

Trabajo 4 : Luzómetro

Asignatura : Bitbloq y Arduino

Profesor : Alberto Valero

Curso : Experto Universitario en Robótica, Programación e Impresión 3D

Universidad UNIR

**Fecha : 18 de diciembre
de 2016**



1 DESCRIPCIÓN

Esta actividad consiste en construir un luzómetro que nos indique si es:

- de día (plena luz)
- atardecer o amanecer (media luz)
- de noche (oscuridad)

El luzómetro se debe calibrar antes de su modo de funcionamiento normal. El proceso debe ser:

1. Al encender el luzómetro debemos calibrarlo. Para ello,
 - usando un potenciómetro llevamos la aguja a la posición de luz e iluminamos el sensor de luz.
 - usando un potenciómetro llevamos la aguja a la posición de media luz e iluminamos parcialmente el sensor de luz.
 - usando un potenciómetro llevamos la aguja a la posición de oscuridad y oscurecemos el sensor de luz.
 - Tanto las posiciones de la aguja como la cantidad de luz se calculan en este proceso y no pueden estar escritos numéricamente en los bloques, son resultado del proceso de calibración.
2. Una vez calibrado , el luzómetro debe mover la aguja a la posición adecuada según la cantidad de luz.

2 ELEMENTOS UTILIZADOS EN LA ACTIVIDAD

Material :

