

Proyecto ELCO Blink Pong



BlinkPong

Grupo 2

**Eduardo González García
Antonio Luis Mata García
Paula Sánchez Almagro
Santiago Muñoz-Chápuli
Francisco Jesús Acién Pérez**

ÍNDICE

Motivación	2
Qué es Blink Pong	2
¿Cómo funciona Blink Pong?	3
Fabricación. Diseños en 3D y Hardware del proyecto	7
Diseño electrónico	7
Diseño Mecánico	11
Guía de usuario	13
Estrategia comercial	15
Dificultades encontradas	16
Red mesh	16
Microcontrolador	16
PCB	17
GitHub del proyecto	17
Presupuesto	17
Proveedores	18
Líneas futuras	18
Referencias	19

Motivación

Este proyecto está claramente marcado por el gran problema que nos hemos enfrentado en la pandemia, el distanciamiento social. Ante una situación tan problemática como no poder acercarnos a nuestros seres queridos, estar confinados en grupos reducidos, hablar a través de una mascarilla. Se nos llenaba la cabeza de formas de aprovechar y disfrutar del tiempo juntos, una vez pasada esta mala época.

Con esto en mente el aluvión de planes e ideas que queríamos hacer cuando desaparezca el dichoso bicho. Ante esto, al aparecer la posibilidad de realizar un proyecto, teníamos claro lo que queríamos: hacer de la vuelta a la normalidad una celebración.

En este enclave nos situamos en la creación de nuestro proyecto Blink Pong, un beer pong inteligente que sea entretenido y ante todo cómodo para, las tan echadas de menos, reuniones de amigos.

Ante todas estas razones, nos embarcamos en el desarrollo de este proyecto, con el que, tras la pandemia, volvamos rápidamente a disfrutar de la normalidad. Una normalidad en la que nuestros clientes disfruten de un juego entretenido, divertido y extraordinario que los lleve a formar unas relaciones que les ayuden a recuperar todo este tiempo perdido.



*De izquierda a derecha: Antonio Luis Mata, Francisco Acien, Eduardo González y Paula Sánchez.
Santiago Muñoz no pudo asistir por motivos sanitarios*

Qué es Blink Pong

Blink Pong es un juego interactivo basado en el clásico juego de beber *Beer Pong* para los jóvenes universitarios que consiste en una serie de posavasos y un controlador del juego.

Los posavasos son capaces de comunicarse con una centralita vía radio creando una red en la que comunican las diferentes situaciones del juego (por ejemplo: en juego o no, esperando a conectar a la red, esperando la asignación de un equipo... Esto se explicará más adelante en el código). Además los posavasos también poseen una tira de LED y un sensor infrarrojo que detecta la presencia de vaso.

La centralita está preparada para, a través de un menú interactivo, poder configurar a través de los botones parámetros como: el número de posavasos en juego, el modo de juego, y durante el juego mostrará la puntuación actual.

Con todo esto, se ha tratado de simular un *Beer Pong* con las mismas reglas de juego. Estas se explicarán más adelante en la Guía de usuario.

¿Cómo funciona Blink Pong?

Para empezar a jugar se configurará el número de posavasos y se iniciará el marcador, con esto queda configurado las tiras de leds y podemos iniciar la partida. Durante el juego se hará una detección cuando un vaso se levante, esto es lo que el posavasos comunica a la centralita vía radio, que se encarga de sumar la puntuación del equipo correspondiente.

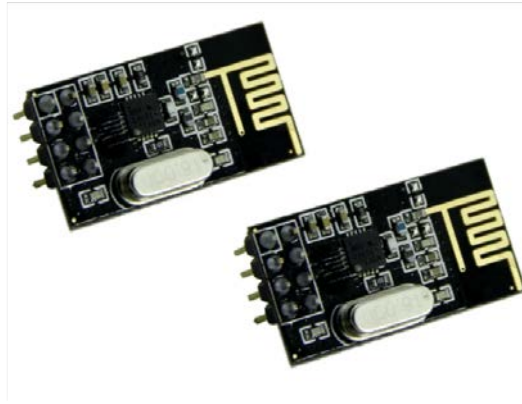
Como ya se ha comentado el proyecto se divide en una centralita y en varios posavasos, actuando respectivamente como maestro y esclavos. El maestro se puede comunicar o bien con un solo esclavo o con todos a la vez mediante una dirección de broadcast, y el esclavo solo se podrá comunicar con el maestro. Para conseguir este tipo de comunicación se hará uso de diferentes cerrojos. Estos bloquean/desbloquean la recepción de los mensajes en función del tipo de dispositivo del que haya sido enviado. En el caso concreto de la centralita, también se pueden encontrar cerrojos cuya finalidad es comprobar que todos los posavasos se encuentran en un estado concreto (por ejemplo: existe un cerrojo *flagTodosConEquipo* que espera a que todos los posavasos hayan enviado un paquete comunicando que se han asignado correctamente a uno de los dos equipos. Una vez que se reciben todos esos paquetes, el cerrojo desbloquea el resto del código del juego).

Todo el proyecto está basado en Arduino UNO pero por restricciones de tamaño, en el prototipo final, el cerebro de los posavasos será el microcontrolador ATmega328p, que es el mismo que controla el Arduino UNO, lo que ayuda a la hora de desarrollar el programa y de tener suficientes conexiones para las funcionalidades actuales y cualesquiera que se quieran implementar en un futuro.

Los componentes necesarios para el proyecto son:

- **NRF24L01 [1]**: la base del proyecto. Gracias a esto y a un protocolo propio, los diferentes agentes del juego (centralita y posavasos) se pueden comunicar de manera

efectiva. La comunicación se da a una frecuencia estándar de entre 2,4 y 2,525 GHz con un ancho de banda de 1 MHz, permitiendo tener hasta 125 canales de comunicación.



NRF24L01

- **Detector de obstáculos infrarrojo [2]:** mediante dos diodos, uno emisor de infrarrojos y uno fotosensible, y un potenciómetro actuando de comparador, permite detectar la presencia o ausencia de un objeto encima del posavasos. Se hará uso de 3 de los 4 pines, los dos de alimentación y el pin D0, ya que el pin A0 se lee de manera analógica y permite medir la distancia exacta a los diodos, función que no es de utilidad para el proyecto, puesto que el umbral de detección ya se puede configurar mediante el potenciómetro. Este sensor solo se emplea en el prototipo, pues en la PCB del diseño del producto final ya están presentes los diodos responsables de esta detección de presencia.



