

CONNECT-BOT

Manual de instrucciones

Cosmina Ioana Nicu - 1454988 Eric Lozano Férriz - 1456846
Antoni Rodríguez Villegas - 1462476 Miquel Martin Ezquerra - 1458791

Mayo 2019

Contenido

1	Material	4
1.1	Obligatorios	4
1.2	Adicionales	4
2	Diseño 3D	5
2.1	Pilares principales	5
2.2	Carrito	5
2.3	Tobogán	6
2.4	Botonera	7
2.5	LCD	8
2.6	Fichas	8
2.7	Cámara	9
3	Esquema de conexiones	10
3.1	Stepper NEMA 17	10
3.2	Servo Motor	11
3.3	Botones y LEDS	11
3.4	LCD	12
3.5	Cámara	12
4	Estructura final	13
5	Librerías necesarias	15
6	Código	16
7	Empezar a jugar	19
8	Robot final	20

Manual de instrucciones

Connect-bot es un robot que compite contra ti en el mítico juego de mesa Connect-4. Consta de hasta 3 niveles distintos de dificultad y un algoritmo que proporciona todo un reto. Además, se comunica contigo a través de mensajes mostrados en su pantalla y tiene una personalidad un tanto "traviesa".

En este documento procederemos a explicar paso a paso como reproducir a la perfección nuestro proyecto de robótica: el robot Connect-Bot.

1 Material

Lo primero que se debe tener bien claro es el material que se necesitará para la creación de nuestro robot.

Se debe tener en cuenta que, si alguno de los componentes no son los mismos especificados a continuación, no encajará en el diseño 3D.

Los clasificaremos en obligatorios y adicionales.

1.1 Obligatorios

- RaspberryPi 3 B+
- RaspberryPi Camera V2
- Stepper NEMA 17
- Controlador stepper A4988
- Servo-motor
- 1 LED RGB y un LED rojo
- 2 pulsadores
- Final de carrera
- Correa
- Poleas
- Power Supply
- LCD
- Filamento de impresión 3D
- Tornillos
- Cables
- Condensador
- Pistola de pegamento
- Juego de herramientas
- Tablero Connect-4
- 3 Ruedas locas
- Resistencias

1.2 Adicionales

- Tubo Termoretráctil
- Loctite SpuerGlue
- Cartulina negra
- Madera de 40x50

2 Diseño 3D

Para poder empezar con nuestro robot deberemos tener listas todas las piezas 3D, las cuales encontraremos en las referencias del documento [1]. Una vez impresas todas, procederemos a unirlas.

2.1 Pilares principales

La estructura principal consta de dos pilares que se unen mediante un encaje como podemos observar en la figura 1a y figura 1b. Además, como veremos más adelante, en el pilar derecho irá el Stepper NEMA 17.



Figure 1: Estructura principal

2.2 Carrito

Para poder crear el carrito se debe enganchar primero de todo la pieza 2a con la pieza 2b con Loctite Superglue o algún pegamento resistente.

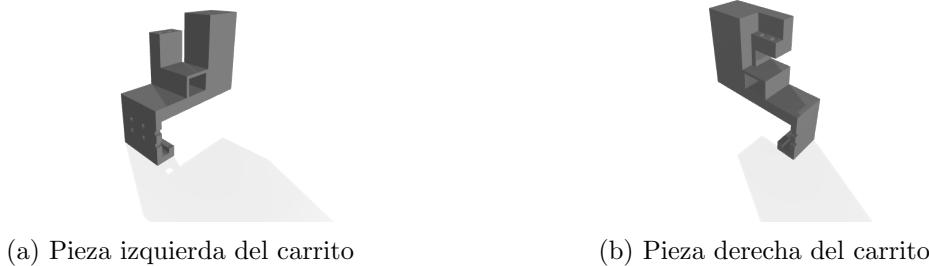
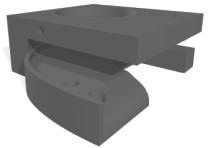


Figure 2: Parte inferior del carrito

A continuación, en la parte superior se encontrarán las siguientes dos piezas que se muestran: la pieza 3a, la cual contendrá el rotor de las fichas 3b y el cual irá atornillado a la pieza 3a.



