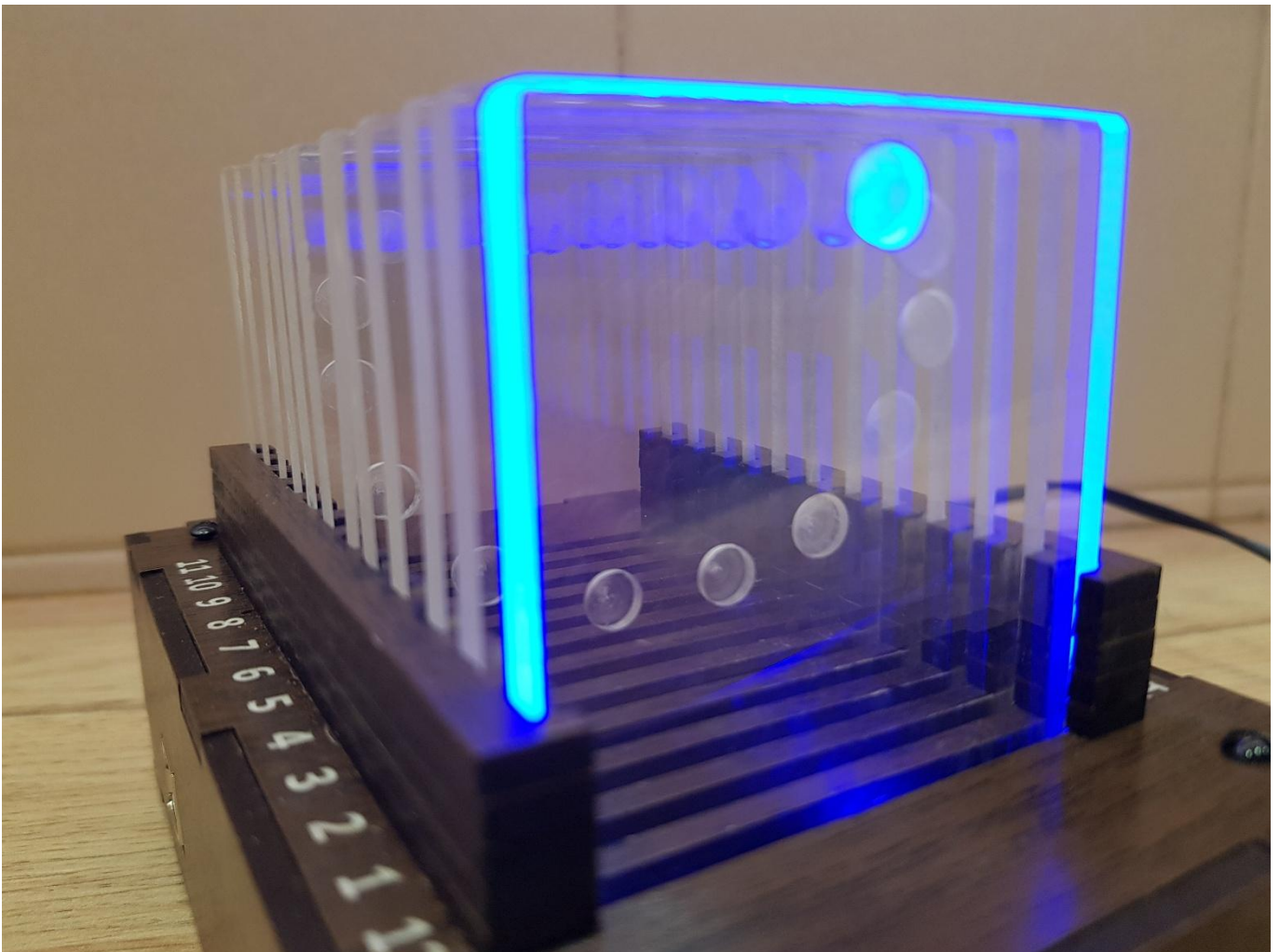


RELOJ DE LUZ FILTRANTE “REFLEJOS”