Sensor de obstáculos infrarrojo

- **Pilas 9V:** batería que se usará en el prototipo para alimentar los Arduinos de los posavasos y la centralita.
- **Pilas AAA:** esta sería la elección para la alimentación de los posavasos en el diseño del producto final. Se ha planteado emplear 3 pilas AAA para alimentar los posavasos con 4,5V.

- **Pantalla LCD 16x2 con I2C [3]:** Para mostrar contenido al usuario en la centralita. El bus I2C se utiliza para economizar el uso de pines en el microcontrolador.



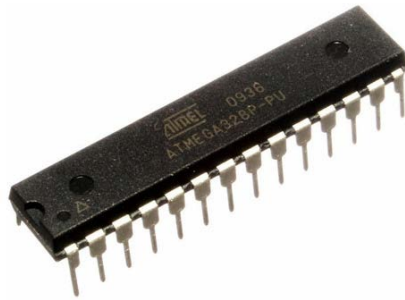
Pantalla LCD 16x2 con I2C

- **Botones:** para seleccionar distintas configuraciones en la centralita.
- **Arduino UNO [4]:** se usará en el prototipo como el microcontrolador tanto de la centralita como de los posavasos. Aun así, en el diseño final, simplemente se usará como maestro de la red mesh (microcontrolador de la centralita).



Arduino UNO

- **ATmega328p [5]:** en el producto final, será el microcontrolador de los posavasos, como antes mencionado, cuenta con el mismo número de pines y la misma memoria que el Arduino UNO.



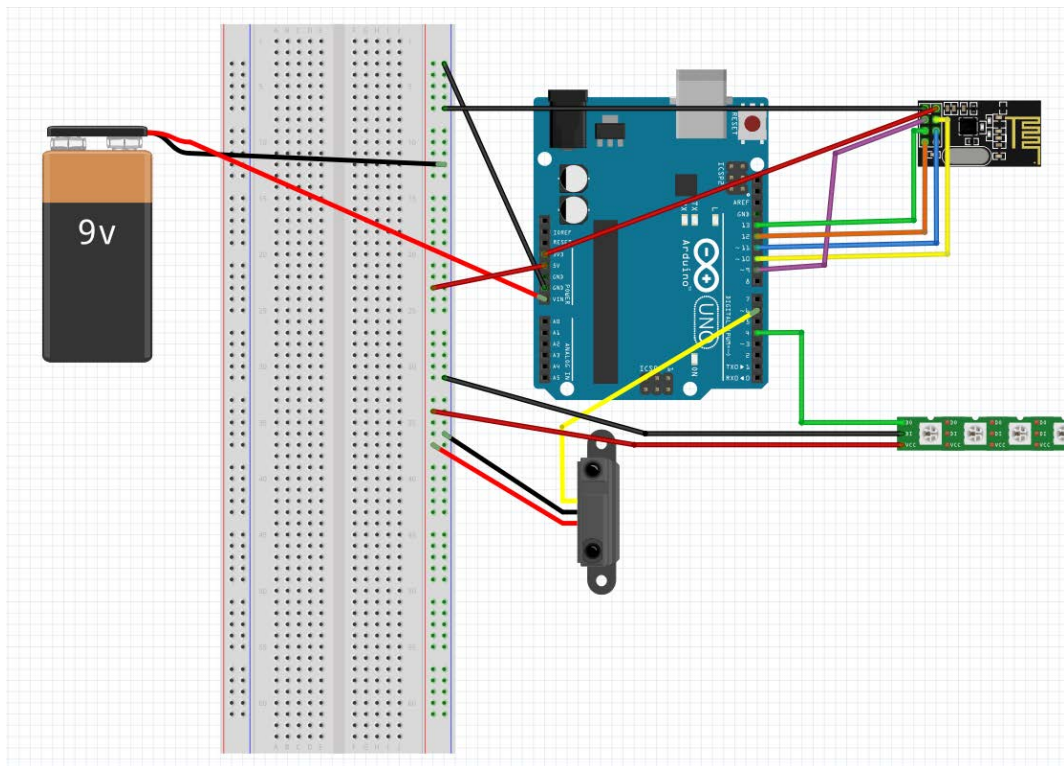
Microcontrolador ATmega328p

- **Tiras de led:** individualmente programables, para maximizar las posibilidades de efectos de luz que se pueden hacer, y con grado de protección IP67, garantizando la máxima resistencia al agua y al polvo.



Tiras de led

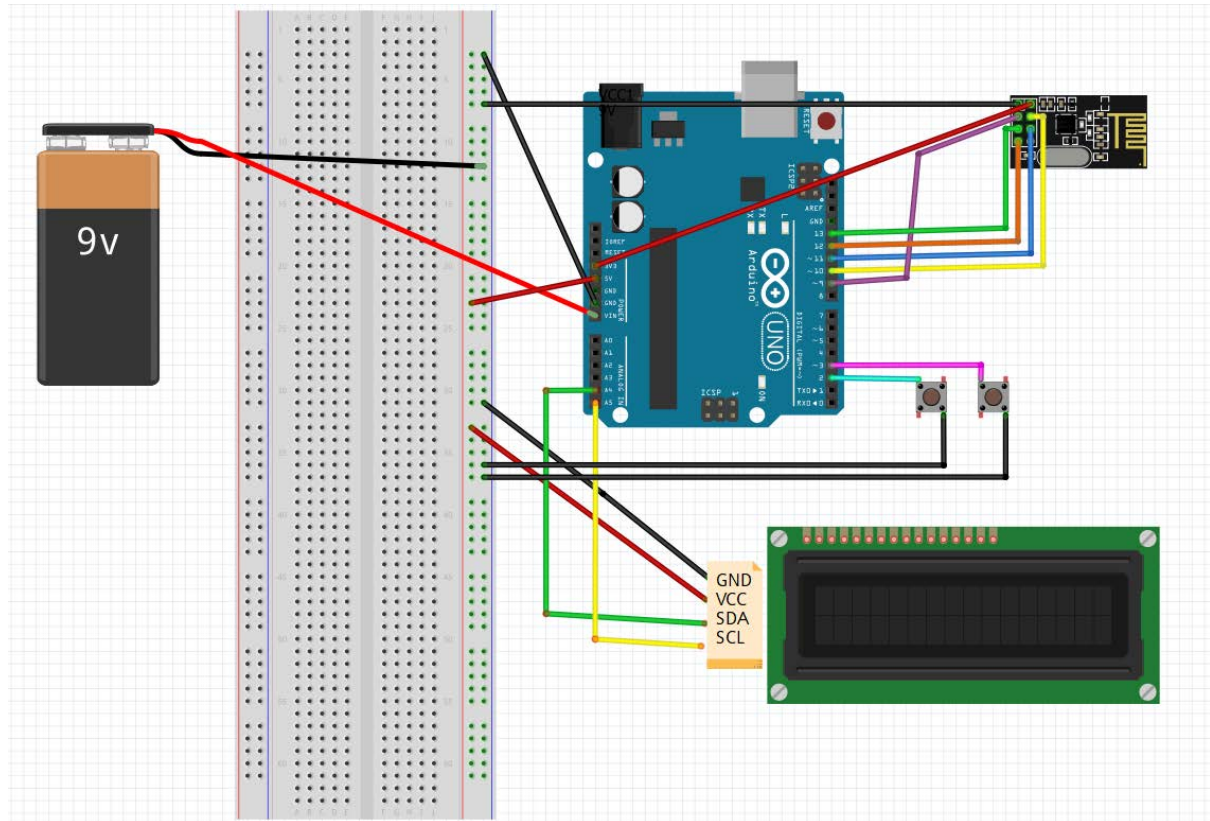
A la hora de hacer pruebas, se llevó a cabo la siguiente disposición de componentes. En el caso de los posavasos realizamos un montaje como el de la siguiente figura.



