

Programación de código en Arduino para movimiento de Servomotores en lenguaje HDL

```
// importamos libreria servo
#include <Servo.h>

// Definimos los objetos
Servo servo1;
Servo servo2;

// Definimos los parámetros para evaluar, max y min posición
int minPulse = 600; // minimum servo position, us (microseconds)
int maxPulse = 2400; // maximum servo position, us

// User input for servo and position
int userInput[3]; // Entrada del usuario para el servo y la posición, 3 bytes
int startbyte; // Iniciar byte, comienza la lectura de entrada
int servo; // Que servo?
int pos; // servo ángulo 0-180
int i; // iteración

// LED on Pin 13 for digital on/off demo
int ledPin = 13;
int pinState = LOW;

void setup()
{
    // Pin digital de cada servo
    servo1.attach(9, minPulse, maxPulse);
    servo2.attach(3, minPulse, maxPulse);

    // Pin indicador
    pinMode(ledPin, OUTPUT);
}
```

```

// Abrir conexion en serie, 9600 baud
Serial.begin(9600);
}
void loop()
{
    // Esperar la entrada en serie(min 3 bytes en buffer)
    if (Serial.available() > 2) {
        // Leer el primer byte
        startbyte = Serial.read();
        //Si es realmente el primer byte (255) ...
        if (startbyte == 255) {
            // ... a continuación obtener los siguientes dos bytes
            for (i=0;i<2;i++) {
                userInput[i] = Serial.read();
            }
            // Primer byte = servo para mover?
            servo = userInput[0];
            // Segundo byte = que posicion?
            pos = userInput[1];
            // Comprobacion de errores
            if (pos == 255) { servo = 255; }
            // Asignar una nueva posición
            switch (servo) {
                case 1:
                    servo1.write(pos); // mover servo1 a la 'pos'
                    break;
                case 2:
                    servo2.write(pos);
                    break;
            }
        }
    }
}

```

```

// LED indicador

case 99:

  if (pos == 180) {
    if (pinState == LOW) { pinState = HIGH; }
    else { pinState = LOW; }
  }

  if (pos == 0) {
    pinState = LOW;
  }

  digitalWrite(ledPin, pinState);
  break;
}
}
}
}
}

```

Programación de código en Python para movimiento de servomotores por medio de interfaz gráfica

```

#Importaciones necesarias...
from Tkinter import *
import tkMessageBox
import serial

usbport = 'COM3'

ser = serial.Serial(usbport, 9600, timeout=1)

#Creación de la ventana
ventana = Tk()

imagen1=PhotoImage(file="servo.gif")
label3=Label(ventana,image=imagen1).place(x=1, y=1)

```

#Función mover

```
def mover():

    n1=int(caja1.get())
    n2=int(caja2.get())

    if(0<=n2<=180):
        ser.write(chr(255))
        ser.write(chr(n1))
        ser.write(chr(n2))

    else:
        tkinterMessageBox.showinfo("ERROR","El angulo debe estar entre 0 y 180
grados")

    print mover.get()
```

#Titulo de la ventana

```
ventana.title("MOVER CAMARA")
```

#Dimensiones (ancho,alto,posicion x,posicion y)

```
ventana.geometry("700x300+700+200")
```

#Creación de la etiqueta para el primer cuadro de texto (Servomotor)

```
var1 = StringVar()
var1.set("NUMERO DEL MOTOR:")
label1 = Label(ventana,textvariable=var1,height = 2)
label1.pack()
```

#Creación de una caja de texto para el número de servomotor

```
numServomotor=StringVar()
caja1=Entry(ventana,bd=4,textvariable=numServomotor)
caja1.pack()
```

#Creación de la etiqueta para el segundo cuadro de texto (Angulo)

```
var2 = StringVar()
var2.set("ANGULO:")
label2 = Label(ventana,textvariable=var2,height = 2)
label2.pack()
```

#Creacion de otra caja de texto para el número del ángulo

```
numAngulo=StringVar()
caja2=Entry(ventana,bd=4,textvariable=numAngulo)
caja2.pack()
```

#Boton para mover cámara

```
boton1 = Button(ventana, text = "MOVER CAMARA", command = mover,width=15)  
boton1.pack()
```

#Cargar la ventana

```
ventana.mainloop()
```

Interfaz en python