Por Pedro Eisman Cabado, octubre 2025

- GUIA DE OPERACIÓN -



1. INTRODUCCIÓN

Mis dos últimos relojes, aunque muy diferentes en estética y funcionalidad, tienen algo en común y es que ambos están gestionados por un Arduino. La mayor dificultad que conlleva usar una de estas placas es que requiere del desarrollo de un programa en lenguaje C para su gestión, lo que exige conocimientos y habilidades en programación. Lamentablemente yo no tengo grandes conocimientos en esta materia lo que limita mucho las cosas. Básicamente lo que he hecho hasta ahora ha sido efectuar pequeños cambios para adaptar programas ya existentes, aunque no he sido capaz de desarrollar un programa completo desde cero.

Para la elaboración del “Fibonacci” conté con la inestimable ayuda de mi amigo Pepe (JAGAR) para adaptar el programa original a las funcionalidades de mi nuevo diseño. Aunque Pepe es informático y domina la materia, ambos tuvimos interacciones con la IA (Inteligencia Artificial) para validar y explorar diversas cuestiones relativas al Arduino y a su programación. Me sorprendió profundamente, por ejemplo, la capacidad de DeepSeek (la IA de China) para efectuar pequeñas modificaciones en el código a partir de unas instrucciones muy básicas. Estas nuevas líneas de programación podían revisarse en simuladores gratuitos que se encuentran fácilmente en internet y cuyo manejo es bastante sencillo. En el caso del reloj “Fibonacci” usé ampliamente la aplicación Wowki.

A partir de aquella experiencia me planteé la posibilidad de desarrollar nuevos relojes con Arduino cuyos diseños y funcionalidades no estuviesen sujetos a las limitaciones de no saber mucho sobre programación. La programación correría a cargo de la IA. Así que tras unas horas explorando la página de “Instructables” de “AutoCAD” pronto terminé con tres posibles proyectos para un nuevo reloj. Desde un reloj que da la hora escrita, pasando por otro en el que la hora se lee en varios voltímetros, hasta llegar al que es objeto de este documento. Un reloj que da la hora proyectando luz filtrada a través de pequeñas placas de metacrilato.

El proyecto original que me ha servido de inspiración se llama “Spatial Desktop Clock” y fue publicado en mayo de 2024. Esta vez no tuve que descargarme ni planos ni programas. Me bastó con ver algunas fotos para imaginar como sería mi nuevo reloj. Eso sí, antes de ponerme manos a la obra en Fusion 360 había que asegurarse de que la IA fuese capaz de desarrollar el programa necesario. Pero además de ello verificar también que con mis técnicas y herramientas disponibles era posible poner en práctica el nuevo concepto de “luz filtrada”.

Verificados los dos puntos anteriores este nuevo proyecto fue fácil de completar. El nuevo reloj cumplía todas las expectativas: un programa desarrollado por la IA y nuevos materiales y efectos luminosos.

Debido a las proyecciones que hace la luz sobre las planchas de metacrilato a este nuevo modelo le puse el nombre de “Reflejos”.