Montaje para pruebas de un Arduino con función de posavasos

Se puede observar la conexión del NRF24L01, la tira de neopixels y el sensor de obstáculos IR al Arduino UNO que en este caso actuaría como posavasos. Para conectar el NRF24L01 nos apoyamos en [6].

En el caso de la centralita, el montaje se muestra a continuación.



Montaje para pruebas de un Arduino con función de centralita

Se puede observar la conexión del NRF24L01, el display LCD 16x2 con I2C y los botones al Arduino UNO que en este caso actuaría como centralita. Para la conexión del display con el Arduino estudiamos el ejemplo de [7].

Fabricación. Diseños en 3D y Hardware del proyecto

Diseño electrónico

Para integrar todos los componentes de los posavasos hemos mandado a fabricar una PCB redonda que se integra dentro de un posavasos impreso en 3D por nosotros mismos. Cómo se vió en la presentación del prototipo, tuvimos problemas a la hora de usar las PCBs

por lo que se usaron arduinos dentro de los posavasos. Esto no provoca un conflicto con la parte comercial, ya que el arduino posee el mismo microcontrolador.



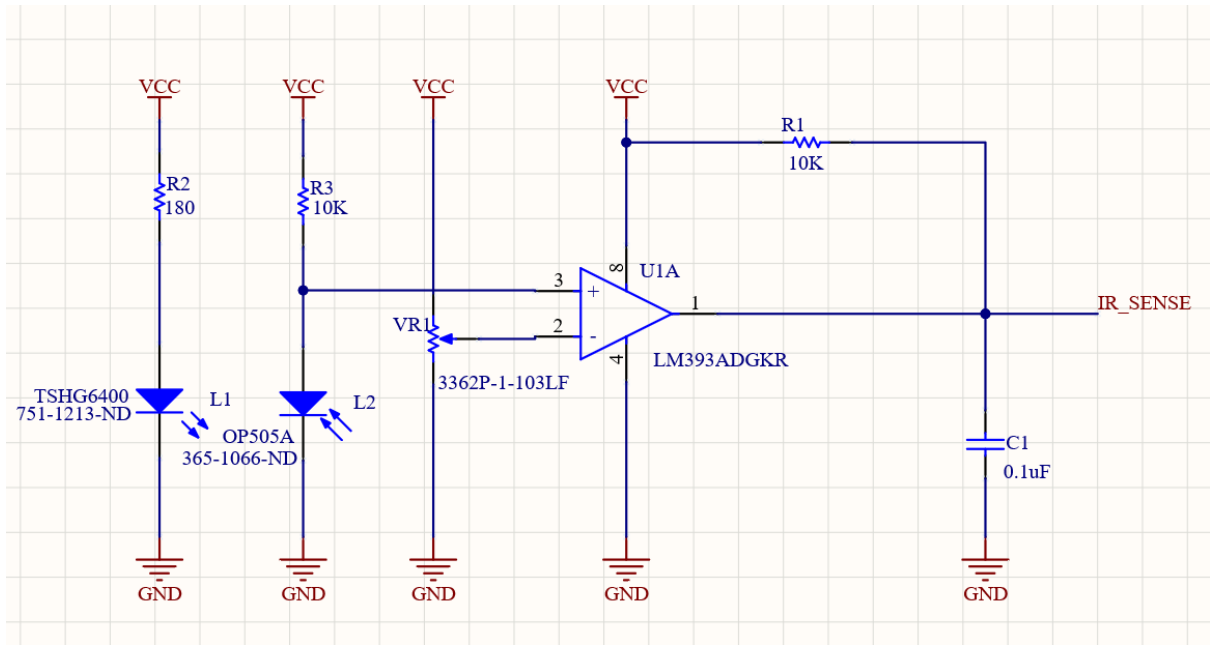
Posavasos



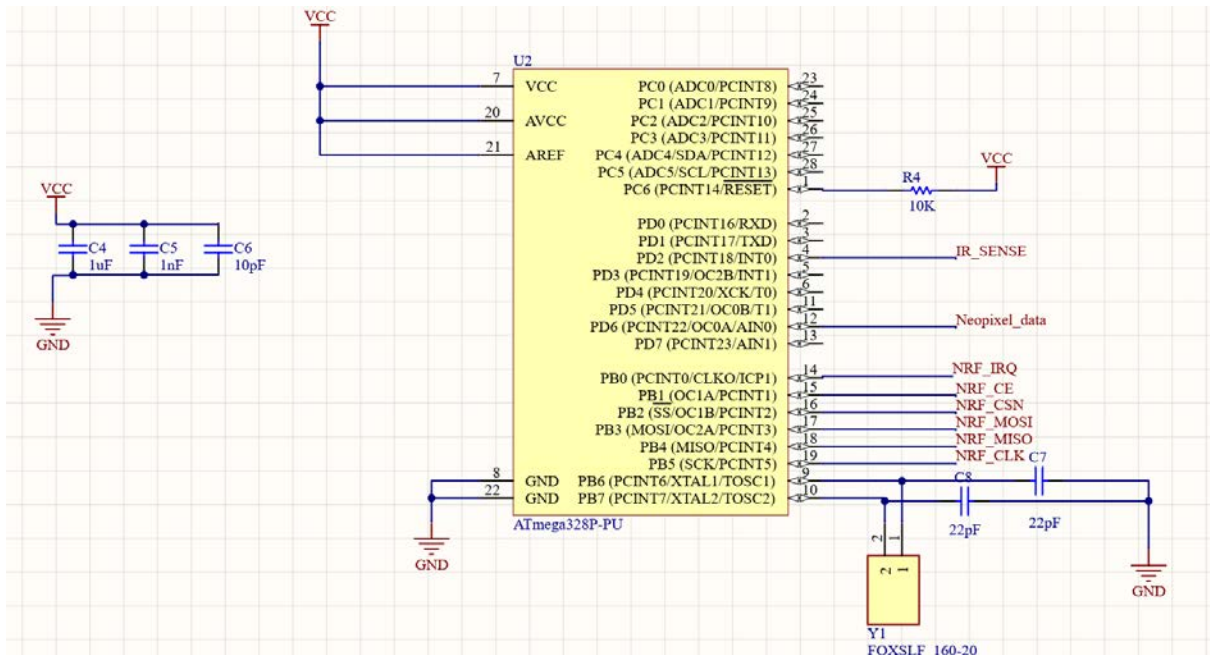
Centralita

