



Universidad Nacional  
de Educación a Distancia



Universidad  
Complutense de Madrid

Escuela Técnica Superior de  
Ingeniería Informática

Facultad de Informática

# Diseño e implementación de un sistema de monitoreo de incubación de huevos con plataforma ESP32 y sensores infrarrojos de temperatura

Sergio Mesa Raya

Director/a: José Sánchez Moreno  
Co-director/a: David Moreno Salinas

Trabajo de Fin de Máster

Máster Universitario  
en Ingeniería de Sistemas y Control

Fecha presentación: 07/09/2024/ Segunda convocatoria]



## **Agradecimientos**

Quisiera dedicar este trabajo de fin de Máster a mi pareja Vanesa, por toda la ayuda y comprensión durante la realización de este.

También a la empresa Pondex del Grupo Vall Companys por su facilidad para la realización de las pruebas industriales dentro de su planta de incubación de Sástago.



## Resumen

Este proyecto aborda la necesidad de monitorear con precisión el parámetro más importante dentro del proceso de incubación de huevos embrionados, esto es: la temperatura de la cáscara. Para ello:

- El desarrollo está centrado en la creación de un sistema técnico avanzado que integra una placa ESP32 y sensores infrarrojos de temperatura MLX90614 para medir y monitorear de manera efectiva esta variable crítica. A este dispositivo en lo sucesivo le denominaremos con el nombre comercial de TEVO, puesto su propósito principal es medir la temperatura del huevo.
- La comunicación de la placa ESP32 con los sensores infrarrojos se realiza en el entorno de Arduino.
- Se ha diseñado una interfaz web para visualizar en tiempo real los datos recopilados en remoto. La misma se realiza en lenguaje HTML embebido en Arduino.
- Una vez realizada la interfaz web, se ha procedido al modelado del dispositivo. Éste tendrá forma de huevo para poder encajarlo en una bandeja de incubación sustituyendo a uno real del proceso, y apuntando a dos huevos contiguos con ambos sensores.
- Por último, se ha fabricado el dispositivo mediante impresión 3D.

**Palabras clave:** incubación, monitorear, cáscara de huevo, ESP32, MLX90614, TEVO.



# Índice general

Agradecimientos .....	3
Resumen .....	5
<b>Índice general.....</b>	<b>7</b>
<b>Índice de figuras .....</b>	<b>9</b>
<b>Capítulo 1 Introducción .....</b>	<b>11</b>
1.1. Motivación .....	11
1.2. Propuesta y objetivos.....	11
1.3. Estructura del documento.....	12
<b>Capítulo 2 Estado del Arte .....</b>	<b>13</b>
2.1. Principios de la incubación artificial .....	13
2.2. La industria de la incubación actual .....	14
2.3. Medidores de cáscara de huevo.....	18
2.4. Aporte del dispositivo objeto del proyecto.....	18
<b>Capítulo 3 Materiales y métodos.....</b>	<b>21</b>
3.1 Componentes .....	21
3.2. Precios de los materiales empleados .....	26
3.3. Métodos y programación realizada .....	28
3.3.1. Conexión placa con sensores .....	28
3.3.2. Interfaz gráfica de TEVO.....	29
3.3.3. Visualización TEVO en terminal.....	32
<b>Capítulo 4 Resultados .....</b>	<b>37</b>
4.1. Temperaturas en monitor serial Arduino.....	37
4.2 Interfaz visualización dispositivo en Arduino.....	37
4.3. Visualización de la interfaz en un terminal .....	38
<b>Capítulo 5 Modelado .....</b>	<b>41</b>
<b>Capítulo 6 Ensamblaje.....</b>	<b>45</b>
<b>Capítulo 7 .....</b>	<b>49</b>
<b>Pruebas industriales.....</b>	<b>49</b>
<b>Capítulo 8.....</b>	<b>51</b>

<b>Conclusiones y trabajos futuros .....</b>	<b>51</b>
<b>Bibliografía y referencias .....</b>	<b>53</b>

# Índice de figuras

Figura 1. Horno de incubación.	13
Figura 2. Incubadora carga múltiple.	15
Figura 3: Incubadora carga simple.	15
Figura 4: Registros temperaturas en carga múltiple.	16
Figura 5: Parámetros proceso carga única.	17
Figura 6: Carros volteados durante incubación.	17
Figura 7: Gráfica de medida de Tempo.	18
Figura 8. Disposiciones de bandejas de incubación.	21
Figura 9. Pines placa SP-Cow ESP32.	22
Figura 10: Precisión estándar de MLX90614.	23
Figura 11. Esquema conexiones elementos.	24
Figura 12. Esquema comunicación I2C.	24
Figura 13. Relaciones componentes dispositivo.	25
Figura 14. Anverso y reverso del cargador.	26
Figura 15. Sensores TECNOIOT MLX90614ESF.	26
Figura 16. Placa ESP32.	27
Figura 17: Batería Litio recargable.	27
Figura 18: Cargador DAOKAI.	27
Figura 19: Inserto impresión 3D.	28
Figura 20: Micro interruptor.	28
Figura 21. Placa y sensor en protoboard.	28
Figura 22. Pruebas en monitor serial de Arduino.	37
Figura 23. Interfaz visualización temperaturas.	38
Figura 24. Visualización en Android.	39
Figura 25: Explosionado piezas carcasa 1.	41
Figura 26: Explosionado piezas carcasa 2.	42
Figura 27: Detalle de TEVO.	43
Figura 28: Prototipos de TEVO y acoples.	45
Figura 29: Modelo inicial y final acople interruptor.	46
Figura 30: Cargador con y sin USB.	46
Figura 31: Comprobación cargador.	47
Figura 32: Soportes de Tevo.	47
Figura 33: Prototipos con soportes.	48
Figura 34: Prototipos Tevo en bandeja incubación.	48
Figura 35: Incubadoras Petersime BioStreamer.	49
Figura 36: Tevo con soportes.	49
Figura 37: Tevo sin soportes.	50
Figura 38: Visualización medición en incubadora.	50



# Capítulo 1

## Introducción

### 1.1. Motivación

Durante años, ha sido ampliamente reconocido que uno de los parámetros más importantes para la incubación industrial junto con el tiempo del proceso es la temperatura de cáscara [1]. Sin embargo, comercialmente no hay muchos dispositivos específicos disponibles para medirla y monitorearla, especialmente en tiempo real, sin tener que esperar a que el proceso haya finalizado o interrumpir parcialmente el mismo para extraer el aparato y descargar los datos.

Debido a mi profesión, la cual está directamente relacionada con este sector, soy consciente de esta problemática particular. Como parte de mi trabajo de fin de Máster en Ingeniería de Sistemas y Control, he decidido realizar un proyecto tangible y físico que pueda tener una salida comercial. Mi objetivo es proporcionar una herramienta adicional a todos los responsables de plantas de incubación, quienes en muchos casos trabajan día y noche con tecnologías obsoletas. El proyecto pretende brindarles una solución que facilite el monitoreo preciso de la temperatura de cáscara de huevo de manera continua y en tiempo real, sin interrumpir el proceso de incubación.

### 1.2. Propuesta y objetivos

El objetivo principal de este proyecto es diseñar un dispositivo capaz de medir la temperatura de la cáscara de hasta dos huevos adyacentes de forma precisa. Además, este dispositivo podría ser utilizado para monitorear las temperaturas durante el almacenamiento en granjas, en la planta de incubación o durante el transporte.

Por tanto, el objetivo de Tevo es transmitir los datos de temperatura a una distancia cercana, dentro de la planta de incubación. Teniendo en cuenta que muchas de estas plantas no cuentan con red Wifi resulta la solución más sencilla implementar una red local a partir de un dispositivo de fácil acceso como un modem. En general, los aislamientos de las incubadoras están compuestos por paneles sándwich con una cámara de aire en el medio, por lo que esto no parece ser un factor limitante a priori. Existen distintas alternativas para implementar la comunicación y enviar de forma efectiva los datos recolectados por el dispositivo a una interfaz donde se puedan visualizar. Una alternativa atractiva para esta implementación podría ser LoraWan, un protocolo de comunicación diseñado para enviar pocos datos a grandes distancias con un consumo energético bajo; el problema de emplear dicho protocolo es la necesidad de utilizar una interfaz de hardware que permita visualizar la información o que integre los protocolos Wifi o Bluetooth, que pueden emplear dispositivos estándar como Smartphone o Tablet para visualizar los datos, algo que parece lo más funcional para el usuario estándar.

Por estos motivos se decide utilizar un dispositivo que integren los protocolos Wifi y Bluetooth, descartando el uso de LoraWan, puesto que requeriría un aparato receptor específico y acarrearía una diferencia notoria en el tamaño y el costo.

Las condiciones interiores de las plantas, con temperaturas cercanas a los 100 °F y una humedad relativa que a menudo supera el 75%, pueden afectar a la durabilidad del dispositivo. Además, es posible que se produzcan condensaciones, especialmente durante los primeros días de incubación, aunque esto no es deseable.

Para asegurar la durabilidad y el rendimiento del dispositivo en estas condiciones, es necesario seleccionar materiales y componentes resistentes a la humedad y a las mencionadas temperaturas. También es importante considerar un sistema de aislamiento adecuado para proteger el dispositivo de los efectos adversos de la condensación.

### 1.3. Estructura del documento

En el **Capítulo 1** se explican los motivos que impulsaron la realización de este proyecto, tanto a nivel personal como industrial, además de establecer los objetivos que se pretenden alcanzar.

En el **Capítulo 2** se realiza un análisis del estado actual de la incubación, explorando su evolución en los últimos años y contextualizando por qué la solución propuesta en este proyecto puede ser de gran utilidad en la mayoría de los casos.

En el **Capítulo 3** se detalla la elaboración del dispositivo, incluyendo la descripción de los materiales utilizados y se proporcionarán argumentos que respalden las soluciones implementadas.

En el **Capítulo 4** se presentan los resultados obtenidos en cada una de las fases del proceso, así como los resultados finales.