(a) Pieza superior del carrito



(b) Rotor de fichas

Figure 3: Parte superior del carrito

Una vez acabado, se añade el tubo contenedor de fichas, obteniendo así el siguiente resultado:

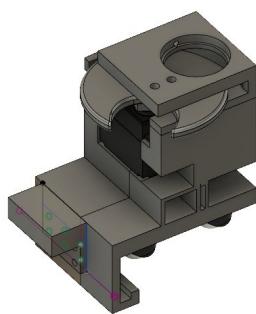


Figure 4: Carrito final

2.3 Tobogán

Por lo que respecta al tobogán, consiste en dos piezas 3D que se pueden enganchar fácilmente con Loctite Super glue entre ellas.



(a) Pieza izquierda del tobogán



(b) Pieza derecha del tobogán

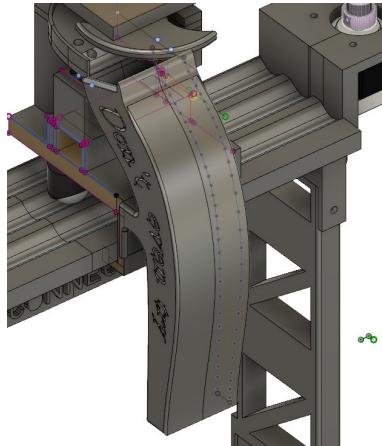
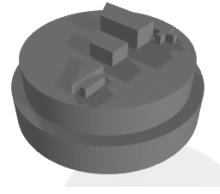


Figure 6: Tobogán

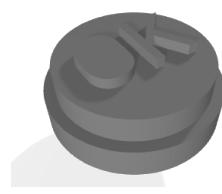
Una vez listas, se atornilla la parte trasera, a la parte delantera del carrito como se puede observar en la figura 6.

2.4 Botonera

Para poder controlar el nivel de dificultad del juego y diferentes aspectos que se verán más adelante, se tendrá una botonera. Para ello las piezas necesarias será los dos botones que se muestran en 7a y 7b.



(a) Botón de selección



(b) Botón OK

Figure 7: Botones

El resultado final de la botonera se puede observar en la figura 8.



Figure 8: Botonera

2.5 LCD

Por otra parte, nos encontramos con una estructura donde poder colocar la LCD. Para ello se atornilla la LCD a la pieza.

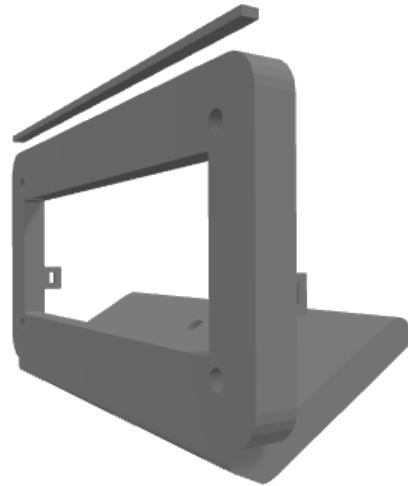


Figure 9: LCD

2.6 Fichas

Como todos sabemos, el juego connect-4 viene con sus respectivas fichas para jugar pero, en este caso, se deben imprimir 21 fichas como las de la figura 10.

Esto es debido a que las fichas deben ir bien encajadas en el rotor encargado de lanzarlas y que, además, el programa necesitará un color que no genere ningún conflicto a la hora de detectar el tablero (como veremos más adelante).



Figure 10: Ficha del robot

2.7 Cámara

Para la Raspberry Pi Camera V2 se ha diseñado un soporte por dos razones importantes:

- Se debe tener sumo cuidado al manejar la cámara, ya que esta es muy delicada.
- Debe estar siempre fija por razones de código.

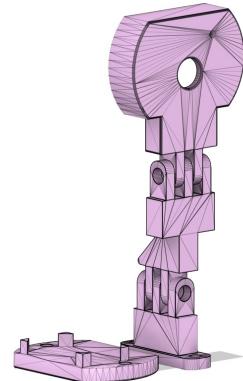


Figure 11: Estructura para Raspberry Pi Camera

3 Esquema de conexiones

Una vez acabado de montar la estructura 3D pasaremos a explicar el esquema de conexiones que debe seguirse.