La PCB del producto final se ha diseñado con Altium Designer [8], un programa profesional para la creación de circuitos electrónicos. El producto final tiene que contar con un microcontrolador, un sensor infrarrojo, un módulo NRF24L01, baterías y botón de encendido y apagado. El microcontrolador del producto final será el Atmega328p, ya que es un microcontrolador sencillo de implementar y que además es el mismo que el del arduino uno, de esta forma nos ahorraremos tiempo de implementación. El sensor infrarrojo se compone de un led infrarrojo y un fototransistor conectado a un comparador que nos dará la lectura de si hay un vaso encima o no. Para que el dispositivo sea lo más compacto posible se ha optado por utilizar pilas AAA directamente ancladas en la parte posterior de la PCB. Para el diseño mecánico posterior se ha tenido en cuenta los agujeros de los tornillos y también dos surcos en la PCB para ensamblar la carcasa.

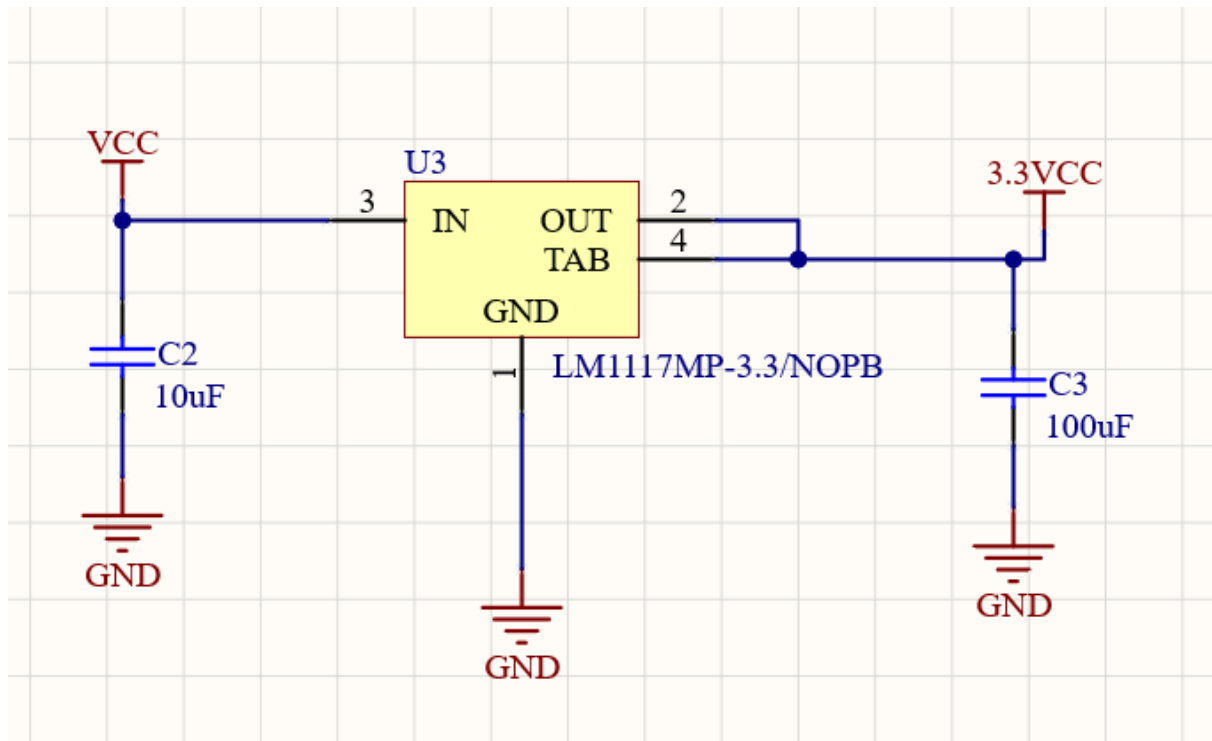
Se desarrolla una descripción del esquemático.



La detección del paso encima del posavasos se hace por medio de un sensor infrarrojo. Se compone de un emisor infrarrojo (L1) que excita un fototransistor conectado a la entrada de un comparador. El comparador se conecta también a un potenciómetro que servirá para ajustar el umbral de detección del aparato. De esta forma a la salida tendremos la lectura del comparador que nos dará la información de si hay algún objeto encima del dispositivo.

