



Franzininho Na FATEC





Aula 05:

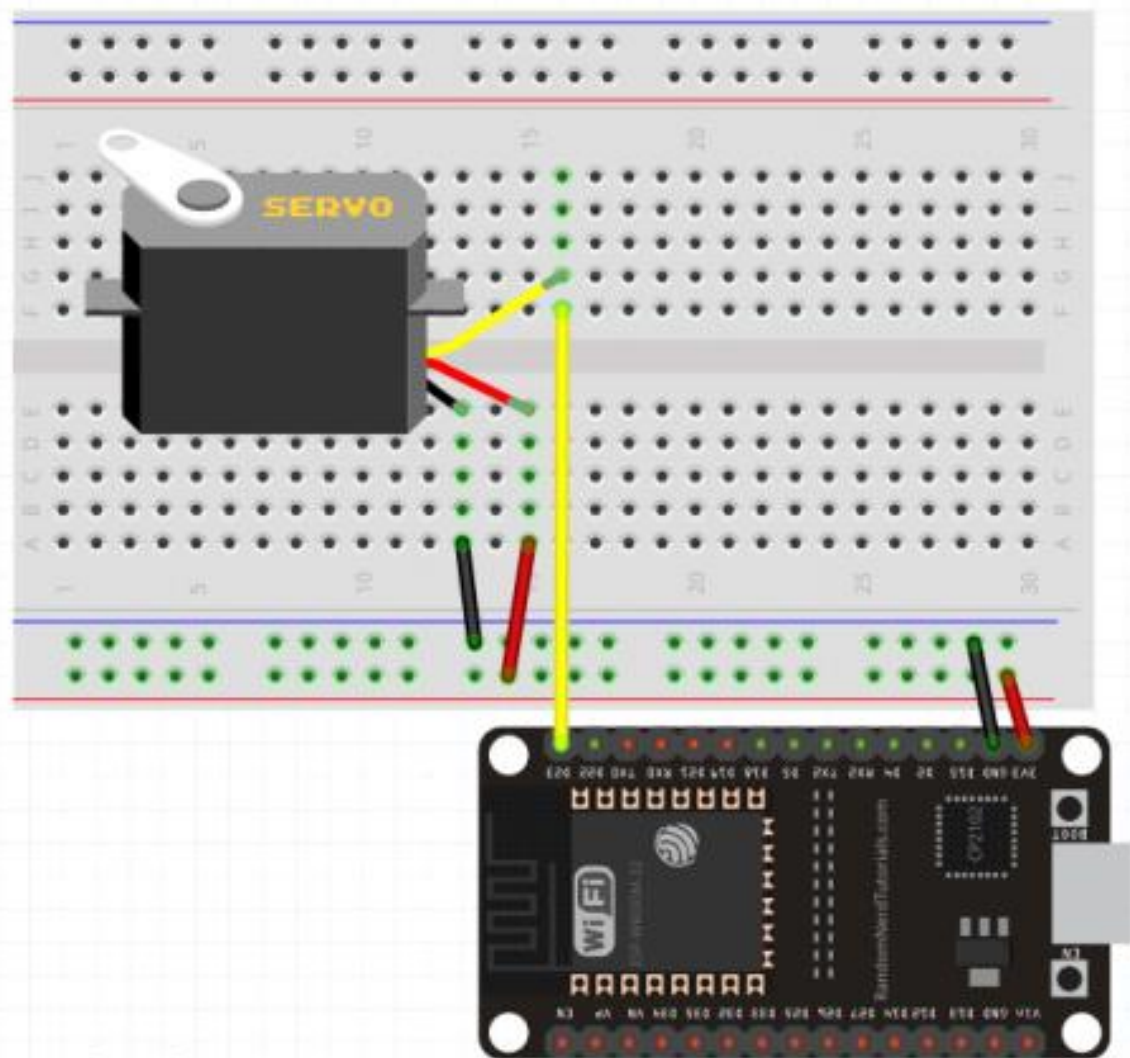
Servo



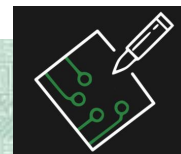
Servo Motor

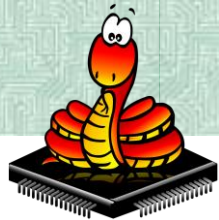
- Um servo motor é um pequeno motor de corrente contínua. Diferentemente dos demais motores DC, onde apenas controlamos a velocidade de giro, conseguimos controlar a posição de um servo, através do PWM.