En el **Capítulo 5** se describen y explican los pasos para el ensamblado de las piezas en el modelado de TEVO.

En el **Capítulo 6** pueden verse los pasos seguidos en el ensamblado de Tevo, con los acoples, soportes y modificaciones sobre el diseño inicial.

El **Capítulo 7** refleja las pruebas de Tevo en una planta de incubación industrial.

En el **Capítulo 8** se nombran las conclusiones derivadas tras las pruebas industriales y que mejoras o trabajos futuros se esperan llevar a cabo para la mejora del dispositivo.

Finalmente, se citan las principales **Bibliografías y Referencias** empleadas para la realización de este trabajo fin de Máster.

# Capítulo 2

## Estado del Arte

### 2.1. Principios de la incubación artificial

La incubación artificial de huevos es una técnica que se remonta al antiguo Egipto, hace 2.000 años, donde se perfeccionó notablemente [2]. Los egipcios desarrollaron hornos de incubación capaces de albergar hasta 4.500 huevos semanalmente. Estos hornos se convirtieron en un referente para otras civilizaciones y dieron origen a una importante industria que sigue vigente en la actualidad. Estas estructuras especializadas mantenían los huevos a una temperatura y humedad adecuadas para permitir el desarrollo y eclosión de los embriones. Incluso hoy en día, como se puede apreciar en la Figura 1, [2], algunos pequeños granjeros de Egipto continúan utilizando esta técnica para incubar sus pollitos.



*Figura 1. Horno de incubación.*

*Fuente: <https://www.atlasobscura.com/articles/egypt-egg-ovens>*

La primera referencia histórica a estos primeros sistemas de incubación fue realizada por Aristóteles, quien escribió que en Egipto "los huevos nacen espontáneamente en el suelo, al ser enterrados en montones de estiércol".

El fraile irlandés Simon Fitzsimons quedó maravillado por los hornos de huevos durante su visita a Egipto como parte de una peregrinación épica que lo llevó desde Irlanda a Tierra Santa a principios del siglo XIV. Fitzsimons consideró que estos hornos eran sobrenaturales y describió su asombro ante ellos de la siguiente manera:

“También en El Cairo, fuera de la Puerta y casi inmediatamente a la derecha... hay una casa larga y estrecha en la que se generan pollitos mediante fuego a partir de huevos de gallina, sin gallos ni gallinas, y en tal número que no se pueden contar”.

Al no mencionar que los huevos previamente habían sido fertilizados por los gallos, Fitzsimons llevó a los lectores a imaginar que las gallinas podían ser “generadas por el fuego”.

Con respecto al proyecto que nos atañe, lo relevante con respecto a esta técnica ancestral es que ya se comprendía que el parámetro más importante para el proceso de incubación es la temperatura de la cáscara, motivo por el cual acercaban los huevos a sus párpados oculares, una de las partes más sensitivas del cuerpo humano, para determinar si deberían quemar más estiércol de camello.

El objetivo de estos hornos era tener una producción de pollitos constante durante todo el año. Por el contrario, los granjeros en buena parte de Europa sólo podían obtener pollitos en primavera y verano, ya que en las épocas más frías las gallinas no conseguían que se alcanzaran las temperaturas de proceso necesarias. No fue hasta la Ilustración cuando ya en Europa se intentó imitar una técnica similar. En 1750, el polímata y físico francés René Antoine Ferchault de Réaumur tuvo ocasión de observar con detalle estos hornos de incubación y trató de replicarlos en Europa sin éxito. Debido a la climatología de esta área de Europa, estos hornos requerían mucha más producción de calor que en Egipto. En cambio, en 1897 el granjero canadiense Lyman Byce ideó la incubadora con lampara de carbón, que utilizaba un regulador eléctrico para mantener la temperatura constante y podemos señalar este punto como el comienzo de la incubación moderna.

## 2.2. La industria de la incubación actual

Para entender el mundo de la incubación actual hay que recurrir a la descripción de las dos tecnologías que siguen coexistiendo hoy en día: la incubación multicarga y la de carga única o simple.

*En una incubadora multicarga normalmente hay desde 3 hasta 6 desarrollos embrionarios distintos, de ahí el prefijo “multi”. Siendo un proceso continuo las dificultades para limpiar y desinfectar las mismas han supuesto que sea una tecnología claramente en retroceso, no suelen aparecer nuevos proyectos de plantas de incubación con la misma. La*

*Figura 2, muestra el interior de una incubadora de estas características [3].*



*Figura 2. Incubadora carga múltiple.*

<https://lohmann-breeders.com/es/sistemas-de-incubacion-etapa-multiple-y-unica/>

*Mientras, en una incubadora de carga simple todos los huevos están en el mismo desarrollo embrionario, cargándose y transfiriéndolos a las nacedoras al mismo tiempo, tras un tiempo de 18 días aproximadamente, teniendo margen para limpiar, desinfectar y secar perfectamente antes de realizar la siguiente carga. Por esto, y otros motivos que se comentan más adelante, es la tecnología predominante en el momento. La*

*Figura 3 muestra una imagen de una incubadora de carga simple [4].*



*Figura 3: Incubadora carga simple.*

<https://www.pasreform.com/en/smartsense>

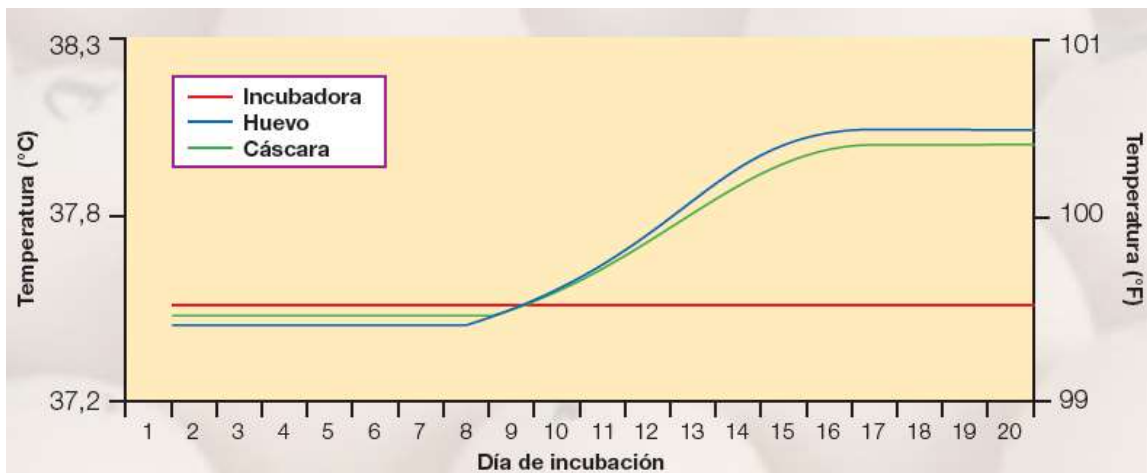
Tanto en uno como en otro tipo de proceso conviene entender que, en términos globales, durante los cinco primeros días es necesario aportar calor, el proceso es endotérmico, mientras que de ahí en adelante el mismo es exotérmico. Sin embargo, en la carga múltiple no podemos

adaptarnos en cada momento a las necesidades del embrión, ya que tenemos desarrollos embrionarios distintos.

Los parámetros principales que habitualmente se controlan en un proceso de incubación actual son:

- Temperatura seca.
- Temperatura húmeda o humedad relativa.
- Volteo de los huevos
- Dióxido de carbono.
- Ventilación.

Teniendo en cuenta que el objetivo durante el proceso de incubación es obtener una temperatura de cáscara lo más constante posible de 100 °F, en la *Figura 4* [5] se reflejan los registros de temperatura en carga múltiple mostrando la problemática que se presenta.

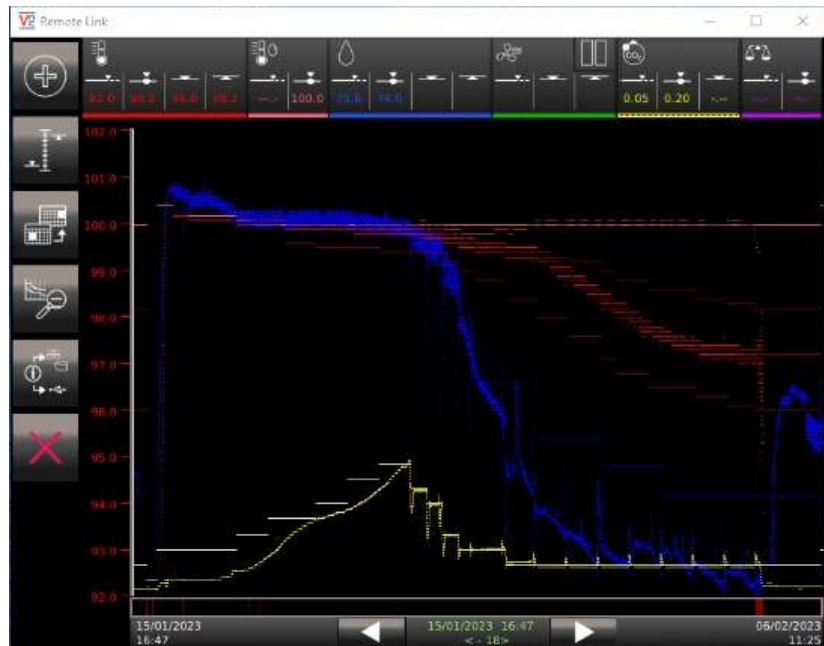


Figura

4: Registros temperaturas en carga múltiple.

Fuente: <https://redkitevets.co.uk/staging/wp-content/uploads/Avian-eggs-slides.pdf>

En carga simple, por el contrario, al tener todos los embriones unas necesidades muy similares en cada momento, sí es posible adaptarse a ellas, tal como se muestra en la *Figura 5*, en una pantalla de proceso de una incubadora modelo Petersime Biostream, donde la línea naranja representa la temperatura de cáscara de huevo, la roja la temperatura seca, la azul, temperatura húmeda y la amarilla la concentración de dióxido de carbono.