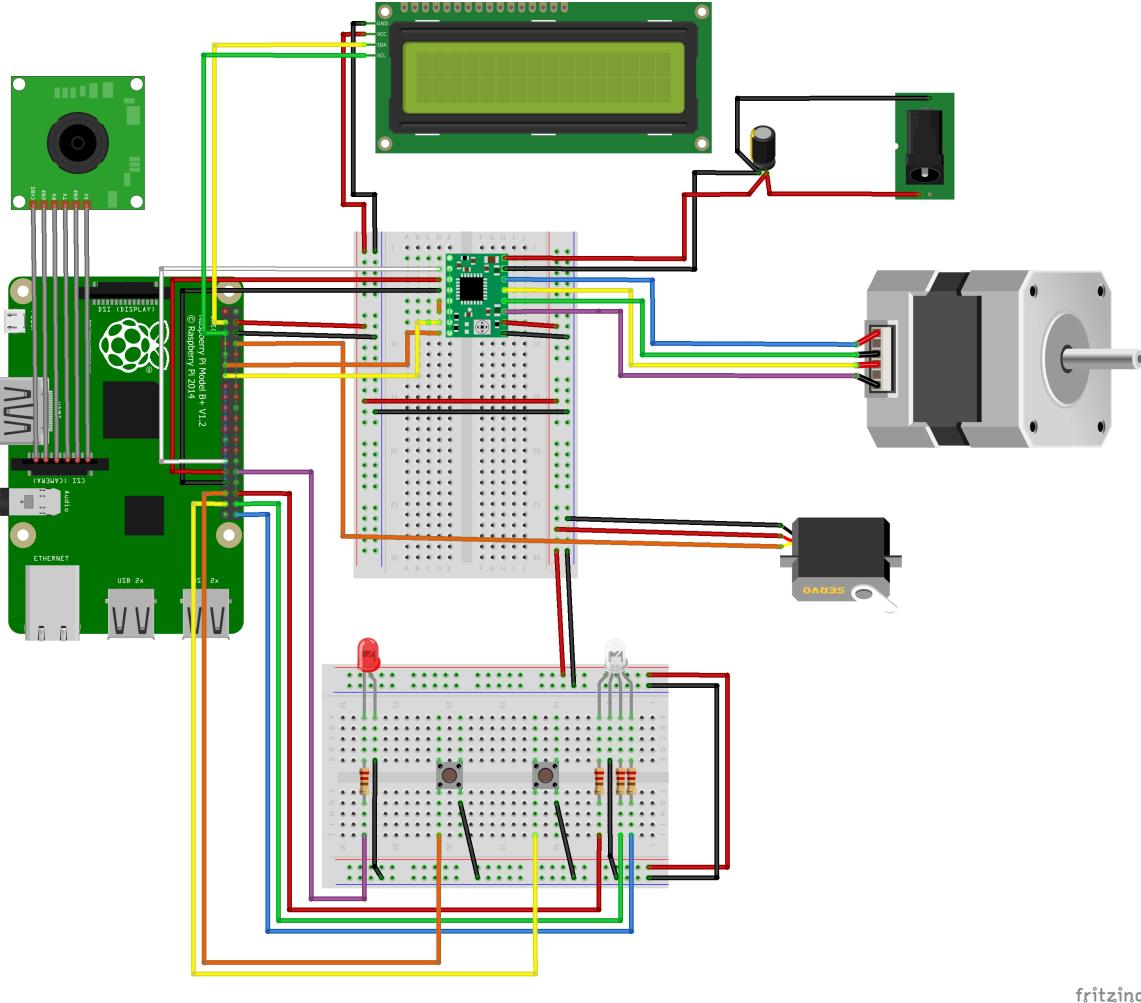


Figure 12: Esquema de conexiones

3.1 Stepper NEMA 17

Probablemente, la parte más complicada en cuanto a conexiones sea el Stepper.

Dicho componente debe estar colocado en la parte superior del pilar derecho de la estructura (véase figura 1b). Para ello estará atornillado.

Primeramente, deberemos conectar una *Power Supply* que nos proporcione los 12 V. necesarios para utilizar el NEMA 17:

1. Cable negro PowerSupply conectado al GND del controlador.

2. Cable rojo PowerSupply conectado al voltaje del controlador.

El condensador debe ser de **100 μ F y 63V** e irá conectado entre el voltaje y el GND del condensador, tal y como se muestra en el esquema.