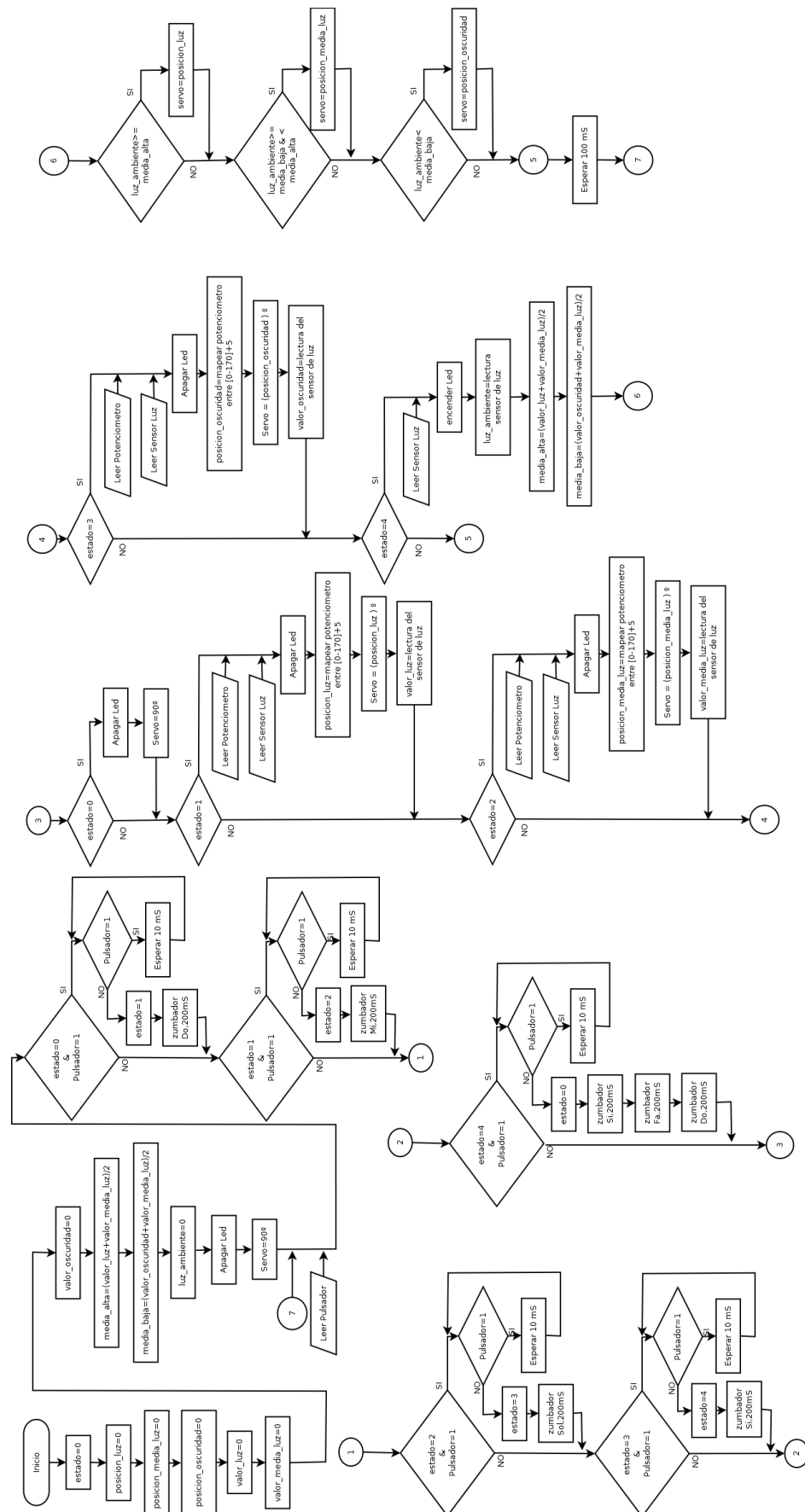
- 1 Placa Bq Zum Core
- 1 Portapilas

- 1 Zum bloq Pulsador
- 1 Zum bloq LED , verde
- 1 Zum bloq Sensor de Luz
- 1 Zum bloq Potenciómetro
- 1 Zum bloq Zumbador
- 1 Mini Servo 0-180º modelo ES08AI
- 16 tornillos M3x10 mm rosca madera
- Pegamento , atornilladores plano y de estrella
- Piezas impresas 3D con PLA en color naranja para la tapa trasera y mesa frontal, y negro para la caja central del Luzómetro y la aguja indicadora .
- Imágenes recortadas en papel para decorar el frontal.

Software :

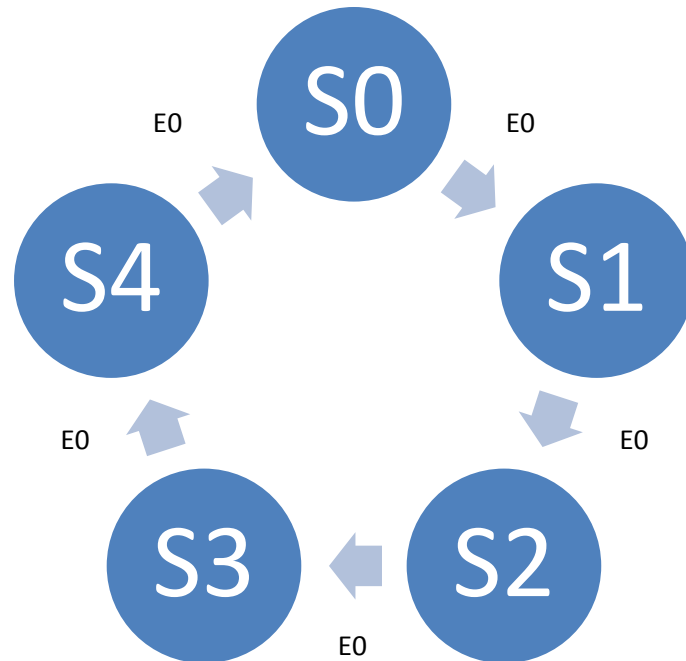
- Bitbloq para realizar la programación del semáforo
- FreeCAD para diseñar las piezas impresas
- Cura 15.04.6 para convertir a gcode
- Dia para realizar el diagrama de flujo

TRABAJO 4 : LUZOMETRO



3.1 EXPLICACION DEL DIAGRAMA DE FLUJO DEL PROGRAMA

Este programa lo he enfocado como si fuese una **máquina de estados** , cuyos estados y eventos son los siguientes :



Descripción de los Estados :

S0 = Luzómetro en estado OFF

S1 = Calibración Estado Luz Nivel Alto

S2 = Calibración Estado Media Luz

S3 = Calibración Estado Oscuridad

S4 = Luzómetro en estado Normal Automático

Descripción de los Eventos :

E0 = Pulsar el Botón

El programa comienza definiendo una serie de variables globales :

estado : almacena el estado en el que se encuentra nuestra máquina

posición_luz : guarda el valor en grados asignado al servo durante la calibración al que se tiene que mover la aguja cuando exista un nivel alto de luz.

posición_media_luz : guarda el valor en grados asignado al servo durante la calibración al que se tiene que mover la aguja cuando exista un nivel medio de luz .

