

Arnau Manyer Vallet

**Sistema de caracterización de sensores capacitivos
impresos sobre substrato flexible**

Trabajo Fin de Grado

dirigido por el Dr. Alfonso José Romero Nevado

dirigido por el Dr. José Luis Ramírez Falo

Grado en Ingeniería Electrónica Industrial y Automática



UNIVERSITAT ROVIRA I VIRGILI

Tarragona

2022

Agradecimientos

Me gustaría empezar expresando mi gratitud a los directores, el Dr. Alfonso José Romero Nevado y el Dr. José Luis Ramírez Falo, por darme la oportunidad de poder asumir un reto tan novedoso, y a la vez plagado de incertidumbre, para realizar mi proyecto de final de carrera. Su ayuda y guía ha sido fundamental para la realización del trabajo, y también para profundizar en la capacidad de gestionar proyectos.

También quería agradecer el apoyo y la ayuda de Xavi Blanch, indispensable en el soporte en las tareas técnicas, así como en la formación sobre la impresora.

Gracias a la Universitat Rovira i Virgili por haber dispuesto los medios y herramientas necesarias para la realización de este proyecto.

Finalmente, a mi familia y amigos. Sin ellos, nada de esto sería posible.

Índice

Contenido

1.	Introducción.....	1
2.	Objetivos	2
3.	Información preliminar	3
3.1.	Especificaciones de los dispositivos utilizados.....	3
3.1.1.	Características del ATmega328p Xplained Mini	3
3.1.2.	Micro pulsador	4
3.1.3.	Pantalla digital de siete segmentos y cuatro posiciones	6
3.1.4.	Características de la impresora Dimatix DMP-2850 (Fujifilm).....	7
3.2.	Sensores capacitivos.....	13
3.3.	Métodos de medida de capacidades	15
3.3.3.	Red RC	15
3.3.4.	Divisor de tensión con condensadores	15
3.3.5.	Oscilador tanque LC.....	17
3.3.6.	Circuito con tensión de salida proporcional al condensador a medir.....	17
3.3.7.	Circuito con puente RC.....	19
4.	Comparativa de los métodos para medir capacidades.....	21
4.1.	Red RC.....	21
4.2.	Divisor de tensión con condensadores.....	24
4.3.	Oscilador tanque LC	26
4.4.	Circuito con tensión de salida proporcional al condensador a medir	27
4.5.	Circuito con puente RC	36
4.6.	Elección del método	37
5.	Prototipo de medidor de capacidades	38
5.1.	Descripción del funcionamiento	39
5.2.	Programa	41
5.3.	Construcción de la maqueta	46
6.	Proceso de impresión	49
6.1.	Operaciones previas a la impresión.....	50
6.1.1.	Preparación del Kapton	50
6.1.2.	Preparación de la tinta de plata.....	52
6.2.	Primeras pruebas	54
6.3.	Pruebas con tinta de plata.....	62
6.4.	Impresiones definitivas con tinta de plata.....	67
7.	Mediciones y resultados.....	75

8. Conclusiones y riesgos	77
9. Webgrafía	79
9.1. Información preliminar.....	79
9.2. Métodos para medir capacidades.....	79
9.3. Programa y maqueta final	79
10. Bibliografía	80

1. Introducción

Vivimos en un mundo en que los avances tecnológicos van sucediendo a un ritmo frenético. Los usuarios cada vez son más exigentes y la competencia más feroz. Por eso es vital que estos progresos cumplan con los criterios de calidad, pero también que sean lo menos costosos posible de desarrollar y producir, y que sean del mínimo tamaño viable.

Este proyecto va a tratar de obtener un capacitor que en un futuro pueda ser empleado como sensor capacitivo sensible a gases químicos, entre otros estímulos (torsión, flexión, compresión o proximidad). Para ello, dispondremos de una impresora capaz de imprimir a nivel micrométrico. Vamos a hacer uso de tinta de plata para hacer de la impresión de la capacidad algo conductivo.

El reto es poder hacer este capacitor lo más pequeño posible, siempre que el funcionamiento sea el adecuado y las mediciones de capacidad sean posibles.

Otra parte del proyecto se va a centrar en escoger y aplicar un método de medida de capacidades funcional y económico, para así poder establecer un sistema de caracterización de estos sensores capacitivos, y, por consiguiente, determinar el grado de cumplimiento de los objetivos de los mismos.

2. Objetivos

Los objetivos de este proyecto son los siguientes:

- Analizar de distintos métodos de medida de capacidades, determinando el más adecuado para el proyecto.
- Desarrollar e implementar del sistema de caracterización de sensores capacitivos mediante un microcontrolador.
- Conocer el funcionamiento de la impresora Dimatix DMP-2850 (Fujifilm).
- Diseñar los patrones adecuados para que resulten en capacidades.
- Imprimir, con la tinta de plata, un capacitor basado en interdigitados.

3. Información preliminar

3.1. Especificaciones de los dispositivos utilizados

3.1.1. Características del ATmega328P Xplained Mini

La ATmega328P Xplained Mini es una placa hardware que tiene el microcontrolador ATmega328P como elemento principal. Éste es un microcontrolador de bajo coste que nos permite realizar ciertas aplicaciones electrónicas.

El hecho que el microcontrolador venga ya con la placa hardware Xplained Mini permite que podamos conectarlo directamente al ordenador y cargarle el código que queramos.

En la figura 1 adjuntamos cómo es la placa Xplained Mini y la función básica de los varios pines:

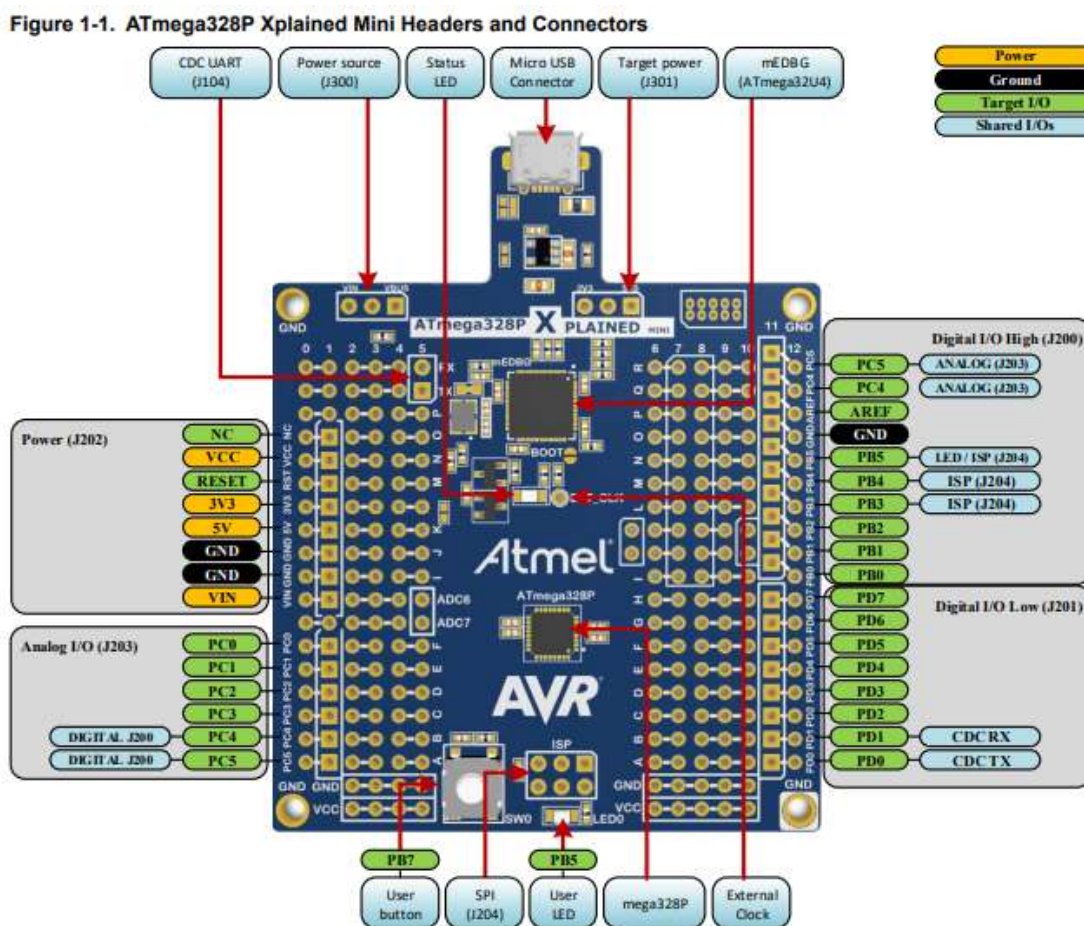


Figura 1. ATmega328P Xplained Mini User Guide, pág. 4.

A continuación, explicaremos la función de los principales pines que vamos a usar en este proyecto. Detallaremos qué hacen, y qué uso le vamos a dar.

Antes, pero, haremos un resumen de las principales características del dispositivo.

Electrónica

- Microcontrolador ATmega328P
- Memoria Flash 32 KBytes
- Memoria EEPROM 1 KBytes

- Memoria RAM 2 KBytes
- Voltaje de operación 5 V
- Pines E/S digitales 14
- Pines E analógicos 6
- Corriente máxima E/S pin 40 mA
- Corriente máxima V_{cc} y GND pines 200 mA
- Temperatura de operación -55 °C a +125 °C
- Frecuencia de reloj 16 MHz

El software que usamos para crear los programas, que posteriormente cargaremos en el microcontrolador, es el **Atmel Studio 7.0**.

Pines de los que haremos uso y la funcionalidad que tendrá en nuestros programas:

- **PC0**: entrada analógica al convertor analógico-digital.
- **PC1**: entrada digital con la que pretendemos detectar los cambios de nivel (alto a bajo o bajo a alto) de la señal (Pin Change Interrupt).
- **PC4**: pin configurado como salida, generador de pulsos.
- **PB1**: pin configurado como salida, generador de ondas cuadradas con un ciclo de trabajo configurable (PWM) (en otros programas configurado de forma distinta).
- **PB5**: entrada digital con la que pretendemos detectar los cambios de nivel (alto a bajo o bajo a alto) de la señal (Pin Change Interrupt).
- **PD0 – PD7**: pines configurados como salida digital.
- **PB0 – PB3**: pines configurados como salida digital.

3.1.2. Micro pulsador

Los micro pulsadores están pensados para poder cerrar un circuito y permitir el paso de corriente en un momento dado al pulsarlo, o al revés.



Figura 2. Imagen de un micro pulsador de la web rs-online.com.

En la figura 2 podemos ver cómo son los micro pulsadores que vamos a usar en este proyecto.

La funcionalidad que van a tener estos dispositivos la vamos a explicar más adelante, ahora nos centraremos en detallar cómo funcionan internamente.

El esquema del conexionado de estos micro pulsadores se puede ver en la figura 3.

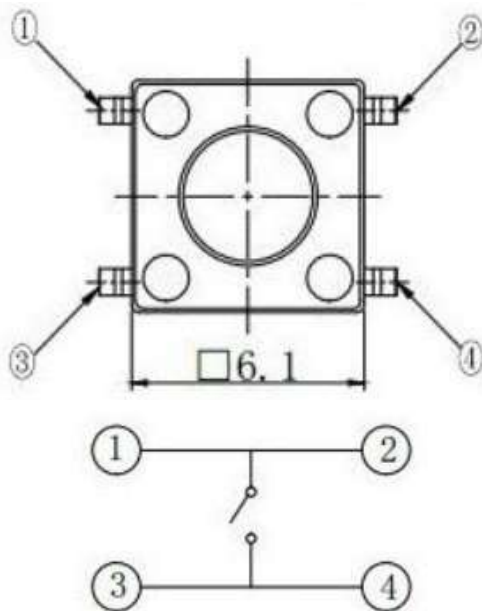


Figura 3. Esquema eléctrico de un micro pulsador de la web www.vistronica.com.

Los pines 1 y 2, y 3 y 4, están cortocircuitados respectivamente.

Al pulsar el botón, permitimos el paso de corriente de los pines 1 y 2 a los pines 3 y 4, o a la inversa. Por eso, lo que tiene sentido es que se conecte la entrada en 3 o 4, por ejemplo, y la salida en 1 o 2. Si lo configuramos así, tendremos el control para hacer pasar corriente a nuestra conveniencia.

Supongamos que usamos los pines 1 y 3 para hacer el conexionado. El esquema quedaría como se muestra en la figura 4.

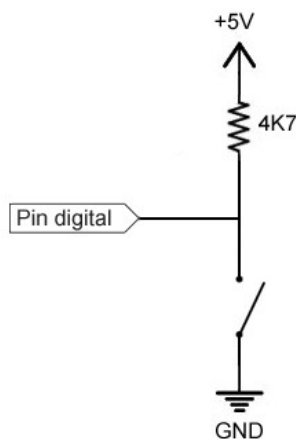


Figura 4. Conexión del micro pulsador de la página luisllamas.es.

Conectamos el pin 3 directamente a tierra y al pin 1 le añadimos una resistencia de Pull-Up antes de conectarlo a alimentación.

Las resistencias de Pull-Up son necesarias porque si no estuvieran, al pulsar el interruptor, estaríamos conectando directamente los valores de 0V y 5V, lo que significa que causaríamos un cortocircuito. Esto provocaría el paso de un elevado de corriente y un rápido calentamiento de componentes y conductores.

La resistencia de Pull-Up fuerza HIGH en el PIN cuando el pulsador está abierto. Cuando este está cerrado el PIN se pone a LOW, la intensidad que circula se ve limitada por esta resistencia.

Es decir, cuando pulsemos el pulsador provocaremos un cambio de HIGH a LOW en el pin del microcontrolador al que lo tengamos conectado. Entonces, en el microcontrolador podremos programar una interrupción que nos avise de este cambio, y poder programar unas determinadas acciones a conveniencia.

3.1.3. Pantalla digital de siete segmentos y cuatro posiciones

En la figura 5 mostramos una imagen de como es este dispositivo para después proceder a explicar su funcionamiento:

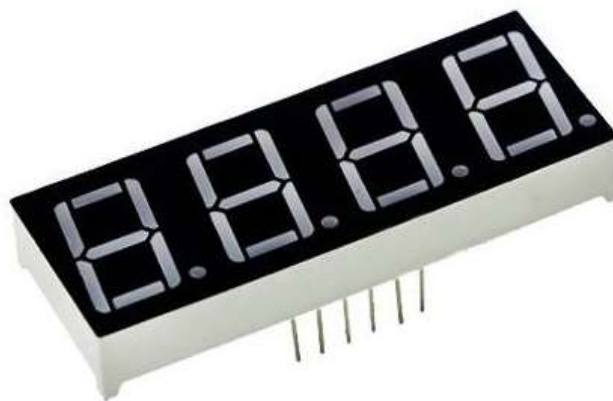


Figura 5. Display 7 segmentos y 4 posiciones de la web createc3d.com.

Esta pantalla funciona conectada al microcontrolador de forma multiplexada. Es decir, al microcontrolador vamos a conectar los siete segmentos más el punto de una posición y los cuatro cátodos de los segmentos, que son LEDS, que nos van a permitir multiplexar la posición.

Vamos a iluminar los segmentos que seleccionemos de una posición en concreto poniendo a 0 el cátodo común de esa posición y a 1 el cátodo de las otras tres posiciones.

En la figura 6 podemos apreciar una captura del esquemático de este display para poder entender mejor el funcionamiento.