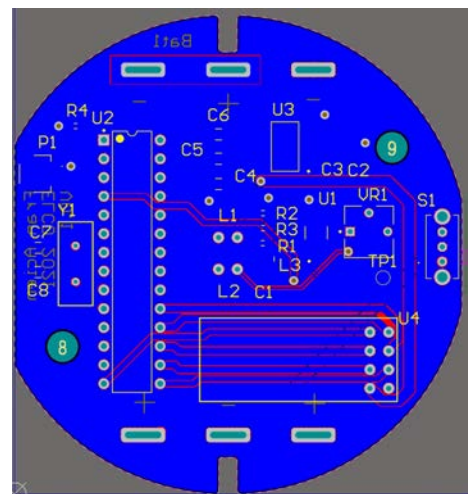
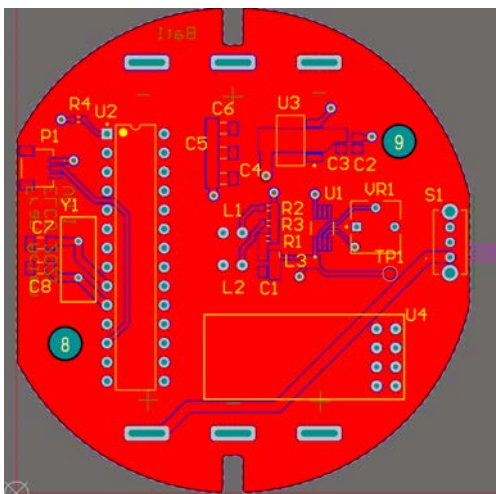


El microcontrolador utilizado en este dispositivo será el Atmega328P. La conexión de este microcontrolador es bastante sencilla, sólo siendo necesario un oscilador de cuarzo, algunos condensadores de desacoplo, y una resistencia pull-up a reset. Las conexiones de los diferentes sensores y actuadores se pueden ver en la imagen.

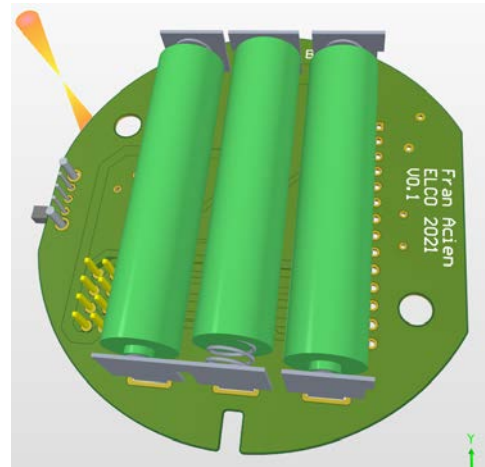
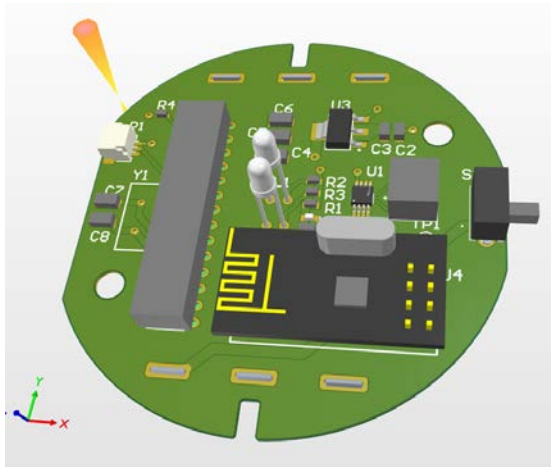


La alimentación del Atmega328p y de los sensores se hace por medio de la alimentación de las 3 pilas AAA conectadas en serie, que nos daría en torno a 4.5V. En cambio, el NRF24L01 se alimenta con una tensión de 3.3 voltios. Esto hace necesario la implementación de un regulador de tensión que nos de a la salida 3.3 voltios para alimentar el NRF. El componente utilizado para esta función es el LM1117 [9] que es un regulador de tensión fijo ampliamente utilizado en otros dispositivos con las misma características que el nuestro.

Los componentes se adquirieron en su mayoría en Digi-Key [10], y la manufacturación la llevamos a cabo nosotros en los laboratorios de investigación de la escuela. Por falta de tiempo y algunos fallos en la manufacturación no fue posible la implementación del dispositivo completo, en cambio, los diferentes módulos de la PCB funcionan como se esperaba. Queda pendiente para posteriores versiones la implementación de la PCB final con el dispositivo.



Diseño de la PCB en Altium Designer de la capa superior e inferior.



Renders de la PCB con Altium Designer



PCB del producto final montada con todos los componentes

Diseño Mecánico

Uno de los objetivos que nos marcamos desde la definición del proyecto fue la de crear un dispositivo pequeño. Como hemos visto en las anteriores secciones, aunque un posavasos puede parecer algo sencillo, nuestra propuesta incluye bastantes elementos. Un reto importante será meter en un dispositivo pequeño un microcontrolador, batería, transceptor de radio, antena y sensor infrarrojo. Para el prototipo inicial se fabricará con una impresora 3D cartesiana, lo que pondrá algunas limitaciones en el diseño.

El diseño mecánico se hizo utilizando el software profesional SolidWorks [11], un programa muy potente para diseño de piezas, simulación térmica, de fluidos, mecánica, etc. Se eligió este programa por las grandes ventajas, además de ser un software profesional ampliamente utilizado en la industria.

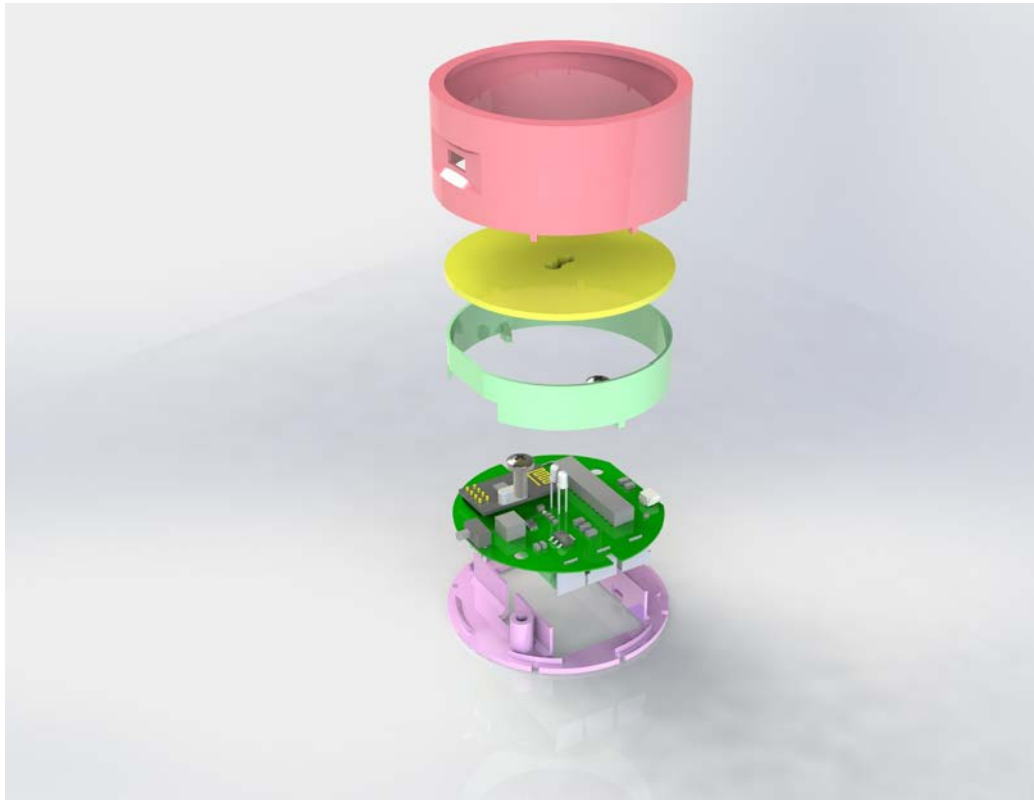
Los requisitos que tiene que cumplir la carcasa del dispositivo será:

- Tiene que tener forma de posavasos
- Tiene que ser fabricable por una impresora 3D. Esto requerirá de no dejar piezas *flotando*.
- La carcasa tendrá que contar con agujeros para el sensor infrarrojo en la parte superior para detectar si hay un vaso. Tiene que haber un agujero en un lateral que tiene que encajar el botón de encendido y apagado. Tiene que contar con un agujero lateral para poder pasar los cables de las luces. Y por último, tiene que contar con una solapa que se pueda poner y quitar para acceder a las pilas.

Con todas estas especificaciones se diseñó la pieza con SolidWorks. El diseño 3D de la PCB generado por Altium Designer que junto con la simulación mecánica de Solidworks hace que los diseños sean muy rigurosos ahorrando fallos en la fabricación. De esta forma no fueron muchas las iteraciones necesarias para conseguir el modelo final. En el desarrollo del modelo de SolidWorks no se encontraron grandes dificultades.



Diseño en 3D de la carcasa de los posavasos



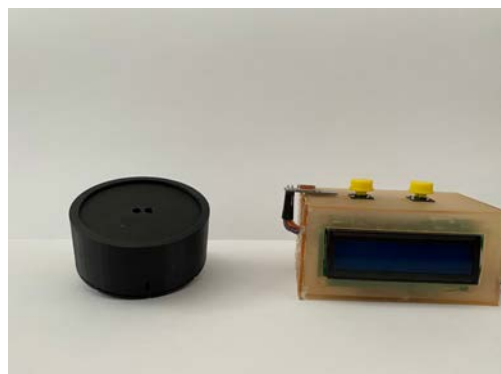
Captura en SolidWorks de la carcasa de los posavasos

Guía de usuario

1. Para comenzar a jugar, lo primero que haremos será encender la centralita y configurar el número de posavasos en juego a través de los botones disponibles:

-Botón de navegación: Sirve para movernos dentro del menú, entre las distintas opciones de juego. Estas opciones son el número de vasos que van a participar en la partida, hemos implementado dos opciones: 12 posavasos (6 para cada equipo) y 20 posavasos (10 para cada equipo). Durante la presentación del dispositivo, se usó como demostración solamente 2 vasos. Pero, el código es completamente escalable y funcional para todas estas configuraciones.

-Botón de selección: Una vez estamos sobre el modo elegido, se presiona este botón para seleccionar la opción de juego deseada y se comenzará a jugar en esta opción.



2. La centralita mostrará un mensaje en la pantalla para conectar los posavasos. Encender los posavasos mediante el botón ON/OFF (en el video presentación del prototipo se correspondía con conectar las pilas de 9V).
3. Los posavasos entrarán individualmente en un estado en el que su tira de leds se pondrá de color verde. Esto se corresponde con la conexión de ese posavasos a la *mesh* del juego creada por la centralita.



4. Cuando la centralita detecta que el número de posavasos conectados coincide con la opción del número de posavasos en juego elegido al principio en el menú de juego, comienza a asignar equipos. Asignará primero el equipo naranja y una vez se hayan asignado correctamente la mitad de los posavasos a este equipo, se asignará el equipo azul. Mientras manda las instrucciones de asignación a cada equipo aparecerá un mensaje informativo en la pantalla comunicando qué equipo se está asignando en ese instante.



5. Una vez todos los posavasos tengan su equipo asignado (leds en color naranja o azul), la centralita avisará que es el momento de colocar los vasos encima de cada posavasos. Además se puede ver un contador en la pantalla que indica el número de posavasos que han comunicado correctamente que tienen vaso y están preparados para jugar.
6. Los posavasos se iluminarán con una animación de luces parpadeantes cuando todos tengan vaso encima indicando así el inicio del juego. A continuación se hará un resumen de las reglas del juego Beer Pong en el que está basado este proyecto.

El objetivo principal es sencillo: los jugadores deben encestar la pelota de ping-pong en el vaso del oponente, el cual está lleno de cerveza o de cualquier tipo de bebida.

Si atinan el tiro, el oponente debe beber el líquido de dicho vaso y retirarlo de la mesa. El equipo ganador es el que encesta en todos los vasos del oponente. Existen ciertas disposiciones y reglas para hacer del juego más formal.

