

Mão na massa: Controlando servos

Como vimos nos vídeos, chegou a hora de conectar os componentes e programar. Caso isso já tenha feito, continue com a próxima atividade.

Antes de montar os servos:

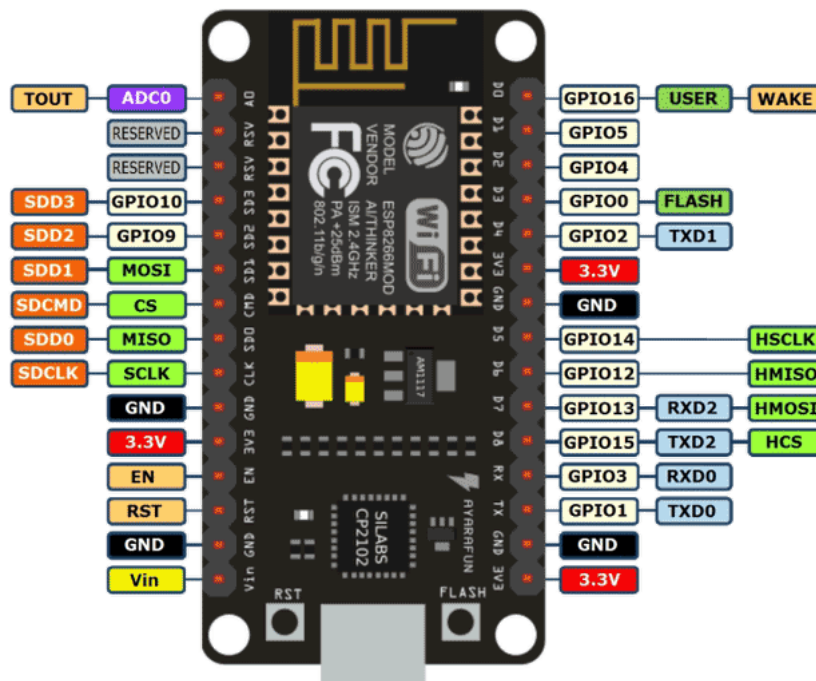
Como vimos no vídeo, usamos a **Arduino IDE** que pode ser baixada no [site do Arduino](https://www.arduino.cc/en/Main/Software) (<https://www.arduino.cc/en/Main/Software>). Neste curso, usamos a versão 1.8.4 da Arduino IDE.

Você deve instalar essa IDE, caso ainda não tenha feito.

Além disso, para usar a placa NodeMCU é preciso carregar uma biblioteca usando o *Gerenciador de Placas* da Arduino IDE. Uma explicação detalhada de como carregá-la se encontra nesse [exercício](https://cursos.alura.com.br/course/iot-com-nodemcu/task/27698) (<https://cursos.alura.com.br/course/iot-com-nodemcu/task/27698>). Carregue essa biblioteca antes de continuar.

Uma vez preparado o ambiente, você pode seguir com os passos abaixo:

- 1) Monte o seu suporte junto com os motores servos.
- 2) Agora conecte os dois motores ao NodeMCU. Para tanto, use os pinos D3 e D4 para controlar os motores. Além disso, conecte os servos com GROUND e 5V do NodeMCU.



- 3) O nosso sketch ainda é bem simples, pois apenas inicializa e posiciona aleatoriamente os motores. Implemente o sketch abaixo:

```
#include <Servo.h>

#define POS_INICIAL 90
#define X_MINIMO 30 // Angulo de X (30 a 150)
```

```
#define Y_MINIMO    50 // Angulo de y (50 a 90)
#define X_INTERVALO 12 // (X maximo - X minimo)/10 = 12
#define Y_INTERVALO 4 // (Y maximo - Y minimo)/10 = 4

Servo servoX;
Servo servoY;

void setup() {
  servoX.attach(D3);
  servoY.attach(D4);
  servoX.write(POS_INICIAL);
  servoY.write(POS_INICIAL);
  Serial.begin(9600);
}

void loop() {

  posiciona_servos();

}

//FUNCOES AUXILIARES

void posiciona_servos(){
  int posicaoX = (random(0,(X_INTERVALO))* 10 + (X_MINIMO));
  int posicaoY = (random(0,(Y_INTERVALO))* 10 + (Y_MINIMO));
  servoX.write(posicaoX);
  servoY.write(posicaoY);
  Serial.print(posicaoX);
  Serial.print(" , ");
  Serial.println(posicaoY);

  delay(2000);
}
```

- 4) Compile tudo e faça o upload para NodeMCU. Os motores devem ficar girando aleatoriamente a cada dois segundos.
- 5) Faça os ajustes para o seu projeto.