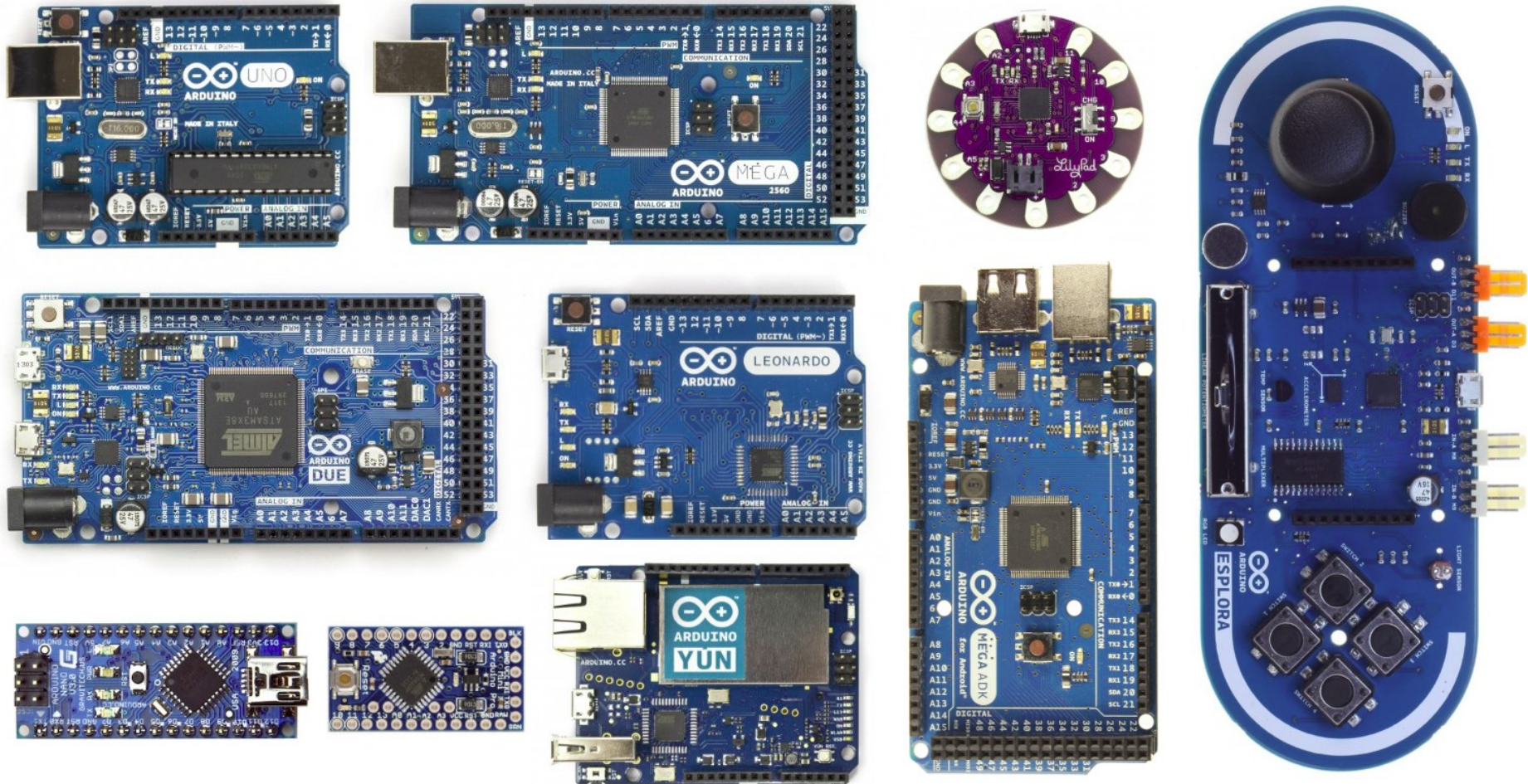


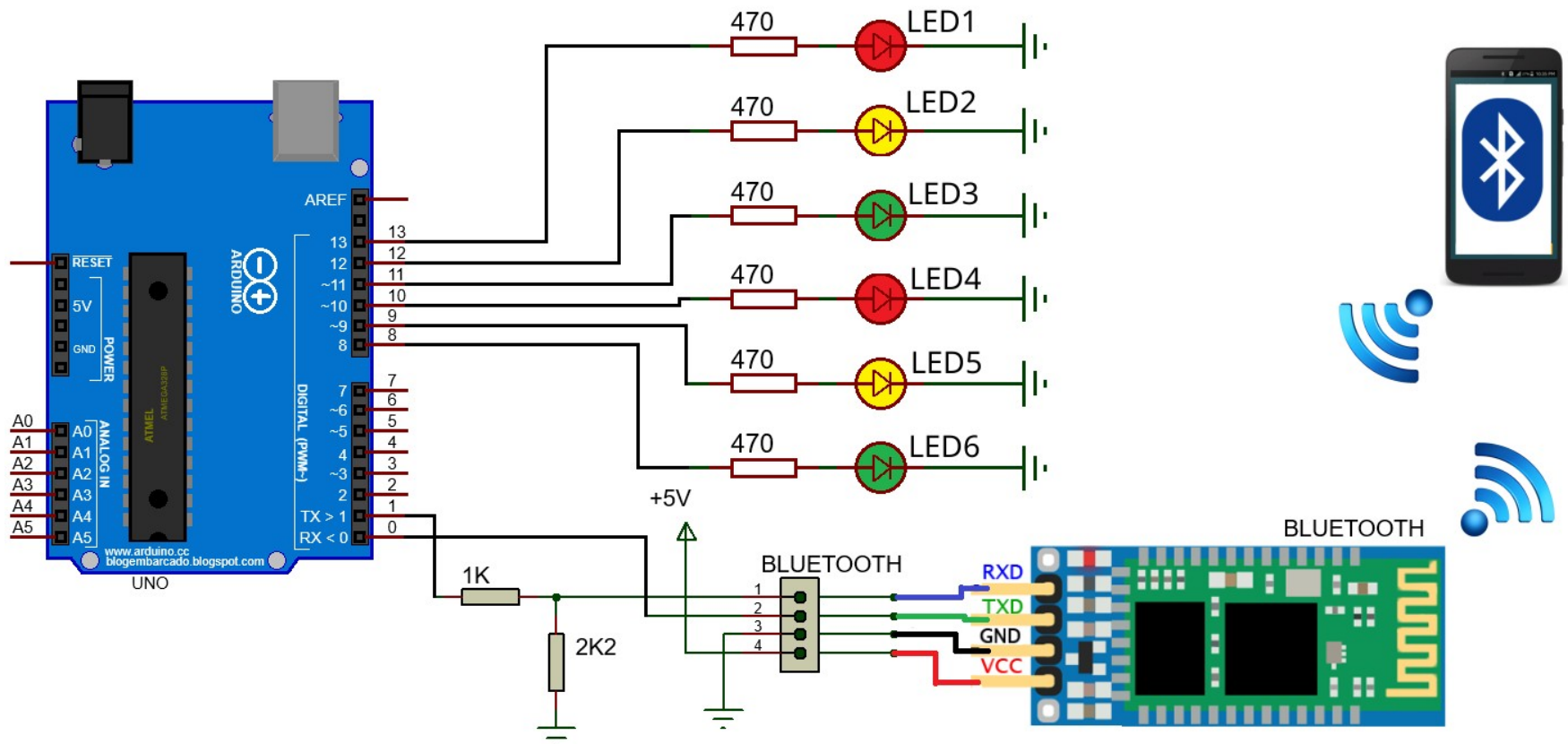


Curso Arduino – Aula 4



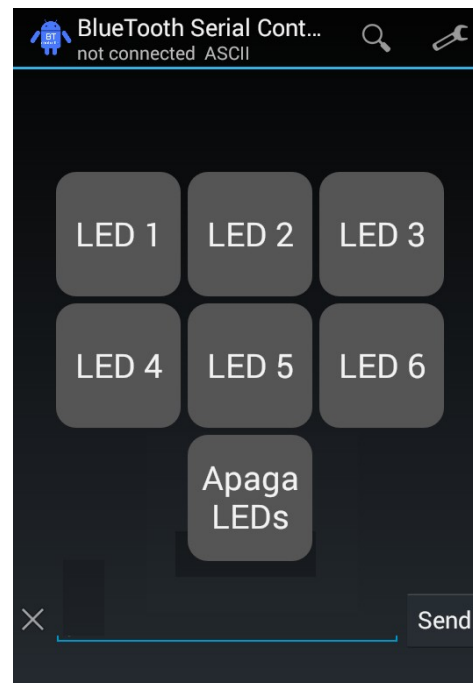
Prática 11 – Acionamento de varios LEDs com Celular via BlueTooth