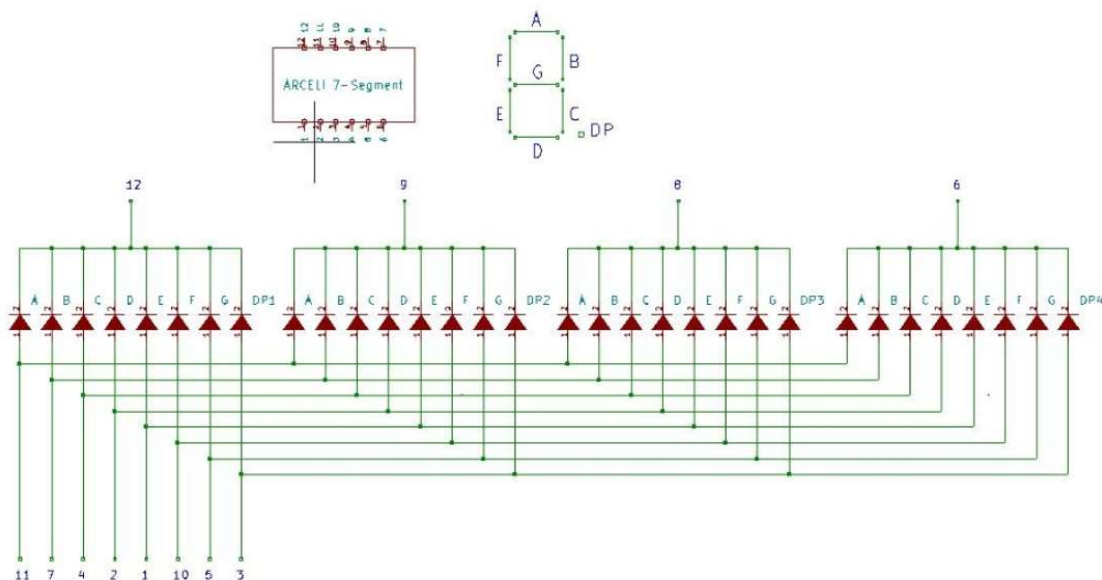


Figura 6. Esquemático del display de la web [electronicathido.com](http://www.electronicathido.com).

Si queremos mostrar un 7 en la posición número 2 empezando por la derecha, vamos a tener que iluminar los segmentos A, B y C. Eso significa que tendremos que dar una señal HIGH a los pines 11, 7 y 4 del display. Los otros pines de los segmentos los vamos a poner en LOW. Para seleccionar la posición 2 lo que tendremos que hacer es poner a nivel alto los pines del display 6, 9 y 12 y a nivel bajo el pin número 8.

Poniendo el pin del cátodo a nivel bajo permitimos el paso de corriente que posibilita la iluminación de esa posición. Si lo ponemos a nivel alto no pasa corriente y, por tanto, no será posible iluminarla.

3.1.4. Características de la impresora Dimatix DMP-2850 (Fujifilm)

La Dimatix DMP-2850 es una impresora de materiales basada en un sistema de deposición de los mismos.

La impresora permite la deposición de materiales fluidos sobre un sustrato usando un cartucho de tinta piezoeléctrico desechable. Esta impresora puede crear y definir patrones en un área de unos 200 x 300 mm y manipular sustratos de hasta 25 mm de grosor con una altura Z ajustable. La temperatura de la platina de vacío, que fija el sustrato en su sitio, se puede ajustar hasta 60°C.

Además, un editor de formas de onda y un sistema de cámara de observación de gotas permiten manipular los impulsos electrónicos al dispositivo de inyección piezoeléctrico para optimizar las características de la gota a medida que es expulsada de la boquilla. Este sistema facilita la impresión de estructuras y muestras para la verificación de procesos y la creación de prototipos.

La característica más exclusiva de este sistema de impresión de sobremesa es el propio cabezal de impresión, una novedad mundial. FUJIFILM Dimatix tiene un cabezal de impresión tipo cartucho basado en MEMS que permite a los usuarios llenar sus propios fluidos e imprimir inmediatamente con el DMP en su propio laboratorio.

En la siguiente página vamos a añadir una fotografía de la impresora (figura 7).



Figura 7. Imagen de la impresora Dimatix DMP-2850.

A continuación, explicaremos los principales aspectos del software que nos permite hacer uso de la impresora. También vamos a detallar cómo se hacen las configuraciones iniciales y previas a las primeras impresiones.

Al abrir el software de la impresora nos sale una pantalla como la de la figura 8.

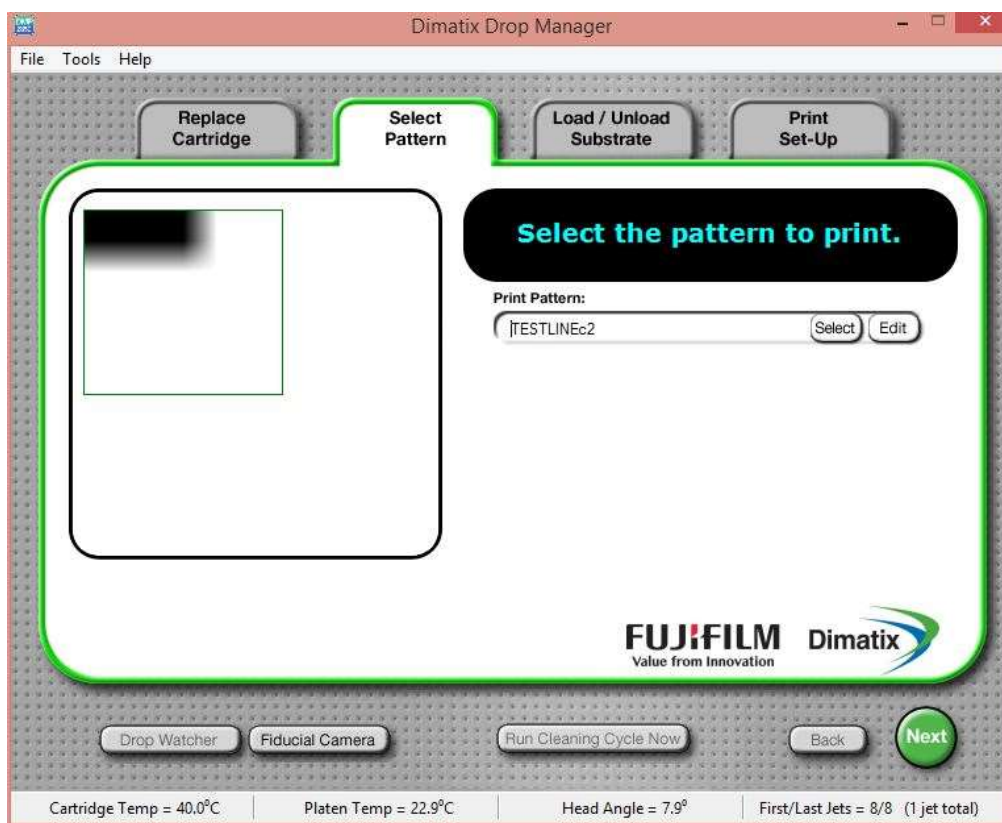


Figura 8. Captura de la pestaña "Select Pattern" del programa.

Como podemos comprobar, arriba tenemos cuatro pestañas principales. En el primer momento que abrimos el programa iremos directos al "Replace Cartidge", en el que tendremos que colocar el cartucho de tinta si no había. Si no lo colocamos no nos dejará avanzar a las siguientes pantallas. También, en esta fase, podemos colocar un filtro limpiador, para que el cartucho pueda ir accediendo a ciclos de limpieza antes de hacer una impresión, o incluso durante la impresión, configurándolo.

Cuando ya tenemos colocado el cartucho, lo segundo que se tendría que hacer es calibrarlo. Para eso vamos a acceder al "Drop Watcher", que está ubicado abajo a la izquierda. Al abrirlo nos aparecerá la pantalla mostrada en la figura 9.

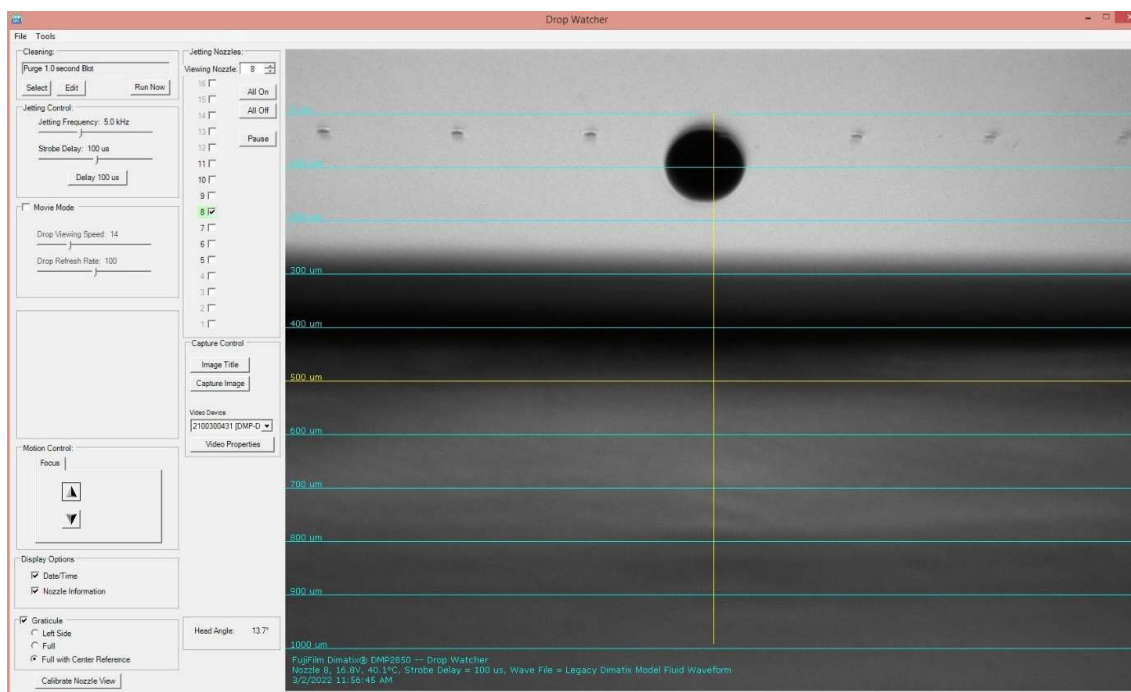


Figura 9. Captura del "Drop Watcher" del programa.

Lo primero que hay que hacer es "Calibrate Nozzle View", en que vamos a calibrar el posicionamiento de estos pequeños orificios que se ven en la imagen, dispuestos de forma horizontal. Son los agujeros por donde va a salir la tinta. En este momento básicamente lo que tendremos que hacer es ajustar la intersección de la línea vertical amarilla con la primera línea horizontal azul al "nozzle" (orificio) número uno y número dieciséis.

Seguidamente, una vez ya hemos realizado la calibración, ajustando la temperatura, la tensión aplicada en el orificio y la forma de onda con la que se eyecta la gota, si la configuración es correcta veremos como empiezan a caer gotas si tenemos activado el "Movie Mode". Entonces lo desactivamos y veremos una gota en la pantalla. Se trata de ajustar esta gota en la intersección de las líneas amarillas modificando el voltaje aplicado. Normalmente esta tensión estará dentro del rango 12 V – 20 V. La tensión, la forma de onda y la temperatura se modifican desde el "Cartridge Settings", que es una ventana que se nos abre directamente con el "Drop Watcher".

En la siguiente página adjuntamos una captura de cómo debería quedar la gota (figura 10). En este caso, solo imprimiremos con uno de los "nozzles" (orificios) por tener un mayor control de lo que imprimimos. A modo de ejemplo, en este apartado, seleccionamos el octavo "nozzle" porque es el que mejor gota eyectaba.

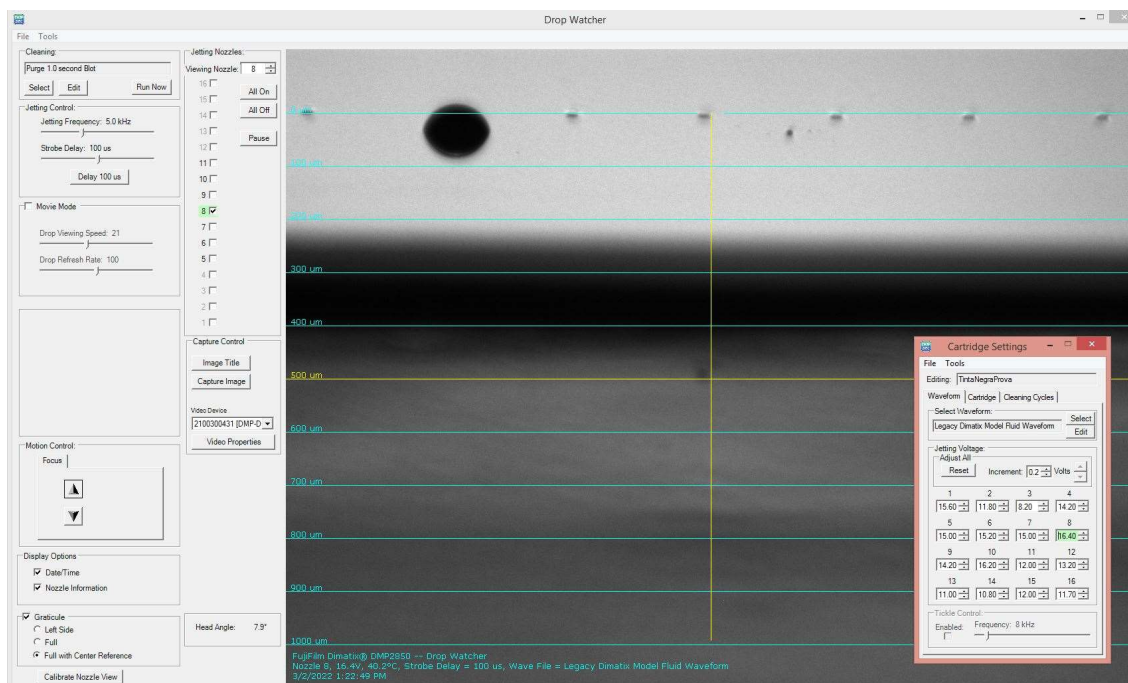


Figura 10. Captura del "Drop Watcher" con la gota situada.

En el "Cartridge Settings" tenemos tres pestañas:

- "Waveform": nos permite configurar la forma de onda en que se va a eyectar la gota y la tensión que queremos aplicar en cada orificio. En este caso el que tenemos seleccionado es el 8.
- "Cartridge": aquí seleccionamos el orificio que queremos que imprima y la temperatura que vamos a aplicar en el mismo.
- "Cleaning Cycles": en esta pestaña configuramos el ciclo de limpieza previo a empezar la impresión e incluso podemos configurar ciclos de limpieza mientras la impresión está en curso.

Es necesario comentar la importancia de que la gota no haga efecto satélite. Es decir, la gota, cuando ya la tenemos encuadrada en la intersección, tiene que ser perfectamente esférica. No tiene que llevar una cola, porque entonces significa que cuando caiga e impacte contra el Kapton va a desparramarse mucho más la tinta y nos va a ensuciar los alrededores más inmediatos de la gota.

Una vez hemos explicado el "Drop Watcher" pasaremos a exponer las funcionalidades de la "Fiducial Camera". Podemos abrirla desde la pestaña que hay al lado del "Drop Watcher", en la pantalla principal. La "Fiducial Camera" es un microscopio, que permite ver con claridad lo que hemos imprimido, ya que, a simple vista, dependiendo del diseño, no podremos distinguir mucho más que puntos.

En la página siguiente, en la figura 11, visualizaremos una captura de lo que podemos ver al abrir la "Fiducial Camera".