*Figura 5: Parámetros proceso carga única.  
Fuente: Propia*

*Otro factor para considerar es la realización de volteo durante el proceso de incubación [6]. El mismo se realiza generalmente en las máquinas incubadoras desde el inicio del proceso hasta el paso de los huevos a las máquinas nacedoras, siendo la recomendación general la rotación de los embriones en ángulos de 45° cada hora [7]. La*

Figura 6 muestra todos los carros de una incubadora en una planta de la compañía Premium Chick en India con un ángulo de 45°.



*Figura 6: Carros volteados durante incubación.  
Fuente: Propia*

## 2.3. Medidores de cáscara de huevo

Conocida desde la antigüedad la importancia de la monitorización de la temperatura de cáscara de huevo, dicha recomendación la podemos encontrar en distintas guías de incubación, desde termómetros médicos de oídos de infrarrojos, o por medio de la sonda de termistor [8].

*No es hasta 2008 cuando una de las compañías punteras del sector de la incubación industrial, Petersime [9] lanza un dispositivo capaz de medir la temperatura de las cáscaras de cuatro huevos contiguos al dispositivo, y con ello regular la temperatura de aire de la incubadora. Tres de estos dispositivos, cableados hasta la placa de control y colocados en unas posiciones fijas, dan una lectura media en pantalla de las temperaturas de las cáscaras, [10]. Un modelo de características y prestaciones similares es el Pilot de la empresa Jamesway, [11], [12], o el mencionado en la*

Figura 7, Tempo de la compañía ChickMaster [13].

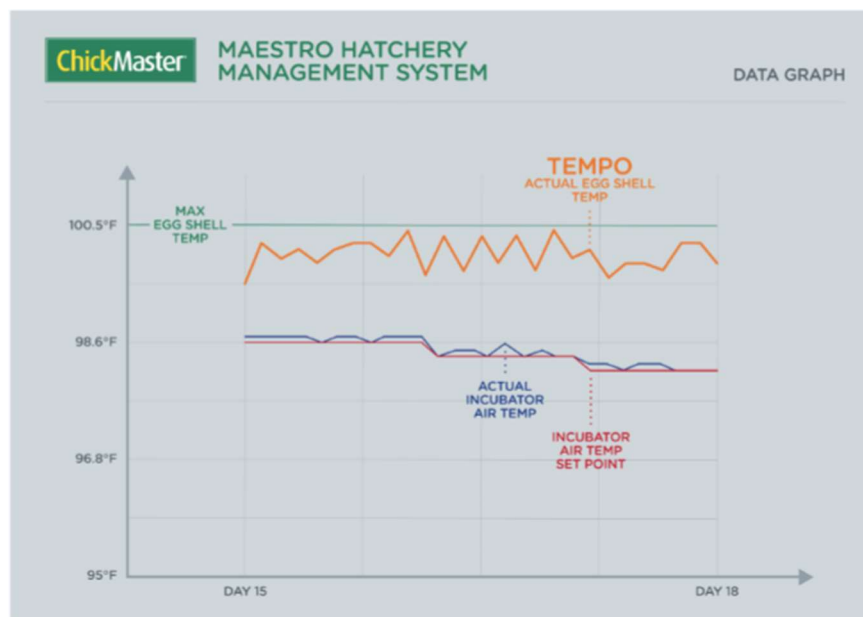


Figura 7: Gráfica de medida de Tempo.  
<https://parts.chickmaster.com/247d-02-5210.html>

## 2.4. Aporte del dispositivo objeto del proyecto

Hay que considerar los siguientes factores para entender el aporte que puede proporcionar TEVO en muchas plantas de incubación:

- Los modelos Ovoscan, Pilot y Tempo sólo funcionan para unos modelos específicos de las marcas Petersime, Jamesway y ChickMaster, respectivamente que no son lo mayoritario en el mercado.
- Aun en los modelos que pueden emplear Ovoscan o Pilot, dichos dispositivos realizan mediciones de unos puntos establecidos, no pudiéndose colocar los sensores en otras zonas de la incubadora. Además, no muestran en pantalla la lectura de cada uno de los

huevos contiguos que miden, sino una media.

- El resto de las soluciones no específicas del mundo de la incubación, como un termómetro médico de oídos de infrarrojos o una sonda termistor con recopilación de datos requiere en el primer caso parar el proceso momentáneamente y realizar medidas manuales, y en el segundo esperar a que éste acabe para ver los datos registrados, no pudiéndose ver insitu.



## Capítulo 3

# Materiales y métodos

### 3.1 Componentes

En este capítulo se comentan los condicionantes y la decisión final para la elección de cada uno de los componentes de TEVO.

El primer factor por considerar fue el soporte en el que iba a trabajar el dispositivo. Como se muestra en la

Figura 8, [14] existen distintos modelos en el mercado de bandejas de incubación con distintas disposiciones, lo cual condicionará el diseño de TEVO.



Figura 8. Disposiciones de bandejas de incubación.

<https://www.bestpoultrytech.com/en/products/setter-trays/setter-tray-154-chicken-eggs>

También hay que considerar que la altura entre bandejas dentro de un carro es algo superior a diez centímetros. Con estos condicionantes se decide que la forma más idónea para el dispositivo es la de un huevo de tamaño medio, conteniendo dos sensores infrarrojos de temperatura dentro del mismo. El dispositivo se podrá orientar de forma que los dos sensores midan a sendos huevos contiguos tanto si la disposición de la bandeja es convencional o en forma de colmena. El mismo ocupará el espacio de un huevo de incubación y no interferirá en el proceso.

Un huevo de incubación para un *broiler*, pollo de crecimiento rápido para producción de carne, de un lote de 54 semanas, ya próximo al fin de su vida productiva, tiene unas dimensiones aproximadas en su parte de mayor diámetro de 62 milímetros de alto por 46 de ancho, por lo que éstas serán las dimensiones que, inicialmente, se tomarán como referencia.

Además de que encaje dentro de estas dimensiones, hay que considerar que en muchas plantas de incubación no disponen de tecnología Wi-Fi (*Wireless Fidelity*), por lo que la placa del dispositivo deberá operar en modo SofAP (*Soft Access Point*) para poder comunicarse con un terminal, ya sea un móvil, una tableta o un ordenador portátil. En caso de que ésta falle,

sería conveniente que disponga de tecnología Bluetooth para garantizar la conectividad. En base a estas características se ha elegido la placa SP-Cow ESP32 que se muestra en la Figura 9.

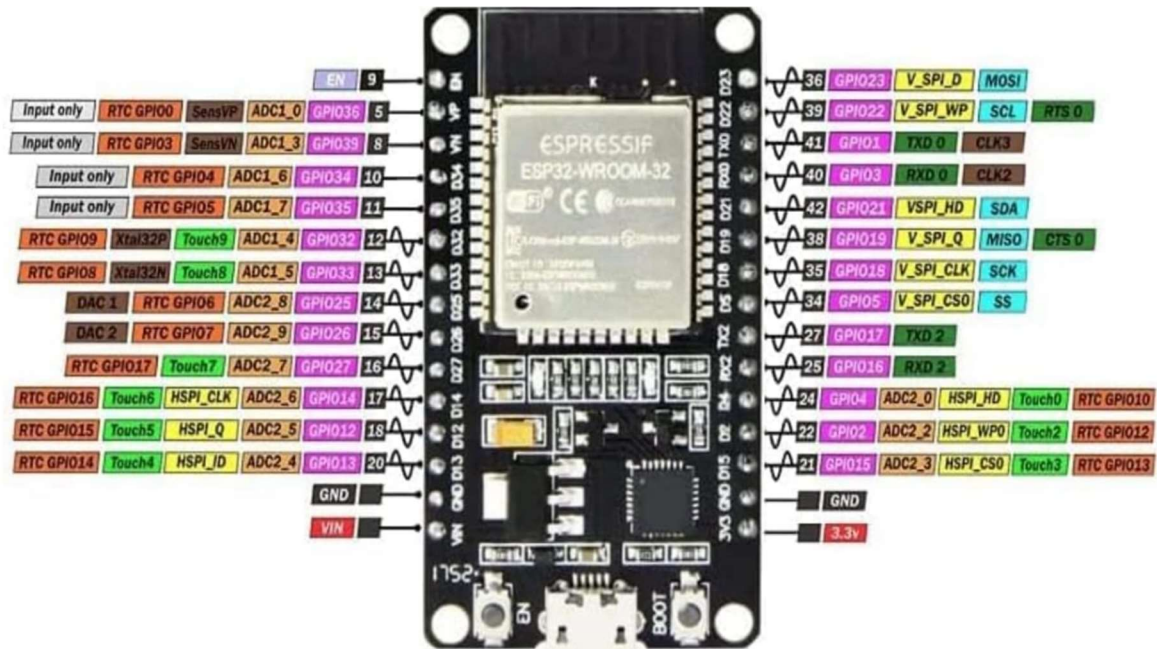


Figura 9. Pines de la placa SP-Cow ESP32.

Fuente: <https://techexplorations.com/guides/esp32/begin/power/>

Comparado con sus antecesores, como el ESP8266-01, el ESP32 ofrece significativas mejoras. Una de las principales ventajas es su doble núcleo, que permite manejar múltiples tareas simultáneamente, mejorando la eficiencia en la ejecución de código y en la gestión de recursos. Además, el ESP32 cuenta con más pines de entrada/salida (I/O), mayor memoria (tanto RAM como flash), y una serie de periféricos adicionales que facilitan la integración con sensores, actuadores, y otros dispositivos externos. Además, la capacidad del ESP32 para entrar en modos de bajo consumo, como el deep sleep, lo convierte en una opción superior para aplicaciones alimentadas por batería, permitiendo que los dispositivos funcionen durante períodos prolongados sin necesidad de recarga o reemplazo frecuente de la batería.