Crie um programa que ligue um LED por vez, controlado pelo Celular via BlueTooth.



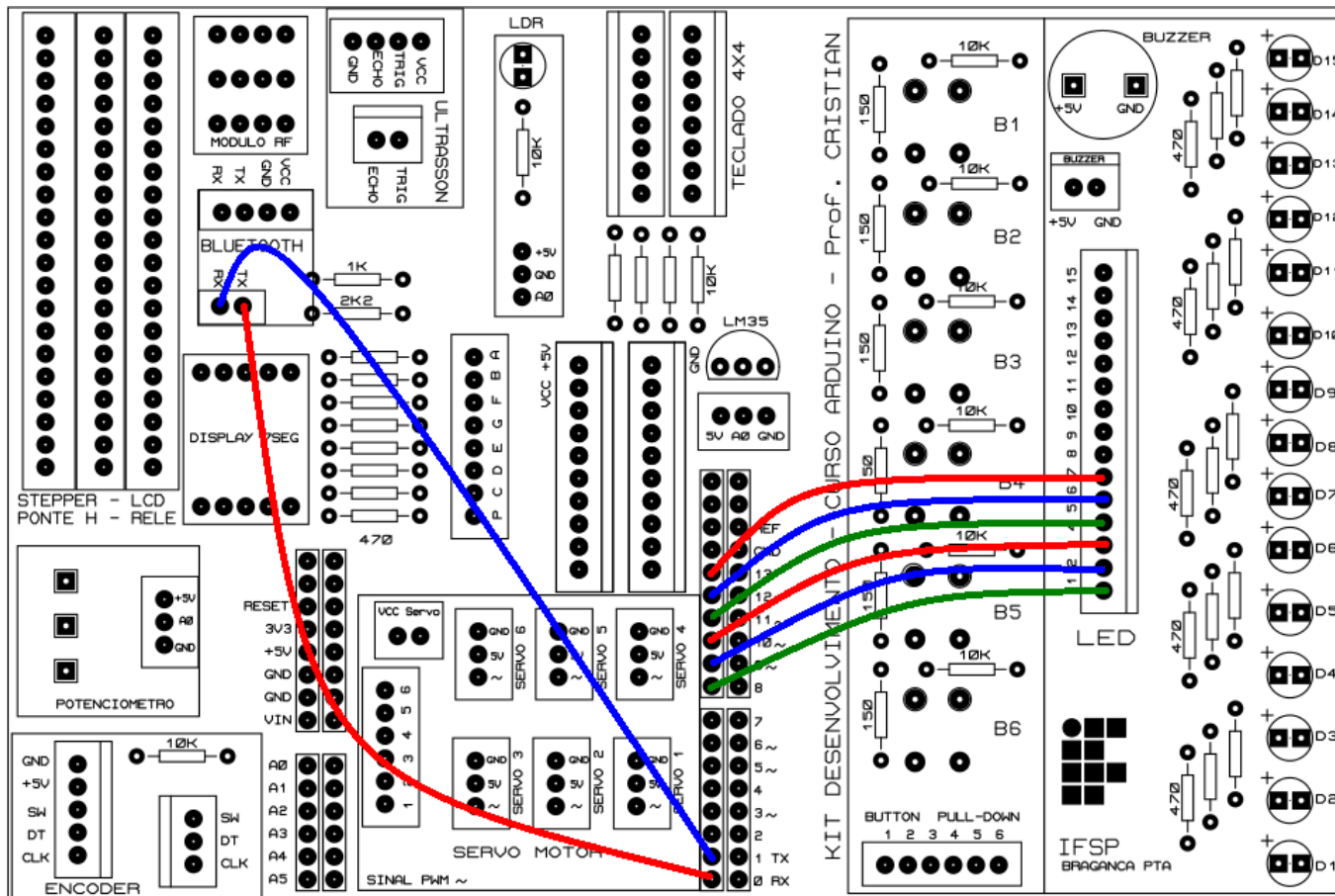
Prática 11 – Acionamento de varios LEDs com Celular via BlueTooth