El Stepper debe estar conectado con su respectivo controlador: el A4988. Se debe seguir explícitamente el esquema de conexión de la figura 12.

1. Cable azul, amarillo, verde y lila del Stepper: se conectan respectivamente a los pines 2B, 2A, 1A y 1B del controlador.
2. Cable rojo y negro del controlador conectado al voltaje y GND de la RaspberryPi.
3. Cable naranja del controlador correspondiente al pin DIR del controlador conectado al pin **GPIO 17** de la RaspberryPi.
4. Cable amarillo correspondiente al pin STEP del controlador conectado al pin **GPIO 27** de la RaspberryPi.
5. Cable naranja en un puente debe conectar SLEEP y RESET del controlador.
6. Cable blanco, rojo y negro corresponden a los pines MS1, MS2, MS3 del controlador: conectados a los pines **GPIO 5,6,13** respectivamente de la RaspberryPi.

3.2 Servo Motor

El servo-motor estará colocado en el carrito atornillado a la parte superior de este.

Para realizar la conexión del servo se debe conectar:

1. Cable rojo conectado a voltaje de la RaspberryPi.
2. Cable negro conectado al GND de la RaspberryPi.
3. Cable naranja conectado al pin **GPIO 14** de la RaspberryPi.

3.3 Botones y LEDs

Para ello utilizaremos otra Protoboard de la misma medida que la botonera de la figura 8 ya que deberá encajar exactamente con los dos botones y los dos agujeros para los LEDs.

De izquierda a derecha en la protoboard tenemos lo siguiente: LED rojo, Botón OK, Botón Nivel y LED RGB.

El LED rojo debe tener una resistencia en la patilla corta ya que corresponde al color rojo que irá conectado al pin **GPIO 12**. Por otro lado, la patilla larga se debe conectar a un GND.

Seguidamente tenemos los dos botones que siguen el mismo esquema de conexión:

1. Patillas de la izquierda conectadas a un pin GPIO de la RaspberryPi (El botón OK al pin **GPIO 19** y el botón de nivel al pin **GPIO 26**).
2. Patillas de la derecha a un GND.

En la derecha de la protoboard tendremos el LED RGB y de izquierda a derecha las conexiones son:

1. Primera patilla LED rojo al pin **GPIO 16** de la RaspberryPi.
2. Segunda patilla a un GND de la RaspberryPi.
3. Tercera patilla, LED verde al pin **GPIO 20** de la RaspberryPi.
4. Cuarta patilla, LED azul al pin **GPIO 21** de la RaspberryPi.

Se debe tener conectado las resistencias a las tres patillas del LED.

3.4 LCD

La LCD tiene incorporado un módulo I2C por lo que hace más fácil su conexión:

1. Cable negro conectado al GND.
2. Cable rojo conectado al voltaje de la RaspberryPi.
3. Cable amarillo correspondiente al SDA conectado al **pin SDA** de la RaspberryPi.
4. Cable verde correspondiente al SCL conectado al **pin SCL** de la RaspberryPi.

3.5 Cámara

Para conectar la cámara a nuestra RaspberryPi se debe encontrar el **puerto J3** que cuenta con una serigrafía en el PCB en la que se lee "CAMERA" (En el esquema de la figura 12 se puede observar). Se debe conectar de manera que, la parte azul de la cámara quede mirando a los puertos de USB de la RaspberryPi tal y como mostramos en la figura 13.



Figure 13: Conexión RaspberryPi Camera V2

4 Estructura final

Una vez que ya tenemos las piezas 3D listas y todos los componentes conectados, procedemos a fijar la estructura del juego en la madera de 40x50cm.

Para ello, se atornilla tanto a la estructura como a la madera en los dos lados de los pilares.