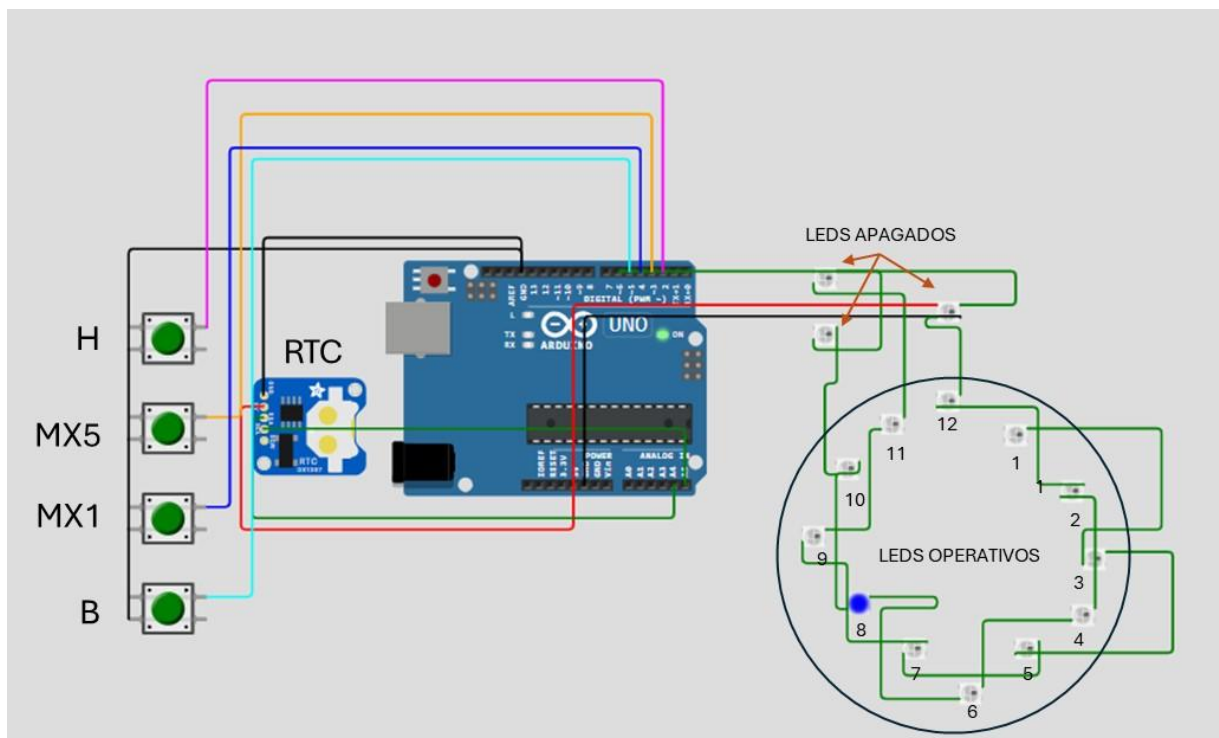
2. EL CIRCUITO Y SU PROGRAMA

Antes de abordar la elaboración del programa para este reloj, estuve haciendo diferentes pruebas con DeepSeek. La verdad es que esta IA funciona muy bien, pero comete algún que otro error y a veces termina complicando mucho las cosas. Es por ello por lo que también interaccioné con ChatGPT, IA con la que quizás es algo más sencillo interactuar, aunque también comete errores y con la que hay que tener igualmente mucha paciencia.

El circuito que había diseñado era muy sencillo. Dos tiras de leds en paralelo y ligeramente desfasadas para completar un total de 12 posiciones (las 12 horas). Dejaría algún led adicional apagado en los extremos para poder acoplarle conectores de forma que las tiras se pudiesen cambiar cuando se fundiesen, al igual que hice con el “Fibonacci”. Por supuesto dispuse de un RTC para almacenar la hora y de varios botones para ajustar tanto la hora como el brillo.

Después de algunas horas iterando con ChatGPT y verificando los progresos en el simulador Wowki, finalmente obtuve un código de 422 líneas que satisfacía mis requisitos. El programa estaba listo por lo que ya se podía abordar la ejecución física del modelo.

Muestro un esquema del circuito en Wowki. Aquí dispuse los leds en forma circular, al igual que en la esfera de un reloj, para facilitar la lectura de los datos y de su comprobación.



La forma en la que se proyecta la hora es como sigue:

- Las horas se dan con parpadeos en las posiciones de 1 a 12. En color azul en caso de ser AM y en color rojo en caso de ser PM.
- Los “minutos por cinco” (parte entera de los minutos divididos por 5) se dan en el mismo lugar que lo daría la manecilla en una esfera de reloj. Por ejemplo, si son las XX:20 parpadea en color verde el led situado a las CUATRO.

- Por último, los minutos restantes se marcan con parpadeos blancos en la misma posición que los “minutos por cinco”.

Con cada nueva hora el reloj ejecuta una espiral de color (azul o rojo según corresponda con AM/PM) para evidenciar el cambio de hora.

