

## Completando o programa

### Transcrição

Já conectamos e testamos cada um dos componentes do nosso projeto individualmente. Os motores, sensor PIR e *laser*, mas vamos revisar e melhorar um pouco do nosso código. Vamos começar comentando novamente onde usamos a serial para imprimir mensagens. Comentamos também o `delay` dentro da função de posicionamento dos motores.

```
#include <Servo.h>

#define POS_INICIAL 90
#define X_MINIMO 30
#define Y_MINIMO 50
#define X_INTERVALO 12
#define Y_INTERVALO 4
#define LASER D1
#define PIR D2

Servo servoX;
Servo servoY;

void setup() {
    servoX.attach(D3);
    servoY.attach(D4);
    servoX.write(POS_INICIAL);
    servoY.write(POS_INICIAL);

    pinMode(LASER, OUTPUT);
    digitalWrite(LASER, LOW);
    // Serial.begin(9600);
}

void loop() {
    int trigger = digitalReader(PIR);
    if(trigger == HIGH){
        liga_laser();
        // Serial.println("Detectado")
    } else {
        desliga_laser();
        // Serial.println("NÃO Detectado");
    }
    delay(1000);
}

// FUNÇÕES AUXILIARES

void posiciona_motores() {
    int posicaoX = (random(0, (X_INTERVALO)) * 10 + X_MINIMO);
    int posicaoY = (random(0, (Y_INTERVALO)) * 10 + Y_MINIMO);
    servoX.write(posicaoX);
    servoY.write(posicaoY);
    // Serial.print(posicaoX);
}
```

```

    // Serial.print("  ", " ");
    // Serial.println(posicaoY);
    // delay(2000);
}

void liga_laser() {
    digitalWrite(LASER, HIGH);
}

void desliga_laser() {
    digitalWrite(LASER, LOW);
}

```

A ideia é que, quando o *laser* seja ativado, os motores também iniciem a movimentação do suporte. Então dentro da função `liga_laser` usaremos a função `posiciona_motores`. Vejamos:

```

void liga_laser() {
    for (int i = 0; i < TEMPO_LASER/2; ++i){
        digitalWrite(LASER, HIGH);
        posiciona_motores();
        delay(2000);
    }
}

```

O laço nos utilizamos para garantir que os motores estarão sempre se reposicionando a cada rodada de tempo. Note que estamos usando uma nova variável, que ainda não definimos. Defina-a com o valor 60.

```
#define TEMPO_LASER 60
```

Testando com um gato de verdade, chegamos a esses valores, mas explicaremos mais detalhadamente agora. O que acontece é: Estamos definindo um tempo de funcionamento para o reposicionamento dos motores, 60 segundos. Mas a movimentação a cada 1 segundo fica muito rápida, por este motivo, dividimos o tempo no laço para ter 30 iterações (60 / 2), mas demos um delay de 2000 (2 segundos) o que dará no mesmo tempo final, ou seja, o reposicionamento será feito a cada 2 segundos, mas estamos usando como base o valor de 1 segundo apenas.

É importante testar bem esses valores, para gatos pode ser um, para tartarugas ser outro e assim por diante. Sinta-se livre para publicar no fórum suas experiências! Nosso código final estará assim:

```

#include <Servo.h>

#define POS_INICIAL 90
#define X_MINIMO 30
#define Y_MINIMO 50
#define X_INTERVALO 12
#define Y_INTERVALO 4
#define LASER D1
#define PIR D2

Servo servoX;
Servo servoY;

void setup() {

```

```
servoX.attach(D3);
servoY.attach(D4);
servoX.write(POS_INICIAL);
servoY.write(POS_INICIAL);

pinMode(LASER, OUTPUT);
digitalWrite(LASER, LOW);
// Serial.begin(9600);
}

void loop() {
    int trigger = digitalReader(PIR);
    if(trigger == HIGH){
        liga_laser();
        // Serial.println("Detectado")
    } else {
        desliga_laser();
        // Serial.println("NÃO Detectado");
    }
    delay(1000);
}

// FUNÇÕES AUXILIARES

void posiciona_motores() {
    int posicaoX = (random(0, (X_INTERVALO)) * 10 + X_MINIMO);
    int posicaoY = (random(0, (Y_INTERVALO)) * 10 + Y_MINIMO);
    servoX.write(posicaoX);
    servoY.write(posicaoY);
    // Serial.print(posicaoX);
    // Serial.print(" , ");
    // Serial.println(posicaoY);
    // delay(2000);
}

void liga_laser() {
    for (int i = 0; i < TEMPO_LASER/2; ++i){
        digitalWrite(LASER, HIGH);
        posiciona_motores();
        delay(2000);
    }
}

void desliga_laser() {
    digitalWrite(LASER, LOW);
}
```