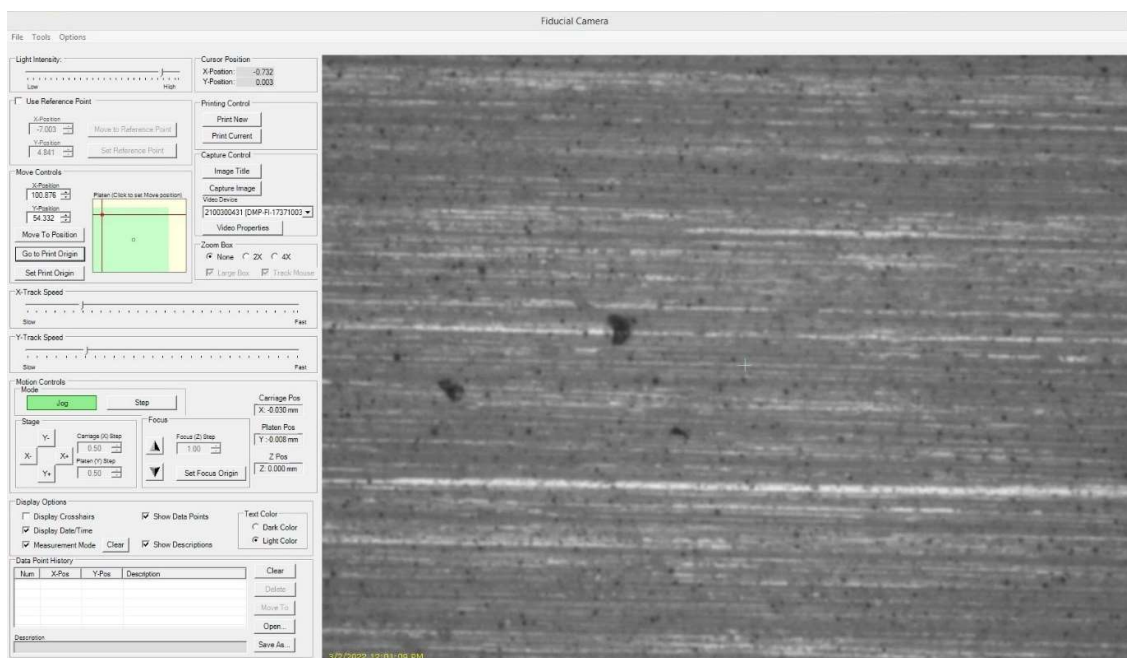


Figura 11. Captura del "Fiducial Camera".

Como ya hemos comentado, esta herramienta nos permite visualizar con detalle las impresiones que realizamos.

A parte, también se usa para fijar el punto origen a partir del cual se comenzará a imprimir. Corresponderá al extremo superior izquierdo del patrón que hayamos hecho. Para fijar este punto de origen tenemos la opción "Set Print Origin".

Ahora podemos volver a las pantallas iniciales del programa, teniendo en cuenta que ya hemos seleccionado el patrón (que tiene que estar en formato .ptf) en la primera captura que hemos adjuntado del programa.

La siguiente pantalla que nos encontramos es la de cargar el sustrato. Tenemos que configurarle el grosor que tiene el Kapton, que es de $50\ \mu\text{m}$. Ponemos más (aproximadamente $53\ \mu\text{m}$). También añadimos la temperatura a la cual calentaremos el sustrato. Lo dejamos por defecto en este caso.

En la página siguiente adjuntamos la captura de esta pantalla del programa (figura 12).

Una vez ya llegamos a la última pantalla, la de "Print Set-Up", después que la impresora ya se haya situado en posición de impresión, podremos darle al botón de "Print", que lo vamos a encontrar en el mismo sitio donde está "Next" en la siguiente captura.

Una vez la impresora nos diga que ya ha finalizado su impresión, podemos abrir el "Fiducial Camera" para visualizar con claridad qué es lo que hemos imprimido exactamente.

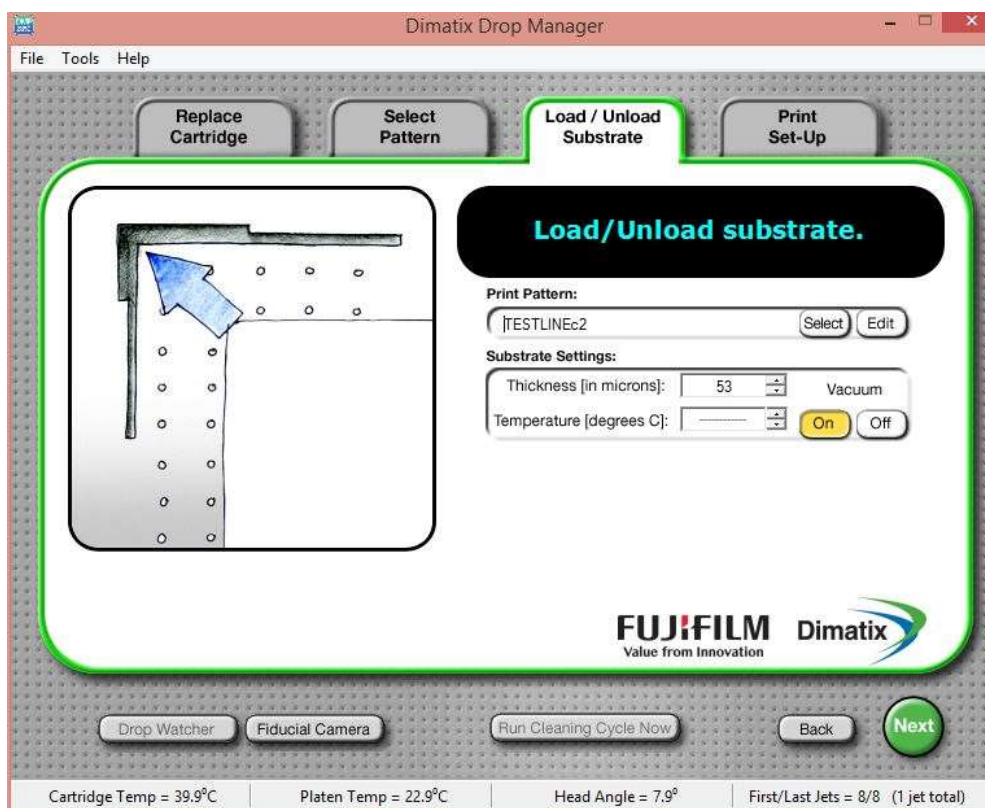


Figura 12. Captura de la pestaña "Load/Unload Substrate" del programa.

3.2. Sensores capacitivos

Los sensores capacitivos son aquellos que reaccionan ante variaciones en la capacidad. Por lo tanto, pueden ser utilizados para medir cualquier parámetro que, al variar, modifique la capacidad de un capacitor.

La capacidad es una propiedad eléctrica que existe entre dos conductores cualesquiera separados por un no conductor. El modelo básico de capacitor son dos placas metálicas separadas por un dieléctrico entre ellas. Un dieléctrico es aquel material que se polariza cuando está en contacto con un campo eléctrico.

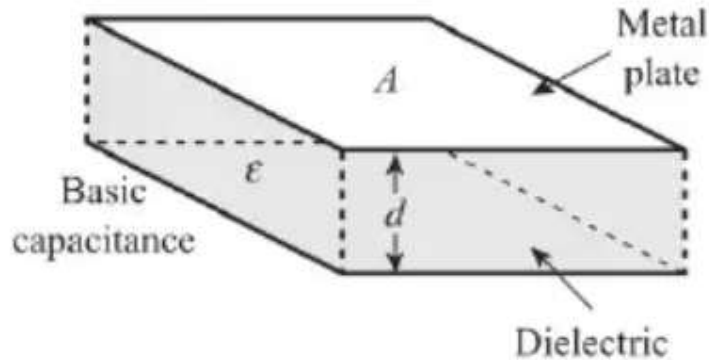


Figura 13. Modelo básico de capacitor.

La fórmula que determina la capacidad de un capacitor es la siguiente:

$$C = \epsilon_0 \cdot \epsilon_r \cdot \frac{A}{d} \quad (1)$$

Donde:

- C es la capacidad
- ϵ_0 es la permitividad del vacío, de aproximadamente $8,85 \cdot 10^{-12}$ F/m.
- ϵ_r es la permitividad relativa del dieléctrico.
- A es la superficie de las placas encaradas.
- d es la distancia entre placas.

Algunos de los sensores capacitivos más comunes son los de proximidad, entre otros.

En los sensores de gases químicos, por ejemplo, el fundamento lo encontramos en el dieléctrico. Si el capacitor no se encuentra bajo la influencia de ningún gas el dieléctrico será el aire, con una permitividad relativa determinada. Asimismo, cuando haya presencia de algún gas, variará la permitividad relativa, y así el valor de la capacidad. Este fundamento, el de cambios en el dieléctrico, se aplica para las otras tipologías de sensor capacitivo.

La estructura más básica de capacidad que podríamos plantear son dos líneas paralelas, pero tendrían que ser muy largas para poder llegar a medir un valor de capacidad apreciable. Tenemos que tener en cuenta que por mucho que la distancia entre líneas sea del orden de decenas de μm , la superficie de las líneas encaradas será la resultante de multiplicar la longitud y el grueso vertical que involucra la superficie efectiva enfrentada entre placas que podamos llegar a conseguir. Eso resulta en un valor muy pequeño de superficie por la influencia del pequeño grueso.

A esta problemática, lo que proponemos es diseñar las capacidades de forma interdigitada, como se muestra en la figura 14.

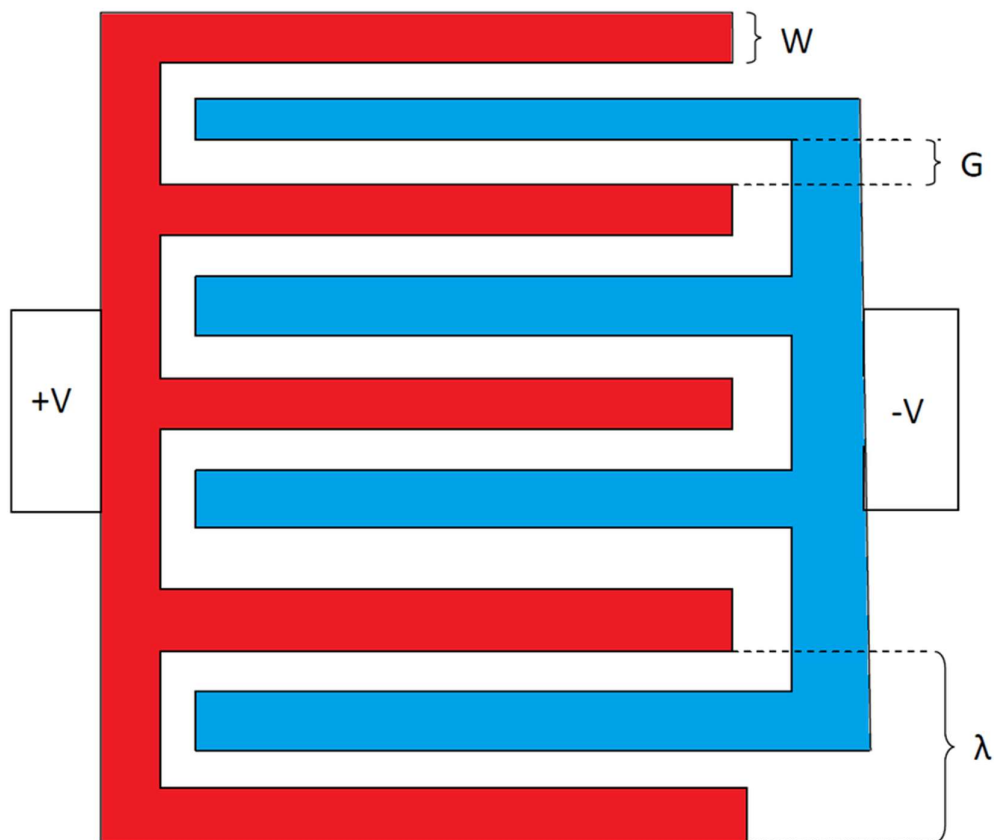


Figura 14. Estructura interdigitada.

Aquí podemos comprobar cómo vamos repitiendo el patrón de líneas paralelas n veces, así la capacidad resultante de encarar dos líneas la tendremos que multiplicar por n .

Las líneas que salen de la izquierda tendrán que ir todas conectadas a un electrodo y no podrán tocar el otro. Las líneas que salen de la derecha se conectarán al electrodo diestro y no podrán tocar el otro.

El objetivo de diseñar así las capacidades es poder trabajar con valores que sean medibles, con una precisión aceptable.

3.3. Métodos de medida de capacidades

En este apartado expondremos algunos de los métodos que se usan para medir capacidades a nivel introductorio. Más adelante, los analizaremos y veremos si son viables para nuestro proyecto. Tenemos que tener en cuenta que mediremos capacidades muy pequeñas, del orden de muy pocos picofaradios, incluso menos.

3.3.3. Red RC

Este método se basa en la curva de carga de una red RC. Se trata de medir la constante de tiempo de la misma mediante un microcontrolador y software. La constante de tiempo la encontramos cuando la tensión del condensador, en un proceso de carga, alcanza el 63,2% de la tensión final.

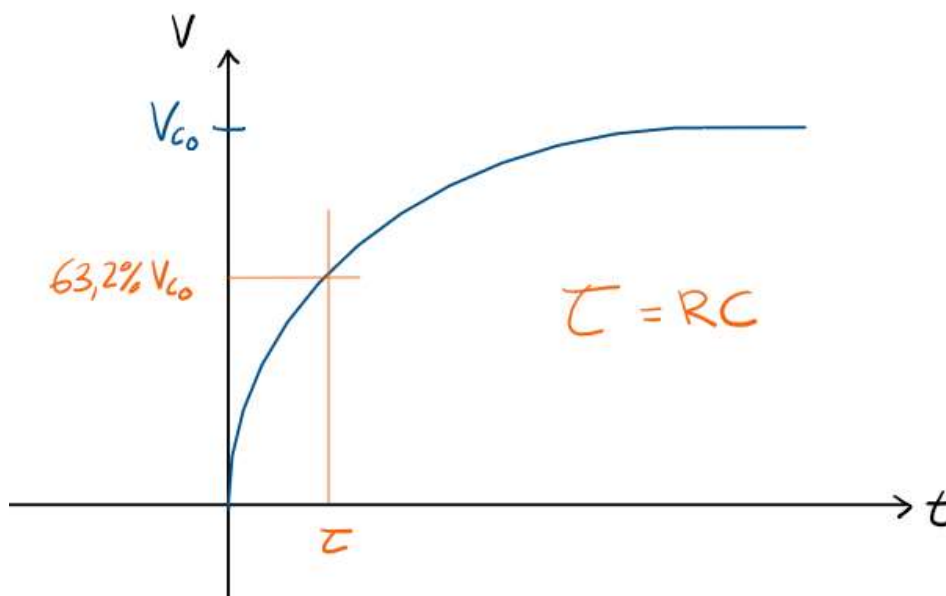


Figura 15. Curva de carga de una red RC.

Mediante un programa alimentaremos una red RC con una onda cuadrada de un período calculado a conciencia para que el condensador de más capacidad que queramos medir le dé tiempo a, al menos, llegar a la constante de tiempo. Con la onda cuadrada cargamos y descargamos la red, sucesivamente, para realizar medidas periódicas. El programa va contando hasta que alcanza el 63,2% de la tensión final, donde ya se puede extraer la constante de tiempo, y con ésta la capacidad (conocemos el valor de la resistencia porque lo fijamos).

3.3.4. Divisor de tensión con condensadores

En ese caso no necesitamos ni resistencia. Se basa en la presencia de un condensador propio del pin al que conectaremos el condensador que queremos medir.