Pongo algunos ejemplos para clarificar mejor lo explicado anteriormente:

- 07:48: Un parpadeo azul a las SIETE, seguido de un parpadeo verde a las NUEVE y tres parpadeos blancos también a las NUEVE.
- 15:12: Un parpadeo rojo a las TRES, seguido de un parpadeo verde a las DOS y dos parpadeos blancos también a las DOS.
- Paso de 16:59 a 17:00. A las 16:59 el reloj parpadea en rojo a la cuatro, seguido de verde a las ONCE y cuatro parpadeos en blanco también a las ONCE. Justo cuando dan las 17:00 el reloj ejecuta una espiral en color rojo. Posteriormente el reloj pulsa una vez en rojo a las DOCE y otra vez esta vez en verde también a las DOCE.

Los botones para establecer la hora van aumentando el valor correspondiente cada vez que se pulsan. En el caso de tener que ajustar los “minutos por uno” estos se visualizan entre las DOCE y las CUATRO (a las DOCE en amarillo para evitar confusiones lo que significa que son cero y en blanco en el resto si son entre uno y cuatro).

Con todos estos requisitos obtuve un código de ChatGPT como he mencionado anteriormente. El mismo se encuentra disponible en el área de descargas de mi página de modelismo naval.

Muestro un pantallazo del programa de carga ARDUINO a modo de referencia.

```
#include <Adafruit_NeoPixel.h>
#include <RTClib.h>
#include <EEPROM.h>

#define PIN_LED 6
#define NUM_LEDS 15

#define BUTTON_HOUR 2
#define BUTTON_MINUTE_5 3
#define BUTTON_MINUTE_1 4
#define BUTTON_BRIGHTNESS 5

Adafruit_NeoPixel strip(NUM_LEDS, PIN_LED, NEO_GRB + NEO_KHZ800);
RTC_DS3231 rtc;

enum State { SHOW_HOUR, WAIT_HOUR_OFF, SHOW_MIN_BLOCK, WAIT_MIN_OFF, SHOW_MIN_EXTRA, WAIT_RESTART };
State currentState = SHOW_HOUR;

unsigned long stateStartTime = 0;
unsigned long blinkStart = 0;
bool blinking = false;
bool ledsOn = false;
int extraBlinkCount = 0;

int lastSecond = -1;
int lastMinute = -1;

bool lastHourButtonState = HIGH;
bool lastMin5ButtonState = HIGH;
bool lastMin1ButtonState = HIGH;
bool lastBrightnessButtonState = HIGH;

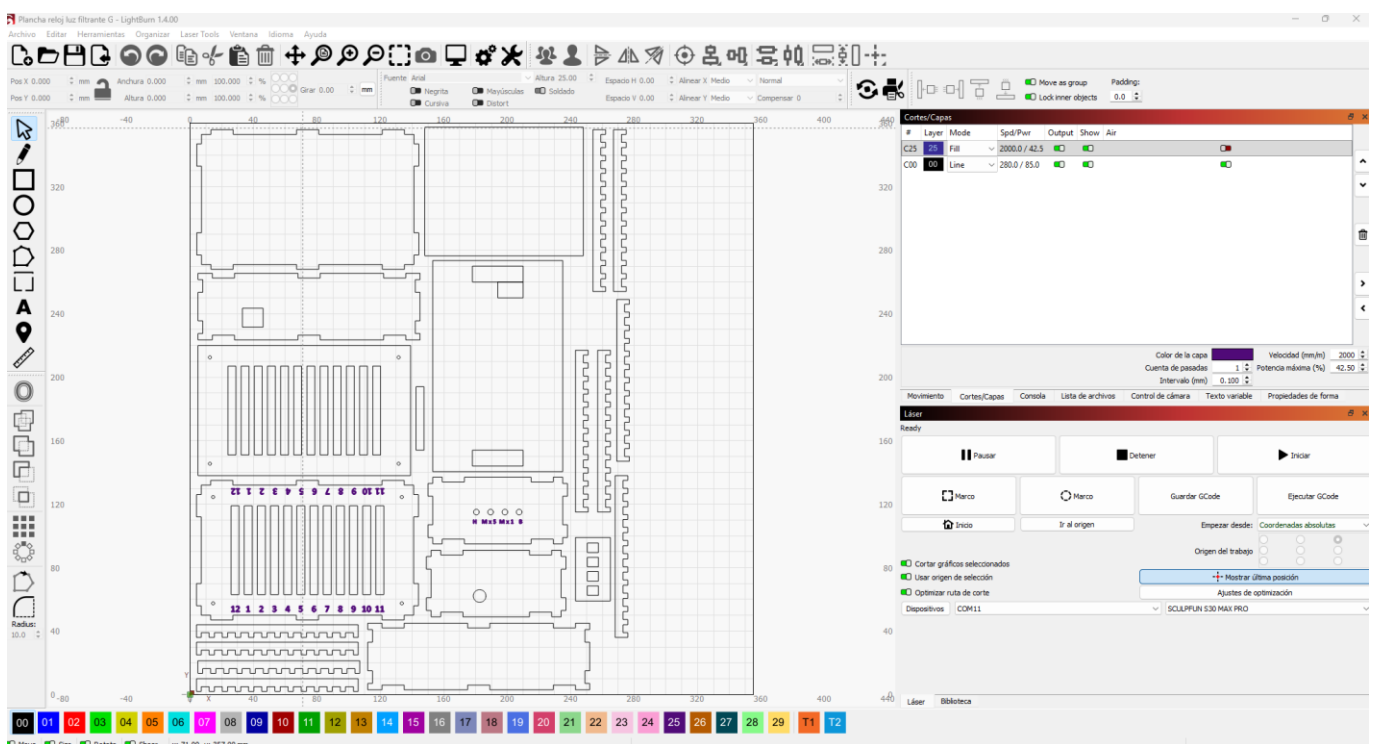
bool adjusting = false;
unsigned long lastAdjustTime = 0;
enum AdjustType { NONE, HOUR, MTN5, MTN1 }; AdjustType = NONE;
```