Puesto que el objeto de los sensores es medir la temperatura de cáscara de huevo durante el proceso de incubación, estos deberán tener una sensibilidad de al menos 0,1 °F y una precisión de 1°F, cuando se está en torno a los 100 °F, tanto de temperatura ambiente como de cáscara de huevo. Si además queremos medir la temperatura del huevo de incubación en otros puntos del proceso, como durante su almacenaje o transporte, los rangos de medida deberán ir desde unos 50 °F a 110 °F. Con esas características y analizando su hoja técnica [15], de la cual se muestra un gráfico de su precisión en la Figura 10, se eligen los sensores MLX90614ESF.

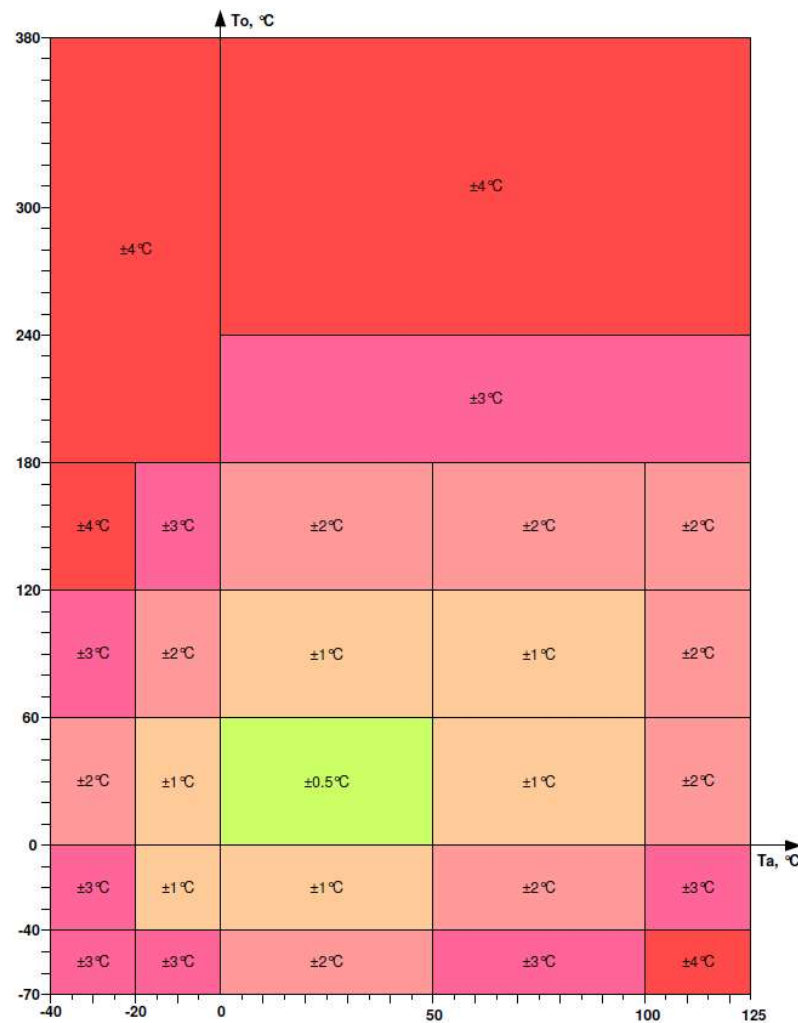


Figura 10: Precisión estándar del sensor MLX90614.

Fuente: <https://programarfácil.com/blog/arduino-blog/termometro-infrarrojo-con-arduino-mlx90614/>

Para conectar los sensores con la placa se efectuará por medio del bus de comunicación I<sup>2</sup>C (*Inter-integrated Circuit*) del ESP32. El protocolo I<sup>2</sup>C permite controlar varios dispositivos conectados únicamente a dos hilos. Estos dos hilos corresponden a la línea SDA (*System Data*) encargada de transmitir los datos y la línea SCL (*Serial Clock*) encargada de enviar los sincronismos de reloj.

A continuación, en la Figura 11 se puede ver el esquema de conexión de los dos sensores, que usarán la comunicación I<sup>2</sup>C donde se puede comprobar que únicamente se emplean esas dos líneas y, por supuesto, la alimentación, VCC, y la referencia a tierra, GND.

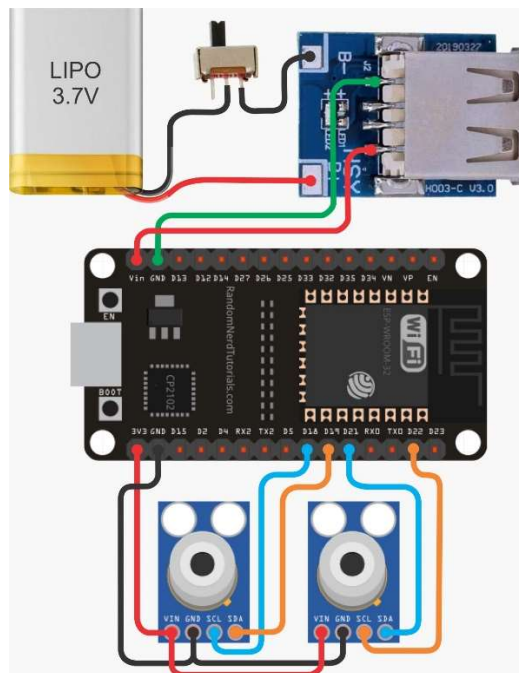


Figura 11. Esquema de las conexiones de los elementos.  
Fuente: Propia.

Dentro de ese bus de comunicación I<sup>2</sup>C, la placa ESP32 hará las funciones de maestro y los sensores MLX90614 de esclavos. Aunque ambos dispositivos pueden tener una dirección única y ser conectado a los mismos puertos, se decide hacerlo en puertos separados, tal como se muestra en la Figura 12, por varios motivos:

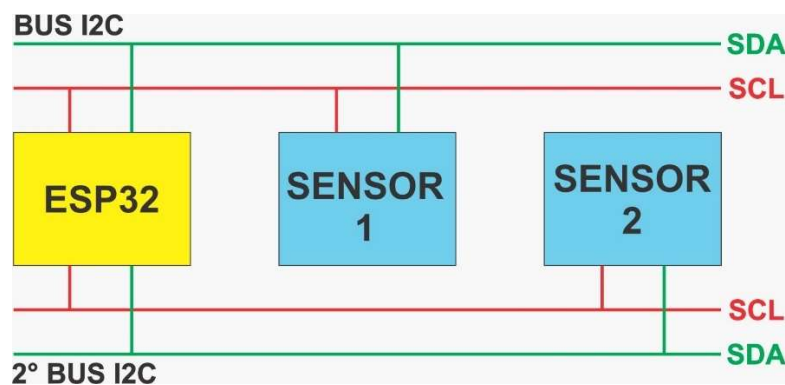


Figura 12. Esquema de la comunicación I2C.  
Fuente: Propia.

1. Como puede verse en la Figura 11, para el dispositivo en cuestión no es necesario emplear más que cuatro pines, dos para cada puerto, de todos los que tiene disponible la placa ESP32 (ver Figura 9), con los que se gestiona la comunicación, por lo que no hay un problema de disponibilidad de estos.
2. La diferencia en el espacio empleado entre cablear distintos puertos y hacer un puente desde los de un sensor al otro es insignificante.
3. En caso de que falle un puerto solamente quedaría inhabilitado un sensor, pudiendo

recibir datos del otro.

En la Figura 13 se presenta un esquema general de las relaciones entre los distintos componentes.

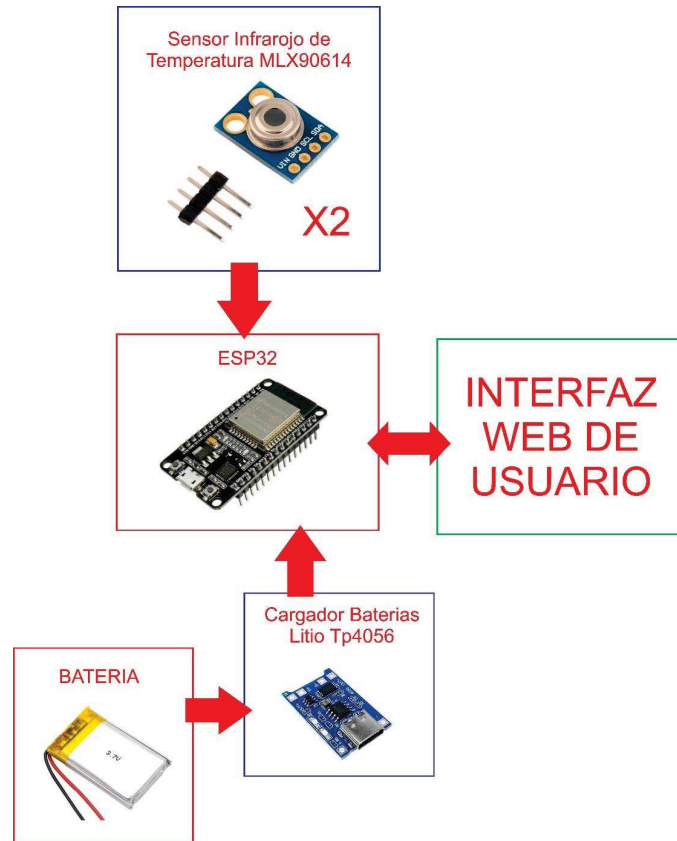


Figura 13. Relaciones entre los componentes del dispositivo.  
Fuente: Propia.

Con respecto al cargador, la Figura 14 en su parte superior muestra por donde se realizará la carga de la batería, mediante un micro-USB (*micro Universal Serial Bus*), y en la parte inferior, por donde se efectuará la salida de corriente. La pieza original, tal como se muestra en dicha figura, contaba con un USB (*Universal Serial Bus*); en la misma, pero se desuelda puesto que no tiene mucho sentido e incluso presenta un impedimento por tamaño para este proyecto.