Figure 14-1. I/O Pin Equivalent Schematic

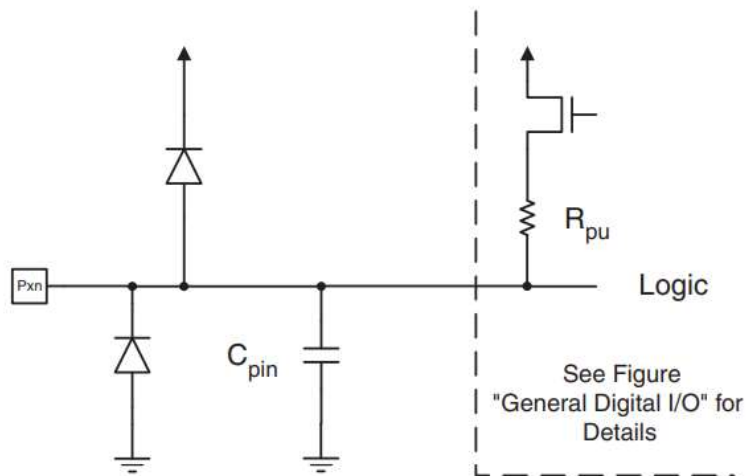


Figura 16. *megaAVR® Data Sheet*, pág. 84.

Cuando tengamos conectado el condensador en el pin correspondiente, nos quedará un divisor de tensión con condensadores, como podemos comprobar en la figura 17, podremos obtener la ecuación 2.

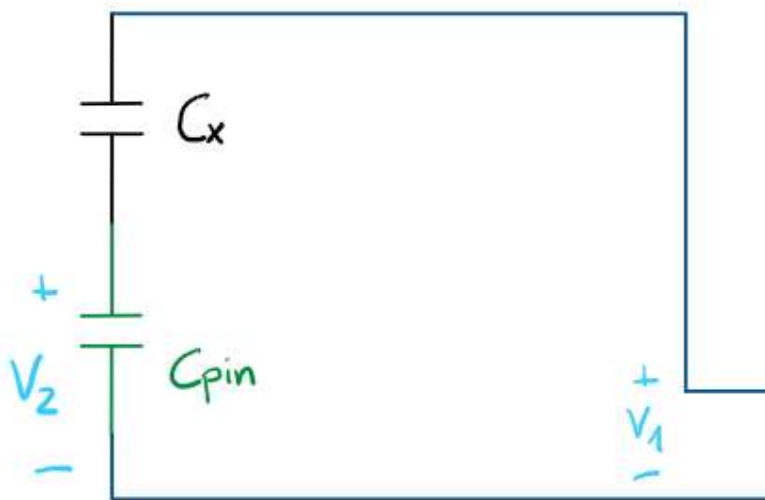


Figura 17. Divisor de tensión con condensadores.

$$V_2 = V_1 \cdot \frac{C_x}{C_x + C_{pin}} \quad (2)$$

3.3.5. Oscilador tanque LC

Consiste, básicamente, en lanzar un impulso a un tanque LC que genere una onda sinusoidal y exponencial decreciente. El tanque resonará a una frecuencia determinada, que, con un valor fijo de inductor, nos permitiría extraer el valor del condensador.

Una vez tengamos la onda sinusoidal debemos convertirla en una onda cuadrada para que el microcontrolador pueda leerla. Esto podemos hacerlo con un comparador. En el microcontrolador se podría configurar una interrupción por cambio en el pin y así detectaremos los cambios de nivel alto a bajo, y viceversa, de la onda cuadrada y consecuentemente, los pasos por 0 de la onda sinusoidal. Consistiría, pues, en realizar una cuenta del tiempo que pasa entre una interrupción y otra (un semiperíodo). A partir de aquí podemos extraer el período y la frecuencia de oscilación del tanque resonante LC.

En la figura 18 podemos ver como es el circuito resonante.

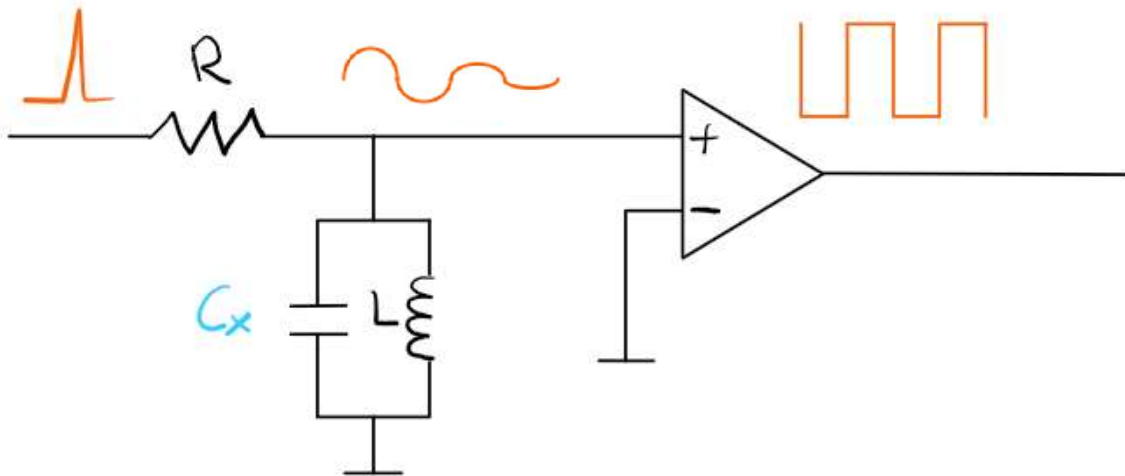


Figura 18. Oscilador con tanque LC.

Una vez conocemos la frecuencia en la que está oscilando el tanque LC podemos aplicar la ecuación 3 para poder extraer el valor de la capacidad.

$$f_o = \frac{1}{2 \cdot \pi \cdot \sqrt{LC}} \quad (3)$$

3.3.6. Circuito con tensión de salida proporcional al condensador a medir

Este circuito estará alimentado por una onda sinusoidal de una amplitud y frecuencia determinada. Tendrá una primera etapa con un amplificador operacional como podemos ver en la figura 19, que aparece en la siguiente página.

A continuación de la figura haremos el análisis del circuito y veremos que, si mantenemos la resistencia de realimentación constante, el valor de la tensión de salida variará de forma lineal con los cambios de capacidad que puedan producirse.

El objetivo final es que un microcontrolador nos vaya tomando muestras de la tensión y nos calcule la capacidad, pero para esto después de esta primera etapa quizás tendremos que amplificar esta tensión (dependiendo del rango de capacidad que queramos medir) y añadir una etapa rectificadora, para que el microcontrolador lea una tensión continua.

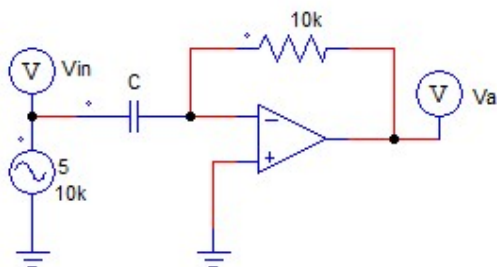


Figura 19. Etapa RC con amplificador operacional.

El análisis de este circuito en el dominio de Laplace es el siguiente:

$$V_{in} \cdot C \cdot s = -\frac{V_a}{10 \cdot 10^3} \quad (4)$$

$$V_a = -10 \cdot 10^3 \cdot V_{in} \cdot C \cdot s \quad (5)$$

Podemos comprobar como la variación de la tensión de salida de esta etapa será lineal a los cambios en el valor de la capacidad.

La siguiente etapa será amplificadora. Dependiendo del rango de capacidades que queramos medir aplicaremos un factor amplificador u otro, cosa que determinaremos más adelante. La etapa amplificadora la podemos ver en la figura 20:

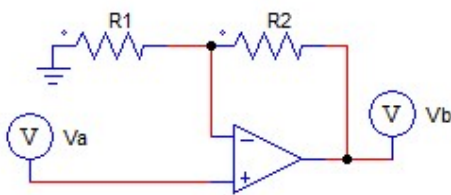


Figura 20. Etapa amplificadora.

El análisis de esta etapa es el siguiente:

$$-\frac{V_a}{R_1} = \frac{V_a - V_b}{R_2} \quad (6)$$

$$-V_a \cdot R_2 = R_1 \cdot (V_a - V_b) \quad (7)$$

$$V_b \cdot R_1 = V_a \cdot (R_1 + R_2) \quad (8)$$

$$V_b = V_a \cdot \left(1 + \frac{R_2}{R_1}\right) \quad (9)$$

Por último, necesitamos una etapa rectificadora que nos pase la tensión alterna a continua. En la siguiente página, en la figura 21, mostraremos esta etapa.

Dicha etapa está basada en un circuito detector de pico con amplificador operacional. Al añadirle este último elemento eliminamos el impacto de la caída de tensión del diodo cuando está polarizado de forma directa (0,7 V aproximadamente).

El diodo se ocupa de rectificar los semiperíodos negativos y hacerlos 0, por lo que hablamos de rectificación de media onda. El condensador se ocupa de mantener la tensión (sin caídas sustanciales) con el valor del pico de la onda.

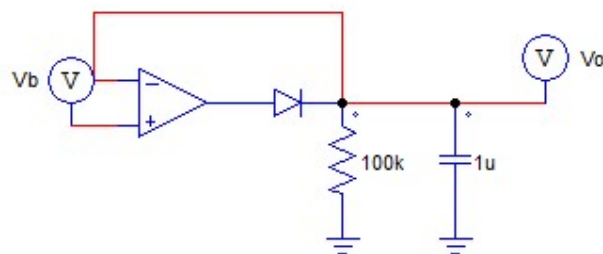


Figura 21. Etapa rectificadora.

Así pues, el circuito, con todas las etapas en cascada, queda configurado tal y como muestra la figura 22:

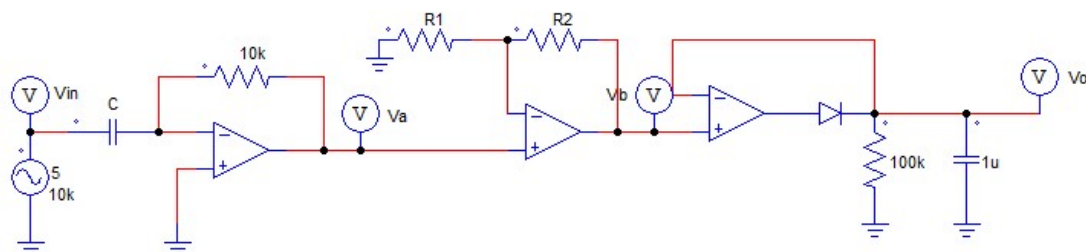


Figura 22. Circuito completo.

3.3.7. Circuito con puente RC

Este circuito estará alimentado por una onda sinusoidal de una amplitud y frecuencia determinada. Tendrá una primera etapa que será un puente RC, como podemos comprobar en la siguiente página, en la figura 23.

Del puente, medimos la tensión diferencial entre patas positivas de los condensadores.

El objetivo final, como en otros métodos, es obtener una tensión continua que varíe de forma lineal a las variaciones de capacidad. Para lograrlo, lo que hacemos es introducir una etapa que constará de un amplificador de instrumentación, al que introduciremos la tensión diferencial mencionada para obtener una tensión a la salida proporcional.

Esta tensión de salida del amplificador de instrumentación no es continua aún, pero por lo menos ya no es diferencial. Para pasar la tensión a continua añadimos una etapa rectificadora como ya hemos explicado en el método anterior.

En la siguiente página, en la figura 23, adjuntamos el esquema del circuito en el PSIM.

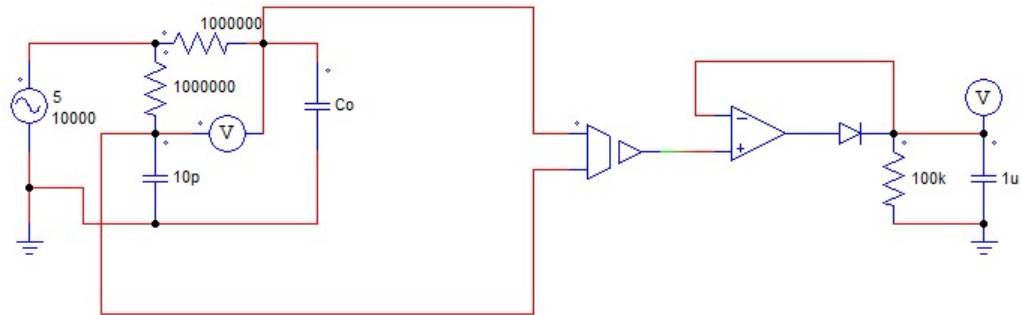


Figura 23. Circuito puente RC completo.

Habiendo presentado los métodos y sus fundamentos, más adelante los compararemos y esclareceremos cuál usar.

4. Comparativa de los métodos para medir capacidades

4.1. Red RC

En la figura 24 podemos ver el esquema del circuito y las conexiones realizadas con la placa del microcontrolador.

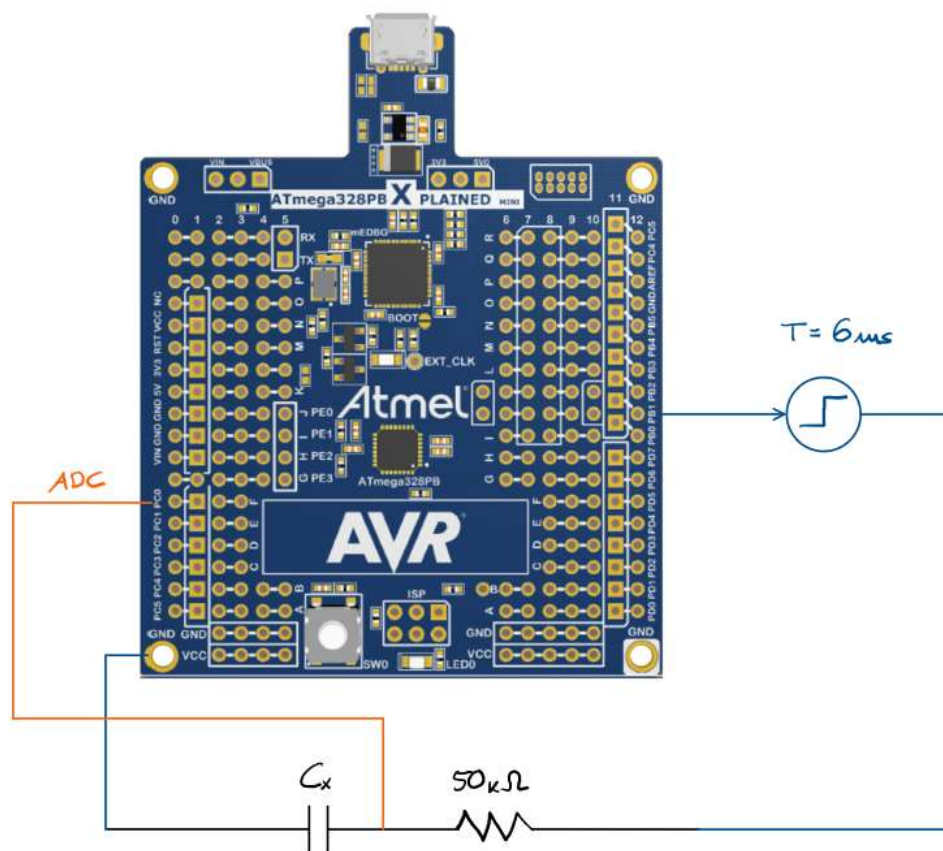


Figura 24. Configuración HW de una red RC.

Debemos tener en cuenta que el microcontrolador tarda en realizar una conversión AD entre $20\text{ }\mu\text{s}$ y $40\text{ }\mu\text{s}$. Si estamos midiendo condensadores a partir de 10 nF y tenemos una resistencia de $50\text{ k}\Omega$, por ejemplo, con el condensador más pequeño tendremos una constante de tiempo ($\tau = RC$) de $500\text{ }\mu\text{s}$. Nos permite realizar más de 10 conversiones hasta la constante de tiempo en el peor de los casos.

El microcontrolador no podrá realizar conversiones más rápido, esto es importante.

Con esta resistencia se pueden medir condensadores a partir de 10 nF .

