

(Opcional) Para saber mais: Melhorando o código parte 2

Vamos modificar nosso código focando no motorBase.

Antes de tudo, vamos importar a biblioteca, para isso, escreva:

```
#include <VarSpeedServo.h>
```

A linha acima deve vir antes de qualquer coisa!

Após importar a biblioteca VarSpeedServo, vamos criar uma variável auxiliar para cada servo motor, logo abaixo do nosso **define** ANGULO_INICIAL_MOTOR

```
int auxBase = ANGULO_INICIAL_MOTOR;
```

Essa será a variável que vai manter nosso braço parado!

A segunda modificação acontece no nosso mapeamento dos servos, agora teremos o seguinte:

```
.  
.   
VarSpeedServo motorBase;  
.   
.
```

Agora modificaremos o nosso método, mas antes vamos dar uma revisada.

Atualmente, temos o seguinte:

Em linguagem humana, lemos nosso analógico, mapeamos essa leitura, transformando-a em um valor dentro do intervalo entre 0 e 180, e então posicionamos nosso motor nessa posição:

```
void moverBase(){  
  int posX = analogRead(joystick1X);  
  posX = map(posX, 0, 1023, 0, 180); //aqui a mágica acontece  
  motorBase.write(posX);  
}
```

Como vimos, isso acontece de maneira muito rápida e não mantém nosso braço parado, ele sempre retorna para o ângulo inicial.

Analisando bem a situação, podemos ver que isso ocorre devido ao loop do arduino, que está o tempo todo chamando todos os métodos e lendo a posição dos analógicos

Para resolver esse problema, teremos que seguir a seguinte linha de raciocínio:

Teremos a leitura inicial do joystick, mas não posicionaremos nosso servo diretamente, antes faremos um teste.

Se a leitura do joystick for maior que determinado valor, vamos incrementar a posição do braço, caso contrário, iremos decrementar, vamos começar:

Manteremos as primeiras duas linhas, lemos a posição e mapeamos.

```
void moverBase(){
  int posX = analogRead(joystick1X);
  posX = map(posX, 0, 1023, 0, 180);
}
```

Escreveremos um `if` onde testaremos a posição do analógico, ficando da seguinte forma:

```
void moverBase(){
  int posX = analogRead(joystick1X);
  posX = map(posX, 0, 1023, 0, 180);
  if(posX > 100){
    // Faz algo
  }
}
```

Devemos também ter uma condicional para o caso de o valor ser menor:

```
void moverBase(){
  int posX = analogRead(joystick1X);
  posX = map(posX, 0, 1023, 0, 180);
  if(posX > 100){
    //Faz algo
  }else if(posX < 80){
    //Faz algo
  }
}
```

Repare que estamos comparando com 100 e 80. Tendo em vista que nosso analógico inicia no 90, demos uma tolerância, já que o mesmo pode ter variações de acordo com o uso.

Agora devemos completar nosso código, removendo o comentário `//Faz algo`.

Dentro dos `ifs`, temos que incrementar ou decrementar nossa posição, utilizando a nossa variável auxiliar, nesse caso `auxBase`.

```
void moverBase(){
  int posX = analogRead(joystick1X);
  posX = map(posX, 0, 1023, 0, 180);
  if(posX > 100){
    auxBase+=10;
    motorBase.write(auxBase,50);
  }else if(posX < 80){
    auxBase-=10;
  }
}
```

```
motorBase.write(auxBase, 50);  
}
```

É de extrema importância que você teste com diferentes valores e veja o que melhor se encaixa para você. Troque também os valores do segundo parâmetro do `write`, e veja que a velocidade do braço também muda.

Lembre-se de fazer a alteração em todo seu código e utilizar uma variável correspondente para cada servo!