

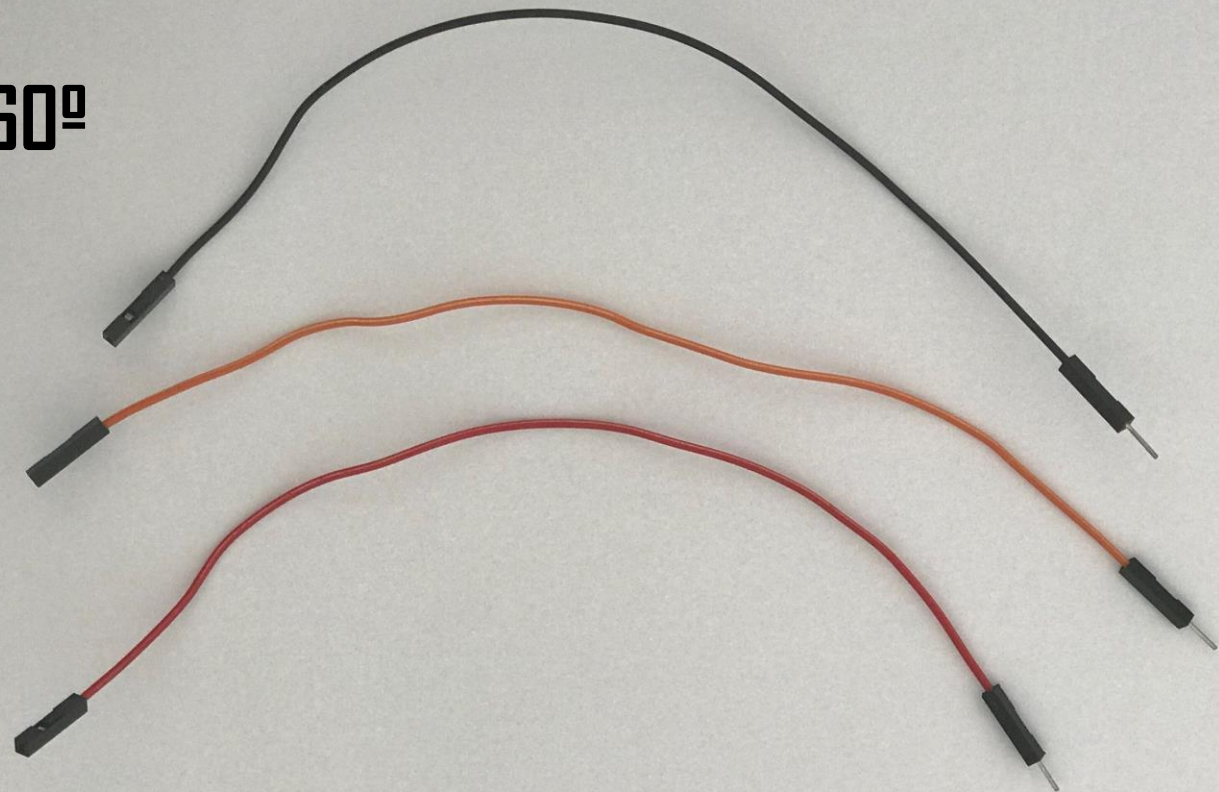
CONFIGURAÇÕES E PROGRAMAÇÕES

MICRO SERVO MOTOR GIRANDO EM 360º

No micro servo temos:

- Fio marrom - GND
- Fio vermelha - VIN
- Fio laranja - D2 (GPIO2)