Si, por ejemplo, queremos medir un condensador de 100 nF (con una resistencia de $50\text{ k}\Omega$), la constante de tiempo es de 5 ms , por lo que el programa no funcionaría correctamente. El semiperíodo de la onda cuadrada es de 3 ms , y, por tanto, el condensador no alcanzaría ni el $63,2\%$ de carga. Deberíamos aumentar el período de la onda cuadrada para solucionarlo. Así pues, este método nos permite medir capacidades grandes a partir de 10 nF , lo único que deberíamos hacer es ir modificando el período de la onda cuadrada para dar el tiempo necesario a que se carguen a un $63,2\%$, al menos, los condensadores.

Si lo que queremos hacer es medir capacidades menores, con este método tendremos problemas.

Puesto que no podemos realizar conversiones a mayor velocidad, lo único que podemos hacer es aumentar la resistencia para mantener constantes de tiempo cercanas a 500 μ s, que es lo mínimo necesario para obtener una medida fiable.

Si por ejemplo quisiéramos medir una capacidad de 10 pF y queremos que la constante de tiempo sea de 500 μ s necesitamos una resistencia de 50 M Ω . Aquí aparece la problemática. Una resistencia tan grande produce una cantidad de ruido inasumible, aparte de que tiene una caída de tensión enorme. Se descarta este método para capacidades tan pequeñas.

Éste es el código que programamos para evaluar este método:

```
#include <avr/io.h>
#include <avr/interrupt.h>
#define R 50

volatile int tics = 0;
volatile int control = 1;
volatile int tao=0, num_tao = 0;
volatile float capacitat;

void ini_timer1_int()
{
    PRR=PRR&0xf7; //Activamos Timer1
    TCCR1A=0x82; //Modo operación PWM, ICR1 TOP y cambio en OCR1A
    TCCR1B=0x1D; //Preescaler 1024
    OCR1A=150; //Ciclo de trabajo de 50 %
    ICR1=300; //Período de 6 ms
    DDRB=0xff; //Configuración PB0-PB7 como salida
}

void ini_timer0_int()
{
    PRR=PRR&0xdf; //Activamos Timer0
    TCCR0A=0x02; //CTC
    TCCR0B=0x02; //Preescaler 8
    OCR0A=20; //Haremos saltar int timer cada 10 us
    TIMSK0=0x02; //Interrupciones Output Compare A
}

void ini_ADC_int()
{
    PRR=PRR&0xfe; //Activamos ADC
    DDRC=0x00; //Configuración PC0-PC7 como entrada
    ADMUX=0x40; //Vcc con condensador externo (incluido en la xplained mini),
    selección canal ADC0 y ADLAR = 0
    ADCSRA=0x8a; //Preescaler 4
    ADCSRB=0;
}

ISR (TIMER0_COMPA_vect) //salta cada 10 us
{
    tics++;
}

ISR (ADC_vect) //salta cuando acaba una conversión
{
    if (ADC < 0x0005) //reinicializar casi al principio del proceso de carga
    {
        control = 1;
        tics = 0;
    }
}
```

```
if (control)
{
    if (ADC>67) //en la medida siguiente de superar el 63,2 % de 5 V
    {
        tao = tao + 10*tics;
        num_tao++;
        if (num_tao == 10)
        {
            capacitat = ((float)tao/(float)num_tao)/R;
            num_tao = 0;
        }
        control = 0; //porque no coja valores durante la descarga
    }
    ADCSRA = ADCSRA | 0x40; //inicio siguiente conversión
}

int main(void)
{
    ini_ADC_int();
    ini_timer1_int();
    ini_timer0_int();
    ADCSRA = ADCSRA | 0x40;
    sei();
    while (1)
    {
    }
}
```

Así pues, se podrían extraer las siguientes conclusiones de este método:

- Ventajas:
 - Es adecuado para capacidades superiores a 10 nF.
 - El código es relativamente fácil de programar.
 - El programa es ampliable a si deben enviarse los valores de capacidad resultantes a partir de alguna comunicación determinada.
- Inconvenientes:
 - No funciona correctamente, o pierde facultades, cuando intentamos medir capacidades inferiores a 10 nF.
 - Hay que ir modificando código, o programar la modificación, conforme cambiemos el rango de capacidades a medir.

4.2. Divisor de tensión con condensadores

En la figura 25 podremos ver el esquema del circuito y las conexiones realizadas con la placa del microcontrolador.

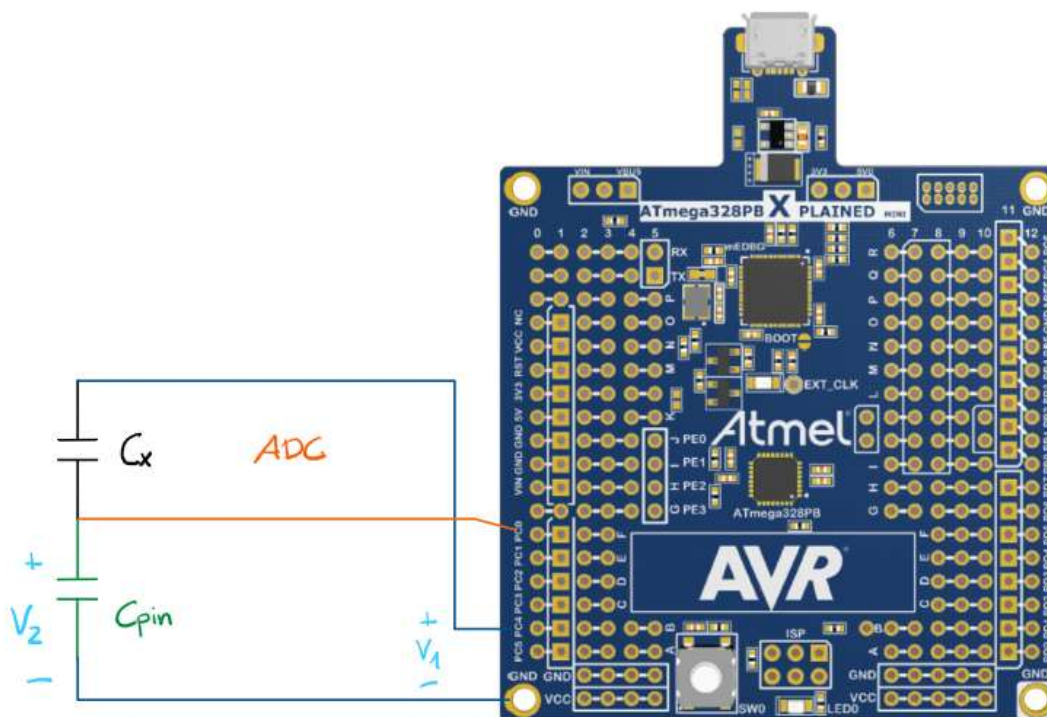


Figura 25. Configuración HW de un divisor de tensión con condensadores.

Desde el pin PC4 subministramos 5 V en un momento dado, ordenamos una conversión, que cuando termina el condensador ya está cargado (se carga mucho más rápido que el tiempo que tarda en realizarse una conversión). Una vez tenemos la tensión V_2 , que es la que nos lee el ADC, podemos aplicar la fórmula y extraer el valor de la capacidad, sabiendo que V_1 es la tensión de fondo de escala y que C_{pin} deberá calibrarse en el laboratorio con un C_x que conozcamos el valor exacto. Una vez que el programa ya tiene el valor de la tensión V_2 configuramos el PC4 a nivel bajo (0 V) y el condensador se empieza a descargar. Para que lo haga de forma más rápida cambiamos el modo de PC0 a salida, así hacemos que la impedancia del pin sea menor, haciendo que el condensador tarde menos en descargarse. Este proceso lo hace en bucle, configurando PC0 como entrada al principio del mismo.

Nos permite medir condensadores de bajas capacidades, de 0,1 pF a 40 pF aproximadamente. Cuando empezamos a medir mayores capacidades el sistema pierde mucha precisión porque el condensador empieza a tardar más en cargarse y puede que termine la conversión ADC antes de que se haya cargado el condensador.

Éste es el código que programamos para avaluar este método:

```
#include <avr/io.h>
#include <avr/interrupt.h>

int main(void)
{
    float capacitat;
    int resultat;
    PRR=PRR&0xfe; //Activamos ADC
    DDRC=0x10; //PCx como entradas excepto PC4, que es salida
    ADMUX=0x40; //Canal PC0 y Vcc con condensador externo (incluido en xplained mini)
    ADCSRA=0x82; //Preescaler 4 y ADC habilitado
    while (1)
    {
        DDRC = DDRC & 0xfe; // PC0 como entrada
        PORTC = PORTC | 0x10; // Configuramos a 1 PC4
        ADCSRA = ADCSRA | 0x40; // Iniciamos conversión
        while ((ADCSRA & 0x10) == 0); // Espera mientras no termine la conversión
        int resultat = ADC;
        PORTC = PORTC & 0xef; // Configuramos a 0 PC4
        DDRC = DDRC | 0x01; // PC0 como salida
        capacitat = (float)resultat * 30 / (float)(1023 - resultat)-cap_parasita;
    }
}
```

Así pues, se podrían extraer las siguientes conclusiones de este método:

- Ventajas:
 - Es adecuado para capacidades entre 0,1 pF y 40 pF.
 - El código es relativamente fácil de programar.
 - El programa es ampliable a si deben enviarse los valores de capacidad resultantes a partir de alguna comunicación determinada.
- Inconvenientes:
 - No funciona correctamente, o pierde facultades, cuando intentamos medir capacidades superiores a 40 pF.
 - Es necesario calibrar el condensador del pin experimentalmente, y el valor de este no tiene por qué ser el mismo en todo momento. Su valor es fluctuante, dentro de una horquilla, pero puede cambiar de una medida a otra.

4.3. Oscilador tanque LC

En la figura 26 podremos ver el esquema del circuito y las conexiones realizadas con la placa del microcontrolador.

Después de mostrar el HW, analizaremos si este método es realmente viable o no. Tendremos que hacer una aproximación del inductor que tenemos que instalar en el circuito para que luego podamos medir capacidades en el rango deseado.

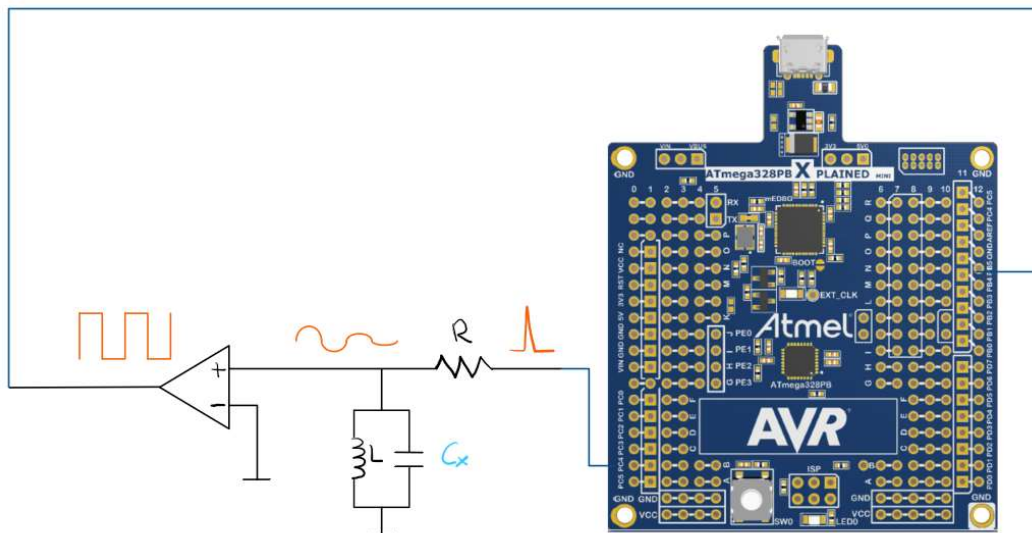


Figura 26. Configuración HW de un oscilador tanque LC.

Ahora que el tema del conexionado y hardware ya está explicado, conviene analizar la viabilidad de este método.

Tomando de base la mejor de las situaciones en cuanto a la frecuencia máxima a la que puede llegar a trabajar el microcontrolador, que en realidad no es real, nos permitirá ver si en el caso ideal lo que hemos planteado es viable. Si no lo es ni con el caso ideal, podemos concluir que no es viable.

Partiendo de que en el caso ideal el microcontrolador puede llegar a contar 16 millones de veces por segundo (imposible), determinamos que al menos pueda tomar 10 medidas de tiempo entre interrupciones por cambio de tensión en el pin. Por tanto, tendríamos una frecuencia del tanque resonante de 1,6 MHz. En caso de que quisiéramos medir un condensador de 1 pF, y resolviendo la ecuación 3, deberíamos colocar un inductor de 9,89 mH.

No hemos hecho ninguna prueba con este método, pero ya se ve que para medir condensadores tan pequeños no será viable porque los inductores que requiere son grandes, y si tenemos mejores opciones esta será mejor descartarla.

4.4. Circuito con tensión de salida proporcional al condensador a medir

En la figura 27 podremos ver el esquema del circuito y las conexiones realizadas con la placa del microcontrolador.

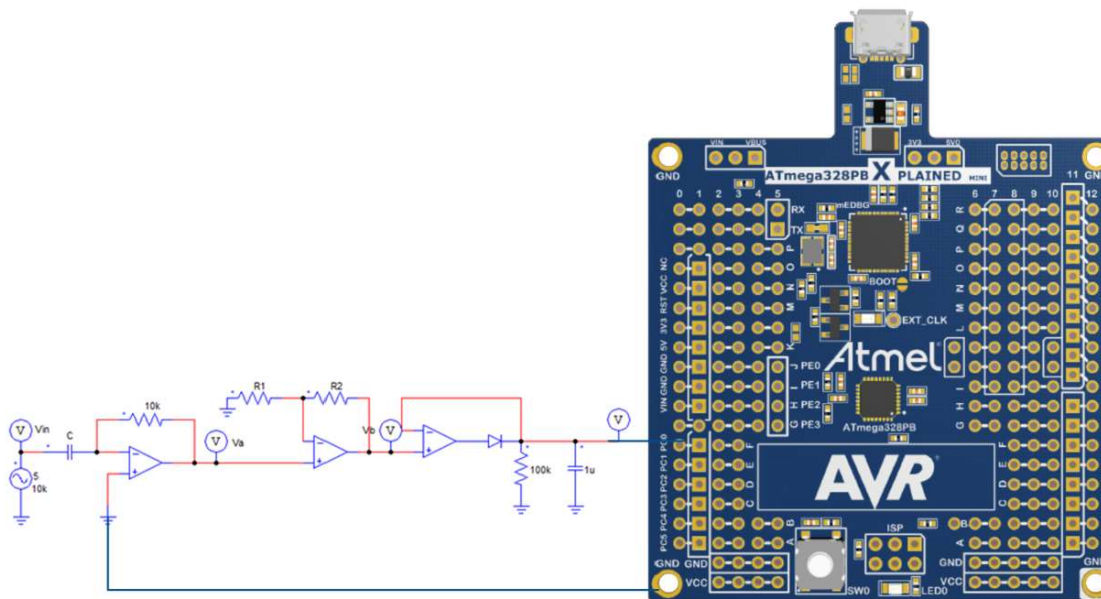


Figura 27. Configuración circuito con salida proporcional al condensador.

Podemos comprobar como conectamos la salida del circuito que ya hemos descrito anteriormente a un pin que configuraremos para que nos haga conversiones analógico-digitales.

Lo que tendremos que determinar aquí es el valor de R_1 y R_2 de la etapa amplificadora según el rango de capacidades que queramos medir.

Empezaremos haciendo el diseño del circuito por un rango de capacidades de $0,1 \text{ pF} - 100 \text{ pF}$. Como la máxima tensión que puede llegar a leer el microcontrolador es de 5 V , tendremos que hacer el diseño de tal forma que cuando tengamos el condensador más grande del rango la tensión de salida sea máxima, y que sea mínima para el condensador más pequeño.