Uma sugestão, é criar os botões para comanda os LEDs, inserindo os comandos correspondente a cada LED. Mas se preferir os comandos podem ser enviados diretamente via teclado.



Prática 11 – Acionamento de varios LEDs com Celular via BlueTooth

A ligação do circuito deve ser realizada como visto a seguir:



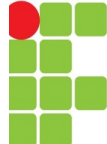
Prática 11

```
char comando;
int LED1 = 13;
int LED2 = 12;
int LED3 = 11;
int LED4 = 10;
int LED5 = 9;
int LED6 = 8;

void setup() {
  Serial.begin(9600);
  pinMode(LED1, OUTPUT);
  pinMode(LED2, OUTPUT);
  pinMode(LED3, OUTPUT);
  pinMode(LED4, OUTPUT);
  pinMode(LED5, OUTPUT);
  pinMode(LED6, OUTPUT);
}

void Apaga_LEDs(){
  digitalWrite(LED1, LOW);
  digitalWrite(LED2, LOW);
  digitalWrite(LED3, LOW);
  digitalWrite(LED4, LOW);
  digitalWrite(LED5, LOW);
  digitalWrite(LED6, LOW);
}
```

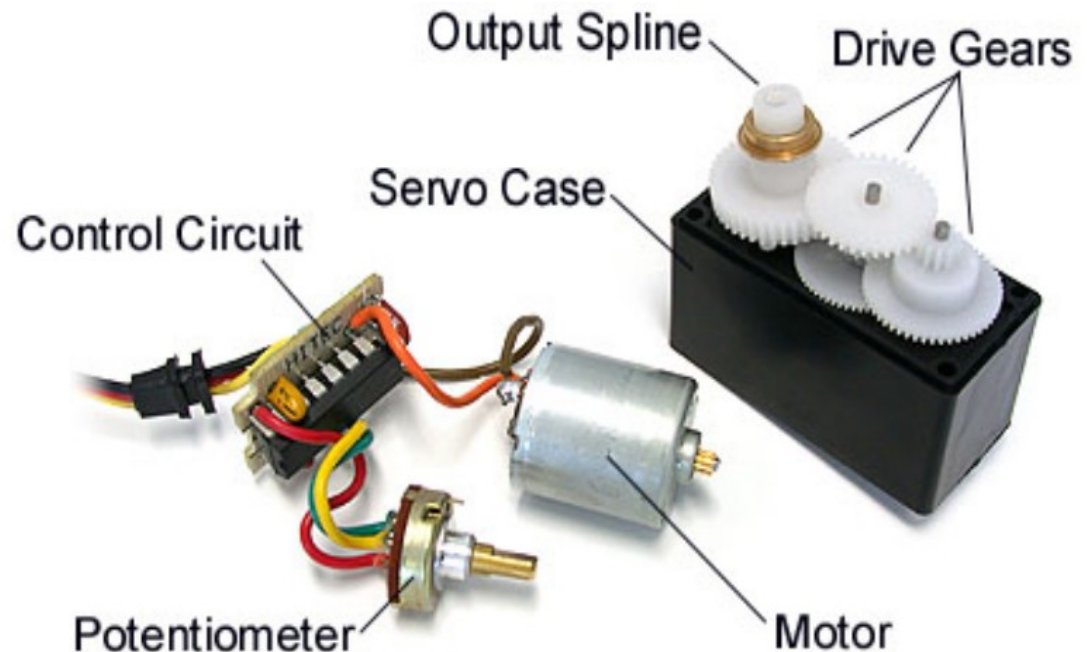
```
void loop() {
  // Enquanto houverem bytes disponíveis;
  while(Serial.available() > 0) {
    // Lê byte do buffer serial;
    comando=Serial.read();
  }
  if(comando=='A') {
    Apaga_LEDs();
    digitalWrite(LED1, HIGH);
  }
  if(comando=='B') {
    Apaga_LEDs();
    digitalWrite(LED2, HIGH);
  }
  if(comando=='C') {
    Apaga_LEDs();
    digitalWrite(LED3, HIGH);
  }
  if(comando=='D') {
    Apaga_LEDs();
    digitalWrite(LED4, HIGH);
  }
  if(comando=='E') {
    Apaga_LEDs();
    digitalWrite(LED5, HIGH);
  }
  if(comando=='F') {
    Apaga_LEDs();
    digitalWrite(LED6, HIGH);
  }
  if(comando=='X') {
    Apaga_LEDs();
  }
}
```