Pegamos três cabos, como os da imagem ao lado



CONFIGURAÇÕES E PROGRAMAÇÕES



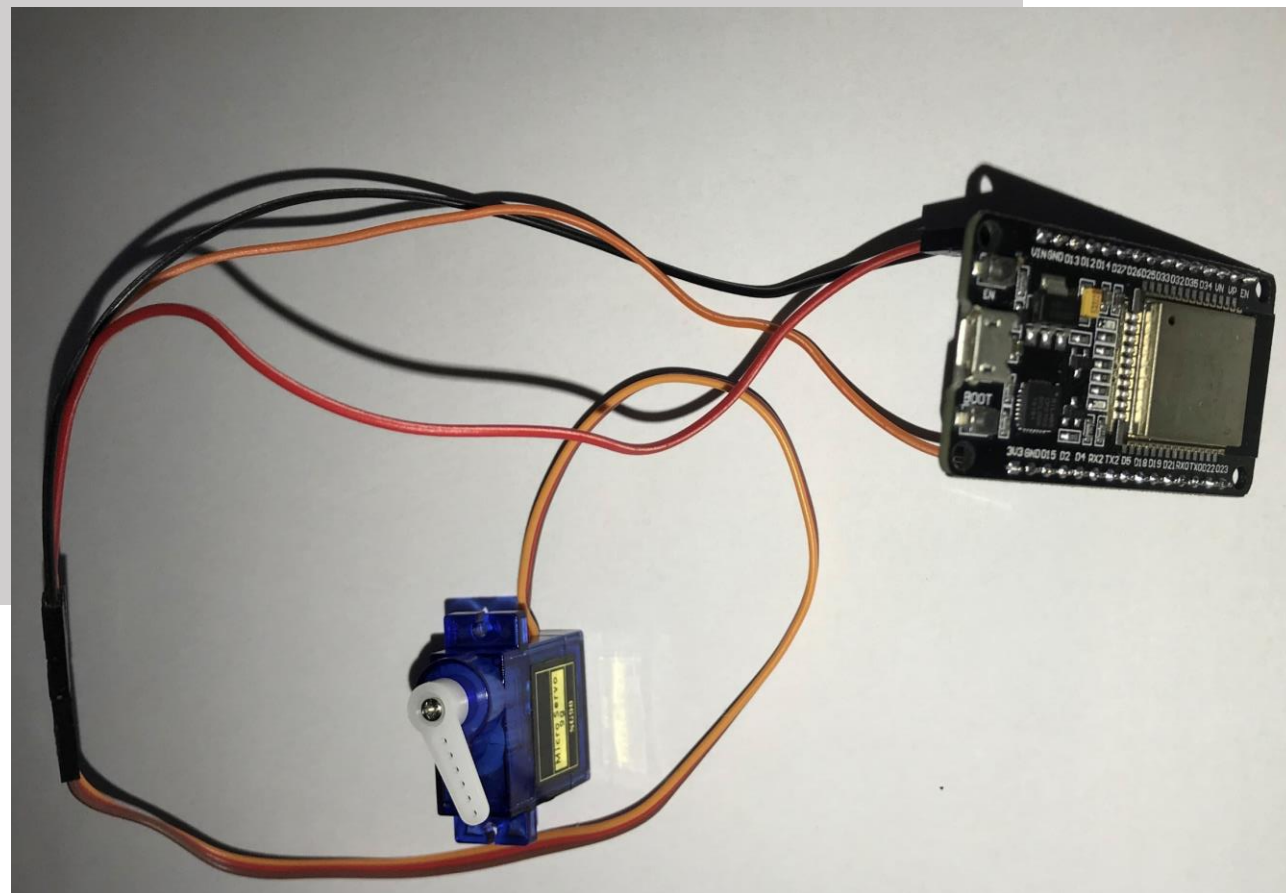
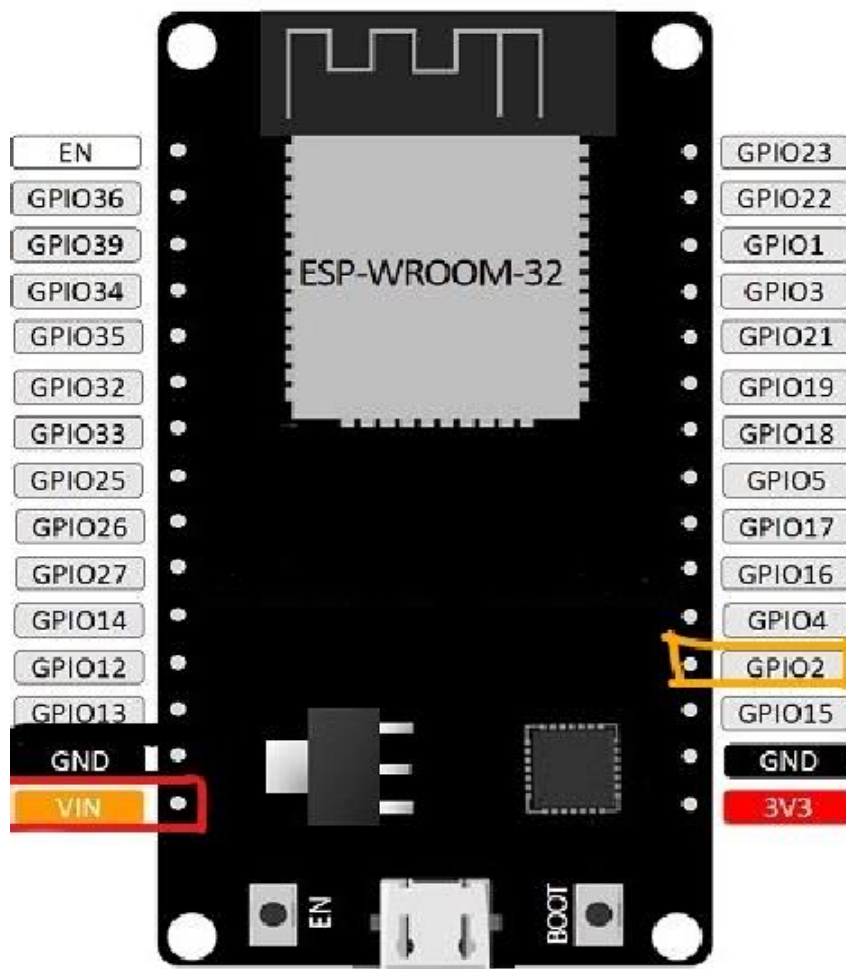
Conectamos cada um no micro servo e em seguida ligamos no ESP32, veja a seguir:

- Fio marrom – Fio Preto
- Fio vermelho – Fio vermelho
- Fio laranja – Fio laranja

CONFIGURAÇÕES E PROGRAMAÇÕES

Agora conectaremos ao ESP32:

- Marrom/Preto - GND
- Vermelho/Vermelho - VIN
- Laranja/Laranja - GPIO2 (D2)





Depois de abrir o uPyCraft, volte ao tópico **CONECTANDO A ESP32 NO UPYCRAFT** e realize os comandos orientados no mesmo. Depois de realizado siga os passos solicitados no tópico atual.

Ao abrir o uPyCraft, pressione **Ctrl + N**, ou selecione "File" e em seguida "**New**". Ao abrir um novo arquivo, transcreva o código e salve.

The screenshot shows the uPyCraft V1.1 IDE interface. The title bar reads "uPyCraft V1.1" with standard window controls. The menu bar includes "File", "Edit", "Tools", and "Help". On the left, a file explorer shows a tree structure with folders: "device", "sd", "uPy_lib", and "workSpace". The main editor window has a tab titled "Servo-motor.py" with a close button. The code in the editor is as follows:

```
1 import machine
2 from machine import Pin, PWM
3
4 p2 = machine.Pin(2)
5 servo = machine.PWM(p2, freq=200, duty=550)
```

On the right side of the editor, there is a vertical toolbar with icons for: creating a new file, opening a file, saving a file, running the program (a green play button), stopping the program (a red stop button), toggling debug mode (a lightning bolt icon), undo, redo, showing a list of errors/warnings, and deleting the current file.

CONFIGURAÇÕES E PROGRAMAÇÕES

Para executar aperte **F5** ou clique no botão **DOWNLOADANDRUN**, o terminal do uPyCraft ficará como abaixo e o servo motor irá dar voltas em 360°.

```
>>>
>>>

>>>

Ready to download this file, please wait!
..
download ok
exec(open("Servo-motor.py").read(),globals())
>>>
```

Se quisermos definir um tempo para o servo motor parar de girar é só importar a biblioteca **"time"** e acrescentar mais duas linhas de código, veja na imagem ao lado.

uPyCraft V1.1

File Edit Tools Help

device
sd
uPy_lib
workSpace

Servo-motor.py X

```
1 import machine
2 from machine import Pin, PWM
3 import time
4
5 p2 = machine.Pin(2)
6 servo = machine.PWM(p2, freq=200, duty=550)
7 time.sleep(5)
8 servo.deinit()
9
```

```
download ok
exec(open("Servo-motor.py").read(),globals())
>>>
>>>

>>>

Ready to download this file, please wait!
..
download ok
exec(open("Servo-motor.py").read(),globals())
```