

RELOJ DE PANTALLA DE TABLILLAS

“TABLEX”

Por Pedro Eisman Cabado, octubre 2025

- GUIA DE OPERACIÓN -

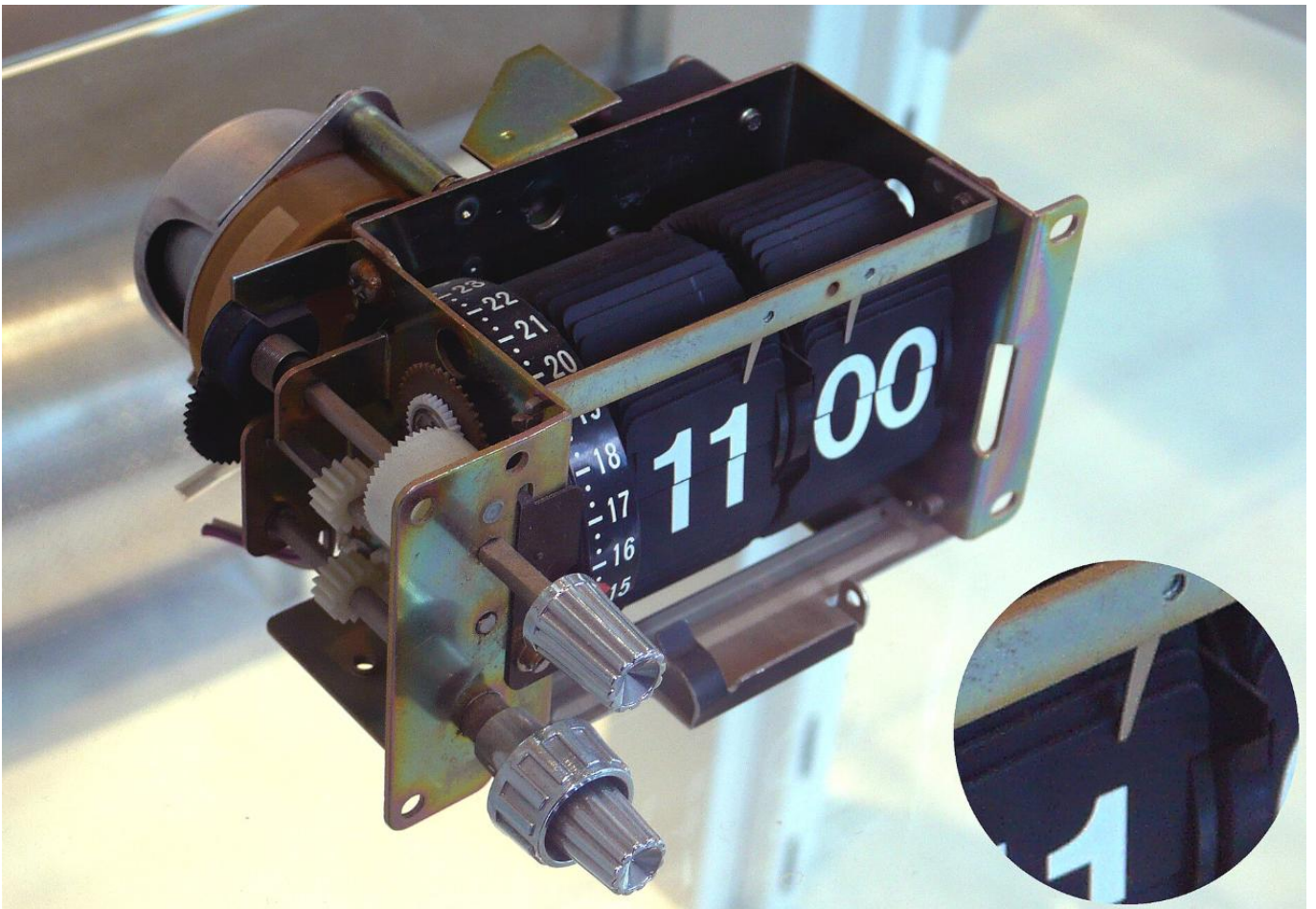


1. INTRODUCCIÓN

Aprovechando el mes de licencia que tengo para el uso avanzado de Fusion 360 he decidido diseñar y elaborar un nuevo reloj. Aunque ya había considerado varias ideas, finalmente me he decantado por un modelo “retro” de pantalla de tablillas ya que conlleva algo de dificultad, al menos mayor que el de las otras opciones que tenía en mente. Es importante experimentar y plantearse retos nuevos, es la mejor forma de seguir aprendiendo.