- 1) **Equipos:** Aunque pueden jugarlo de manera individual, las reglas oficiales establecen dos personas por equipo.
 - 2) **Disposición de los vasos.** Como ya lo mencionamos, son 10 vasos por equipo. Éstos deben ubicarse de manera triangular (4-3-2-1), donde la punta del triángulo apunta al jugador oponente. Los vasos deben ubicarse en el centro de la mesa.
 - 3) **Contenido de los vasos:** Oficialmente, los 10 vasos se llenan con agua y aparte se tienen los vasos con cerveza. Esto tiene una razón higiénica, pues así se evita que los jugadores ingieran la cerveza donde cayó la pelota (que suele caerse al suelo). Pero comúnmente, se llenan los vasos con 3/4 de cerveza, y el vaso extra con agua para limpiar la bola.
 - 4) **Lanzamiento:** El jugador debe lanzar la pelota al otro extremo de la mesa donde están los vasos del oponente. Éste debe mantener una posición centrada y siempre estar detrás de la mesa (no en sus alrededores); debe usar una sola mano; puede inclinarse, pero sin tocar la mesa (con ninguna parte del cuerpo); y puede encestar de manera directa (sin rebote), con rebote en la mesa (sólo un rebote), o un rebote fuera de la mesa, pero este último debe rebotar e ir directo al vaso.
 - 5) **Turnos:** Cada equipo tiene dos lanzamientos por turno, es decir, uno por persona del equipo. El equipo rival debe esperar a que acabe el turno del otro para beber los vasos donde cayó la bola.
 - 6) **Distracción:** Se puede distraer al rival ya sea con gestos, hablando, etc, pero siempre manteniendo su posición detrás de la mesa. Se prohíbe bloquear la visibilidad de los vasos, ir al otro lado del rival, soplar o cualquier cosa que cree una condición de aire inestable.
 - 7) **Penalizaciones:** Los jugadores oponentes no deben tocar la pelota mientras está en movimiento. No deben interceptarse ni menos cogerla antes de que toque el vaso. En caso de que esto ocurra, el equipo que lanzó la pelota puede pedir la retirada del vaso oponente.
-
7. Durante el juego cuando entra una pelota en uno de los posavasos, el jugador levanta el vaso para beber, en ese momento el posavasos lo detecta, cambia de color, y manda un mensaje a la centralita, que quita un punto al equipo de ese vaso.
 8. El juego continúa hasta que un equipo haya conseguido el máximo de puntos, en ese momento acaba el juego y los leds lo informan con un espectáculo de luces.

Estrategia comercial

Nuestro producto está dirigido a un público joven, universitario en principio. Inicialmente nos daremos a conocer en la ETSIT, hablaremos con las asociaciones universitarias como los clubes de nuestra escuela. Una vez hecho esto, podremos extendernos

a otras asociaciones de distintas escuelas y universidades de Madrid. Donde veamos que interese, pondremos carteles en estas universidades.

Cuando hayamos recaudado suficiente, podremos publicitarnos a gran escala con a través del marketing digital, con anuncios en redes sociales como TikTok, Instagram y Youtube. Podremos grabar vídeos cortos en los que se explica el funcionamiento del producto.

Para la distribución de nuestro producto crearemos una página web donde se podrá comprar online el producto, con distribución en la península y Baleares (<https://eduarho13.wixsite.com/blinkpong>). En esta misma web tendremos un servicio post-venta para solucionar posibles errores y devoluciones.

Ofreceremos dos paquetes al público, el primero consta de 4 posavasos y una centralita, y el segundo de 12 posavasos y una centralita. Los precios comerciales serán 39,95€ y 89,95€ respectivamente. Desglosando estos precios, serían unos 5€ por posavasos (PCB montada + carcasa de plástico) y centralita otros 5€. Esto nos dejaría unos beneficios de 14,95€ para el pack de 4 posavasos y 24,95€ para el pack de 12. Estos precios y beneficios pueden verse desglosados más adelante en el apartado de presupuesto.

Dificultades encontradas

Red mesh

Las principales dificultades que encontramos con el envío y recibo de paquetes en la mesh de radiofrecuencia generada por los NRF24L01 fueron:

- Por una parte la complejidad a la hora de evitar una congestión de paquetes. Conseguimos solucionarlo consiguiendo, mediante el uso de protocolo propio, que los dispositivos solo tuviesen visible en la red los paquetes que fuesen enviados para ellos.
- Encontramos en las pruebas muchos problemas a la hora de los envíos de paquetes por parte de los esclavos. Por lo que, se desarrolló un método que comprobaba cada cierto periodo de tiempo si se había enviado correctamente el paquete y en caso contrario volvía a intentar el envío.
- Otro problema que se encontró fue la dificultad en ocasiones puntuales de conectarse a la *mesh*. Con un reseteo general de la centralita y los posavasos se solucionaba sin grandes problemas.

Microcontrolador

Aunque el diseño electrónico propuesto consta de un Atmega328P, la idea original del proyecto era utilizar un attiny85 [12]. El Attiny85 y el Atmega328p [5] son de la misma familia de microcontroladores AVR pero con diferentes capacidades. El diseño original contó con la utilización del attiny85 pero la pequeña capacidad de la memoria RAM y ROM nos obligaron a cambiar la idea original y utilizar un microcontrolador más grande. Una ventaja que tiene este cambio de microcontrolador es que el Atmega328P es el mismo microcontrolador que el del Arduino UNO, asegurándonos que funciona perfectamente en esta arquitectura.

PCB

El diseño de la PCB se vio con algunas dificultades durante el desarrollo. Como hemos mencionado anteriormente, se ha desarrollado una PCB para el prototipo final, pero que por falta de tiempo y algunos fallos de diseño no se ha podido implementar completamente. El esquemático original funcionaba perfectamente pero a la hora de soldar los componentes superficiales, con métricas 0603 en algunos componentes, no se soldaron suficientemente bien. La falta de experiencia, y la dificultad de soldar estos componentes sin stencils hizo que no se pudieran soldar las PCBs a tiempo para implementar todo el sistema. Haciendo algunas pruebas rápidas de los diferentes sistemas de la PCB, podemos determinar que funciona cómo estaba diseñado, pero que faltaría implementar todo el sistema expuesto anteriormente. Para solucionar este problema haría falta volver a soldar las PCBs y dedicar tiempo en integrar todo el prototipo final.

GitHub del proyecto

El código del proyecto, tanto el código del maestro como el de los esclavos puede verse y descargarse en el siguiente enlace https://github.com/paulitasa/ELCO_BlinkPong.

Presupuesto

A continuación mostramos en tablas los precios desglosados por centralita y por posavasos.