Para simplificar el análisis consideraremos la amplitud de las señales a la hora de hacer los cálculos, sabiendo que al final lo que hace el rectificador es detectar el pico de la onda y mantenerlo con la mínima variación de tensión posible. Aquí consideraremos la amplitud de la señal V_b como la entrada (en tensión continua) al conversor analógico – digital del microcontrolador.

Entonces, recordemos las expresiones de V_a y V_b :

$$V_a = -10 \cdot 10^3 \cdot V_{in} \cdot C \cdot s \quad (5)$$

$$V_b = V_a \cdot \left(1 + \frac{R_2}{R_1}\right) \quad (9)$$

Para el condensador de 100 pF tenemos que tener en la salida, o en este caso, en V_b , una tensión de 5 V .

Sustituyendo en la ecuación 4 la amplitud del voltaje de entrada (5V), y el valor de capacidad que nos interesa:

$$V_a = -10 \cdot 10^3 \cdot 5 \cdot 100 \cdot 10^{-12} \cdot s$$

Si buscamos el módulo de la amplitud de V_a , sabiendo que la frecuencia es de 10 kHz:

$$|V_a| = 10 \cdot 10^3 \cdot 5 \cdot 100 \cdot 10^{-12} \cdot 2 \cdot \pi \cdot 10 \cdot 10^3$$
$$|V_a| = 0,31 \text{ V}$$

Ahora, se trata de que esta tensión, a través de la etapa amplificadora, pase a ser de 5 V con la ecuación 8.

$$5 = 0,31 \cdot \left(1 + \frac{R_2}{R_1}\right)$$
$$1 + \frac{R_2}{R_1} = 16,13$$

Se puede escoger una R_1 de 1 k Ω y una R_2 de 15 k Ω , para que la tensión de salida no sea exactamente 5 V, sino un poco menos.

El condensador rectificador lo escogemos mediante ensayos con PSIM, seleccionando el más pequeño posible (por cuestiones de coste) que nos permita mantener unos niveles de tensión continua estables para que las sucesivas lecturas del conversor analógico – digital no difieran mucho entre sí. El valor de condensador rectificador seleccionado es de 1 microfaradio.

Si aplicamos este factor de amplificación obtendríamos una tensión de salida (según la ecuación 8):

$$V_b = 0,31 \cdot (16)$$
$$V_b = 4,96 \text{ V}$$

A partir del PSIM se obtiene un poco menos de tensión, seguramente por ser la versión demo y por el modelo matemático que usa, igualmente la calibración real del circuito se tiene que hacer experimentalmente con un condensador a medir de valor previamente conocido. Asimismo, el PSIM nos da una aproximación de cómo funciona el circuito y su supuesta validez, sujeta a las múltiples situaciones que aparecen realmente.

Así pues, la tensión de salida que nos ofrece PSIM es la mostrada en la figura 28:

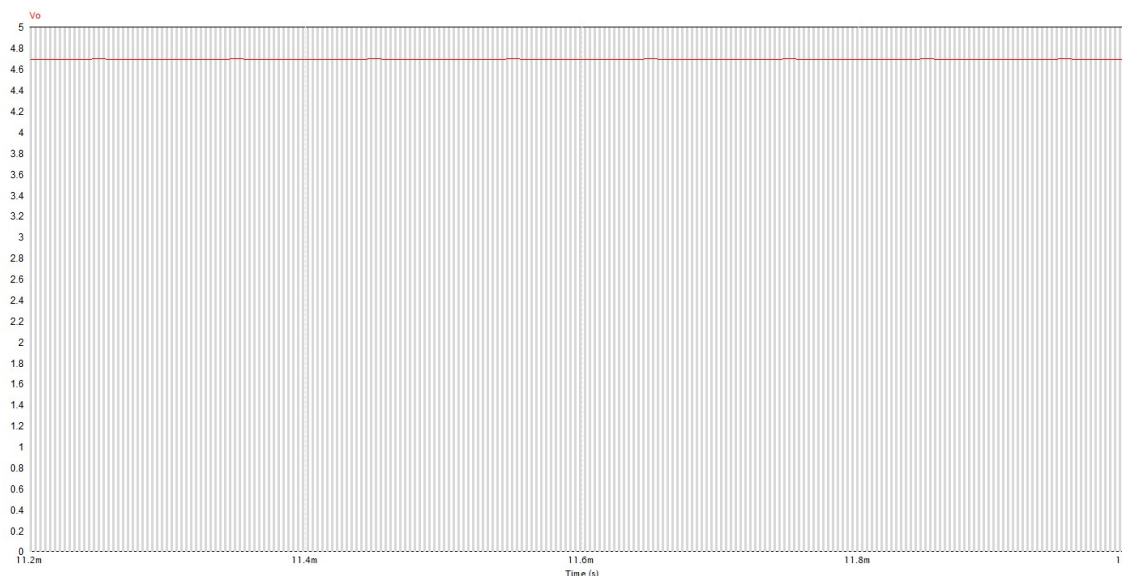


Figura 28. Tensión de salida con condensador de 100 pF.

También, para comprobar que la tensión de salida es lineal al condensador que queremos medir, vamos a obtenerla cuando el condensador es de 10 pF. En la figura 29 deberíamos obtener 0,47 V en V_0 .

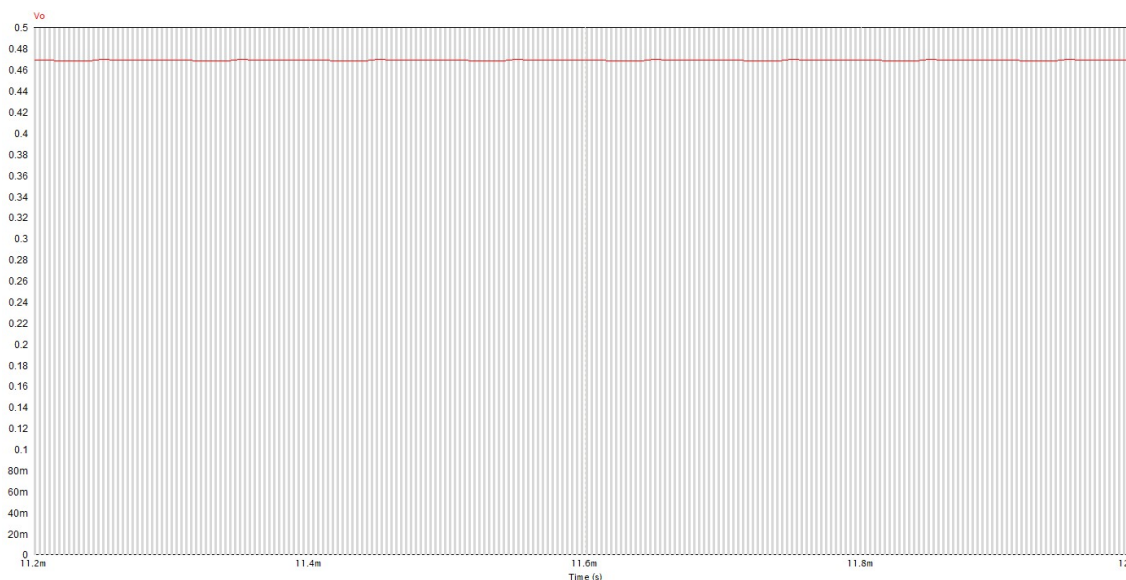


Figura 29. Tensión de salida con condensador de 10 pF.

Se confirma la linealidad del condensador que queremos medir con la tensión de salida.

Asimismo, las capacidades que vamos a medir, seguramente van a ser más pequeñas que 10 pF. Por esto, sería necesario diseñar un circuito que midiera hasta 10 pF, incluso otro que lo hiciera hasta 1 pF. Así, ya tendríamos diseñadas varias opciones por lo que podamos encontrarnos a la hora de imprimir capacidades.

Lo único que tendremos que hacer es modificar la etapa amplificadora. Repetimos los cálculos de una forma muy similar a como hemos hecho en el anterior caso.

De la ecuación 4 extraemos V_a :

$$V_a = -10 \cdot 10^3 \cdot 5 \cdot 10 \cdot 10^{-12} \cdot s$$

$$|V_a| = 10 \cdot 10^3 \cdot 5 \cdot 10 \cdot 10^{-12} \cdot 2 \cdot \pi \cdot 10 \cdot 10^3$$

$$|V_a| = 0,031 V$$

Ahora, se trata de que esta tensión, a través de la etapa amplificadora, pase a ser de 5 V con la ecuación 8.

$$5 = 0,031 \cdot \left(1 + \frac{R_2}{R_1}\right)$$

$$1 + \frac{R_2}{R_1} = 159,15$$

Se puede escoger una R_1 de 1 k Ω y una R_2 de 158 k Ω , para que la tensión de salida no sea exactamente 5 V, sino un poco menos.

En la figura 30, en la siguiente página, añadiremos la captura de PSIM donde podemos visualizar la tensión de salida con la configuración del circuito preparado para medir en un rango hasta 10 pF.

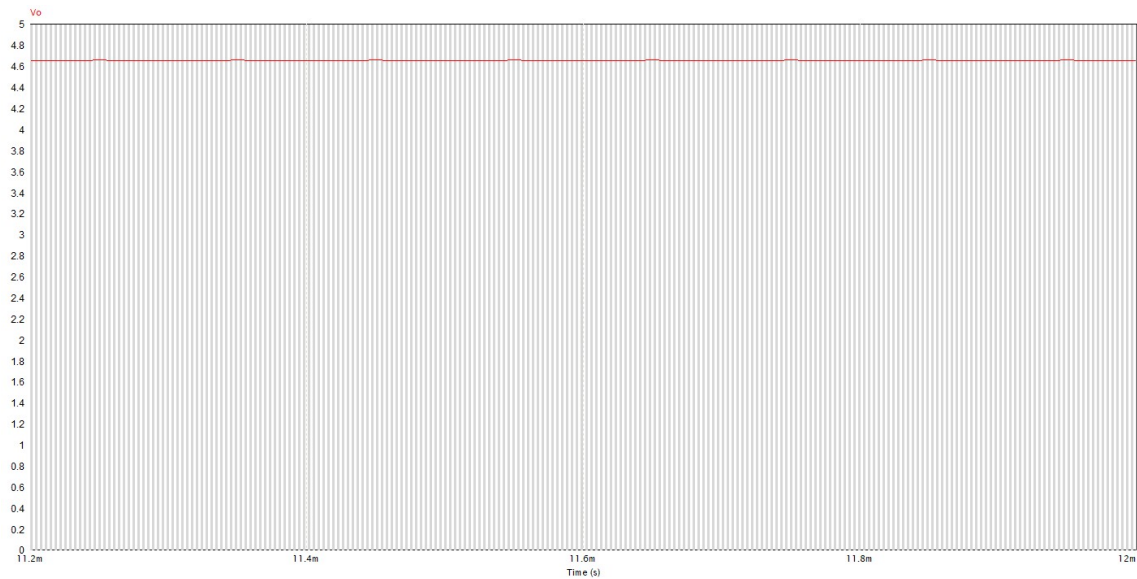


Figura 30. Tensión de salida con condensador de 10 pF.

Ya, por último, si quisiéramos diseñar una configuración de circuito con un rango hasta 1 pF, para medir capacidades muy pequeñas, haríamos lo siguiente:

De la ecuación 4 extraemos V_a :

$$V_a = -10 \cdot 10^3 \cdot 5 \cdot 1 \cdot 10^{-12} \cdot s$$

$$|V_a| = 10 \cdot 10^3 \cdot 5 \cdot 1 \cdot 10^{-12} \cdot 2 \cdot \pi \cdot 10 \cdot 10^3$$

$$|V_a| = 0,0031 \text{ V}$$

Ahora, se trata de que esta tensión, a través de la etapa amplificadora, pase a ser de 5 V con la ecuación 8.

$$5 = 0,0031 \cdot \left(1 + \frac{R_2}{R_1}\right)$$

$$1 + \frac{R_2}{R_1} = 1591,55$$

Se puede escoger una R_1 de 1 k Ω y una R_2 de 1,58 M Ω , para que la tensión de salida no sea exactamente 5 V, sino un poco menos.

En la figura 31 añadiremos la captura de PSIM con la tensión de salida de este último circuito cuando estamos midiendo 1 pF.

A continuación, también, adjuntaremos el programa básico que usaríamos para leer los valores de la tensión de salida, y a partir de aquí poder obtener las capacidades que queremos medir.

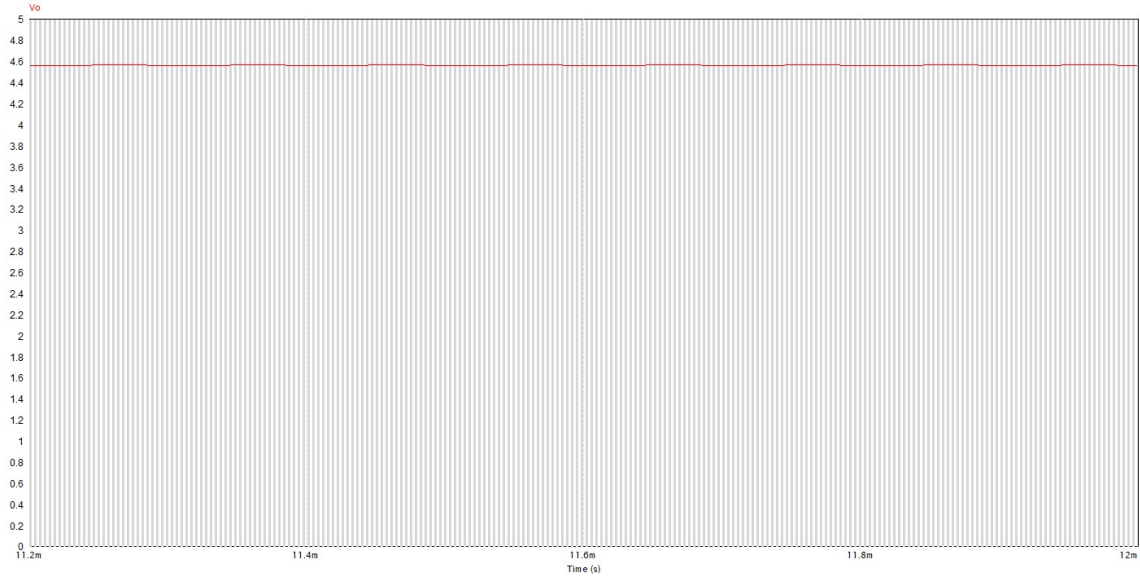


Figura 31. Tensión de salida con condensador de 1 pF.

El código del programa es el siguiente:

```
#include <avr/io.h>
#include <avr/interrupt.h>

volatile int tics = 0, contar = 1, tensio;
volatile float capacitat, capacitat_acum, capacitat_real;

void ini_timer2_int()
{
    PRR=PRR&0xbf; //Activamos Timer2
    TCCR2A=0x02; //CTC
    TCCR2B=0x07; //Preescaler 1024
    OCR2A=16; //Haremos que int timer salte cada 1 ms
    TIMSK2=0x02; //Interrupciones Output Compare A
}

void ini_ADC_enq()
{
    PRR=PRR&0xfe; //Activamos ADC
    DDRC = 0x00; //PCx como entrada
    ADMUX=0x40; //Canal PC0 y Vcc con condensador externo (incluido xplained mini)
    ADCSRA=0x82; //Preescaler 4 y ADC habilitado
}

ISR (TIMER2_COMPA_vect) //Salta cada 1 ms
{
    if (contar) tics++;
    if (tics == 10) //Para que le de tiempo a estabilizarse
    {
        contar = 0;
        tics = 0;
    }
}

int main(void)
{
    ini_timer2_int();
    ini_ADC_enq();
```

```
while (1)
{
    if (contar == 0)
    {
        for (int i = 0; i<100; i ++)
        {
            ADCSRA = ADCSRA | 0x40; //Iniciamos conversión
            while ((ADCSRA & 0x10) == 0); //Esperamos que acabe
            tensio = ADC;
            capacitat = (1000*(float)tensio)/996;
            capacitat_acum = capacitat_acum + capacitat;
        }
        capacitat_real = capacitat_acum/100;
    }
}
```

Este método lo hemos probado también en el laboratorio para calibrarlo en base a la experiencia real. Asimismo, pero, nos hemos encontrado muchas dificultades a la hora de medir las capacidades más pequeñas. A continuación, explicaremos por qué.