Este modelo me ha permitido además aprovechar los motores pasos a paso que adquirí cuando construí el reloj “Campanilla”. Por su puesto he contado con Chat GPT para la elaboración del código que se necesita para que este nuevo reloj funcione.

Las pantallas de tablillas se usaban mucho hace varias décadas, sobre todo en aeropuertos y en estaciones de tren. Todo el mundo recuerda el sonido tan particular que hacían cuando refrescaban la información, todo un clásico. Y en su día también se fabricaban y vendían relojes que se basaban en esta tecnología. De hecho, todavía se pueden encontrar modelos a la venta en internet. Su pantalla tan característica en la que se muestra la hora de forma muy clara y visible los hace todavía muy atractivos.



Antes de ponerme manos a la obra tuve que investigar un poco acerca de la mecánica que llevan estos relojes. La página y videos que Scott Bezek tiene al respecto me han sido de gran utilidad para tener en cuenta diversas cuestiones que son esenciales en la elaboración de este reloj.

<https://github.com/scottbez1/splitlap>

2. CONSIDERACIONES EN EL DISEÑO

La primera cuestión que surge es la de cómo se puede acoplar un motor paso a paso, que típicamente realiza múltiplos de 64 pasos por vuelta, a un reloj que necesita dar 60 minutos y 24 horas. Lamentablemente no se obtiene una división entera entre los pasos que da el reloj y los que se necesitan. Este problema ya se me planteó en la elaboración del “Campanilla” y en aquella ocasión lo resolví con dos engranajes, uno de 15 y otro de 16 dientes de forma que $64 \cdot 15 / 16 = 60$. El uso de engranajes hace más complejo el proyecto a la par que más voluminoso, pero por supuesto es una opción que puede ser viable.

La otra cuestión importante era la de decidir el número de tablillas a usar. Después de darle muchas vueltas decidí que el reloj llevaría dos rollos de tablillas, uno para dar las horas (de 24 tablillas) y otro para dar los minutos (de 60 tablillas). Está claro que se pueden disponer de más rollos, por ejemplo, uno para cada dígito, pero entonces se necesitan más motores y como veremos posteriormente más pines en el Arduino, lo que limita bastante las opciones. Pero lo más importante es que deseaba tener un número de tablillas lo suficientemente grande para asegurar que el reloj haría el sonido tan típico al rotar rápidamente. Pero a mayor número de tablillas más grande debía ser el rollo para poder acoplarlas sin que se rompieran ni que tropezaran. Había que considerar todos estos factores.

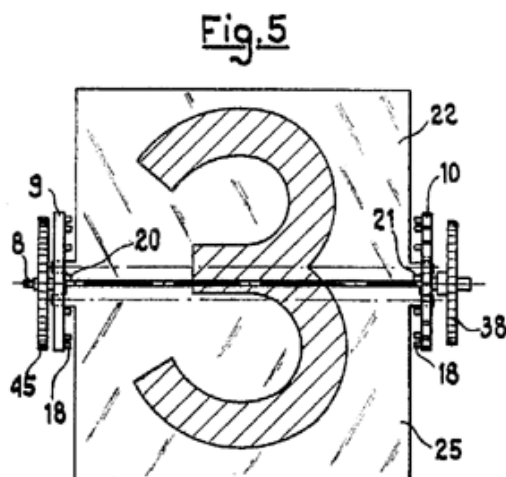
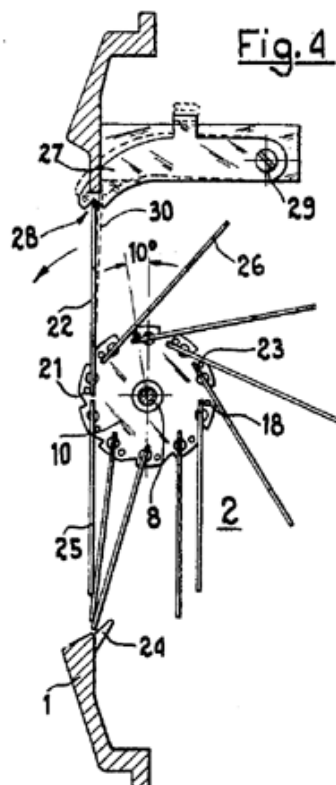
En relación con el desacoplo de los pasos del motor vs los requisitos horarios, la opción por la que me decanté fue la de hacer una resincronización periódica de los rollos de tablillas. Voy a explicar esto en más detalle.