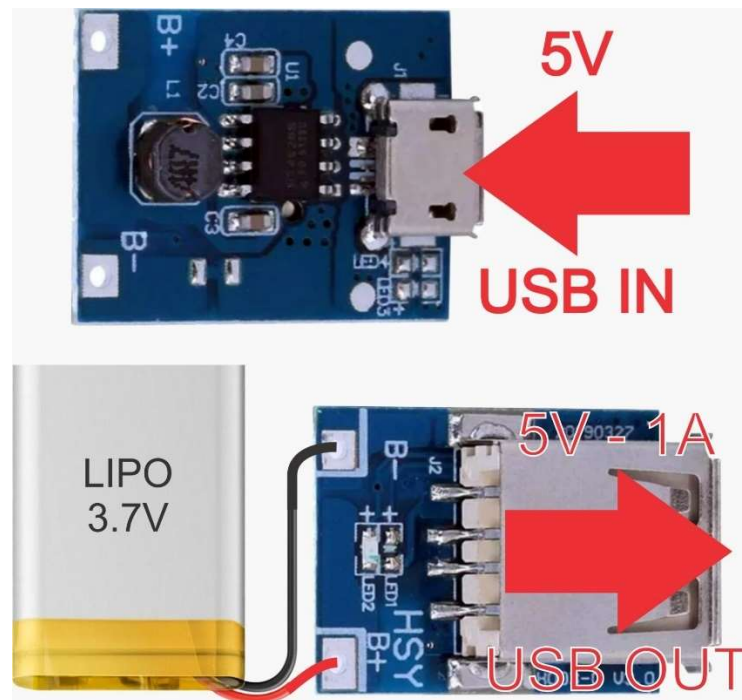


Figura 14. Anverso y reverso del cargador.  
Fuente: Propia

### 3.2. Precios de los materiales empleados

A continuación, se enumeran los materiales empleados en la construcción del prototipo junto con su precio y correspondientes figuras:

- TECNOIOT 2pcs GY-906 MLX90614 Non-Contact Infrared Temperature Sensor Module MLX90614ESF: 33,04 €.



Figura 15. Sensores TECNOIOT MLX90614ESF.  
Fuente: <https://www.amazon.es/Hailege-MLX90614-Non-Contact-Temperature-Raspberry/dp/B07YKNQQ7P>

- SP-Cow ESP32 módulo Wi-Fi placa de desarrollo, ESP-WROOM-32 ESP-32 ESP-32S 2.4GHz Wi-Fi + Bluetooth modo dual. Módulo de antena con Ultra Low Power (1 Pack): 10,59 €.



Figura 16. Placa ESP32.

Fuente: <https://forum.arduino.cc/t/alimentar-esp32-sin-el-puerto-usb/523034>

- Batería de litio recargable Mod. Xyh 902030 de 3.7V con capacidad de 460 Mah, tamaño 9 mm. x 20 mm. x 30 mm: 4,9 €.



Figura 17: Batería Litio recargable.

Fuente: <https://www.amazon.es/Bateria-Recargable-902030-Capacidad-Tama%C3%B1o/dp/B09TPNJZ91>

- DAOKAI 1 circuito de carga para dispositivos de 5 V, incluye protección de carga y módulo de potencia de aumento 134N3P, conector micro-USB: 1,12 €

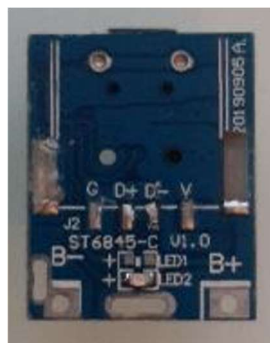


Figura 18: Cargador DAOKAI.

Fuente: Propia.

- 4 piezas M3 Insertos de juego de calor roscado impresión 3D, tuercas de latón moleteadas (M3\*4x5mm): 0,27 €.



Figura 19: Inserto para la impresión 3D.

Fuente: <https://www.amazon.es/Impresi%C3%B3n-moleteadas-inserci%C3%B3n-proyectos-impresi%C3%B3n/dp/B0BYVLP4R>

- 1 micro interruptor deslizante Spdt, 3 pines, 5 posiciones, Mini ss12f15, ss12f44, 1p2t, 5mm, palanca pequeña, Terminal de montaje PCB para Arduino: 0,1 €.



Figura 20: Micro interruptor.

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

### 3.3. Métodos y programación realizada

En este apartado se explica cómo se ha realizado la comunicación entre los componentes y los elementos terminales, a la par que se muestran las soluciones elegidas.

#### 3.3.1. Conexión placa con sensores

A continuación, se implementa físicamente en una *protoboard* las conexiones de los sensores con la placa ESP32, primero únicamente con uno de ellos, tal como se muestra en la Figura 21, y se desarrolla su código en Arduino (Listado 1).

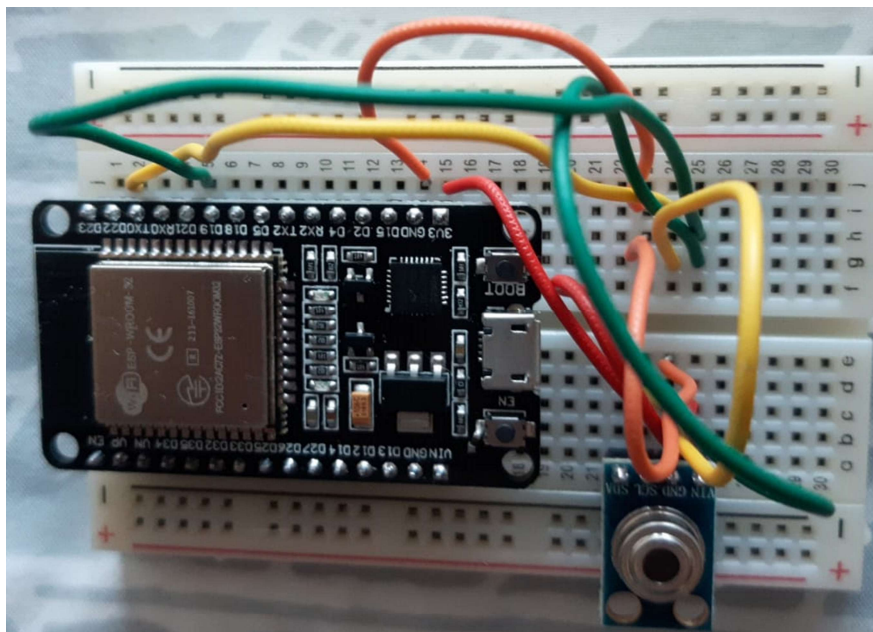


Figura 21. Placa y sensor en protoboard.

Fuente: Propia.

```

#include <Wire.h>
#include <Adafruit_MLX90614.h>

Adafruit_MLX90614 mlx1;
Adafruit_MLX90614 mlx2;

void setup() {
  Serial.begin(9600);

  // Inicializar el bus I2C para cada sensor
  Wire.begin(21, 22); // Puertos I2C diferentes, puedes ajustar estos pines
según tu configuración
  mlx1.begin();

  Wire1.begin(18, 19); // Puertos I2C diferentes, puedes ajustar estos pines
según tu configuración
  mlx2.begin(MLX90614_I2CADDR,&Wire1);
}

void loop() {
  // Leer temperaturas de ambos sensores
  float temp1 = mlx1.readObjectTempF();
  float temp2 = mlx2.readObjectTempF();

  // Imprimir temperaturas
  Serial.print("Temperatura Sensor 1: ");
  Serial.print(temp1);
  Serial.println(" °F");

  Serial.print("Temperatura Sensor 2: ");
  Serial.print(temp2);
  Serial.println(" °F");

  delay(1000);
}

```

Listado 1: Código de Arduino que muestra lecturas en el monitor serial.

### 3.3.2. Interfaz gráfica de TEVO

El resultado de la ejecución de dicho código puede verse en la Sección “4.1. Temperaturas en monitor serial Arduino”. La siguiente fase es crear una interfaz gráfica en HTML (*Hypertext Markup Language*) que permita visualizar los sensores. En el Listado 2 se muestra el código para la citada interfaz.

```

<!DOCTYPE html>
<html>

```

```
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width">
  <title>TEVO</title>
  <style>
    body {
      font-family: monospace;
      background-color: rgb(20, 165, 212);
      text-align: center;
    }
    h1 {
      color: white;
      text-align: center;
      margin: 30px;
      font-size: 22px;
    }
    .barra {
      height: 200px;
      width: 200px;
      margin: auto;
      border-radius: 50%;
      overflow: hidden;
    }
    .cont {
      height: 100px;
      width: 200px;
    }
    .circulo_centro {
      width: 150px;
      height: 150px;
      background-color: rgba(0, 21, 38, 50);
      border-radius: 50%;
      margin: auto;
      position: relative;
      margin: -175px auto 50px;
    }
    .rectangulo {
      width: 200px;
      height: 100px;
      background-color: rgba(0, 21, 38, 50);
      position: relative;
```

```

    margin: -125px auto 50px;
  }
  .ContSensor {
    display: inline-block;
    background-color: rgba(0, 21, 38, 50);
    border-radius: 15px;
    padding: 5;
    text-align: center;
    width: 280px;
    margin-top: 10px;
    margin-bottom: 10px;
    margin: 10px;
    position: relative; /* Agregamos posición relativa para los contenedores */
  }
  .titulo-negro {
    color: black;
    text-align: center;
    margin: 30px;
    font-size: 24px;
  }
  .linea-vertical {
    position: absolute;
    width: 2px;
    height: 100px;
    background-color: rgb(20, 165, 212);
    left: 50%;
    top: 25%; /* Ajustamos la posición para que esté sobre la barra y el círculo_centro */
    transform: translateX(-50%);
    z-index: 1;
  }
</style>
</head>

<body>
  <h1 class="titulo-negro">Visualización de Temperatura</h1>

  <div class="ContSensor">
    <h1>Temperatura 1</h1>
    <div class="linea-vertical"></div> <!-- Línea vertical agregada -->
    <div class="barra" style="transform: rotate(100deg);">
      <div class="cont" style="background-color: rgb(143, 136, 136);"></div>

```

```

    <div class="cont" style="background-color: rgb(180, 0, 0);"></div>
</div>
<div class="circulo_centro"></div>
<div class="rectangulo"></div>
<h1 style="font-size: 22px; position: relative; margin: -115px auto 40px;">24.9
°F</h1>
</div>

<div class="ContSensor">
  <h1>Temperatura 1</h1>
  <div class='linea-vertical'></div> <!-- Línea vertical agregada -->
  <div class="barra" style="transform: rotate(100deg);">
    <div class="cont" style="background-color: rgb(143, 136, 136);"></div>
    <div class="cont" style="background-color: rgb(180, 0, 0);"></div>
  </div>
  <div class="circulo_centro"></div>
  <div class="rectangulo"></div>
  <h1 style="font-size: 22px; position: relative; margin: -115px auto 40px;">24.9
°F</h1>
</div>

</body>

</html>

```

Listado 2: Código HTML con la interfaz de Tevo.