posición_oscuridad : guarda el valor en grados asignado al servo durante la calibración al que se tiene que mover la aguja cuando exista un nivel bajo de luz .

valor_luz : guarda el valor recogido del sensor de luz durante la calibración a nivel alto de iluminación.

valor_media_luz : guarda el valor recogido del sensor de luz durante la calibración a nivel medio de iluminación.

valor_oscuridad : guarda el valor recogido del sensor de luz durante la calibración a nivel bajo de iluminación.

media_alta : Calcula el valor medio de las variables valor_luz y valor_media_luz .

media_baja : Calcula el valor medio de las variables valor_media_luz y valor_oscuridad .

luz_ambiente : Guarda el valor recogido por el sensor de luz durante el estado 4 de funcionamiento automático .

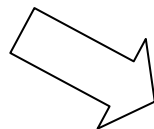
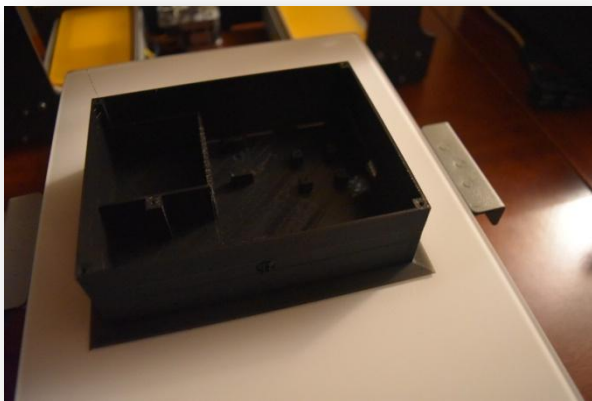
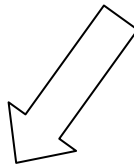
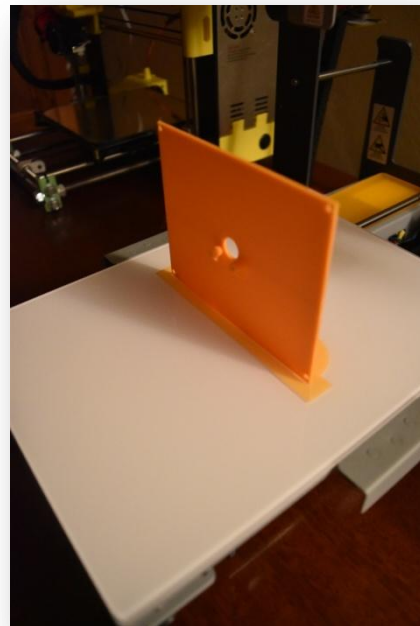
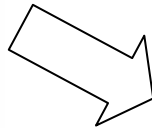
A continuación se lee el estado del pulsador y en función de ese valor “verdadero “ o “ falso “ en que se encuentre **el pulsador** y del **estado** en que se encuentre en ese momento mi sistema (0:4) , el programa va atravesando una serie de condicionales que primero establecen la **LÓGICA** de cambio de estados , para después y en función de dicho estado indicarle que **ACCIONES** ha de ejecutar , dependiendo del estado en que se encuentre :