Es importante disponer de un mecanismo para “memorizar” la posición en el rollo. Si el reloj se desconecta de la red o pierde la hora ¿cómo sabe en qué parte del rollo se encuentran las 12 por ejemplo? Para resolver este problema se suele utilizar un sensor magnético. En la rueda donde van acopladas las tablillas se incrusta un imán y cada vez que este imán pasa por delante del sensor puede ser detectado. Esto establece un punto de referencia en el rollo, pongamos por ejemplo que la hora/minuto “00” se corresponda cuando el imán está justo frente al sensor.

Teniendo en cuenta lo anterior, es decir, que los rollos van a disponer de un sensor magnético, ahora había que resolver el problema de los pasos. Cada vez que el motor mueve uno de los rollos las paletas se van adelantando o atrasando debido al desfase de 64 pasos versus los 60 o 24 requeridos. Llega un momento en que el desfase acumulado hace que bien salten dos tablillas de golpe o bien que se necesiten dos pasos para que pueda pasar la tablilla. Ese desfase no es muy problemático si el rollo da una vuelta completa, pero genera los problemas descritos con anterioridad al cabo de un determinado número de vueltas. Una opción para ir eliminando los desfases que se van acumulando es hacer un “reinicio” del rollo, es decir, obligar al rollo a girar hasta el imán para reiniciar el conteo de pasos. Y eso es justo lo que decidí hacer en el diseño, un reinicio cada vez que los rollos completaran una vuelta. De esta forma, además conseguía marcar una especie de “campanadas” cada hora y cada día.

Otra cuestión no menor era la del disponer del mecanismo para poder poner el reloj en hora. Los relojes antiguos solían llevar motores continuos que hacían que el rollo girase uniformemente y no por saltos como con los de motores paso a paso. En aquellos relojes antiguos la puesta en hora se hacía manualmente, girando unas ruedas como se ilustra en la fotografía anterior. En nuestro caso lo más sencillo era incorporar unos pulsadores que avanzaran los motores paso a paso. Y así dispuse de un botón para ajustar las horas y otro para ajustar los minutos.

Las últimas consideraciones, básicamente mecánicas, tienen que ver con asegurar el sonido de las tablillas al chocar unas con otras evitando que queden oscilando en el aire cada vez que saltan. Estos aspectos se van resolviendo y afinando conforme se va construyendo el modelo.



Mi idea original fue realizar el modelo con tablillas negras y números en blanco. Busqué planchas de poliestireno o similar de color negro y solo las encontré de 1 milímetro de espesor. Hice pruebas con esas tablillas, pero resultaban demasiado gruesas. Así que finalmente me decanté por el poliestireno blanco de 0.5 mm de espesor que vengo utilizando ampliamente en modelismo naval y también en mi reloj "Fibonacci".

3. EL CIRCUITO Y SU PROGRAMA

En el simulador Wowki no se encuentran muchos de los componentes que usa este reloj por lo que es muy difícil simular su funcionamiento. En Wowki no se encuentra ni el sensor magnético ni el motor paso a paso 28BYJ-48. Lo máximo que se puede hacer es una simulación con motores servos, con pulsadores y con el clásico RTC. Así que no he podido hacer muchas pruebas en Wowki en la elaboración de este modelo.

Una vez que dispuse de un programa muy básico, elaborado por Chat GPT, hice algunas comprobaciones con servos en Wowki. Pero pronto me vi forzado a elaborar un prototipo e ir probando diferentes versiones del programa según iba avanzando. No poder disponer de un monitor serie para saber la hora que estaba calculando el sistema ha sido un hándicap muy importante. Esto sin duda ha retrasado mucho los avances porque ha sido un poco como trabajar a ciegas.

Chat GPT es fantástico, pero puede ser al mismo tiempo desesperante. El programa inicial, siguiendo mis indicaciones detalladas, solo le tomó unos pocos segundos crearlo. Mover motores, usar sensores Hall y almacenar la hora en el RTC DS3132 fue coser y cantar para Chat GPT. Pero cuando incorporamos botones para cambiar la hora o le pedimos a Chat GPT que hiciera las recalibraciones al reconectar a la corriente o al cambiar la hora, entonces se convirtió en una pesadilla.

Han sido muchas horas, muchas iteraciones las que he tenido que hacer para obtener un programa realmente fiable y estable. He estado cerca de arrojar la toalla en varias ocasiones. Cuando algo que

queríamos solucionar llegaba a funcionar, otra parte que ya estaba resuelta volvía a fallar. Hemos dado muchas vueltas en círculo.