### 3.3.3. Visualización TEVO en terminal

A continuación, se integró el código de la interfaz de visualización efectuado en HTML dentro del de lectura de los sensores de Arduino, de manera que las temperaturas mostradas fueran las temperaturas medidas por los mismos, y refrescadas cada dos segundos. Además, se crea la red WI-FI, “TEVO\_1”, que producirá la placa, para poder conectarse desde un móvil u otro terminal y se configura un estado “modem sleep” para aumentar la duración de la batería, ya que en las primeras pruebas se comprueba que, sin éste, la misma funciona únicamente durante unas dos horas y media. El Listado 3 muestra el código completo:

```

include <WiFi.h>
#include <WebServer.h>
#include <Wire.h>
#include <Adafruit_MLX90614.h>
#include <esp_wifi.h>

```

```

Adafruit_MLX90614 mlx1;
Adafruit_MLX90614 mlx2;

const char* ssid = "TEVO_1"; // Nombre del punto de acceso
IPAddress ip(192, 168, 1, 22); // Creamos la dirección IP fija
IPAddress PuertaEnlace(192, 168, 1, 1);
IPAddress MascaraRed(255, 255, 255, 0);
WebServer server(80); // Puerto 80 para el servidor web

void setup() {
  Serial.begin(9600);

  // Inicializar el bus I2C para cada sensor (SDA,SCL)
  Wire.begin(21, 22); // Puertos I2C diferentes, puedes ajustar estos pines se-
  gún tu configuración
  mlx1.begin();

  Wire1.begin(19, 18); // Puertos I2C diferentes, puedes ajustar estos pines
  según tu configuración
  mlx2.begin(MLX90614_I2CADDR, &Wire1);

  // Iniciar el punto de acceso
  WiFi.softAP(ssid);
  WiFi.softAPConfig(ip, PuertaEnlace, MascaraRed);
  Serial.println("Punto de acceso creado");

  IPAddress IP = WiFi.softAPIP();
  Serial.print("IP del ESP32: ");
  Serial.println(IP);

  server.on("/", handleRoot); // Ruta raíz
  server.begin();

  // Configurar el WiFi en modo modem sleep
  WiFi.setSleep(WIFI_PS_MIN_MODEM);
}

void loop() {
  server.handleClient();
  delay(10); // Mantenerse despierto 10 ms
  server.handleClient();
  // Entrar en modem sleep
  Serial.println("Entrando en modem sleep...");
  esp_wifi_set_ps(WIFI_PS_MIN_MODEM); // Activar modem sleep
  delay(10 * 1000); // Dormir por 10 segundos
  // Salir de modem sleep
  Serial.println("Saliendo de modem sleep...");
  esp_wifi_set_ps(WIFI_PS_NONE); // Desactivar modem sleep
}

```

```

void handleRoot() {
  // Leer temperaturas de ambos sensores
  float temp1 = mlx1.readObjectTempF();
  float temp2 = mlx2.readObjectTempF();
  float temp1map = map(temp1, 32, 168, 0, 180);
  float temp2map = map(temp2, 32, 168, 0, 180);
  String page = "<!DOCTYPE html>"
    "<html>"
    "<head>"
    "<meta charset='utf-8'>"
    "<meta http-equiv='refresh' content='2'>" // Actualizamos la
página cada 2 s
    "<meta name='viewport' content='width=device-width'>"
    "<title>TEVO</title>"
    "<style>"
    "body {"
    "font-family: monospace;"
    "background-color: rgb(20, 165, 212);"
    "text-align: center;"
    "}"
    "h1 {"
    "color: white;"
    "text-align: center;"
    "margin: 30px;"
    "font-size: 22px;"
    "}"
    ".barra {"
    "height: 200px;"
    "width: 200px;"
    "margin: auto;"
    "border-radius: 50%;"
    "overflow: hidden;"
    "}"
    ".cont {"
    "height: 100px;"
    "width: 200px;"
    "}"
    ".circulo_centro {"
    "width: 150px;"
    "height: 150px;"
    "background-color: rgba(0, 21, 38, 50);"
    "border-radius: 50%;"
    "margin: auto;"
    "position: relative;"
    "margin: -175px auto 50px;"
    "}"
    ".rectangulo {"
    "width: 200px;"
    "height: 100px;"
    "background-color: rgba(0, 21, 38, 50);"

```

```

"position: relative;"
"margin: -125px auto 50px;"
}"
".ContSensor {"
"display: inline-block;"
"background-color: rgba(0, 21, 38, 50);"
"border-radius: 15px;"
"padding: 5;"
"text-align: center;"
"width: 280px;"
"margin-top: 10px;"
"margin-bottom: 10px;"
"margin: 10px;"
}"
".titulo-negro {"
"color: black;"
"text-align: center;"
"margin: 30px;"
"font-size: 24px;"
}"
"</style>"
"</head>"
"<body>"
"<h1 class='titulo-negro'>Visualización de Temperatura</h1>"
"<div class='ContSensor'>"
"<h1>Temperatura 1</h1>"
"<div class='barra' style='transform: rotate("
+ String(temp1map) + "deg);'>"
"    <div class='cont' style='background-color:
rgb(143, 136, 136);'></div>"
"    <div class='cont' style='background-color:
rgb(180, 0, 0);'></div>"
"    </div>"
"    <div class='circulo_centro'></div>"
"    <div class='rectangulo'></div>"
"    <h1 style='font-size: 22px; position: rel-
ative; margin: -115px auto 40px;'>"
"        + String(temp1) + " °F</h1>"
"    </div>"
"    <div class='ContSensor'>"
"        <h1>Temperatura 2</h1>"
"        <div class='barra' style='transform: rotate("
+ String(temp2map) + "deg);'>"
"            <div class='cont' style='background-color:
rgb(143, 136, 136);'></div>"
"            <div class='cont' style='background-color:
rgb(180, 0, 0);'></div>"
"            </div>"
"            <div class='circulo_centro'></div>"
"            <div class='rectangulo'></div>"

```

```
        "<h1 style='font-size: 22px; position: rel-  
ative; margin: -115px auto 40px;'"  
        + String(temp2) + " °F</h1>"  
        "</div>"  
        "</body>"  
        "</html>";  
  
    server.send(200, "text/html", page);  
}
```

Listado 3: Código completo.

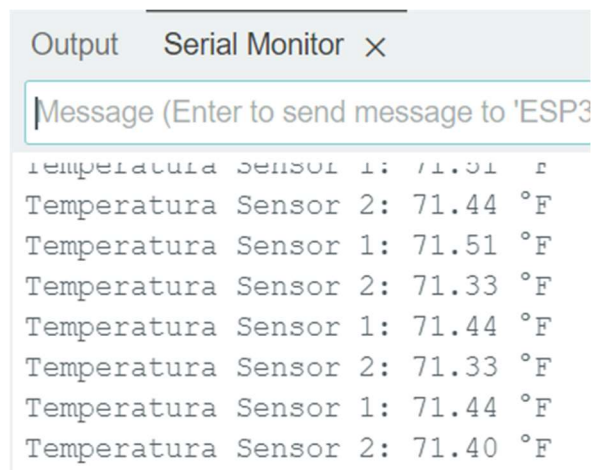
El resultado se muestra en el Apartado 4.3. Visualización de la interfaz en un terminal

# Capítulo 4

## Resultados

### 4.1. Temperaturas en monitor serial Arduino

Tras distintas pruebas y modificaciones del código de Arduino mostrado en el Apartado 3.3.1, se obtienen los resultados mostrados en la Figura 22, medidos en una sala en torno a 22 °C, donde podemos comprobar que ambos sensores se comunican perfectamente con la placa.

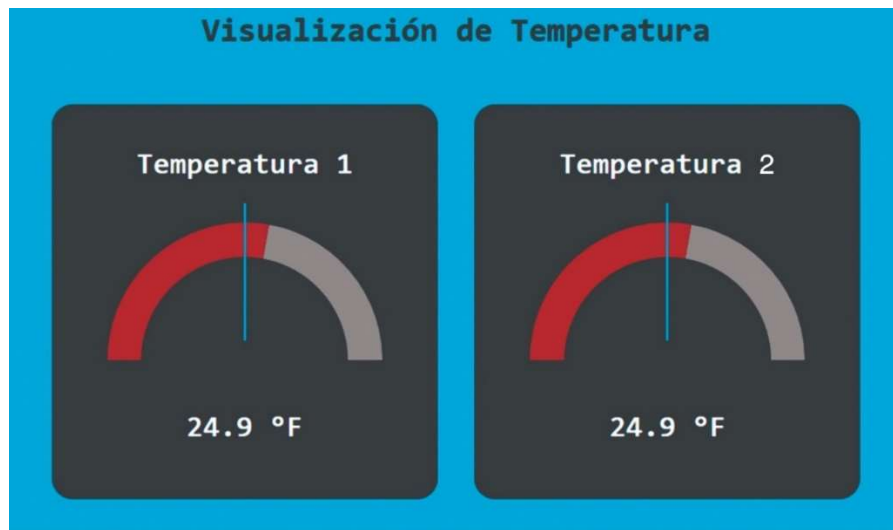


```
Output Serial Monitor x
Message (Enter to send message to 'ESP3
Temperatura Sensor 1: 71.51 °F
Temperatura Sensor 2: 71.44 °F
Temperatura Sensor 1: 71.51 °F
Temperatura Sensor 2: 71.33 °F
Temperatura Sensor 1: 71.44 °F
Temperatura Sensor 2: 71.33 °F
Temperatura Sensor 1: 71.44 °F
Temperatura Sensor 2: 71.40 °F
```

*Figura 22. Pruebas en monitor serial de Arduino.  
Fuente: Propia.*

### 4.2 Interfaz visualización dispositivo en Arduino

La Figura 23 muestra el resultado de la implementación del código HTML embebido en Arduino, mostrado en el Apartado 3.3.2, con objeto de representar gráficamente la lectura de ambos sensores.