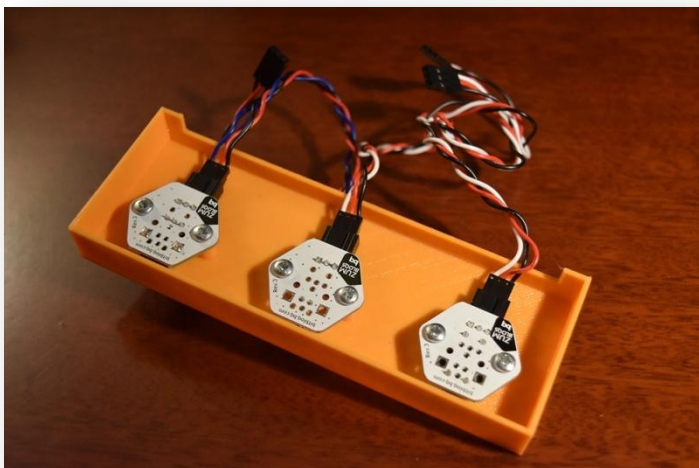
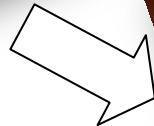
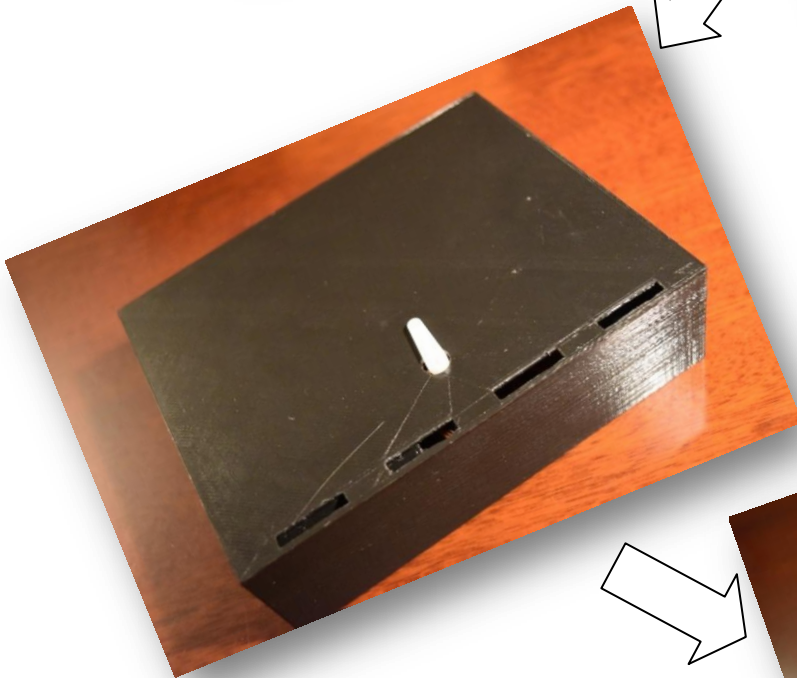
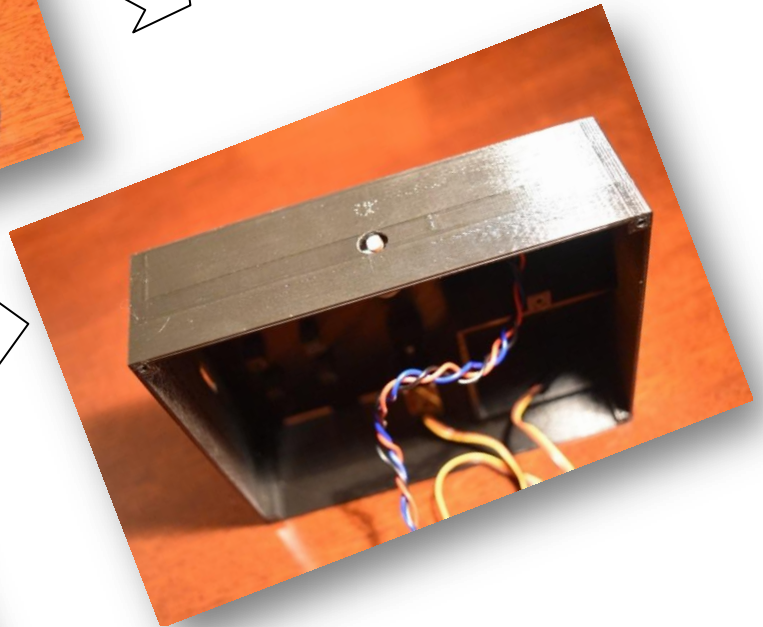
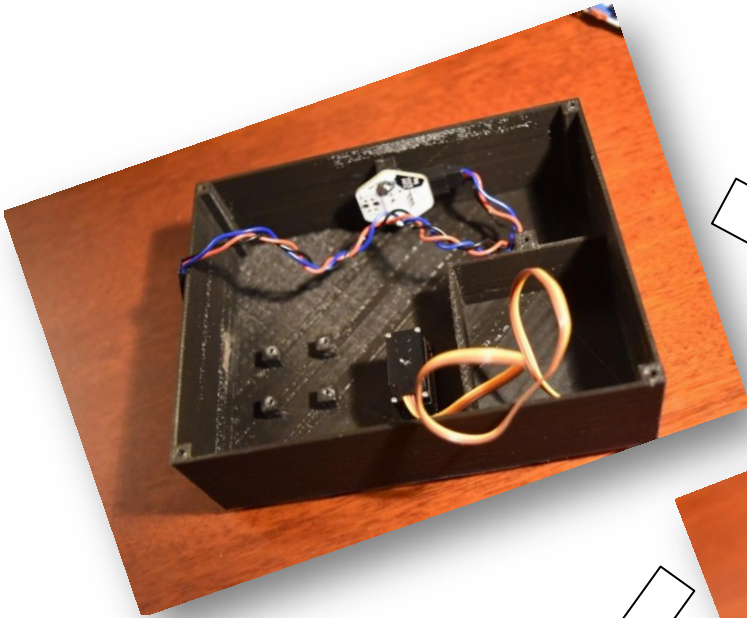
El estado 0 es de inactividad OFF