Servomotores

Os servomotores são pequenos dispositivos constituídos basicamente por:

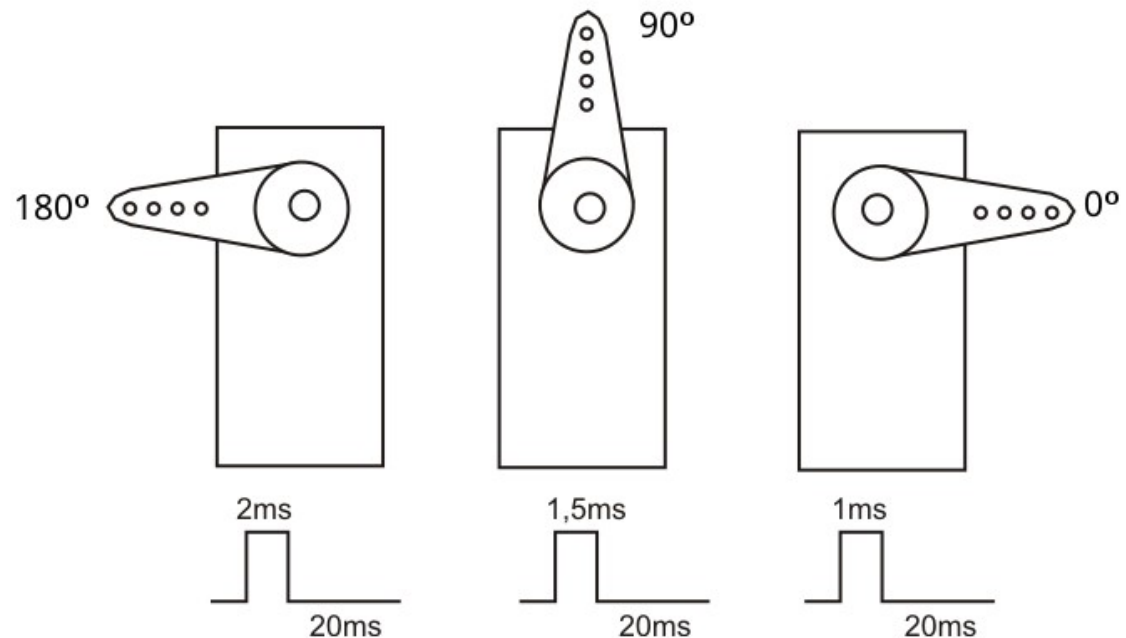
- Potenciômetro que roda com o eixo do servo;
- Circuito eletrônico de controle;
- Motor DC;
- Conjunto de engrenagens;
- Três fios exteriores de ligação.



Servomotores



Um servo é utilizado para controlar movimentos angulares, tipicamente, entre 0° e 180° pois não é mecanicamente capaz de rodar muito mais devido à existência de um potenciômetro e de um batente mecânico na engrenagem.



Servomotores



Controle de ângulo do eixo de um servo:

O ângulo do eixo dos servos é determinado pela duração do pulso (tempo **ON** e tempo **OFF**) que se aplica na entrada de comando.

Este pulso é um sinal PWM (Modulação por Largura de Pulso), sistema que consiste em gerar uma onda quadrada em que se varia a duração do pulso.

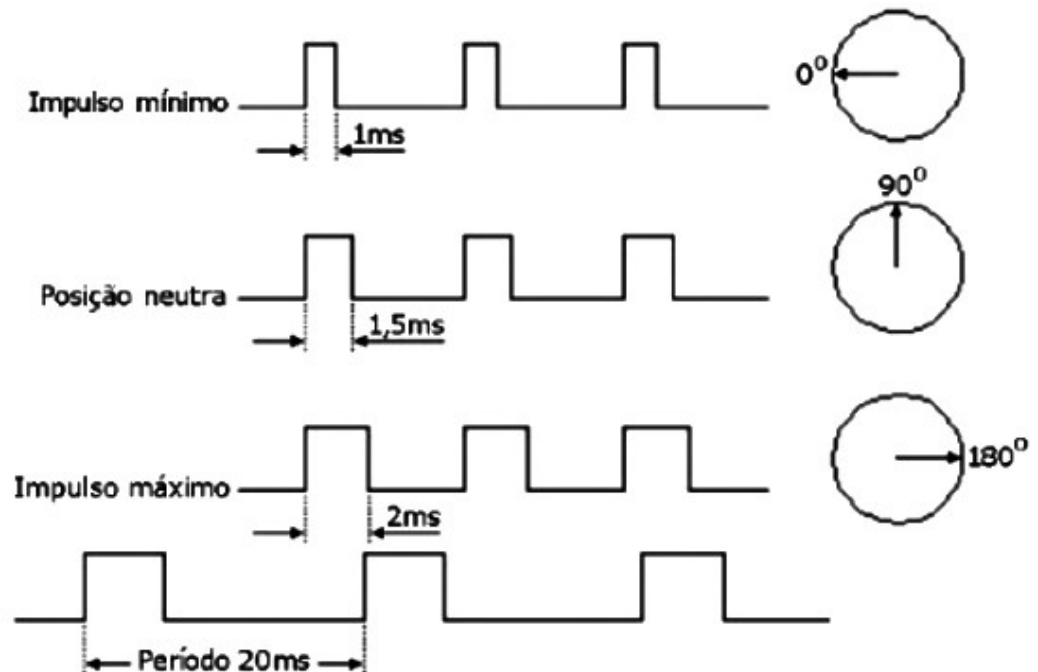
Servomotores



Controle do ângulo do eixo de um servo:

Para que o servo funcione corretamente, os impulsos devem ser aplicados a cada 20ms (50 Hz).