fritzing

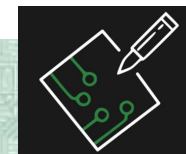




Servo Motor

- Nesse programa, vamos aprender a como usar um servo motor e posicioná-lo no ângulo que desejamos. Para isso, escreva o código a seguir:
- Começaremos impontando as funções Pin e PWM do módulo machine, que nos permitira acessar os pinos da placa e configurá-los:

```
1  from machine import Pin, PWM
```



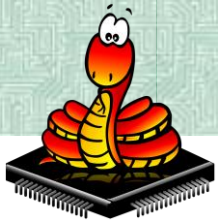


Servo Motor

- Também, necessitaremos da função sleep da biblioteca time que, utilizaremos como para intervalos de temporização:
- E, a biblioteca nativa do Python, math. Que traz diversas funções matemáticas e usaremos a de arredondamento:

```
2  from time import sleep
3  import math
```





Servo Motor

- Agora, criaremos uma função chamada converter onde necessitaremos de 5 parâmetros: valor que desejamos converter, entrada mínima, entrada máxima, saída mínima e saída máxima.

```
5 def converter(x, in_min,in_max,out_min,out_max):  
6     |   conta = math.trunc((x - in_min) * (out_max - out_min) / (in_max - in_min) + out_min)  
7     |   return conta
```





Servo Motor

- A segunda função, criaremos com o intuito de enviar o ângulo desejado para o servo motor.
- No servo utilizado no experimento, o intervalo vai de 20 a 120. Essa função, pede apenas um parâmetro, o ângulo que desejamos posicionar o nosso servo motor.

```
9  def angulo(x):  
10     angulo = converter(x, 0,4095,20,120)  
11     print('angle',angulo)  
12     servo.duty(angulo)
```





Servo Motor

- Configuraremos o pino como uma saída PWM, nota que definiremos uma frequência de 50Hz nesse caso, pois é a frequência padrão de servomecanismo:

```
13
14  servo = PWM(Pin(23), freq = 50) #frequência de servomecanismo
15
```





Servo Motor

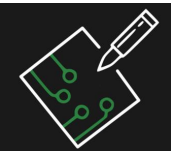
- Por fim, vamos para o laço de repetição infinita. Onde a cada dois segundos, trocaremos a posição do servo:

```
15  while True:
16      angulo(0)
17      sleep(2)
18      angulo(45)
19      sleep(2)
20      angulo(90)
21      sleep(2)
22      angulo(135)
23      sleep(2)
24      angulo(180)
25      sleep(2)
```





```
1  ∨ from machine import Pin, PWM
2    from time import sleep
3    import math
4
5  ∨ def converter(x, in_min,in_max,out_min,out_max):
6      conta = math.trunc((x - in_min) * (out_max - out_min) / (in_max - in_min) + out_min)
7      return conta
8
9  ∨ def angulo(x):
10     angulo = converter(x, 0,180,20,120)
11     servo.duty(angulo)
12
13     servo = PWM(Pin(23), freq = 50) #frequência de servomecanismo
14
15  ∨ while True:
16     angulo(0)
17     sleep(2)
18     angulo(45)
19     sleep(2)
20     angulo(90)
21     sleep(2)
22     angulo(135)
23     sleep(2)
24     angulo(180)
25     sleep(2)
```