Es muy importante explicarle bien a Chat GPT lo que se busca y más aún mostrarle las inconsistencias de sus análisis cuando fallan las cosas. En caso contrario Chat GPT se precipita en sus propuestas.

Pero finalmente conseguimos una versión operativa. Muestro aquí algunas líneas a modo de ejemplo:

```
#include <Wire.h>
#include "RTClib.h"

RTC_DS3231 rtc;

// --- Pines ---
const int motorPinsHour[4] = {10, 11, 12, 13};
const int motorPinsMinute[4] = {4, 5, 6, 7};
const int hallPinHour = 9;
const int hallPinMinute = 3;
const int btnHour = A0;
const int btnMinute = A1;

// --- Parámetros ---
const int STEPS_PER_REV = 4096; // 28BYJ-48
const int FLAPS_HOUR = 24;
const int FLAPS_MINUTE = 60;
const float STEPS_PER_FLAP_H = STEPS_PER_REV / (float)FLAPS_HOUR;
const float STEPS_PER_FLAP_M = STEPS_PER_REV / (float)FLAPS_MINUTE;
const int stepDelayUs = 1200;

// --- Secuencia half-step ---
const int seq[8][4] = {
  {1,0,0,0}, {1,1,0,0}, {0,1,0,0}, {0,1,1,0},
  {0,0,1,0}, {0,0,1,1}, {0,0,0,1}, {1,0,0,1}
};

// --- Estado ---
int seqPosH = 0, seqPosM = 0;
int flapHour = 0, flapMinute = 0;
bool lastBtnHour = HIGH, lastBtnMinute = HIGH;
int lastRTCHour = -1;
int lastRTCMinute = -1;

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

  for(int i=0;i<4;i++){
    pinMode(motorPinsHour[i], OUTPUT);
    pinMode(motorPinsMinute[i], OUTPUT);
  }
  pinMode(hallPinHour, INPUT_PULLUP);
  pinMode(hallPinMinute, INPUT_PULLUP);
  pinMode(btnHour, INPUT_PULLUP);
  pinMode(btnMinute, INPUT_PULLUP);

  Wire.begin();
  rtc.begin();
```

Las cuestiones relacionadas con el uso del Arduino UNO también han influido en el diseño final del reloj. Cada motor, que lleva su propio controlador, requiere de cuatro pines al margen del positivo a 5V y de la toma de tierra. A estos pines hay que sumar los que requieren los pulsadores, los sensores HALL y el RTC. Son muchos pines para un Arduino UNO, aunque no exceden su capacidad (por muy poco). Eso sí, en caso de pretender incluir un tercer motor o más pulsadores, entonces Arduino UNO sería insuficiente y habría que plantearse el uso del Arduino MEGA, por ejemplo. Estas consideraciones también fueron muy relevantes en el diseño del modelo final.

Hoy, antes de empezar a escribir esta guía, he contactado de nuevo con Chat GPT. Aunque me había desconectado, el programa recordaba las dificultades que habíamos tenido los días atrás. Le he pedido sugerencias para darle un nombre a este reloj y me ha dado varias entre las cuales figuraba “Tablex”. Me ha hecho gracia así que me he decidido por ese nombre. Parafraseando a Chat GPT: *“Suena como una marca real, mezcla el lado mecánico de las tablillas con el lujo y precisión de un Rolex, pero con un toque muy maker”*.