A duração do pulso
determina o
posicionamento
do servo.



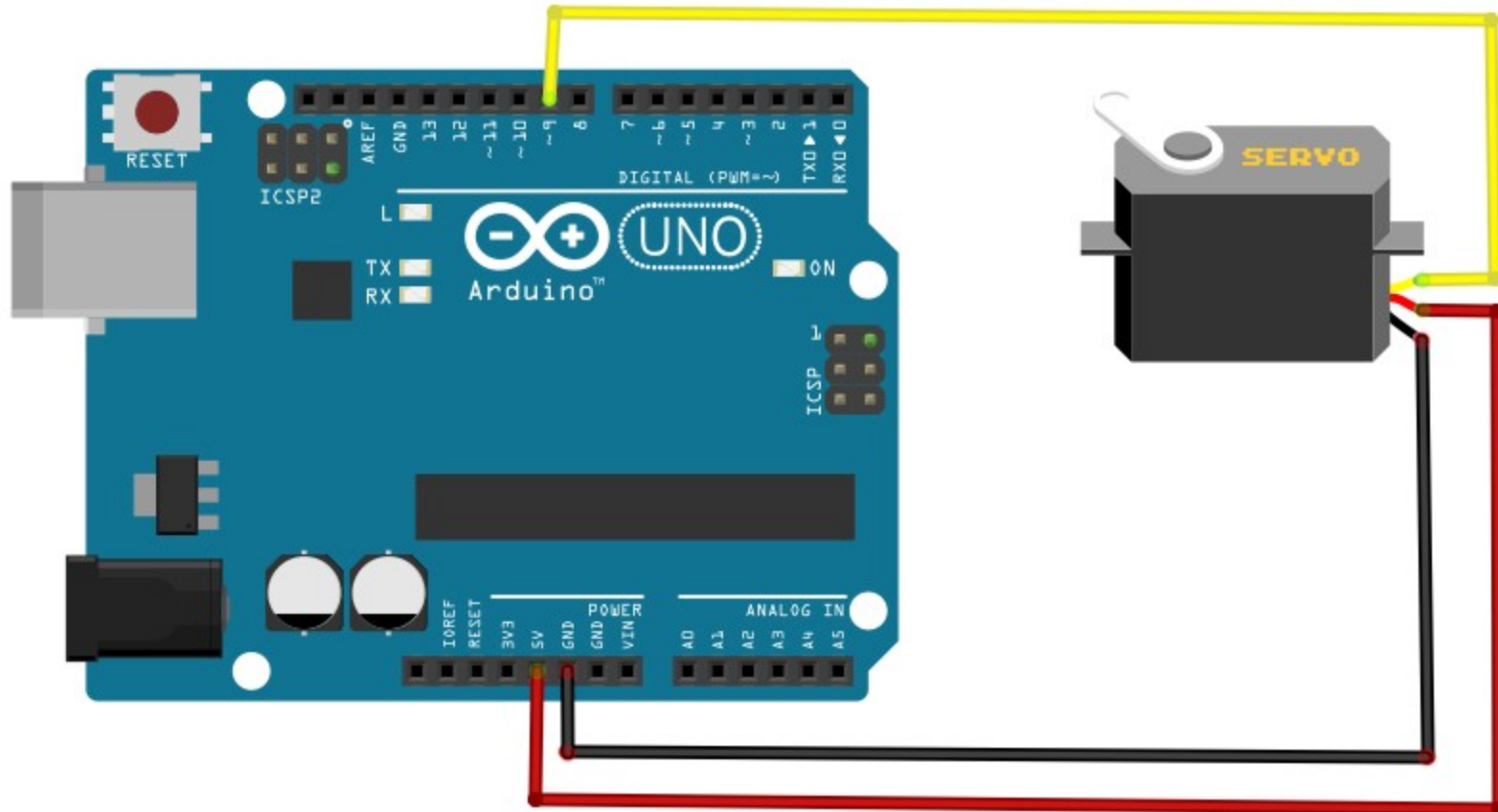
Servomotores

Em nossas práticas utilizaremos o Micro Servo 9g com força de 1,2 kg/cm no seu eixo.