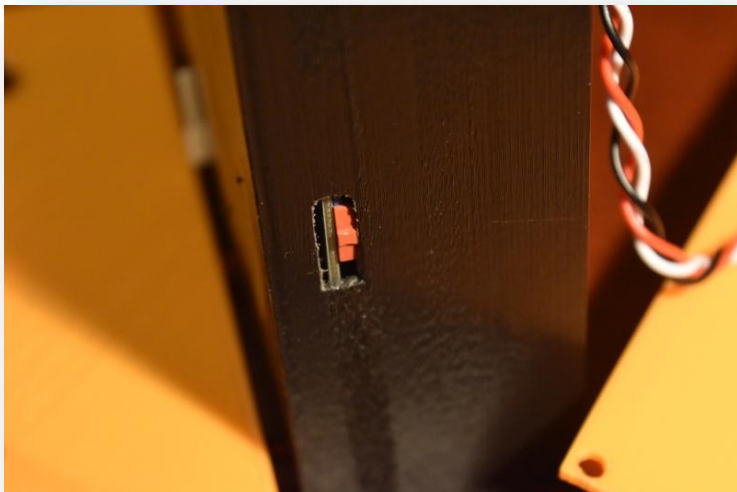
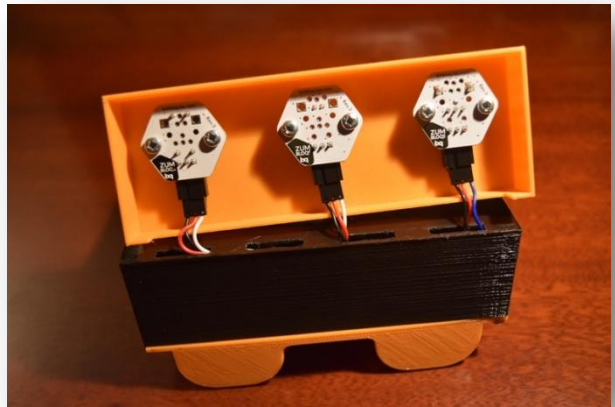
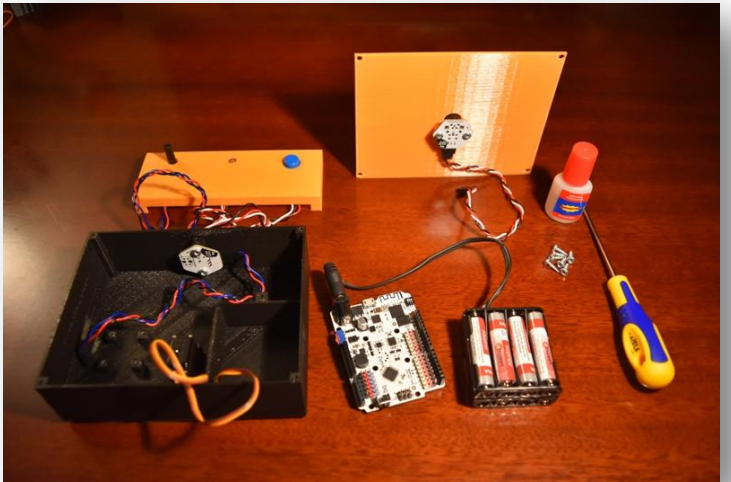
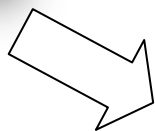
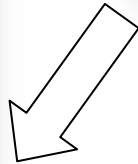
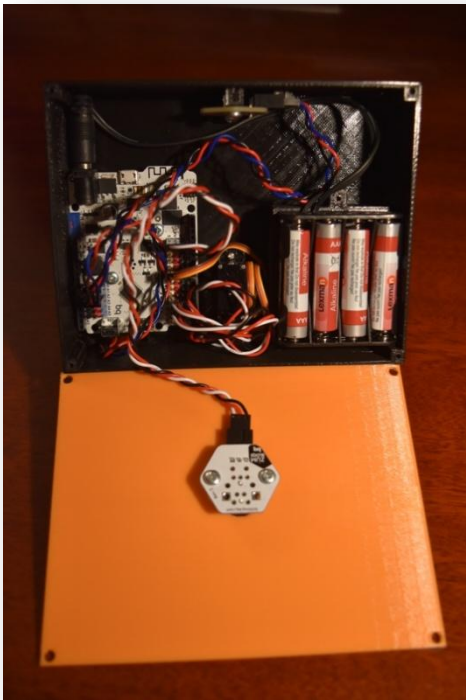
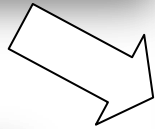
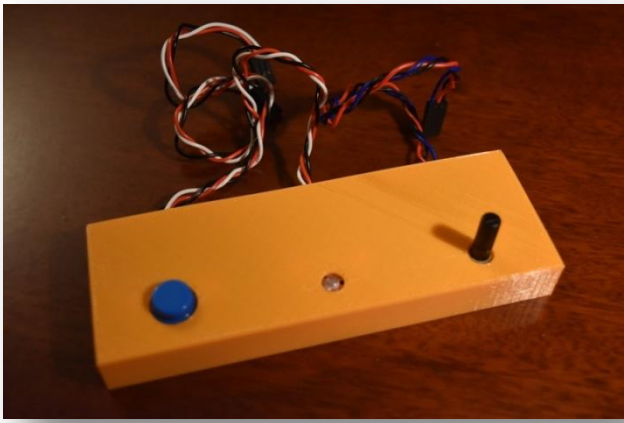
En los estado 1 , 2 y 3 se pretende calibrar el aparato para que memorice tanto la posición del servo como los niveles de luz que identifican las tres situaciones posibles : Luz alta , media y baja .