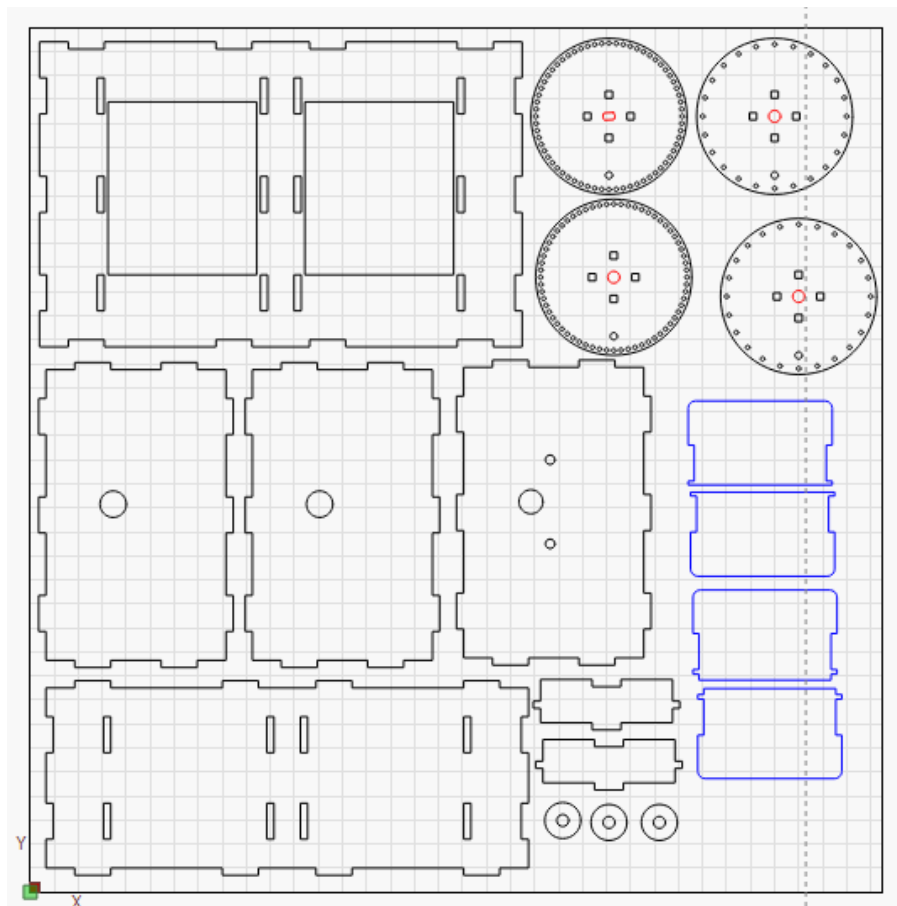
Dado que Chat GPT ha jugado un papel protagonista en el desarrollo de este proyecto ¡qué menos que usar una de sus propuestas!

4. MATERIALES Y ESTRUCTURA

Para la elaboración del reloj se han utilizado los siguientes materiales:

- Plancha de MDF sin tratar y con acabado tipo Wengué de 3 mm de espesor
- Cuadradillo de samba de 15*15 mm² de sección
- Plancha de poliestireno de 0.5 mm de espesor
- Vinilo adhesivo permanente
- Tubo de aluminio de 5 mm
- Rodamientos
- Varilla de ayus de 3mm
- Placa de Arduino Uno R3
- Imanes de neodimio de 3 mm de diámetro
- Sensores Hall KY003
- Pulsadores de 6mm*6mm
- Placa RTC DS3231 con pila CR 2032
- Fuente de alimentación de 9V
- Conector Jack hembra para fuente de alimentación
- Jumpers y cables varios
- Tapa poros, barniz mate en espray, tirafondos y fieltros protectores

Como es habitual, la estructura que alberga la electrónica está elaborada con plancha de MDF de 3 mm de espesor, la que he cortado con mi Sculpfun S30. Su diseño lo he realizado en Fusión 360 para poder exportarlo a LightBurn y desde allí grabar y cortar las piezas. También he usado la cortadora laser para elaborar las tablillas, así como los números de vinilo adhesivo.



Como finalmente me decanté por utilizar tablillas de poliestireno blanco, para la carcasa he utilizado el MDF con acabado Wengué a fin de obtener un buen contraste. Teniendo en cuenta la disposición de los elementos en el interior, he diseñado un modelo que es totalmente desmontable, es decir, no hay piezas pegadas (a excepción de la estructura que alberga a los interruptores) y se puede acceder a cualquier parte del interior tan solo desatornillando la carcasa. Unos cuadradillos de samba sirven para el anclaje de la tornillería.