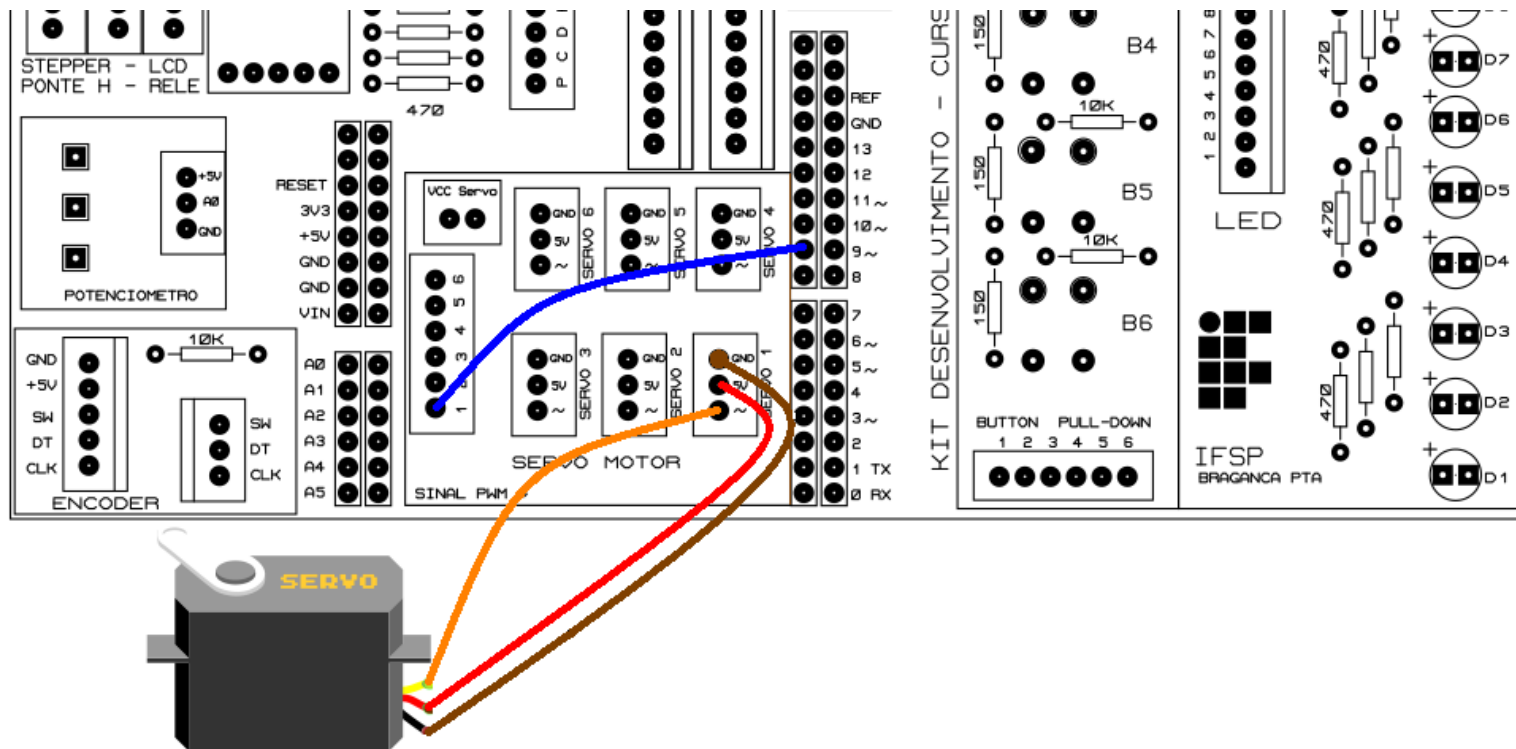
Prática 12 – Controle do servomotor, via Monitor Serial:

Devido ao seu baixo consumo de corrente elétrica, é possível ligar os servo motores direto no Arduino.



Prática 12 – Controle do servomotor, via Monitor Serial:

A ligação do circuito deve ser realizada como visto a seguir:



Prática 12 – Controle do servomotor, via Monitor Serial:

Escreva no monitor serial o ângulo que deseje que o servo execute entre 0° e 180°.

```
#include<Servo.h> // Inclui a biblioteca Servo.h

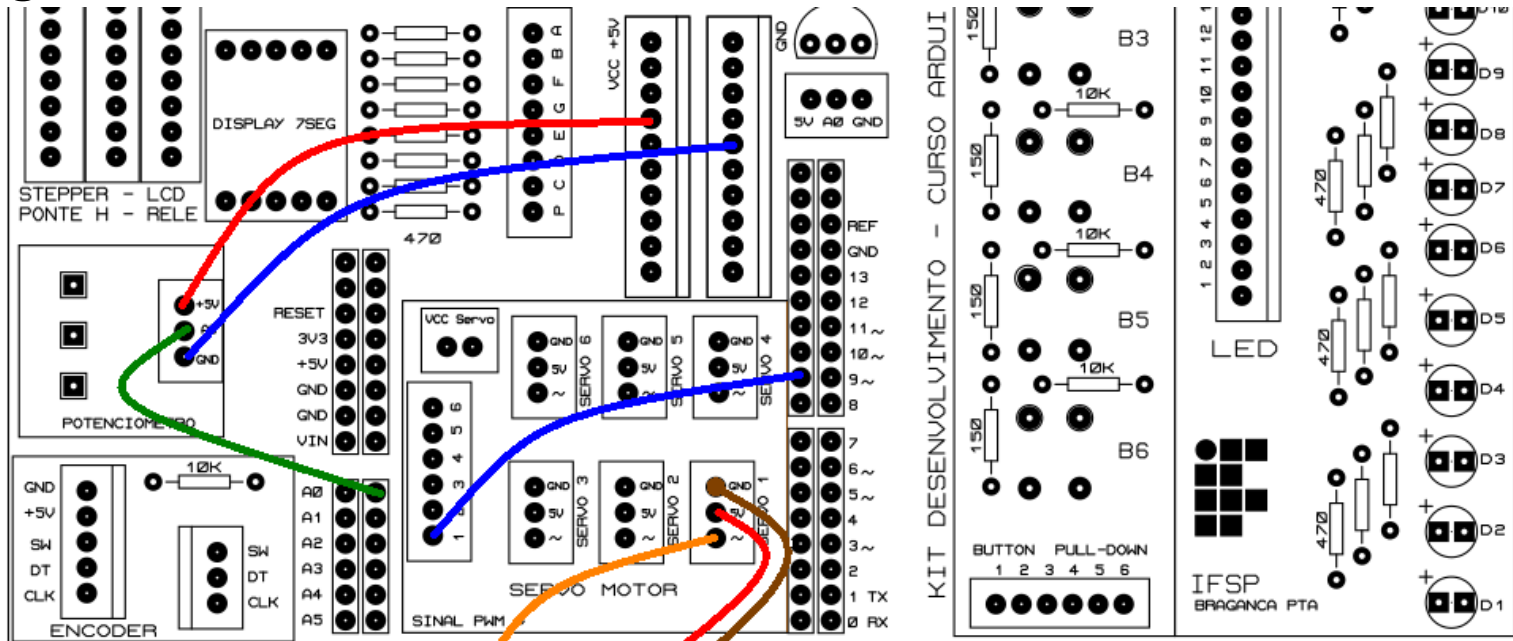
Servo eixo_X; // Cria objeto para controlar o servo base
int graus=90;

void setup(){
  Serial.begin(9600);
  eixo_X.attach(6); // Define que o servo esta ligado a porta 6
}

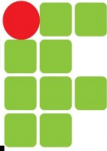
void loop(){
  if (Serial.available()>0){ // Se hover bits disponvel
    graus = Serial.parseInt(); // Retorna o primeiro número inteiro do buffer serial
    eixo_X.write(graus); // Move o eixo do servo, de acordo com o angulo
    Serial.print("Angulo = ");
    Serial.println(graus);
  }
}
```