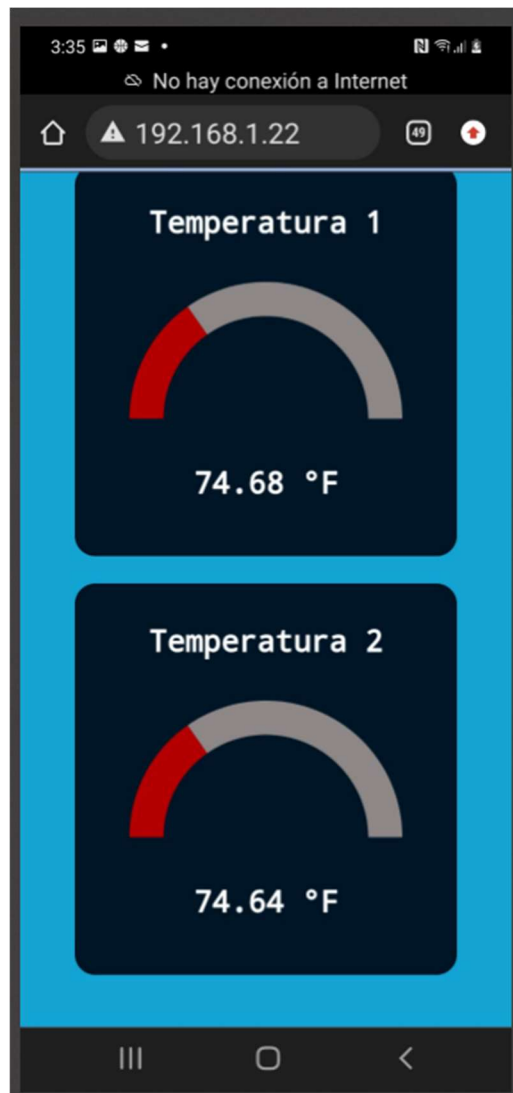


*Figura 23. Interfaz visualización temperaturas.  
Fuente: Propia*

### **4.3. Visualización de la interfaz en un terminal**

La Figura 24 muestra cómo se visualiza la interfaz de TEVO en un teléfono Android. La comunicación se ha producido mediante la red Wi-Fi producida por la placa ESP32. El programa es el mostrado en el apartado “Visualización TEVO en terminal”.

El mensaje “No hay conexión a Internet” refleja el hecho de que el dispositivo móvil se está conectando directamente al servidor alojado en la placa ESP32 y por esta razón se requiere especificar la IP configurada en la misma.



*Figura 24. Visualización en Android.  
Fuente: Propia.*

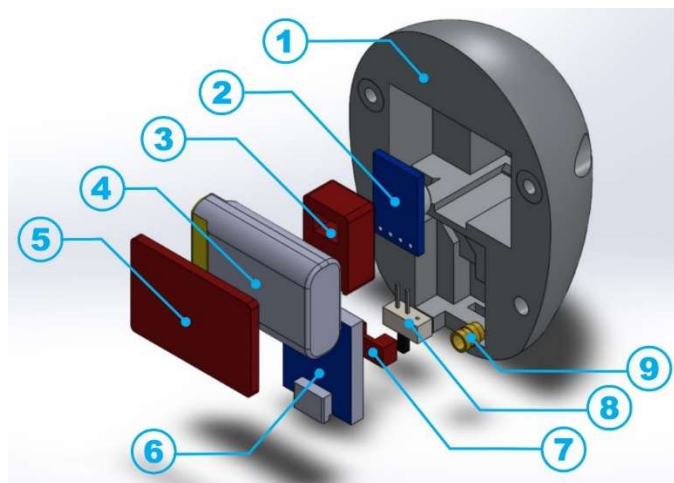


# Capítulo 5

## Modelado

Aunque en el presente documento aparecen separados los procesos de modelado y ensamblaje, en realidad se trató más de un proceso iterativo que lineal, en el cual después de terminar el modelo inicial y empezar a ensamblar hubo que modificar algunas piezas con el objetivo de coincidir con las tolerancias reales de los elementos dispuestos en el interior del dispositivo. Este proceso iterativo se puede observar en detalle en el Capítulo 6 Ensamblaje.

Como se ha comentado anteriormente, TEVO exteriormente presentará la forma de un huevo de 62 mm x 46 mm. Por tanto, el primer criterio para el modelado de éste va a ser el espacio. Para facilitar el montaje y el ensamblado se separará la carcasa en dos mitades simétricas. La Figura 25 muestra un explosionado de la primera de ellas.



*Figura 25: Explosionado piezas carcasa 1.  
Fuente: Propia.*

En la misma Figura 25 se muestran numeradas:

- 1: La carcasa 1 de TEVO.
- 2: El sensor infrarrojo de temperatura MLX90614.
- 3: El acople de sujeción del sensor de temperatura.
- 4: La batería.
- 5: El acople de sujeción de la batería.
- 6: El cargador de la batería.
- 7: El acople de sujeción del interruptor.
- 8: El interruptor general.
- 9: El inserto para un roscado M3.

Para medir correctamente la temperatura de cáscara de su huevo contiguo, el sensor debe estar colocado a una altura ligeramente superior a la mitad del huevo siempre considerando como base la parte más puntiaguda. Tal como se verá en la Figura 26, donde va la placa ESP32, todos los demás elementos, a diferencia del segundo sensor, por disponibilidad de espacio deberán estar ubicados en la mitad representada en la Figura 25. El cargador se decide colocar con su micro-USB lo más centrado posible y en la parte baja de TEVO, para evitar entrada de agua o de líquido proveniente de la explosión de un huevo cercano, ya que si el cargador se posicionara en la parte superior el líquido entraría directamente al interior del dispositivo por efecto de la gravedad. Con el condicionante de falta de espacio, la batería se coloca en la parte superior del cargador, ya que es el único lugar que queda libre y, de esta forma, el conexionado con el mismo se puede realizar con facilidad. El interruptor, por el mismo criterio del cargador, se colocará en la parte inferior, al lado de éste. Además, para evitar consumo cuando el dispositivo esté apagado, éste cortará la comunicación entre la batería y el cargador, por lo que esa ubicación, al lado de ambos, también facilita su conexión.

Con respecto a la conexión entre las dos mitades del dispositivo, mostradas en el ensamblaje de la Figura 25 y la Figura 26, se realizará mediante seis cables: dos de alimentación de la batería hacia el ESP32 y cuatro de la placa al sensor de la Figura 25, dos de alimentación a 3,3 voltios, y dos de datos.

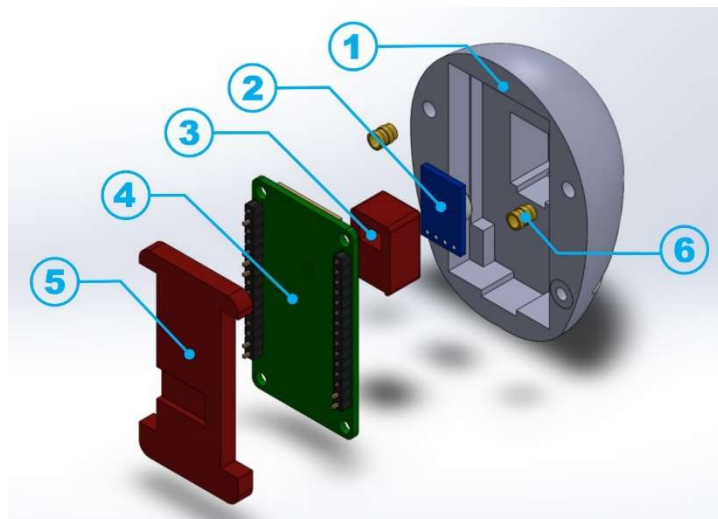
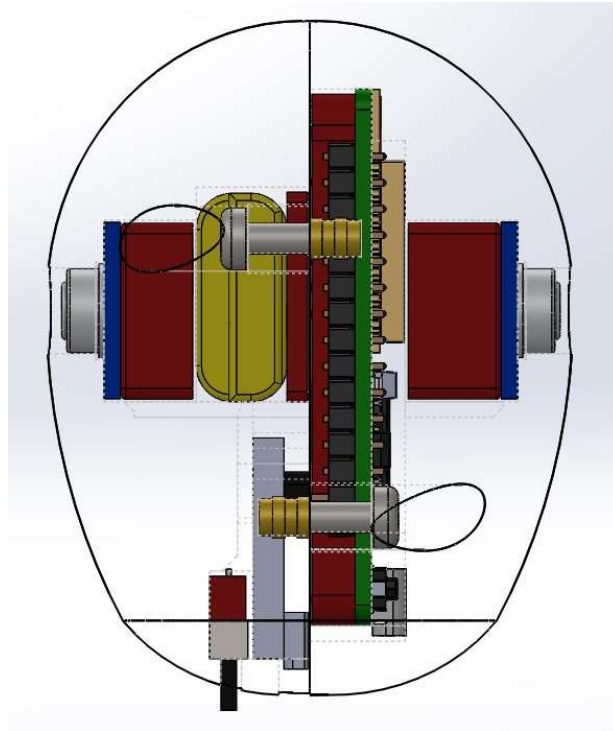


Figura 26: Explosión de las piezas de la carcasa 2.  
Fuente: Propia.