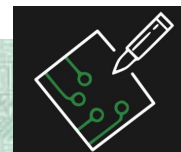
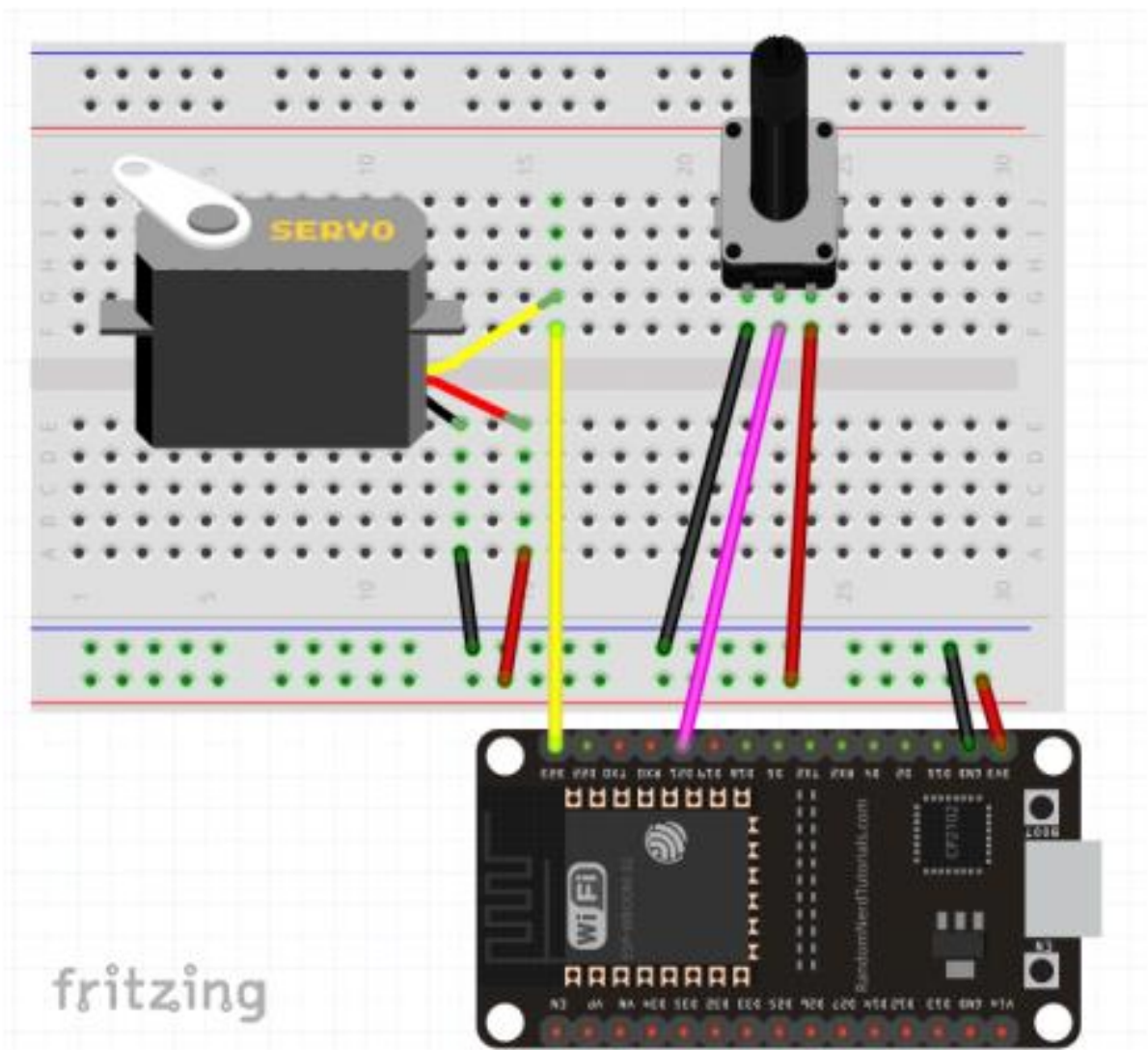




Exercícios

- 1) Assim como fizemos no ultimo capitulo com a intensidade luminosa do led, movimente o servo de 0° a 180° e depois decemente o valor.
- 2) Acrescente um potenciômetro a sua montagem e, controle o ângulo do servo de acordo com a posição do potenciômetro.







```
1  from machine import Pin, PWM, ADC
2  from time import sleep
3  import math
4
5  def converter(x, in_min,in_max,out_min,out_max):
6      conta = math.trunc((x - in_min) * (out_max - out_min) / (in_max - in_min) + out_min)
7      return conta
8
9  def angulo(x):
10     angulo = converter(x, 0,4095,20,120)
11     print('angle',angulo)
12     servo.duty(angulo)
13
14     servo = PWM(Pin(23), freq = 50) #frequência de servomecanismo
15
16     pot = ADC(Pin(32))
17     pot.atten(ADC.ATTN_11DB)
18     pot.width(ADC.WIDTH_12BIT) #seta 12 bits(faixa de 0 - 4095)
19
20     while True:
21         angle = pot.read()
22         print(pot.read())
23         angulo(angle)
24         sleep(0.1)
```