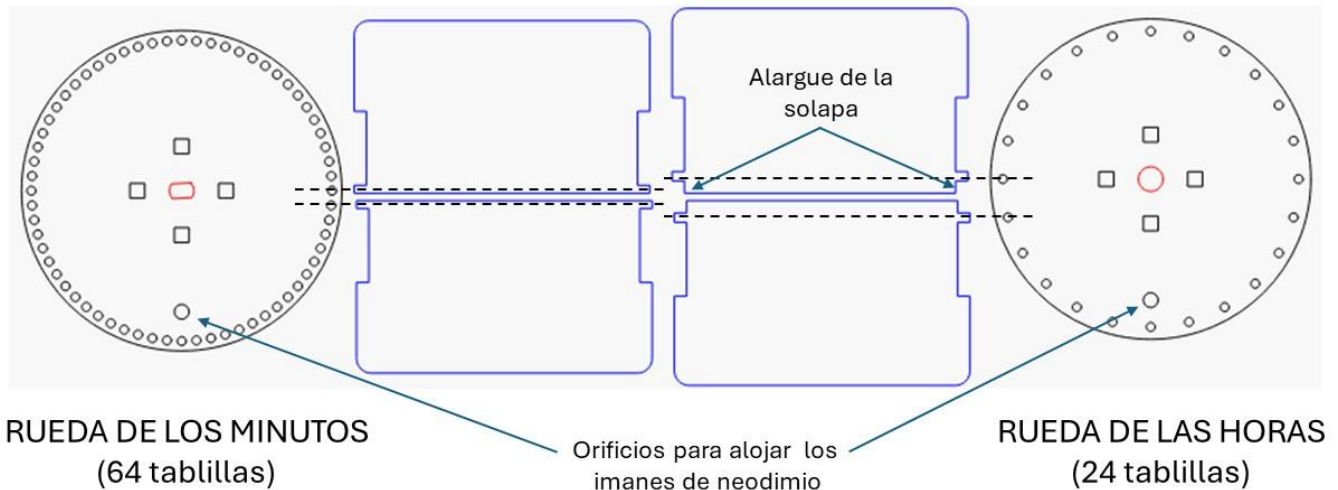


El corte de poliestireno con laser de diodo deja mucha rebaba en la pieza cortada ya que es un material plástico. Para un buen acabado he tenido que lijar suavemente cada tablilla para eliminar dichas rebabas. Además, estas pantallas de tablillas suelen ir en acabado mate para evitar reflejos que pudieran distorsionar su lectura. Así que el lijado cumple las dos funciones.

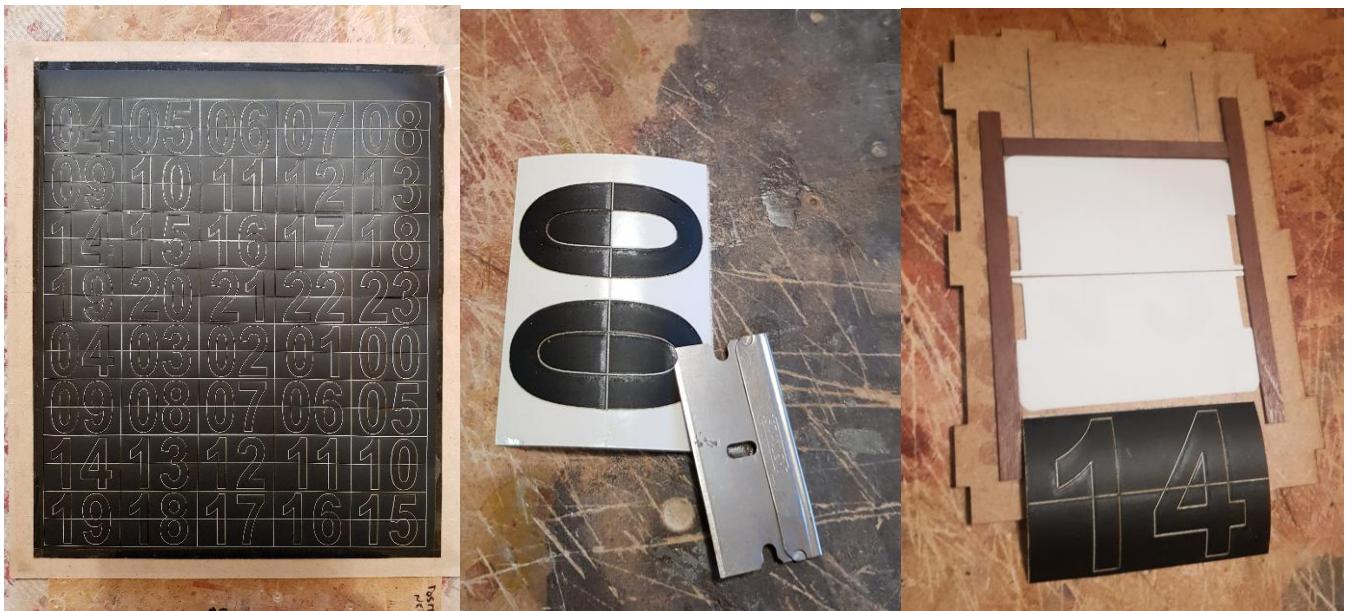
El tamaño de la rueda del rollo que alberga las tablillas está limitado por el número de orificios que se precisan para colocar las mismas y por el diámetro de estos orificios. Esto es lo que determina realmente el tamaño que va a tener el reloj. Al colocar las tablillas queda un espacio entre la mitad superior y la inferior. Lo ideal es que ese espacio no sea muy grande para que los números o letras no queden muy distorsionados. Si los orificios están muy espaciados entonces el hueco que queda entre las tablillas es demasiado grande. Ese era el caso de la rueda de 24 tablillas. Como había diseñado las ruedas del mismo tamaño por simplificación, los orificios en la rueda de la hora quedaban más separados que en la de los minutos. Para eliminar el problema de un hueco excesivo entre las tablillas del rollo de las horas he tenido que alargar un poco las mismas, tal como se muestra en el esquema adjunto.

