

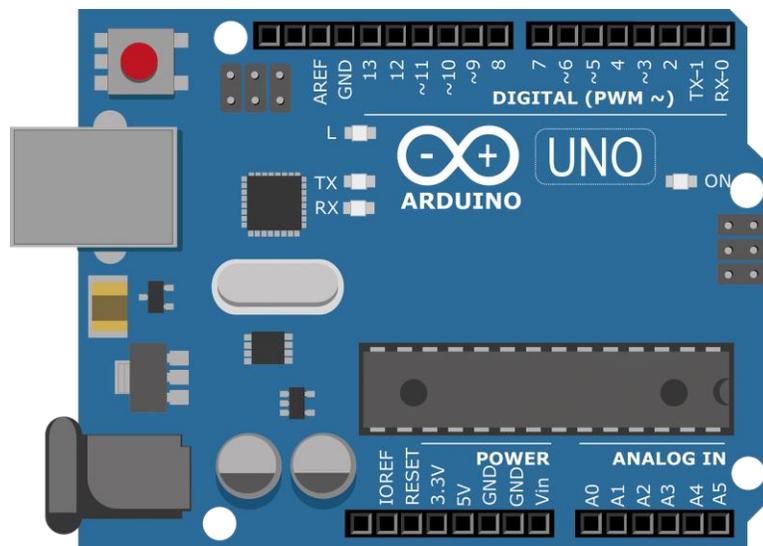
Introdução Arduino

“O Arduino é uma plataforma de prototipagem eletrônica *open-source* que se baseia em hardware e software flexíveis e fáceis de usar. É destinado a artistas, designers, hobbistas e qualquer pessoa interessada em criar objetos ou ambientes interativos.

O Arduino pode *sentir* o estado do ambiente que o cerca por meio da recepção de sinais de sensores e pode interagir com os seus arredores, controlando luzes, motores e outros atuadores. O microcontrolador na placa é programado com a linguagem de programação Arduino, baseada na linguagem Wiring, e o ambiente de desenvolvimento Arduino, baseado no ambiente Processing. Os projetos desenvolvidos com o Arduino podem ser *autônomos* ou podem comunicar-se com um computador para a realização da tarefa, com uso de *software* específico (ex: Flash, Processing, MaxMSP).”

A plataforma é formada por *Hardware* e *Software*.

O hardware é composto por uma placa de prototipagem na qual são construídos os projetos.

