

Controle do carrinho via Python

Transcrição

Com a base do nosso carrinho montada, podemos dar início à implementação do script Python que irá controlá-lo. Na *home* do Raspberry Pi, vamos criar uma pasta chamada **piobot**, que será onde ficará guardados os nossos scripts:

```
mkdir ~/piobot
cd ~/piobot
```

Dentro dessa pasta, vamos criar o arquivo **controle.py**. Para editar o arquivo, você pode usar os editores de texto **vi** ou **nano**, ou até utilizar a interface gráfica do Raspberry Pi via VNC, como vimos no curso anterior.

Vamos começar importando as bibliotecas necessárias:

```
import RPi.GPIO as GPIO
import time
import getch
```

Aqui a novidade é a biblioteca **getch**, que irá ler os *inputs* do teclado e os transformará em comandos para o carrinho. Veremos detalhes sobre essa biblioteca mais à frente.

Agora dizemos que vamos usar como referência no script a **pinagem física** do GPIO. Vamos remover também as mensagens de alerta:

```
import RPi.GPIO as GPIO
import time
import getch

GPIO.setmode(GPIO.BOARD)
GPIO.setwarnings(False)
```

Faremos agora o mapeamento dos pinos, dando nome aos mesmos:

```
import RPi.GPIO as GPIO
import time
import getch

GPIO.setmode(GPIO.BOARD)
GPIO.setwarnings(False)

F_DIREITA = 16
F_ESQUERDA = 11
T_DIREITA = 18
T_ESQUERDA = 13
```

Com os pinos com os seus devidos nomes, vamos criar uma função que defina-os como saída, ou seja, como **OUTPUT** :

```
import RPi.GPIO as GPIO
import time
import getch

GPIO.setmode(GPIO.BOARD)
GPIO.setwarnings(False)

F_DIREITA = 16
F_ESQUERDA = 11
T_DIREITA = 18
T_ESQUERDA = 13

def setup_motor():
    GPIO.setup(F_DIREITA, GPIO.OUT)
    GPIO.setup(F_ESQUERDA, GPIO.OUT)
    GPIO.setup(T_DIREITA, GPIO.OUT)
    GPIO.setup(T_ESQUERDA, GPIO.OUT)
```

Com os motores inicializados, podemos agora mover o nosso carrinho. Vamos criar duas funções, uma que move nosso carrinho para frente e outra para a direita. Começaremos com a função que move o carrinho para frente. Ela colocará os dois pinos responsáveis pelo movimento de ir em frente em alta, aguardará 0.4 segundos e colocará os pinos em baixa novamente:

```
import RPi.GPIO as GPIO
import time
import getch

GPIO.setmode(GPIO.BOARD)
GPIO.setwarnings(False)

F_DIREITA = 16
F_ESQUERDA = 11
T_DIREITA = 18
T_ESQUERDA = 13

def setup_motor():
    GPIO.setup(F_DIREITA, GPIO.OUT)
    GPIO.setup(F_ESQUERDA, GPIO.OUT)
    GPIO.setup(T_DIREITA, GPIO.OUT)
    GPIO.setup(T_ESQUERDA, GPIO.OUT)

def move_frente():
    GPIO.output(F_DIREITA, GPIO.HIGH)
    GPIO.output(F_ESQUERDA, GPIO.HIGH)
    time.sleep(0.4)
    GPIO.output(F_DIREITA, GPIO.LOW)
    GPIO.output(F_ESQUERDA, GPIO.LOW)
```

Ou seja, quando a função for chamada, nosso carrinho dará um impulso, que irá durar 0.4 segundos, e depois irá parar.

Para criar a função que move o carrinho para a direita, precisamos ter atenção, pois precisamos mover somente o motor da **esquerda** para frente:

```
import RPi.GPIO as GPIO
import time
import getch

GPIO.setmode(GPIO.BOARD)
GPIO.setwarnings(False)

F_DIREITA = 16
F_ESQUERDA = 11
T_DIREITA = 18
T_ESQUERDA = 13

def setup_motor():
    GPIO.setup(F_DIREITA, GPIO.OUT)
    GPIO.setup(F_ESQUERDA, GPIO.OUT)
    GPIO.setup(T_DIREITA, GPIO.OUT)
    GPIO.setup(T_ESQUERDA, GPIO.OUT)

def move_frente():
    GPIO.output(F_DIREITA, GPIO.HIGH)
    GPIO.output(F_ESQUERDA, GPIO.HIGH)
    time.sleep(0.4)
    GPIO.output(F_DIREITA, GPIO.LOW)
    GPIO.output(F_ESQUERDA, GPIO.LOW)

def move_direita():
    GPIO.output(F_ESQUERDA, GPIO.HIGH)
    time.sleep(0.2)
    GPIO.output(F_ESQUERDA, GPIO.LOW)
```

Com essas funções criadas, no próximo vídeos iremos complementar o script, adicionando mais funcionalidades, entre elas a captura de comandos do teclado!