En la Figura 26, se pueden ver numeradas:

- 1: La carcasa 2 de TEVO.
- 2: El sensor infrarrojo de temperatura MLX90614.
- 3: El acople de sujeción del sensor de temperatura.
- 4: La placa ESP32.
- 5: El acople de sujeción de la placa.
- 6: El inserto para un roscado M3.

Mientras que en la Figura 27 se aprecian con mayor detalle las piezas de las que se compone TEVO una vez ensamblado.



*Figura 27: Detalle de TEVO.  
Fuente: Propia.*



## Capítulo 6

### Ensamblaje

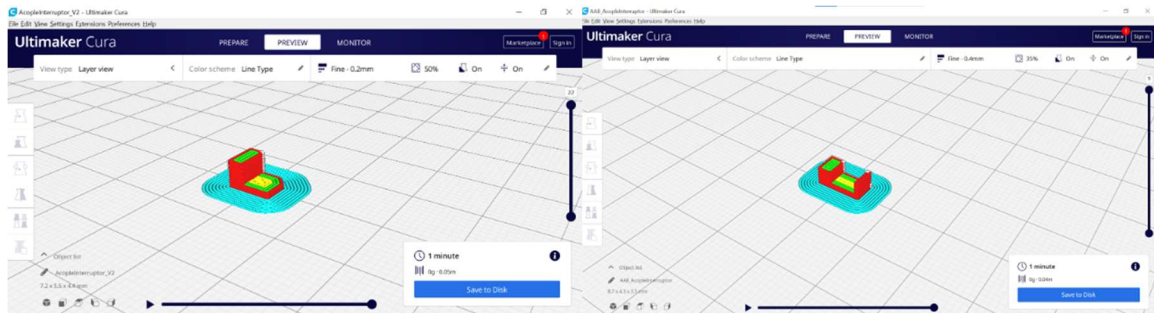
Una vez finalizado el modelado se realizan distintas impresiones en 3D, con una altura de capa de 0,2 mm y un porcentaje de relleno del 50%, tanto la carcasa como los acoples de sujeción de los componentes. Distintos componentes de los prototipos y ensamblado son mostrados en la Figura 28.



*Figura 28: Prototipos de TEVO y acoples.  
Fuente: Propia.*

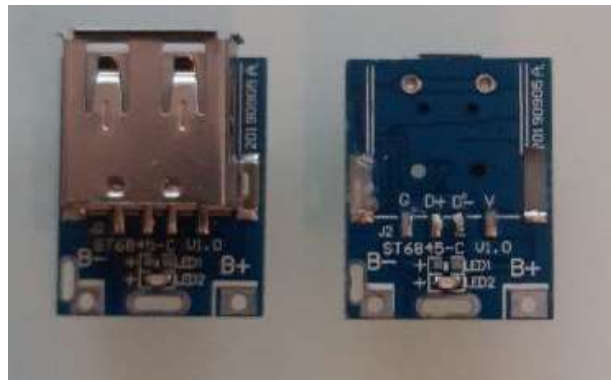
Al iniciar el proceso de ensamblaje del dispositivo se hizo evidente la necesidad de redimensionar alguna pieza, en especial los componentes internos, en función de las tolerancias reales de las diferentes conexiones. La Figura 29 es un ejemplo de estas modificaciones, en donde se observa que se redujo el tamaño de la pieza original y se modificó la forma con el objetivo de que encajara de forma precisa en el ensamblaje. Este tipo de

modificaciones refleja que el proceso de modelado y ensamblaje no fue lineal, sino que se trató de un proceso iterativo en donde a medida que se iba ensamblando el dispositivo se realizaban los ajustes dimensionales necesarios para mejorar el encaje de todas las piezas.



*Figura 29: Modelo inicial y final del acople del interruptor.  
Fuente: Propia.*

Con respecto al cargador, como ya se mostraba en el Capítulo 5 Modelado, hay que retirar la pieza hembra del conector USB ya que no se va a utilizar y no hay espacio dentro de TEVO para el mismo. En la Figura 30 podemos ver a la izquierda la pieza original y a la derecha, una vez retirado el cargador hembra USB.



*Figura 30: Cargador con y sin conector USB.  
Fuente: Propia.*

Durante el ensamblado se realizan diversas verificaciones, como el mostrado en la Figura 31, en la que se comprueba que el cargador funciona, conectando su salida a una placa de Arduino, que funciona a 5V.



*Figura 31: Comprobación cargador.  
Fuente: Propia.*

Uno de los problemas que TEVO con casi toda seguridad va a encontrarse en casi todos los modelos de máquinas incubadoras es el relacionado con el volteo [7]. Aunque este se realice de forma suave, va a suponer una rotación de éste dentro de la bandeja de incubación, modificando la dirección en la que los sensores medirán. Para evitar la rotación de este, se diseñan varios soportes para las bandejas como los que se muestran en la Figura 32, los cuales se probarán posteriormente.



*Figura 32: Soportes de Tevo.  
Fuente: Propia.*

La Figura 33 y Figura 34 muestran respectivamente los prototipos de Tevo con distintos tipos de soportes como los mostrados en la Figura 32, y colocados en una bandeja de incubación.



*Figura 33: Prototipos con soportes.  
Fuente: Propia.*



*Figura 34: Prototipos de Tevo en la bandeja de incubación.  
Fuente: Propia.*

# Capítulo 7

## Pruebas industriales

Las pruebas industriales se llevaron a cabo en la planta de incubación de la empresa avícola Pondex, perteneciente al grupo Vall Companys, en su planta de la localidad de Sástago, Zaragoza, el día 31 de julio de 2024. La Figura 35 muestra imágenes del pasillo de incubación y de las máquinas de la marca Petersime modelo BioStreamer en las que se testó Tevo.



*Figura 35: Incubadoras Petersime BioStreamer.  
Fuente: Propia.*

Como puede verse en la Figura 36, se colocó Tevo con varios soportes en las bandejas de incubación comprobándose que su fijación es adecuada pero también que la altura con la que queda con respecto a los huevos que ha de medir es superior a la deseada.



*Figura 36: Tevo con soportes.  
Fuente: Propia.*

Debido a esto se decide colocar Tevo sin soportes, tal como aparece en la Figura 37, comprobando que su fijación es adecuada, incluso tras realizar varios volteos [6] en la incubadora.



*Figura 37: Tevo sin soportes.  
Fuente: Propia.*

También se comprueba y muestra en la Figura 38 que el prototipo de Tevo colocado dentro de la incubadora mide y transmite correctamente.



*Figura 38: Visualización de la medición en incubadora.  
Fuente: Propia.*

# Capítulo 8

## Conclusiones y trabajos futuros

Una vez realizado todo el prototipado de Tevo, con su modelado y ensamblaje, y llevadas a cabo pruebas industriales en una planta de incubación podemos llegar a las siguientes conclusiones:

- Tevo cumple su objetivo principal de proporcionar datos de temperatura de cáscara de huevo en tiempo real, pudiendo ser mostrada en un Android o cualquier otro dispositivo con acceso a Wi-Fi.
- El dispositivo se adapta a una bandeja industrial, siendo suficientemente robusto como para soportar las condiciones específicas de una incubadora, incluso no siendo un problema el volteo [6].

Sin embargo, para hacer más atractivo el dispositivo se trabajará en las siguientes mejoras:

- Una posibilidad para aumentar la independencia de la batería sería ofrecer la alternativa de que el Wi-Fi del dispositivo no funcionara constantemente sino cada cierto tiempo, y subiera los datos a una nube como Google Drive o Microsoft OneDrive.
- Aunque Tevo encajó dentro del alveolo o bandeja de incubación lo haría más si fuera menos ancho, por lo que debería implementarse este cambio de diseño en el siguiente prototipo.



# Bibliografía y referencias

- [1] H. J. Wijnen, H. van der Brand, A. Lammers, I. A. M. van Roover-Reijrink, C. W. van der Pol, B. Kemp and R. Molenaar, “Effects of eggshell temperature pattern during incubation on primary immune organ development and broiler immune response in later life,” *Poultry Science*, vol. 99, no. 6, pp. 3020-3029, 2020.
- [2] V. Traverso. (2019, March). [Online]. Available: <https://www.atlasobscura.com/articles/egypt-egg-ovens>.
- [3] Lohmann-Breeders. (2024, March 31). [Online]. Available: <https://lohmann-breeders.com/es/sistemas-de-incubacion-etapa-multiple-y-unica/>.
- [4] Royal Pas Reform, [Online]. Available: <https://www.pasreform.com/en/smartsense>.
- [5] H. N. S. Tazawa, «Response of egg temperature, heart rate and blood pressure in the chick embryo to hypothermal stress,» *Journal of Comparative Physiology B*, vol. 155, pp. 195 - 200, 1985.
- [6] D. C. Deeming, «The role of egg turning during incubation,» *Avian Biology Research*, vol. 2, n° 1-2, pp. 67-71, 2013.
- [7] D. C. Deeming, «Characteristics of unturned eggs: critical period, retarded embryonic growth and poor albumen utilisation,» *British Poultry Science*, vol. 30, n° 3, pp. 591-606, 1989.
- [8] Aviagen, *Uso correcto de los registradores de datos Tinytag*, 2022, pp. 51 - 52.
- [9] Petersime, [Online]. Available: <https://www.petersime.com/>.
- [10] Agrinews. (2017, April 20). [Online]. Available: <https://www.youtube.com/watch?v=7gpuVznjxnc>.
- [11] N. Kinsley, «Jamesway’s Pilot measures incubator conditions,» *Poultry World*, 04 May 2010.
- [12] Jamesway. [Online]. Available: <https://jamesway.com/storage/2019/04/Pilot-System-ESP-for-Screens.pdf>.
- [13] ChickMaster. [Online]. Available: <https://parts.chickmaster.com/247d-02-5210.html>.
- [14] Bestpoultrytech. (2016). [Online]. Available: [www.bestpoultrytech.com/en/products/setter-trays/setter-tray-154-chicken-eggs](http://www.bestpoultrytech.com/en/products/setter-trays/setter-tray-154-chicken-eggs).
- [15] Microelectronic Integrated Systems, *MLX90614 family*, 13 ed., 2019, p. 34.