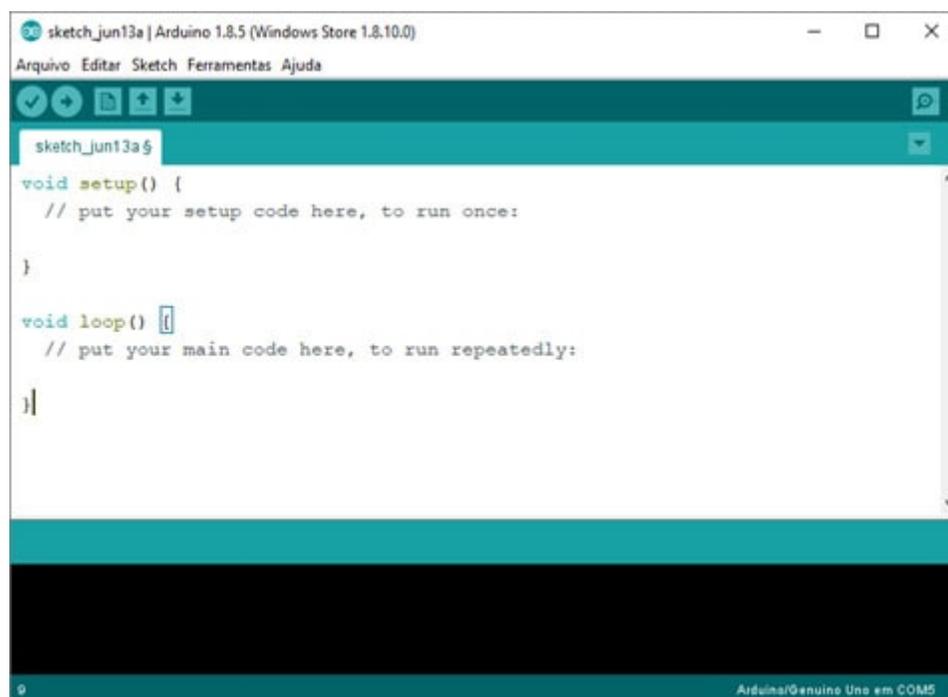


- 14 pinos de entrada e saída digital (pinos 0-13):
 - Esses pinos podem ser utilizados como entradas ou saídas digitais de acordo com a necessidade do projeto e conforme foi definido no *sketch* criado na IDE.
- 6 pinos de entradas analógicas (pinos A0 – A5):
 - Esses pinos são dedicados a receber valores analógicos, por exemplo, a tensão de um sensor. O valor a ser lido deve estar na faixa de 0 a 5 V onde serão convertidos para valores entre 0 e 1023.
- 6 pinos de saídas analógicas (pinos 3, 5, 6, 9, 10 e 11):
 - São pinos digitais que podem ser programados para ser utilizados como saídas analógicas, utilizando modulação PWM.

A alimentação da placa pode ser feita a partir da porta USB do computador ou através de um adaptador AC. Para o adaptador AC recomenda-se uma tensão de 9 a 12 volts.

O *software* é uma IDE, que é executado em um computador onde é feita a programação, conhecida como *sketch*, na qual será feita *upload* para a placa de prototipagem Arduino, através de uma comunicação serial. O *sketch* feito pelo projetista dirá à placa o que deve ser executado durante o seu funcionamento.

A IDE pode ser baixada gratuitamente no site do Arduino, onde pode ser escolhida a melhor opção de download conforme plataforma utilizada.

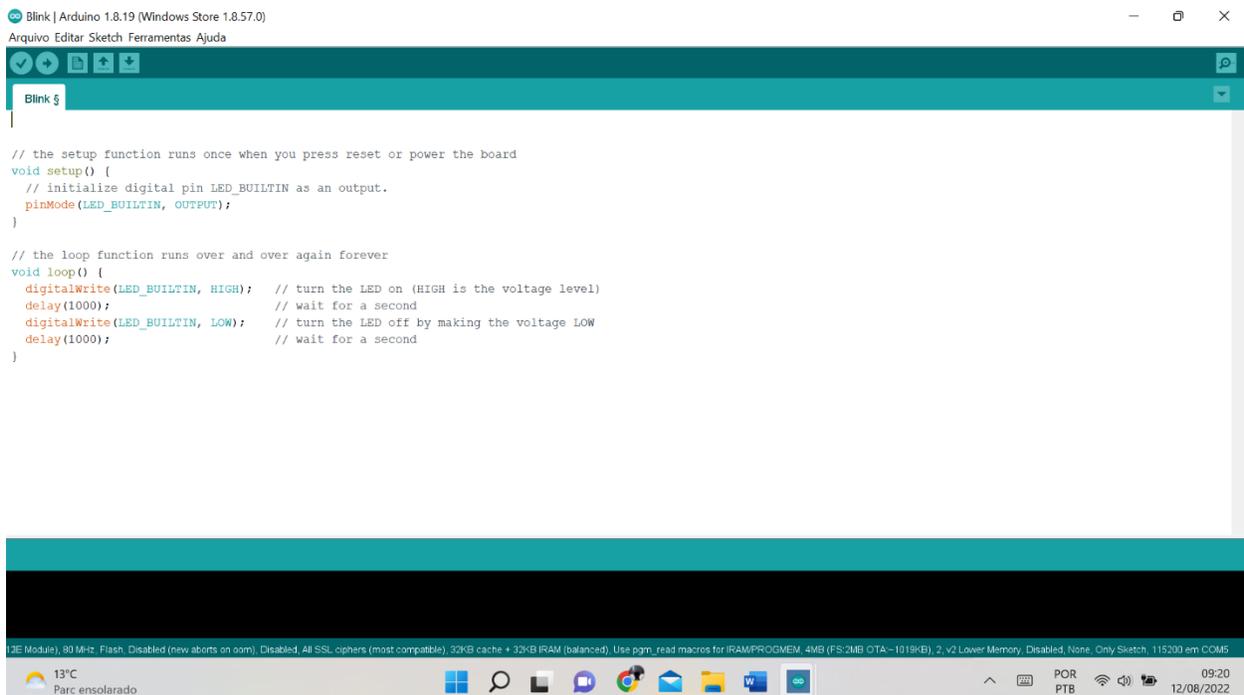
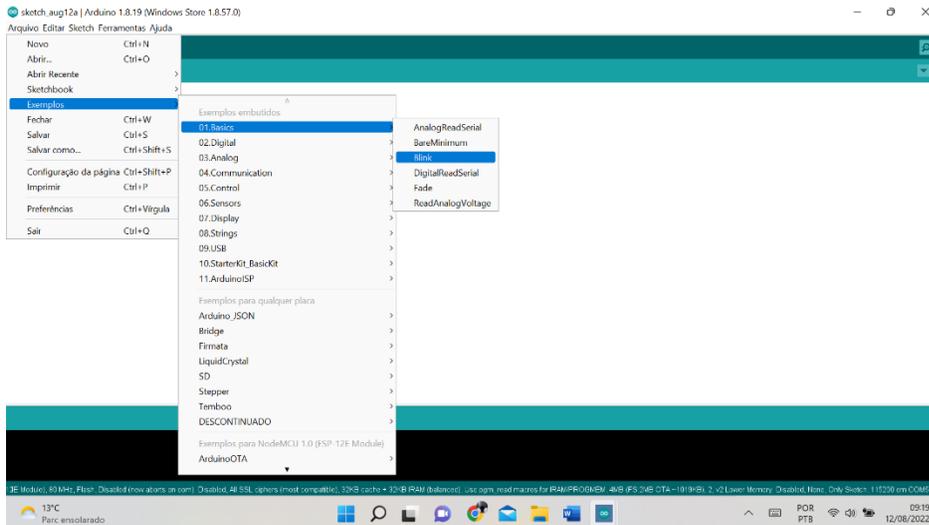