Prática 13 – Controle do servomotor, por um potenciômetro:

A ligação do circuito deve ser realizada como visto a seguir:



Prática 13 – Controle do servomotor, por um potenciômetro:



```
#include<Servo.h> // Inclui a biblioteca Servo.h

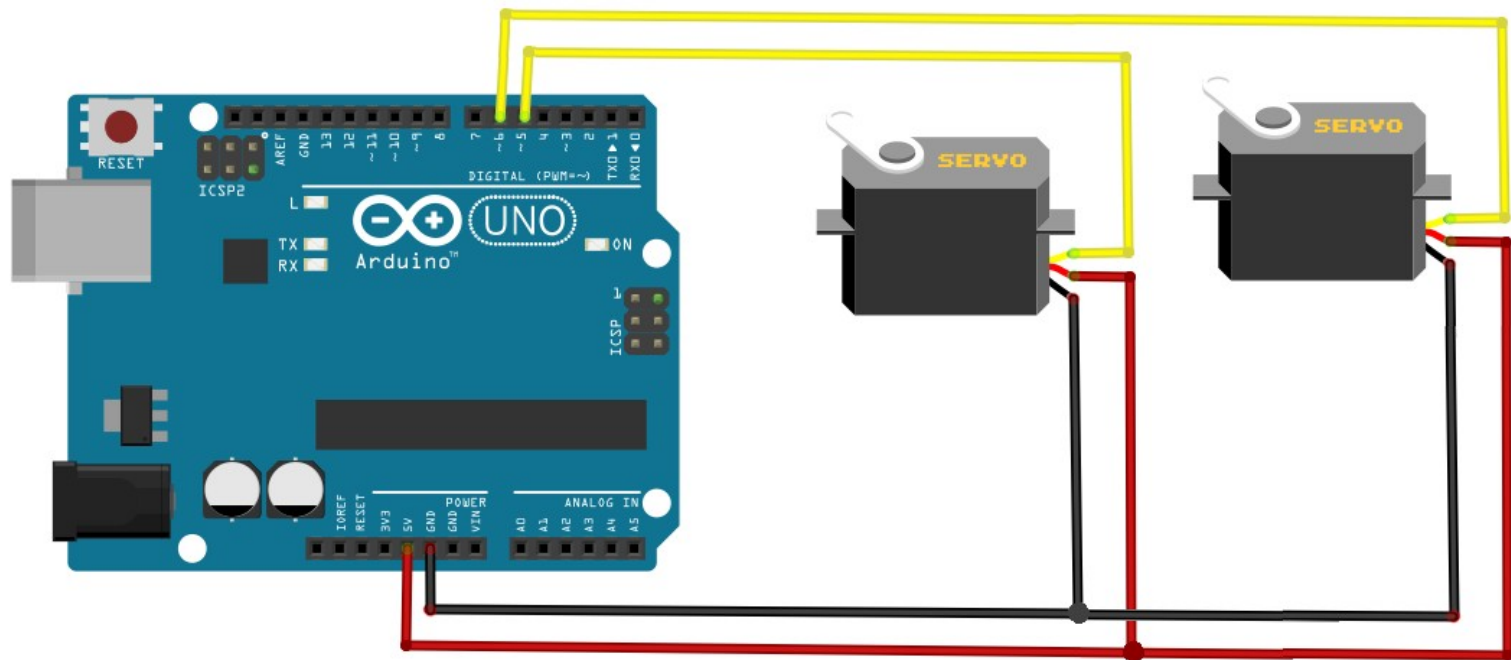
Servo eixo_X; // Cria objeto para controlar o servo base
int graus;
int pot;

void setup(){
  Serial.begin(9600);
  eixo_X.attach(6); // Define que o servo esta ligado a porta 6
}

void loop(){
  pot=analogRead(0); // Le o valor do potenciometro no pino A0
  graus=map(pot,0,1023,0,180); // mapea o valor lido entre 0 e 180 graus
  eixo_X.write(graus); // Move o eixo do servo, de acordo com o angulo
  Serial.print ("Valor Potenciometro = ");
  Serial.print (pot);
  Serial.print ("      Angulo = ");
  Serial.println (graus);
  delay(15);
}
```