Hemos hecho las pruebas con el circuito que mide en un rango hasta 100 pF y hemos empezado visualizando la tensión de salida cuando el condensador de capacidad "desconocida" es de 47 pF. Obtenemos lo que podemos ver en la figura 32.

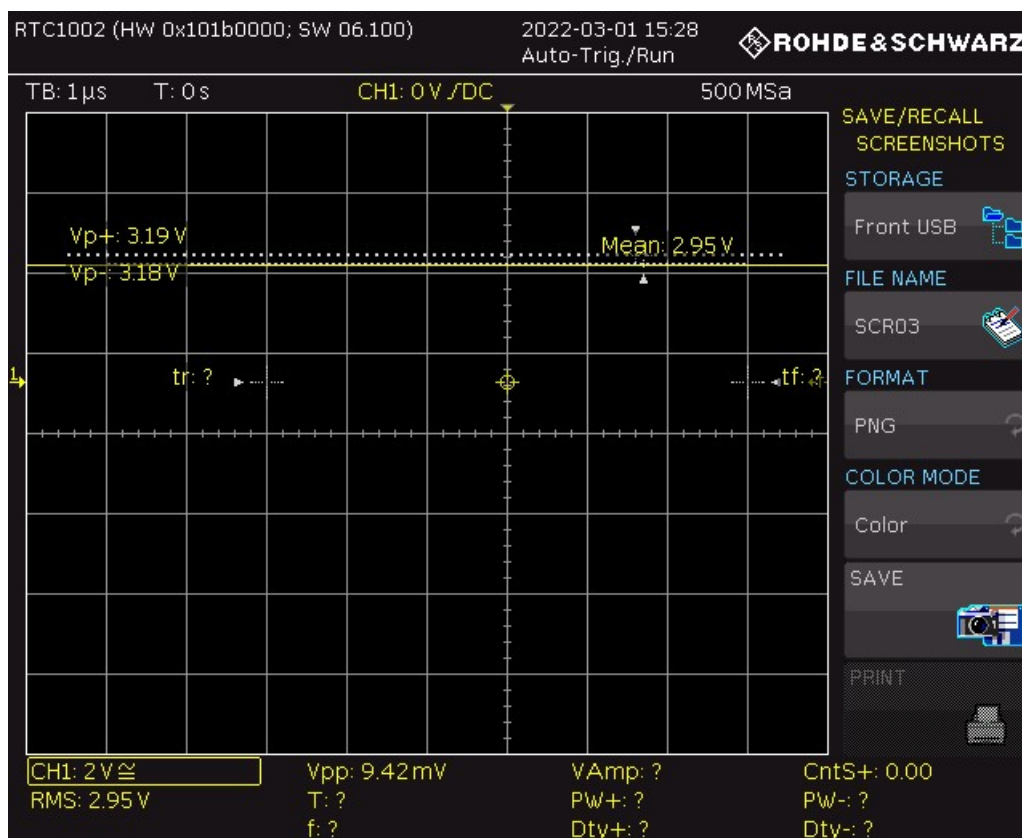


Figura 32. Tensión de salida real con condensador de 47 pF.

Hay que comentar una cosa aquí. El simple hecho de acercar el dedo al osciloscopio para mandarle guardar la captura hace aumentar la tensión de salida. La tensión real de salida para 47 pF fue de 2,7 V.

Tenemos que tener en cuenta, también, que la resistencia R_2 que hemos colocado en la etapa amplificadora en el circuito experimental es de 15,8 k Ω . Lo configuramos así también en el PSIM.

También adjuntaremos a continuación, en la tabla 1, las capacidades nominales con las reales medidas con un LCR, medidas que también se ven influidas por los efectos parásitos.

CAPACIDAD NOMINAL	CAPACIDAD REAL
47 pF	45,3 pF
33 pF	32,9 pF
10 pF	10,1 pF
4,7 pF	5,1 pF
2,2 pF	2,7 pF

Tabla 1. Capacidades de prueba medidas con LCR.

Con el valor real del condensador de 47 pF, en el PSIM se obtiene lo que podemos observar en la figura 33.

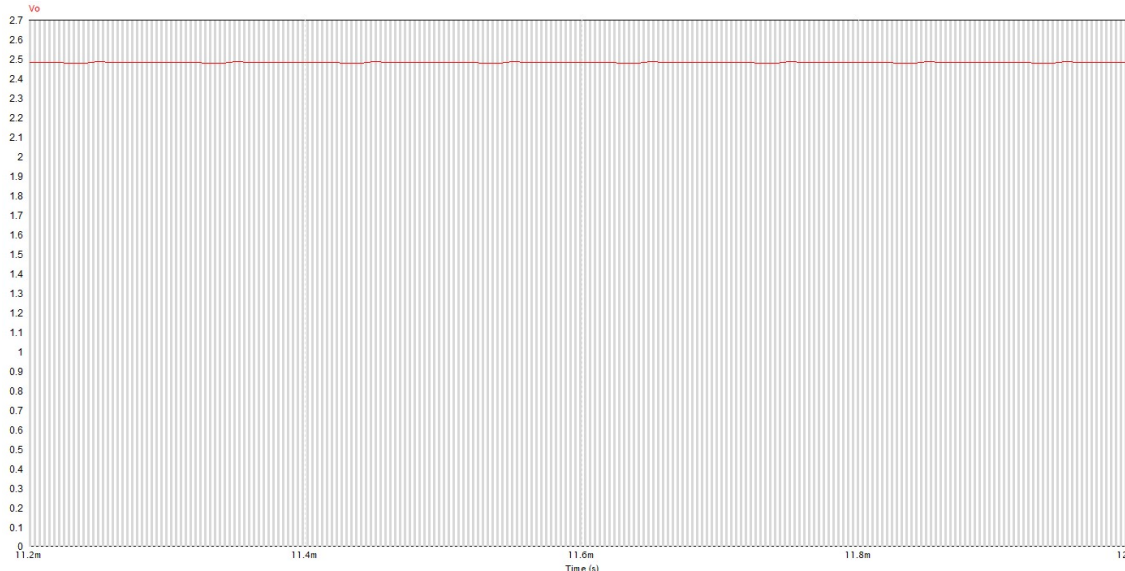


Figura 33. Tensión de salida PSIM con condensador de 47 pF.

Podemos ver como la tensión no es la misma exactamente, pero es normal porque el circuito de PSIM no tiene en consideración muchos aspectos que aparecen en la realidad, sobre todo en circuitos más complejos.

A continuación, adjuntaremos, en la tabla 2, la tensión de salida que hemos obtenido experimentalmente al medir los condensadores de prueba, y la tensión que nos aparece en el PSIM.

CAPACIDAD NOMINAL	CAPACIDAD REAL	TENSIÓN REAL	TENSIÓN PSIM
47 pF	45,3 pF	2,7 V	2,23 V
33 pF	32,9 pF	1,97 V	1,62 V
10 pF	10,1 pF	1,15 V	0,5 V
4,7 pF	5,1 pF	1,07 V	0,25 V
2,2 pF	2,7 pF	0,9 V	0,11 V

Tabla 2. Comparación tensiones reales - PSIM.

El gráfico 1, que resulta de comparar las capacidades reales con la tensión obtenida en el PSIM es el siguiente:

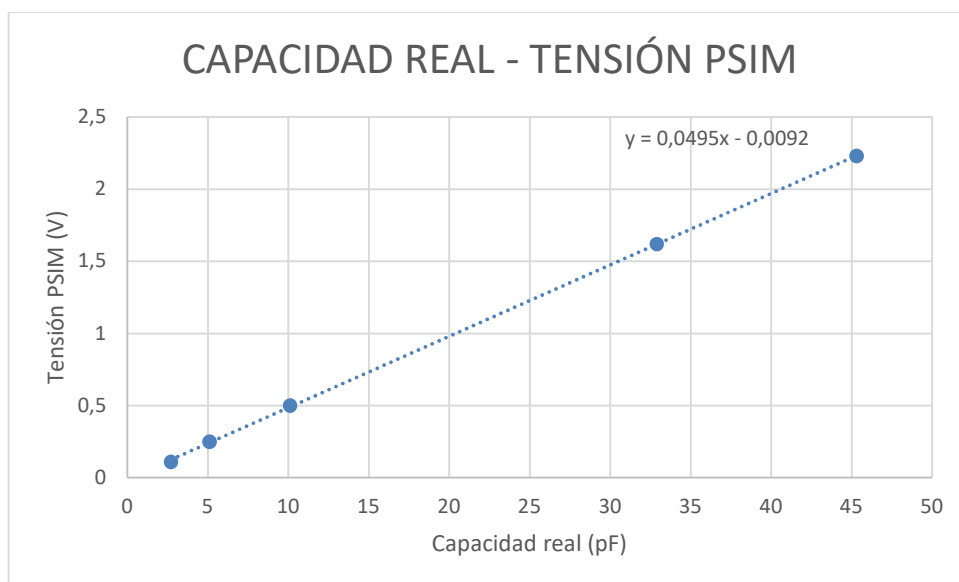


Gráfico 1. Capacidad real – Tensión PSIM.

Ya podemos ver como idealmente existe una linealidad casi perfecta entre la capacidad y la tensión de salida del circuito. Eso es positivo, porque permite hacer cálculos de forma directa, y sacar inmediatamente la capacidad incógnita.

En la realidad, pero, como ya veremos, esta línea de tendencia cambia, pero nos sigue permitiendo hacer cálculos de forma bastante fácil. Asimismo, pero, la capacidad obtenida no se acercará tanto al valor real como si lo midiéramos con un LCR.

Además, el propio circuito ya tiene una capacidad parásita implícita y en el caso que no pongamos ningún condensador para medir y dejemos el circuito abierto entre estos dos puntos, seguimos obteniendo una tensión de salida, bastante grande. Esto hace que sea muy difícil medir capacidades pequeñas, más bajas de 15 - 20 pF, porque son más fuertes las capacidades parásitas que la que queremos medir.

En la siguiente página adjuntaremos el gráfico 2, que compara la capacidad real con la tensión de salida real y la captura del osciloscopio cuando no ponemos ninguna capacidad para medir (figura 34).

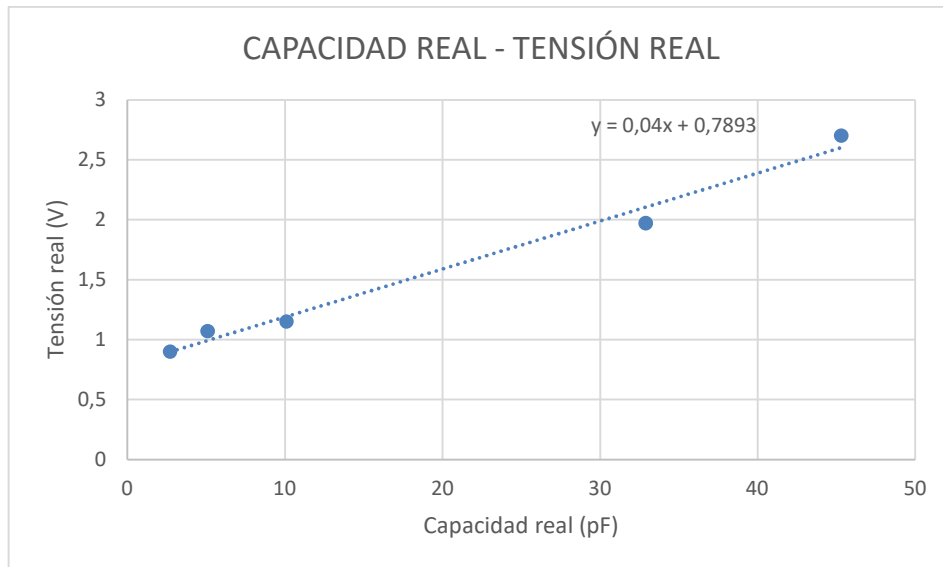


Gráfico 2. Capacidad real – Tensión real.

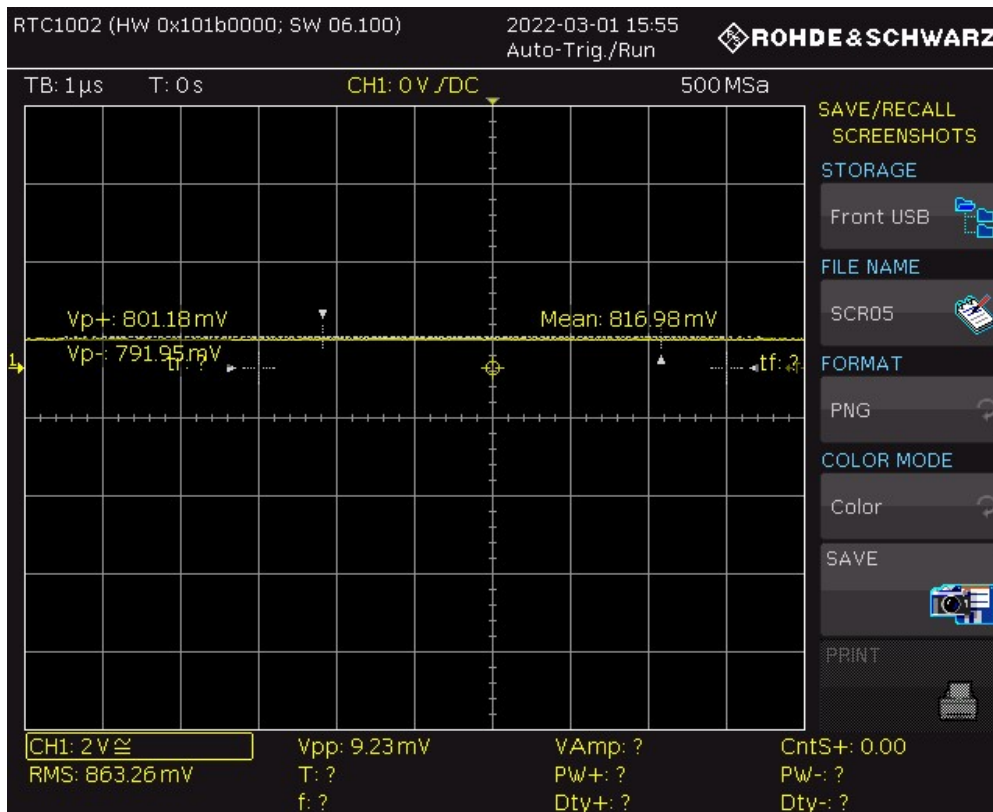


Figura 34. Tensión de salida real sin condensador a medir.

Este valor es oscilante entre 700 mV y 900 mV, aproximadamente.

También hay que tener en cuenta que el LCR, si no ponemos ninguna capacidad a medir, de por sí ya mide 3,2 pF de capacidades parásitas.

4.5. Circuito con puente RC

En la figura 35 podremos ver el esquema del circuito y las conexiones realizadas con la placa del microcontrolador.