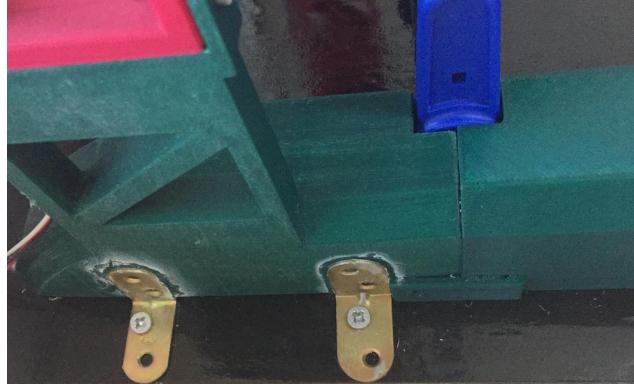


Figure 14: Estructura atornillada a la madera

En cuanto a la botonera, la RaspberryPi y la LCD se fijan con Superglue.

Por otro lado, tenemos la Protoboard con el controlador A4988, la conexión del final de carrera que, con la madera que ha sobrado, se aprovecha para crear una caja donde guardarla. Esto lo podemos observar en la figura 15.

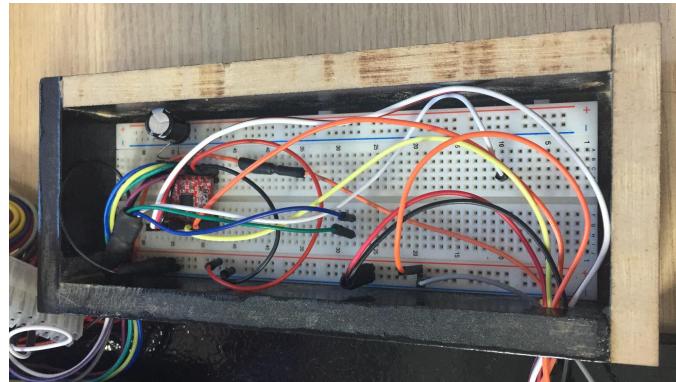


Figure 15: Caja de madera para la Protoboard

Esta caja irá enganchada en un lateral de la madera y llevará una tapa la cual también servirá para depositar las fichas del jugador.

Por otro lado, para poder tener todos los cables asegurados, juntos y sin que molesten a la hora de jugar, se debe unir las conexiones con tubo termo retráctil consiguiendo así un acabado como el siguiente:

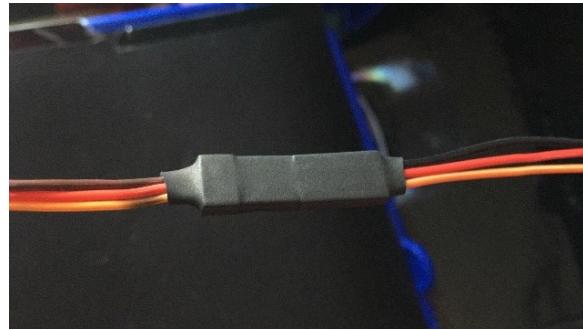


Figure 16: Cable con acabado termo retráctil

Una vez acabados todos los retoques, el robot se puede observar en la siguiente figura:

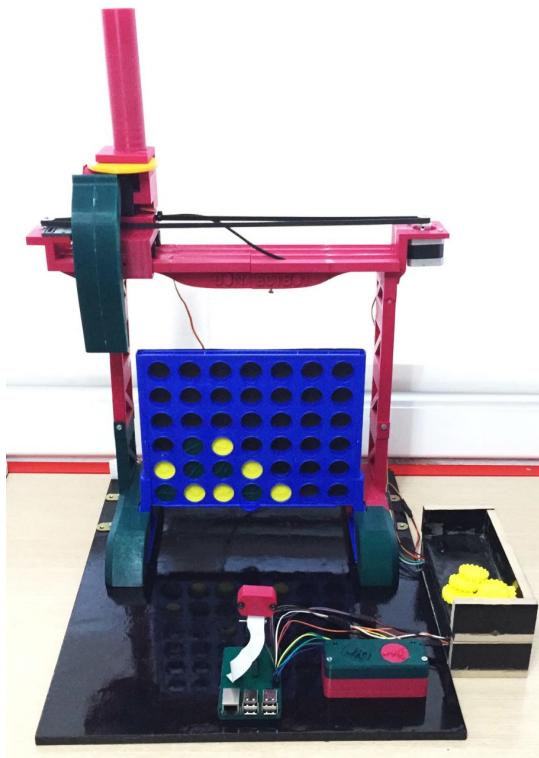


Figure 17: Robot Connect-4

5 Librerías necesarias

Antes de empezar a explicar el programa en sí, se debe tener todas las librerías instaladas.