Centralita	Precio Unitario por 10000U	Precio Por Dispositivo
1 Carcasa	0,3	0,3
1 Controlador (Arduino)	3	3
1 NRF24L01	0,45	0,45
1 LCD Display	0,53	0,53
3 Botones	0,1	0,3
1 conector Pila 9 V	0,507	0,507
TOTAL	4,887	5,087

Posavasos	Precio Unitario por 10000 U	Precio Por Dispositivo
1 PCB	0,24569	0,24569
1 Cristal de Cuarzo	0,13	0,13
1 Condensador 0.1µF	0,04543	0,04543
1 Condensador 10µF	0,04928	0,04928
1 Condensador 1µF	0,04084	0,04084
1 Condensador 1000pF	0,04691	0,04691
1 microcontrolador ATmega328p	1,46	1,46
1 Comparador	0,11446	0,11446
1 Cabezal de conector	0,0389	0,0389
3 Contactos para pilas	0,073205	0,219615

1 Resistencia 100 Ohms	0,00101	0,00101
2 Resistencias 10 KOhms	0,00751	0,01502
1 Zócalo DIP	0,11871	0,11871
1 Conector Hembra-Macho 8 Pines	0,25074	0,25074
1 Potenciómetro 10 kOhms	0,094	0,094
1 Interruptor ON/OFF	0,20586	0,20586
1 LED	0,0808	0,0808
1 Condensador 10pF	0,06276	0,06276
1 Condensador 100 µF	0,085	0,085
2 Resistencias 22 kOhms	0,00287	0,00574
2 Condensadores 22 pF	0,02415	0,0483
1 NRF24L01	0,45	0,45
20 cm LEDs individualmente programables IP67	0,87	0,87
1 Carcasa	0,4	0,4
TOTAL	4,898079	5,079019

Producto	Precio de Fabricación	Precio de Venta	Margen de Beneficio
Pack Centralita + 4 Posavasos	25,40 €	39,95 €	57,28%
Pack Centralita + 12 Posavasos	66,04 €	89,95 €	36,21%

Proveedores

El producto final se estructura en diferentes elementos, que serán pedidos a diferentes proveedores.

- Componentes electrónicos: todos los componentes electrónicos utilizados en el receptor se pedirán en Digi-Key [10], y todos los precios especificados para los componentes se han obtenido de esta página.
- Placa PCB: para la realización de la placa del receptor, se enviará a fabricar a JLCPCB [13].
- Botones, PLA, Arduino y display LCD: tanto los componentes mencionados como el plástico para la impresión 3D se pedirán a AliExpress [14].

Líneas futuras

Durante el desarrollo del proyecto, nos encontramos con una serie de ideas que nos gustaría implementar en el producto final:

- **Water resistance**

La primera mejora sería realizar los posavasos para que sean estancos. Durante el desarrollo del prototipo no fue necesario incluir esta propiedad. Pero una vez desarrollado el producto final, se efectuará un esfuerzo por la empresa que ensambla cada posavasos para crearlo estanco. Una vez se implemente cada posavasos con PCBs esto se simplifica enormemente.

- **Reset y Modos de juego**

Tras la implementación de la versión estanca del producto, se enfocarán los esfuerzos en mejorar la centralita. Aparecerá un botón de reset para solucionar posibles problemas en la creación de la red e incluso tendrá la función de reiniciar el juego.

Además se plantearán propuestas de nuevos modos de juego según los intereses de nuestros consumidores. Algunos de estos nuevos modos podrían ser: modo contrarreloj, parecido a los relojes de los ajedrecistas; modo entrenamiento, donde todos los posavaso tendrían el mismo color o modo torneo, en el que la centralita organizaría los enfrentamientos e hiciese un seguimiento de los resultados.

- **Leaderboards**

Se plantea un desarrollo de una app para llevar un recuento de las partidas que juegue una persona, con la que la propia centralita vuelque sus resultados dentro de nuestros servidores y creemos un ranking con el que fomentar la competencia entre los usuarios y el uso de nuestro producto.

- **Nuevos posavaso**

Como última ampliación de nuestro prototipo se propone desarrollar el juego de tal manera que nuevos posavaso se pudieran ir añadiendo una vez adquirido un pack. La idea sería añadir a los posavaso un código que permitiese, al encenderse por primera vez, enlazarse a un “set” de Blink Pong ya inicializado siempre y cuando este estuviese esperando a comenzar a jugar con la opción “enlazar nuevo” activada. Así, adoptará este posavaso como suyo, de tal manera que nuestro producto fuera escalable y se pudieran ir adquiriendo posavaso según las necesidades de nuestro cliente.

Referencias

[1] El datasheet correspondiente al NRF24L01 es el siguiente:

https://www.sparkfun.com/datasheets/Components/SMD/nRF24L01Pluss_Preliminary_Product_Specification_v1_0.pdf

[2] El diagrama correspondiente al sensor IR es el siguiente:

<https://www.openimpulse.com/blog/wp-content/uploads/wp-content/uploads/wpsc/downloadables/TCRT5000-Sensor-Module-Circuit-Diagram.pdf>

[3] El datasheet correspondiente al display LCD 16x2 con I2C es el siguiente:

<https://opencircuit.shop/resources/file/da88acc1702a90667728fcf4ac9c75c455475706466/I2C-LCD-interface.pdf>

[4] El datasheet correspondiente al Arduino UNO es el siguiente:

<https://datasheet.octopart.com/A000066-Arduino-datasheet-38879526.pdf>

[5] El datasheet correspondiente al ATmega328p es el siguiente:

https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-7810-Automotive-Microcontrollers-ATmega328P_Datasheet.pdf

[6] Montaje NRF24L01. COMUNICACIÓN INALÁMBRICA A 2.4GHZ CON ARDUINO Y NRF24L01:

<https://www.luisllamas.es/comunicacion-inalambrica-a-2-4ghz-con-arduino-y-nrf24l01/>

[7] Montaje display lcd 16x2 con I2C y Arduino. LCD 16×2 por I2C con Arduino usando solo dos pines:

<https://www.geekfactory.mx/tutoriales/tutoriales-arduino/lcd-16x2-por-i2c-con-arduino/>

[8] Página web oficial del entorno de diseño electrónico Altium:

<https://www.altium.com/es>

[9] El datasheet correspondiente al LM1117 es el siguiente:

<https://www.alldatasheet.es/datasheet-pdf/pdf/197818/SEOUL/LM711.html>

[10] Página web oficial de compra de componentes electrónicos Digi-Key:

<https://www.digikey.es/>

[11] Página web oficial del software profesional SolidWorks:

<https://www.solidworks.com/es>

[12] El datasheet correspondiente al attiny85 es el siguiente:

https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-2586-AVR-8-bit-Microcontroller-ATtiny25-ATtiny45-ATtiny85_Datasheet.pdf

[13] Página web oficial de JLCPCB:

<https://jlcpcb.com/>

[14] Página oficial de AliExpress:

<https://es.aliexpress.com/>