En el dibujo nuestro también la posición de los imanes de neodimio.

TABLILLAS



En este reloj es la primera vez que uso vinilo adhesivo permanente. Generalmente este material tiene softwares y máquinas cortadoras específicas, aunque en mi caso he utilizado la cortadora láser de diodo. Encontrar los parámetros adecuados no ha sido complicado, aunque hay que tener precaución de no traspasar el papel donde va pegado el vinilo. Luego se usa un filme especial transparente con muy poco adhesivo para arrancar del papel el contorno de vinilo ya cortado. Yo me he hecho un marco para facilitar la labor de pegado a las tablillas.



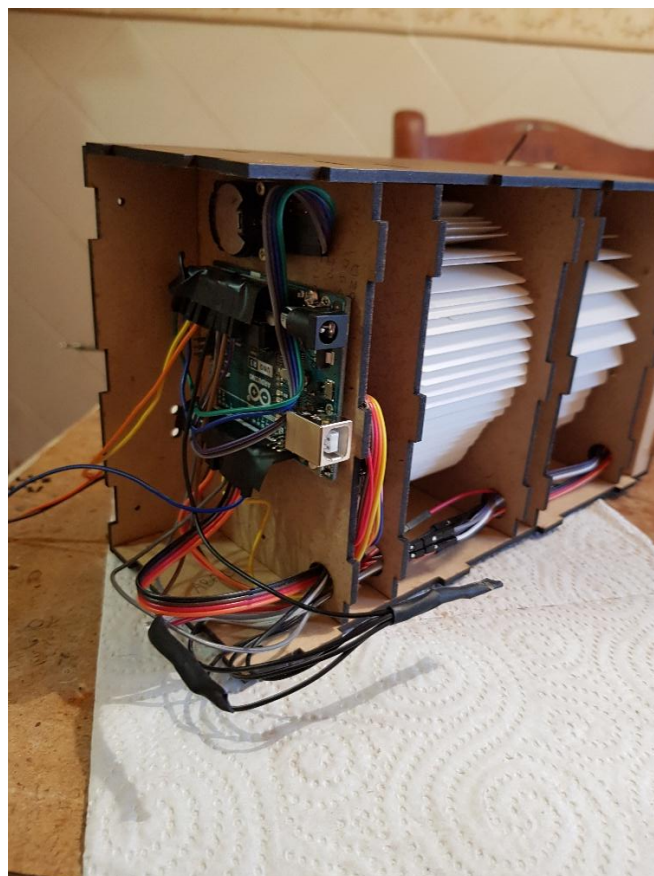
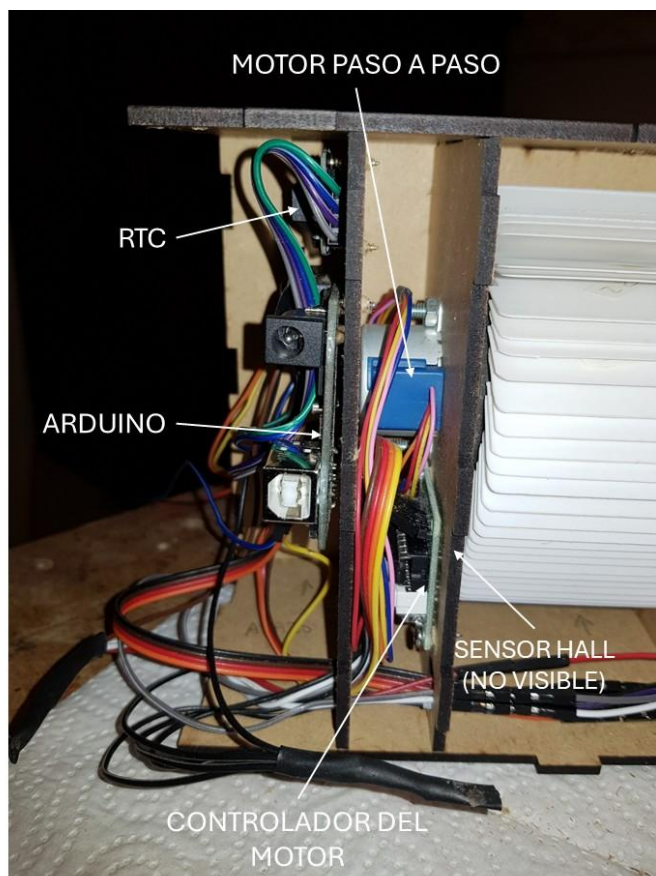
Otro detalle curioso que ya me surgió en mi reloj anterior es que en el MDF con acabado Wengué no se ven bien los grabados, no destacan al ser oscuros como el color del MDF. Para resolver esto pensé en usar cera blanca que queda insertada en el hueco del grabado y que luego queda oculta (protegida) bajo la capa de barniz. El resultado es muy satisfactorio.