Como todos sabemos, la RaspberryPi es como un mini ordenador por lo que, lo primero de todo y más importante, es la instalación de su Sistema Operativo. Para ello se instalará **Raspbian**.

Una vez que se tenga instalado Raspbian podemos proceder a instalar *Python* y *OpenCV* (véase guía de instalación [6]).

A continuación, se deberá instalar librerías importantes como:

- NumPy
- GpioZero
- ColorZero
- Matplotlib

Por último, recomendamos la instalación de la plataforma TeamViewer para un control remoto desde nuestro portátil.

6 Código

Una vez que se tiene todo lo referente a la estructura del robot construida, los componentes conectados y todo a nuestro gusto, procedemos a ejecutar el juego en sí.

Para ello primero de todo se debe entender la arquitectura Software que tenemos

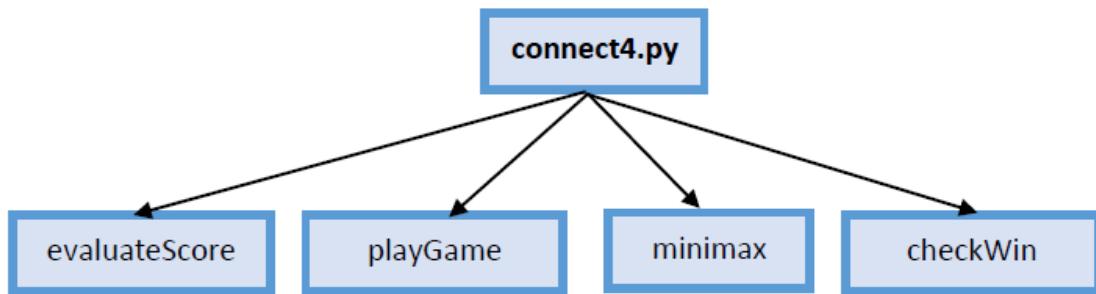


Figure 18: Estructura software

El programa principal del robot es el *connect4.py* en el cual podemos encontrar distintas funciones que en su conjunto conforman el algoritmo del juego. Además, también existen llamadas a algunos componentes del robot en ciertos puntos claves durante el enfrentamiento humano-máquina.

Las funciones principales son las siguientes:

- ***evaluateScore***

Esta función es la encargada de evaluar el tablero mediante el cálculo de una heurística definida.

- ***playGame***

Ejecuta el bucle principal en el que se lleva a cabo todas las operaciones de juego: tirar ficha, control de turnos, calcular columna a jugar, etc.

- ***minimax***

Algoritmo que se encarga de encontrar la mejor jugada posible según el estado del tablero y la dificultad de la partida.

- ***checkWin***

Función que comprueba si se ha finalizado la partida o no. Además, en caso de haberse acabado, comprueba quien es el ganador o si ha habido empate.