3. MATERIALES Y ESTRUCTURA

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

- Plancha de MDF de 3 mm de espesor
- Cuadradillo de samba de 15*15 mm² de sección
- Plancha de plexiglass de 5 mm de espesor
- Plancha de contrachapado de abedul de 2 mm de espesor
- Placa de Arduino Uno R3
- 4 pulsadores de 6mm*6mm
- Placa RTC DS3231 con pila CR 2032
- Tiras WS2812B de leds RGB direccionables a 5v
- Fuente de alimentación de 9V
- Conector Jack hembra para fuente de alimentación
- Conectores para tiras de led
- Jumpers y cables varios
- Tubos termo-retráctiles
- Estaño de soldadura
- Cola blanca y cianocrilato
- Tapaporos y barniz satinado en spray
- Tirafondos
- Filtros protectores

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 como siempre lo he realizado en Fusión 360 para poder exportarlo a LightBurn y desde allí grabar y cortar las piezas.



Hasta ahora utilizaba el MDF clásico que tintaba después de lijarlo suavemente. En este reloj he probado una plancha de MDF que ya viene preparada en color Wengue. El tratamiento de esta es el habitual, un poco de tapa poros, un suave lijado y posterior acabado en espray satinado. Como la madera es bastante oscura los grabados no se visualizan bien. Es por ello por lo que he rellenado los huecos de estas zonas grabadas con cera blanca antes de aplicar el barniz en espray.