Finalmente hemos obtenido una caja para el reloj de dimensiones 25*14*9 (en centímetros). Un tamaño algo voluminoso para un reloj de mesa, pero todo condicionado por las cuestiones de diseño y por las limitaciones que he explicado anteriormente.

Muestro unas fotos del interior del reloj ya montado donde indico cómo y donde se ubican los distintos elementos que lleva el reloj.





Esta guía, así como el video explicativo de este reloj están disponibles en mi página web de modelismo naval.

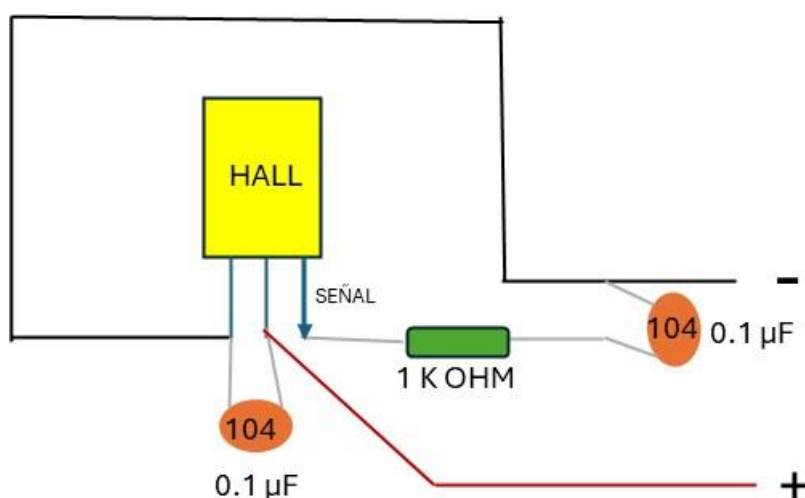
<https://modelismo-naval-de-pedro-eisman-cabado6.webnode.es/>

5. AJUSTES EFECTUADOS EL 02/11/2025

Tras unos días de funcionamiento del reloj, se han detectado dos problemas que precisaban de una solución urgente.

El primero tenía que ver con fallos intermitentes de los sensores Hall. Si se cambiaba el reloj de lugar o dependiendo de la hora del día, el reloj no se reiniciaba correctamente, los sensores Hall no respondían. Esto en un principio me pareció un gran misterio al que denominé “Poltergeist” ¿Cómo era posible que en una habitación los sensores funcionasen y en otra no? En realidad, el problema lo ocasionaban interferencias magnéticas en la señal que llegaba a los sensores, bien por causas ambientales o bien producidas por los propios cables de los motores.

Para resolver este inconveniente he dispuesto unos capacitadores cerámicos y unas resistencias en cada sensor Hall conforme se muestra en el siguiente esquema:



Estos elementos funcionan como desacople de los motores con la fuente de alimentación y dan un filtrado a la señal del Hall sin afectar su funcionamiento. Tras su instalación el reloj no ha vuelto a mostrar un comportamiento errático.

La segunda cuestión tenía que ver con el “homing” de los rollos después de dar las campanadas. La versión del código con la que estaba trabajando el reloj no hacía el reinicio de forma correcta por lo que los rollos se desalineaban tras unas horas de funcionamiento. Este era un problema muy serio y el caso es que he necesitado de algunas iteraciones con Chat GPT (bastantes horas a fuerza de ser sincero) para obtener un código ajustado y eficaz. Pero ha sido totalmente necesario ya que de lo contrario el reloj no funcionaba como era debido. En mi página web he colgado la nueva versión del código que por fin reinicia los rollos de tabllas después de cada campanada de forma satisfactoria.