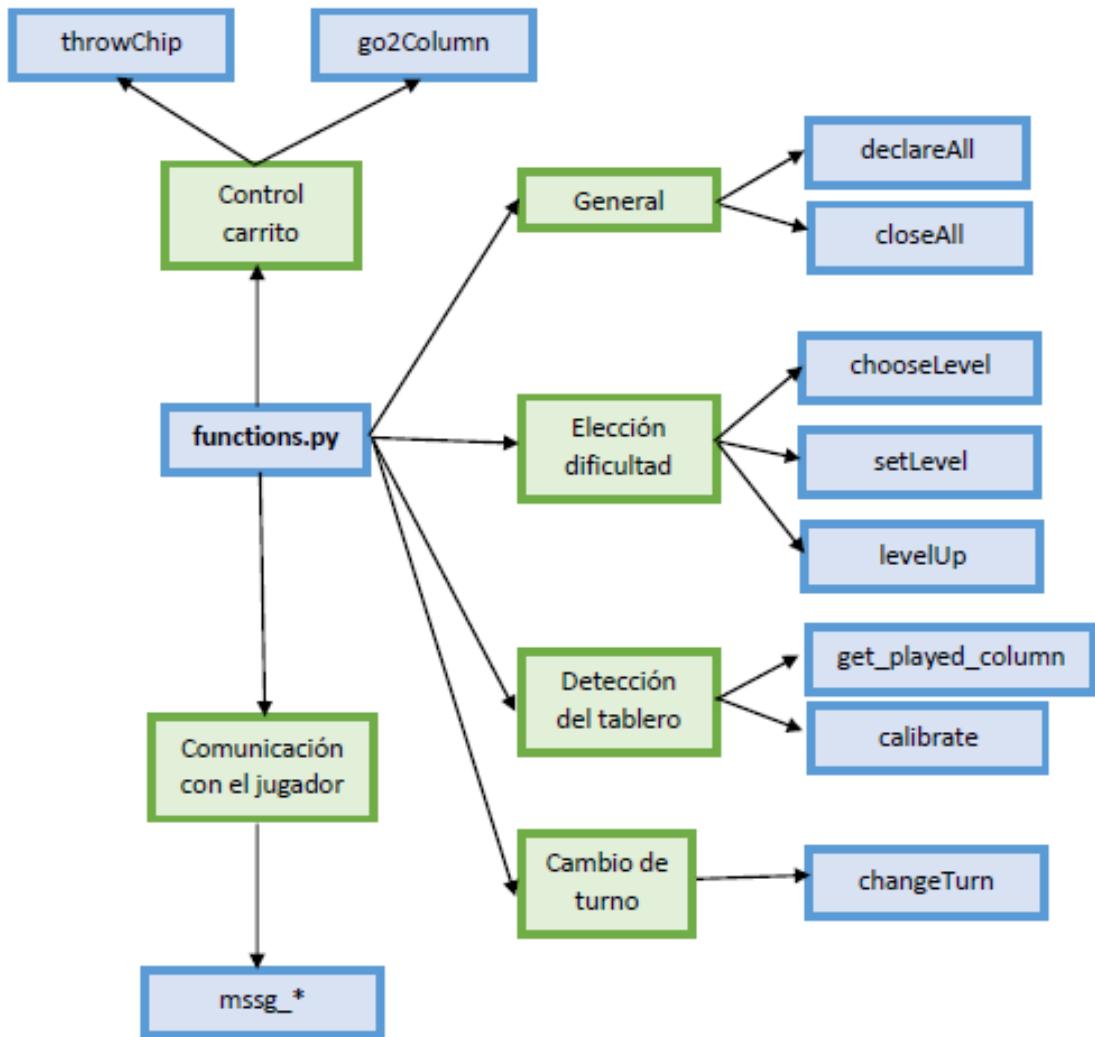


Figure 19: Estructura software

El archivo *functions.py* contiene todas las funciones encargadas de interactuar con los componentes del robot, para ello la librería que se ha utilizado ha sido GPIOZERO [5].

Las funciones principales son las siguientes:

- ***throwChip***

Esta función se encarga de rotar el servo, lanzando así la ficha al tobogán.

- ***go2column***

Función encargada de llevar el carrito hasta la columna especificada para a continuación poder colocar la ficha en el tablero. Una vez colocada dicha ficha, el carrito regresa a la posición *home*.

- ***declareAll***

Declaración de todas los componentes utilizados, asignando a cada componente los pines de las *Raspberry Pi* especificados en el esquema de conexiones (véase sección 3).

- ***closeAll***

Se procede a liberar todos los pines utilizados por los componentes terminando así la conexión con estos.

- ***chooseLevel***

Se muestra un mensaje en la pantalla de la LCD comunicando al jugador que debe escoger un nivel de dificultad. Para esto último, llama a la función *setLevel* explicada a continuación.

- ***setLevel***

Permite al jugador escoger la dificultad de la partida mediante dos botones: uno de selección y otro para finalizar.

- ***levelUp***

Función encargada de incrementar el nivel de dificultad cada vez que se pulsa el botón de selección.

- ***get_played_column***

Se realiza una fotografía del tablero mediante la cual obtendremos la columna en la que el jugador ha colocado su ficha.

- ***calibrate***

Función encargada de calibrar la cámara para evitar problemas de iluminación.

- ***changeTurn***

Se enciende/apaga el led de turno con la finalidad de indicar así los distintos turnos de cada jugador.

- ***mssg_****

Estas funciones muestran mensajes aleatorios por la pantalla LCD según el estado de la partida y el momento.

