado a um dos pinos de conexão de entrada e saída digitais da placa.
- Ele está conectado ao pino digital 13. Isso limita o uso do pino 13 a ser uma saída.
- No entanto, o LED consome apenas uma pequena quantidade de corrente, de modo que você ainda pode conectar outras coisas a esse pino de conexão.

# Curiosidade

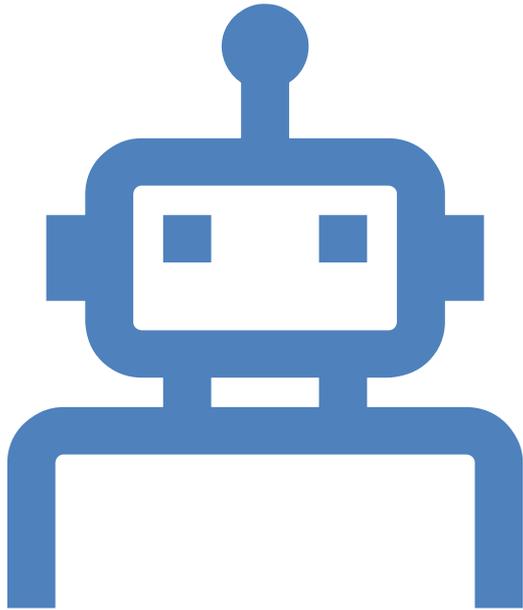


# Atividade prática



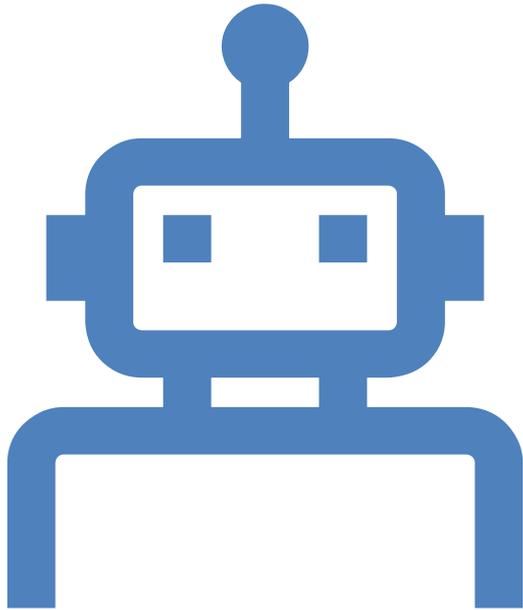
- Modifique o programa para:
- Fazer o LED piscar mais rápido (0,5s).
- Fazer o LED piscar mais devagar (2s).
- Desafio: adicionar um segundo LED e alternar o piscar.

# Atividade prática



- Criar um sinalizador de código morse SOS com LEDs piscantes.
  - Em código Morse,
    - S = ...
    - O = ---
  - Portanto, a sequência  
... --- ...  
é a codificação de SOS.

# Atividade prática



- Criar um sinaleiro de trânsito, com 3 LEDs (vermelho, amarelo, verde), ficando 10 segundos verde, 5 segundos amarelo e 15 segundos vermelho.
- Desafio: Adicionar um segundo sinaleiro, sincronizados como sinaleiros verdadeiros.