Abaixo são identificados os ícones de atalho da IDE:

- **Verify**
 - Verifica se existe erro no código digitado.
- **Upload**
 - Compila o código e grava na placa Arduino se corretamente conectada;
- **New**
 - Cria um novo *sketch* em branco.
- **Open**
 - Abre um *sketch*, presente no sketchbook.
- **Save**
 - Salva o *sketch* ativo
- **Seria monitor**
 - Abre o monitor serial.

Exemplo Simples (Blink)

A placa Arduino Uno já possui um Led ligado ao pino digital 13 que pode ser utilizado para o teste, e na IDE podemos carregar o exemplo *Blink*:



Código:

```
// the setup function runs once when you press reset or power the board  
void setup() {  
  // initialize digital pin LED_BUILTIN as an output.  
  pinMode(LED_BUILTIN, OUTPUT);  
}  
// the loop function runs over and over again forever  
void loop() {  
  digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)  
  delay(1000); // wait for a second  
  digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the voltage LOW  
  delay(1000); // wait for a second  
}
```

Analisando o código:

A função setup () é executada na inicialização do programa e é responsável pelas configurações iniciais do microcontrolador, tal como definição dos pinos de I/O, inicialização da comunicação serial, entre outras.

A função loop () será onde ocorrerá o laço infinito da programação, ou seja, onde será inserido o código que será executado continuamente pelo microcontrolador.

Dentro do *loop* principal está o código que fará o led ligado pino 13 piscar em intervalos de 1 segundo.

A função digitalWrite(led, HIGH); coloca o pino em nível lógico 1, ligando o led.

A função delay(1000); aguarda o tempo de 1000 ms, ou seja, 1 segundo para que possa ser executada a próxima instrução.

A função digitalWrite(led, LOW); coloca o pino em nível lógico 0, desligando o led.

E novamente é esperado 1 segundo com a função delay();

O loop é repetido infinitamente enquanto a placa estiver ligada.