7 Empezar a jugar

Una vez se han configurado todos los componentes e instalado todo lo necesario, será muy sencillo ejecutar el juego y enfrentarse así al robot en una partida al *connect-4*.

A continuación se describen de manera simple los pasos a seguir para empezar a jugar contra el robot:

1. Encender la *Raspberry Pi*.
2. Conectar la *Power Supply*.
3. Asegurarse que estén todas las conexiones de igual manera que las especificadas en la sección 3.
4. Ejecutar el archivo *connect4.py*.
5. Seguir los pasos indicados en la pantalla LCD.
6. Asumir la derrota ante el gran Connect-bot.

8 Robot final

A continuación se mostrará una imagen final de cómo debería quedar el robot y un link a un vídeo con una representación de éste [7].

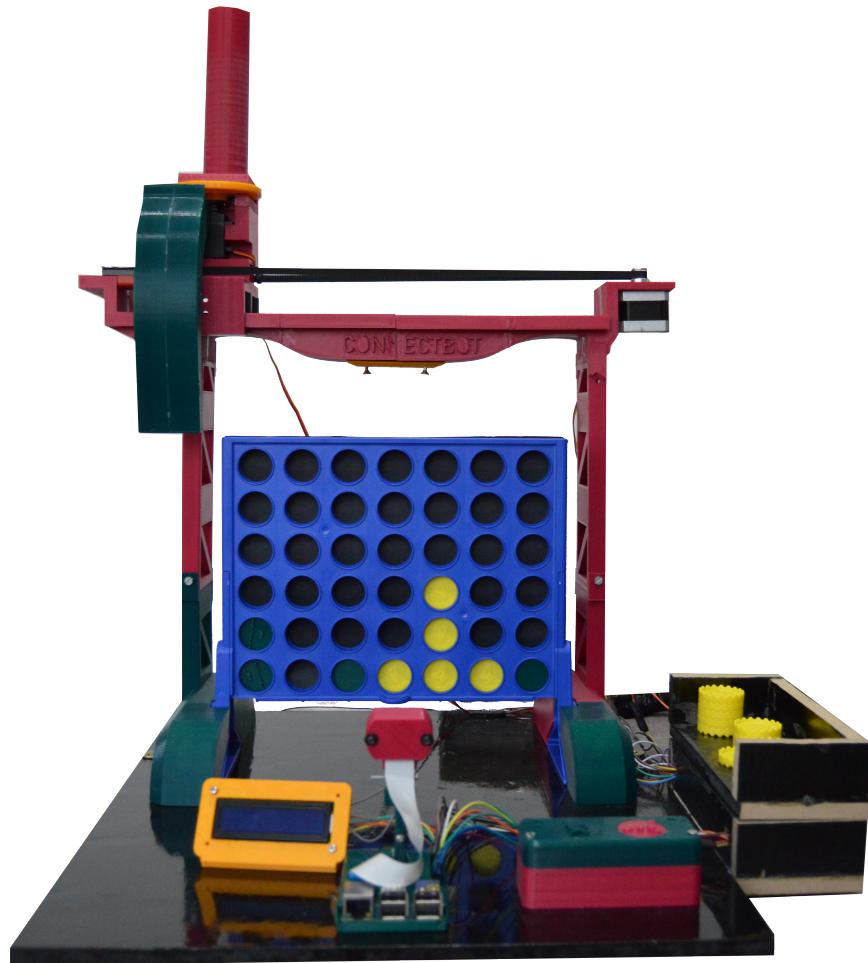


Figure 20: Robot final

Referencias

- [1] Piezas 3D y código en GitHub
<https://github.com/Connectbott/Connect-Bot>
- [2] Visualización del diseño 3D.
<https://a360.co/2P12U1F>
- [3] Especificaciones RaspberryPi 3B+
<https://www.raspberrypi.org/products/raspberry-pi-3-model-b-plus/>
- [4] Instalación sistema operativo Raspbian.
<https://www.luisllamas.es/installar-raspbian-en-raspberry-pi-con-etcher/>
- [5] GpioZero
<https://gpiozero.readthedocs.io/en/stable/>
- [6] Guía de instalación de OpenCV
<https://pysource.com/2018/10/31/raspberry-pi-3-and-opencv-3-installation-tutorial/>
- [7] Video Connect-Bot <https://vimeo.com/338985197>