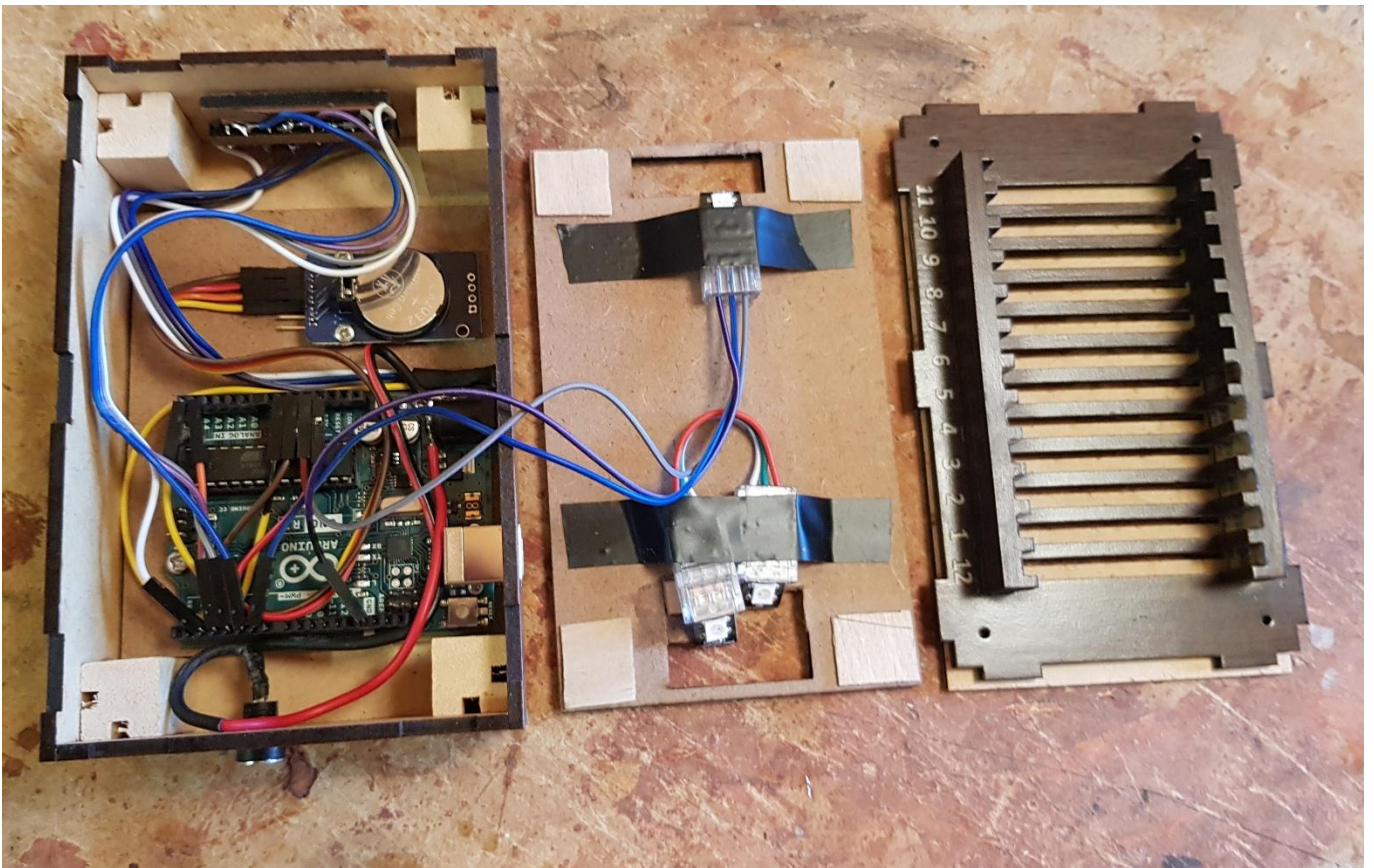


Otra de las cuestiones que necesitaba verificar era lo de la luz filtrante. Hice pruebas con planchas de metacrilato de 2 mm porque son fáciles de cortar con el cúter. Sin embargo, los resultados no eran del todo satisfactorios.

Posteriormente probé con planchas de 5 mm de espesor y me llevé la grata sorpresa de lo fácil que resulta cortarlas con mi cortadora de sierra circular. Eso sí, conviene luego lijar suavemente los bordes con la lijadora de disco. Los taladros se hacen fácilmente con una broca para metal.



En las siguientes fotografías se muestran el interior del reloj, así como el modelo terminado.



Como se puede apreciar, la lectura de la hora se puede hacer bien mirando el reloj frontalmente, donde la luz se marca en los orificios dispuestos como en la esfera de un reloj, o bien en los bordes laterales de las planchas, para lo cual se han grabado las doce horas en cada costado.