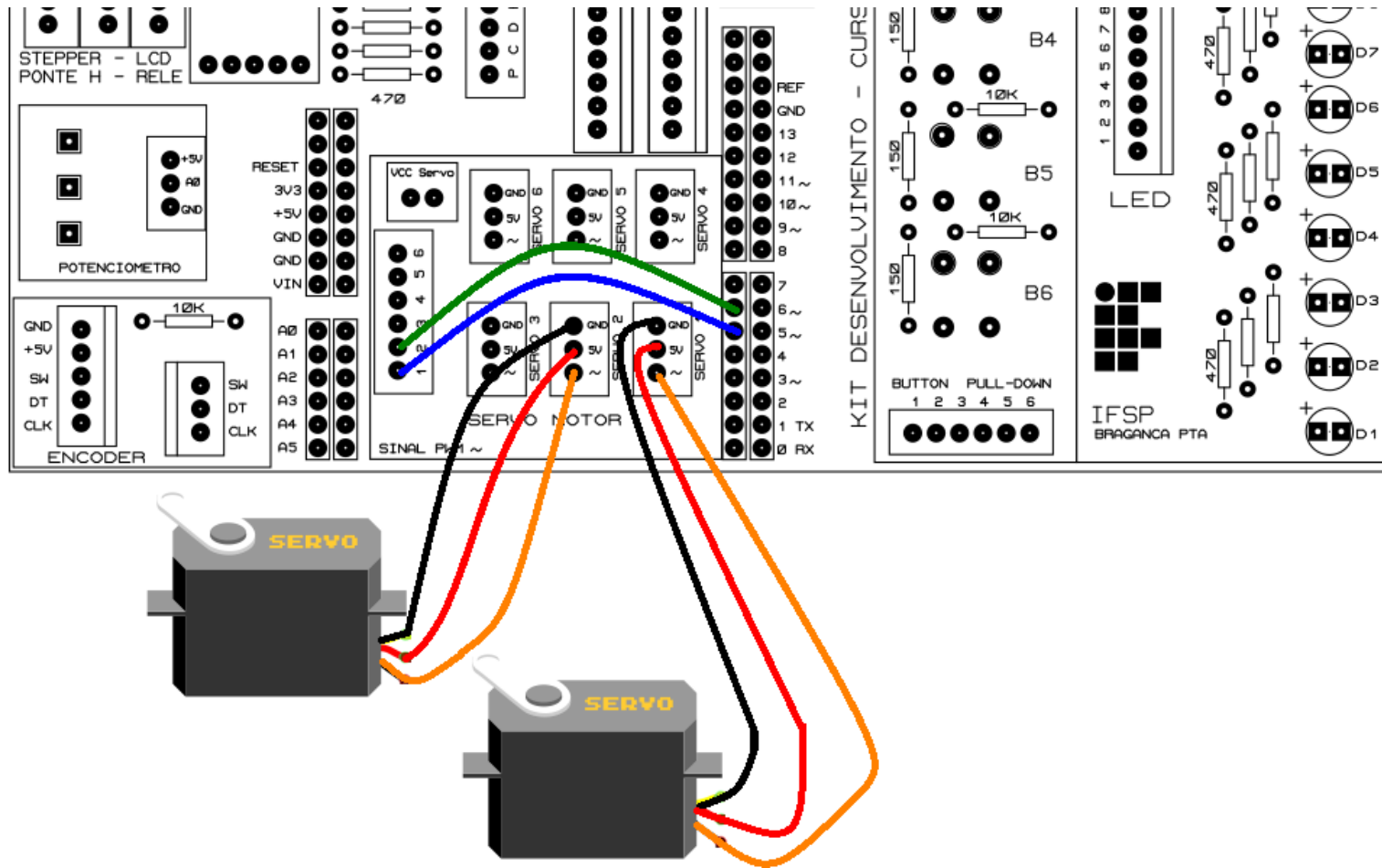
Prática 14 – Controle de dois servomotor simultâneo

No programa a seguir temos o **eixo X** e **eixo Y** representado pelos dois servo, onde controlamos o ângulo de cada um, enviando o valor e o eixo correspondente juntos. Exemplo: 180x, 90y, etc...



Prática 14 – Controle de dois servomotor simultâneo

A ligação do circuito deve ser realizada como visto a seguir:



```
#include <Servo.h>
```

```
Servo eixo_x; // Cria objeto para controlar o servo no eixo X
```

```
Servo eixo_y; // Cria objeto para controlar o servo no eixo Y
```

```
int graus=90;
```

```
int valory = 90;
```

```
int valorx = 90;
```

```
int pos=0;
```

```
int letra;
```

```
void setup() {
```

```
  Serial.begin(9600);
```

```
  eixo_x.attach(6); // Define que o servo esta ligado a porta 6
```

```
  eixo_y.attach(5); // Define que o servo esta ligado a porta 5
```

```
}
```

```
void loop() {
```

```
  if (Serial.available() > 0) { // Se hover bits disponvel
```

```
    graus = Serial.parseInt(); // Retorna o numero inteiro do buffer serial
```

```
    letra = Serial.read(); // lê inteiro recebido pela comunicação série
```

```
    if (letra == 'x') { // Se a letra apos angulo for x
```

```
      eixo_x.write(graus); // Move o eixo X do servo, de acordo com o angulo
```

```
      delay(15);
```

```
    }
```

```
    Serial.print("Posicao do eixo X em = ");
```

```
    Serial.print(graus);
```

```
    Serial.println(" graus");
```

```
    if (letra == 'y') { // Se a letra apos angulo for y
```

```
      eixo_y.write(graus); // Move o eixo Y do servo, de acordo com o angulo
```

```
      delay(15);
```

```
    }
```

```
    Serial.print("Posicao do eixo Y em = ");
```

```
    Serial.print(graus);
```

```
    Serial.println(" graus");
```

```
  }
```

```
}
```


Prática 14 – Controle de dois servomotor simultâneo



Desafio - Faça o controle dos dois eixos X e Y dos servo motores do programa anterior pelo Bluetooth, com ângulos pré-definido .

Segue sugestão da tela do controle Bluetooth ao lado.

