

03

Funções para inicializar os servos

Transcrição

Para implementar o programa final do braço-robô, vamos utilizar como base o código já feito no aula 4:

```
#include <Servo.h>

#define ANGULO_INICIAL_MOTOR 90

// --- Mapeamento dos Joysticks ---
#define joystick1X A0
#define joystick1Y A1

// --- Mapeamento dos Servos ---
Servo motorBase;

void setup() {
  motorBase.attach(5);
  pinMode(joystick1X, INPUT);
  motorBase.write(ANGULO_INICIAL_MOTOR);
}

void loop() {
  int posX = analogRead(joystick1X);
  posX = map(posX, 0, 1023, 0, 180);
  motorBase.write(posX);
  delay(100);
}
```

Vamos começar definindo os pinos do segundo joystick, que serão os pinos **A2** e **A3**, para os eixos X e Y, respectivamente:

```
// --- Mapeamento dos Joysticks ---
#define joystick1X A0
#define joystick1Y A1
#define joystick2X A2
#define joystick2Y A3
```

No mapeamento dos servos, vamos adicionar o restante dos motores:

```
// --- Mapeamento dos Servos ---
Servo motorBase;
Servo motorBraco1;
Servo motorBraco2;
Servo motorGarra;
```

Agora, vamos criar uma função que mapeia os pinos dos servo motores, eles ficarão nos seguintes pinos:

- `motorBase` : pino 5
- `motorBraco1` : pino 6
- `motorBraco2` : pino 9
- `motorGarra` : pino 10

Logo, a função ficará assim:

```
void mapearPinosDosMotores() {  
    motorBase.attach(5);  
    motorBraco1.attach(6);  
    motorBraco2.attach(9);  
    motorGarra.attach(10);  
}
```

Para finalizar este vídeo, vamos criar uma função para definir o ângulo inicial de todos os motores para 90°, lembrando que já temos a constante `ANGULO_INICIAL_MOTOR` com esse valor guardado:

```
void inicializarAngulosDosMotores() {  
    motorBase.write(ANGULO_INICIAL_MOTOR);  
    motorBraco1.write(ANGULO_INICIAL_MOTOR);  
    motorBraco2.write(ANGULO_INICIAL_MOTOR);  
    motorGarra.write(ANGULO_INICIAL_MOTOR);  
}
```

No próximo vídeo, vamos chamar essas funções dentro da função `setup()` e continuar incrementando o nosso código.