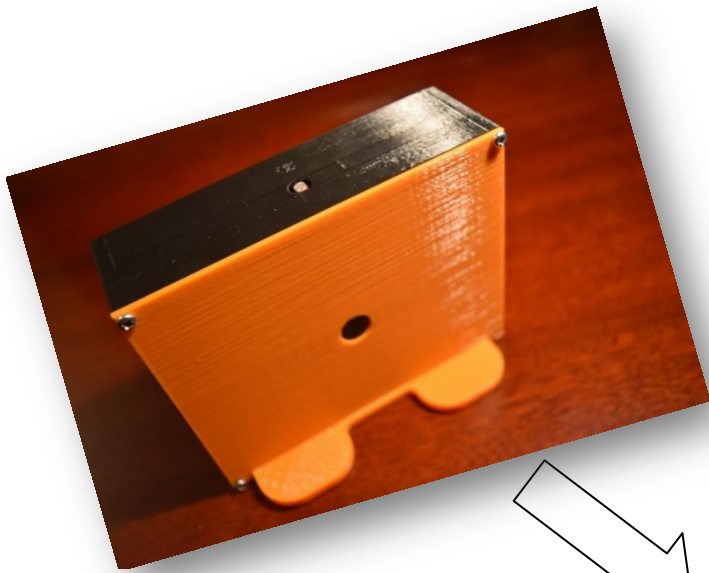
En el estado 4 , de funcionamiento autónomo , dependiendo de la luz recibida por el sensor de luz el programa mueve el servo a la zona correspondiente del panel frontal. Para ello , dividimos las lecturas posibles del sensor en tres intervalos que se clasifican según si son menores que el valor `media_baja` , entre `media_baja` y `media_alta` , o superiores a `media_alta` .

4 SECUENCIA DE FOTOS DE LA REALIZACIÓN PRÁCTICA DE LA ACTIVIDAD









5. ALGUNOS DETALLES DE LAS PIEZA DISEÑADAS EN FREECAD Y LOS ARCHIVOS GENERADOS PARA PODER IMPRIMIR EN MI IMPRESORA 3D .