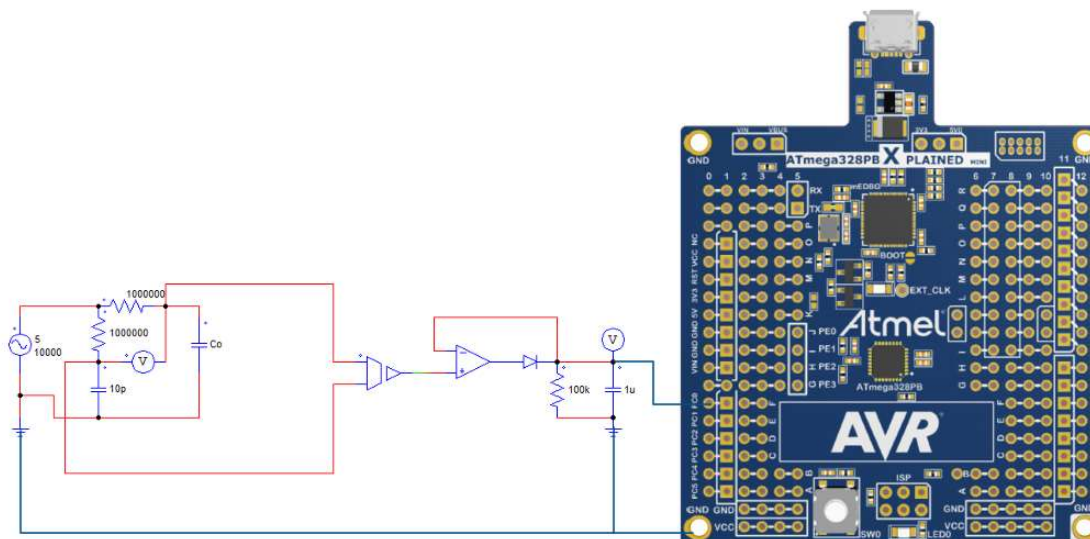


Figura 35. Configuración circuito con puente RC.

Configuramos un "Parameter Sweep" para que el C_0 de este circuito varíe de 1 pF a 10 pF, aumentando el valor de 1 en 1 pF.

La figura 36 muestra qué obtenemos cuando simulamos con el PSIM este método.

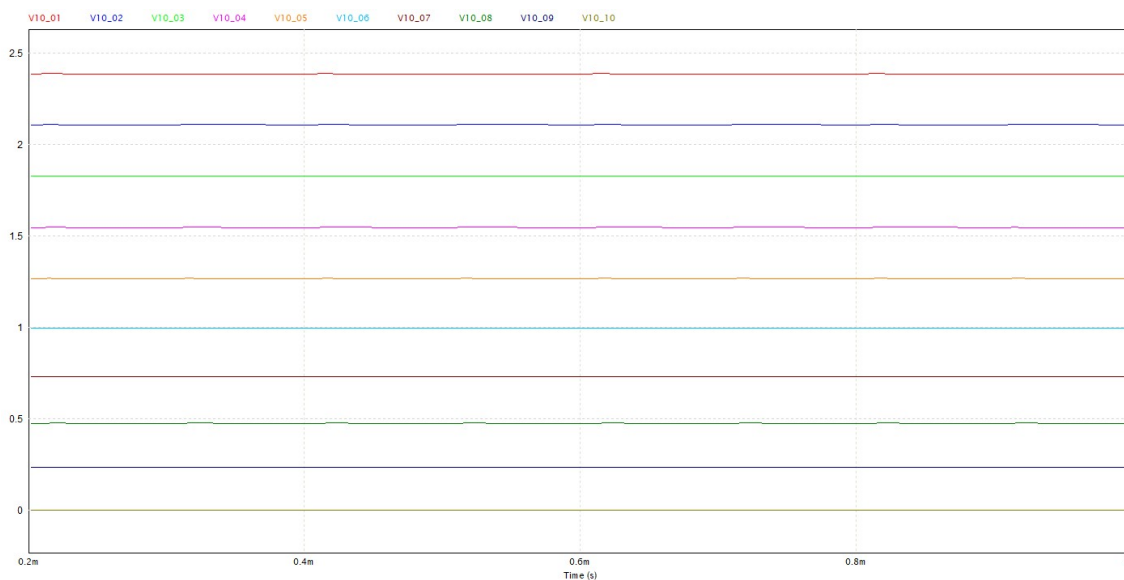


Figura 36. Tensión de salida PSIM con condensadores de 1 a 10 pF.

La tensión de salida con el condensador de 1 pF es V10_01 y con el condensador de 10 pF es V10_10.

A continuación, podemos ver el gráfico 3, que refleja los resultados PSIM anteriores.

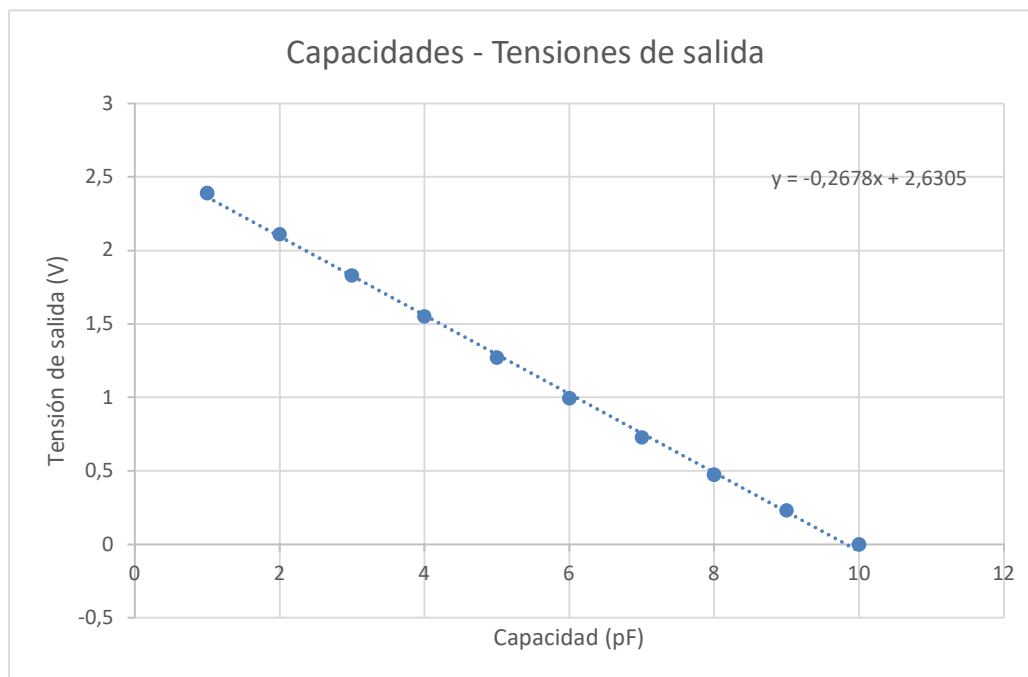


Gráfico 3. Capacidad – Tensión PSIM puente RC.

Puede demostrarse, de forma teórica, la linealidad entre las variaciones de capacidad y la tensión de salida.

Tendremos que comprobar, ahora, su funcionamiento real, donde seguramente tendremos influencia de capacidades parásitas.

Los resultados obtenidos de implementar este método en el laboratorio no son satisfactorios. No hemos conseguido extraer valores de tensión de salida que nos permitan establecer una relación clara capacidad – tensión. Asimismo, pero, sí que hemos observado la sensibilidad del circuito al cambio de capacitor, pero el ruido y las capacidades parásitas lo han hecho no funcional.

4.6. Elección del método

Dada la circunstancia que las capacidades que seguramente vamos a medir serán muy pequeñas, el método más adecuado sería el segundo, que se basa en cargar y descargar con impulsos dos condensadores dispuestos en divisor de tensión.

Las pruebas realizadas de forma experimental confirman que este método es el que ofrece los resultados más precisos.

El condensador del PIN que necesitábamos configurar tiene un valor de 30 pF. Ha sido establecido realizando pruebas con varios condensadores incógnita de valor conocido. También hemos podido experimentar como este valor de condensador de pin no ha variado en las pruebas realizadas.

5. Prototipo de medidor de capacidades

Para disponer de un dispositivo funcional capaz de medir capacidades del orden de pF, y una vez ya hemos escogido el método que nos permite realizar estas medidas con la máxima precisión posible, vamos a programar y construir un prototipo de medidor de capacidades "casero", para poder hacer uso del mismo una vez hayamos impreso capacidades con la impresora.

5.1. Descripción del funcionamiento

La idea general es que el prototipo haga las tres siguientes funciones básicas:

- Medición de capacidades.
- Mostrar por pantalla el valor de la capacidad en cuestión.
- Posibilidad de ajustar un offset o nivel 0 para contrarrestar el efecto de las capacidades parásitas que pueden aparecer en cualquier medición.

En la figura 37, vamos a adjuntar un esquema general con las conexiones que vamos a realizar a la placa del microcontrolador.

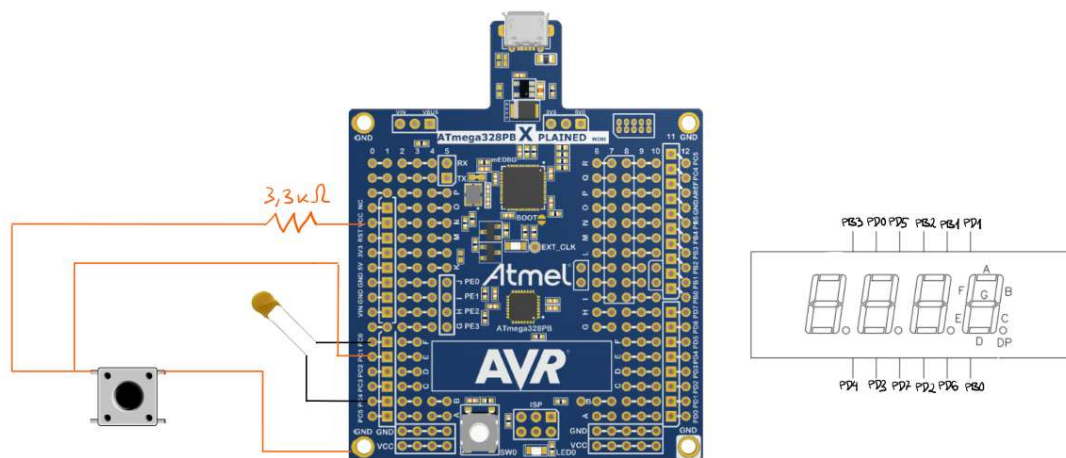


Figura 37. Esquema del conexionado del prototipo.

Por lo que al proceso de medición de la capacidad se refiere, el microcontrolador va a estar tomando medidas en bucle según el algoritmo descrito en el apartado 4.2 de este documento. Vamos a modificar el código para que pueda comunicarse con otras partes del programa, pero esto lo veremos en el siguiente punto.

El pulsador lo vamos a utilizar para avisar al microcontrolador que queremos que nos fije la capacidad actual como referencia cero. Por ejemplo, si no estamos midiendo ninguna capacidad, pero aun así aparece en la pantalla 0,2 pF de capacidades parásitas, podemos apretar el pulsador para que fije esos 0,2 pF como nivel 0. Vamos a configurar una interrupción por cambio de señal en el pin PC1 para que el microcontrolador sepa que hemos pulsado el botón. Es muy importante que pulsemos el botón cuando no tengamos ninguna capacidad que queramos medir conectada, porque si no fijaría el valor de esa capacidad como nivel 0. La gestión que hacemos en el código es hacer una medición de capacidad normal, pero guardando el resultado en una variable que llamaremos `cap_parasita`. Entonces, cuando se calcule la capacidad real se restará dicha variable.

El funcionamiento del display estará gestionado por un timer, configurado a conciencia para que la frecuencia de multiplexado sea la idónea para poder ver los números de forma correcta. Esto se ha hecho experimentalmente.

Lo que hace el timer es, cada vez que entra en la interrupción, selecciona una posición a iluminar, y se manda, mediante una función, los segmentos que se tienen que activar para mostrar el número deseado en esa posición. La posición va aumentando para que se haga un ciclo de la primera a la cuarta, en que reiniciamos la variable que nos va a permitir seleccionar la posición y vuelve a empezar.

Desde otro timer se calculará el valor medio de 100 medidas para que por pantalla veamos un dato que, aunque sea entre valores muy próximos, oscile lo menos posible. Cuando se ha llegado a 100 medidas, se va a descomponer el valor obtenido de la capacidad en cuatro caracteres (decenas, unidades, décimos y centésimos) para que puedan ser mandados a mostrar por pantalla.

En el siguiente apartado vamos a adjuntar el código comentado para entender mejor el funcionamiento del programa.

5.2. Programa

A continuación, podemos comprobar cuál es el código que hemos empleado.

```

#include <avr/io.h>
#include <avr/interrupt.h>

//DECLARACIÓN DE CONSTANTES
//N0...N9 corresponde a la codificación en hexadecimal de los segmentos que hay que
iluminar del número 0 al 9. Por ejemplo, en el caso del 0, se tienen que iluminar todos
los segmentos menos el punto y el del medio. Tal y como lo hemos configurado tendremos
que dar señal a PD0, PD1, PD2, PD3, PD4 y PD5. Tendremos que configurar el PORTD con
0x3F.

#define N0 0x3F
#define N1 0x06
#define N2 0x5B
#define N3 0x4F
#define N4 0x66
#define N5 0x6D
#define N6 0x7D
#define N7 0x07
#define N8 0x7F
#define N9 0x67

//D1...D4 corresponde a la codificación de las posiciones de la pantalla, de derecha a
izquierda. Por ejemplo, la tercera posición está conectada a PB2. En el código tendremos
que poner a nivel bajo este PIN y a nivel alto el resto. Podemos configurar PORTB como
el inverso de 0x04. El 0x04 lo definimos como la constante D3

#define D1 1
#define D2 2
#define D3 4
#define D4 8

//DECLARACIÓN DE VARIABLES GLOBALES

volatile int calibracio = 0, residu = 0, n = 0, i = 0;
volatile float capacitat, capacitat_acum, capacitat_final, cap_parasita;
volatile char c1, c2, c3, c4, selec_display = 1, control = 0;

//DECLARACIÓN DE LA FUNCIÓN mostrarNum

void mostrarNum(char numero);

//DECLARACIÓN Y DEFINICIÓN DE LA FUNCIÓN ini_timer1_int
//Programamos el timer1 en modo normal, con un preescaler de 1024 (f = fcpu/1024) y
configurando que salten las interrupciones cada vez que el conteo llega a 50. Si hacemos
cálculos: ftimer1 = 16M/1024/50 = 312,5. ttimer1 = 1/312,5 = 3,2 ms

void ini_timer1_int()
{
    PRR=PRR&0xf7; //Activamos Timer1
    TCCR1A=0x00; //Normal mode
    TCCR1B=0x0D; //Preescaler 1024 i CTC mode
    OCR1A=50;
    TIMSK1=0x02; //Interrupciones Output Compare A
}

//DECLARACIÓN Y DEFINICIÓN DE LA FUNCIÓN ini_timer1_int
//Programamos el timer2 en modo normal, con un preescaler de 1024 (f = fcpu/1024) y
configurando que salten las interrupciones cada vez que el conteo llega a 16. Si hacemos
cálculos: ftimer1 = 16M/1024/16 = 976,56. ttimer1 = 1/312,5 = 1 ms

```

```
void ini_timer2_int()
{
    PRR=PRR&0xbf; //Activamos Timer2
    TCCR2A=0x02; //Normal mode y CTC
    TCCR2B=0x07; //Preescaler 1024
    OCR2A=16;
    TIMSK2=0x02; //Interrupciones Output Compare A
}

//DECLARACIÓN Y DEFINICIÓN DE LA FUNCIÓN ini_PORTC_int
//Programamos la interrupción por cambio de nivel alto a bajo, o viceversa, en el PIN
PC1

void ini_PORTC_int()
{
    PCICR=0x02; //Activamos Pin Change Int notification grupo PCINT8-14
    PCMSK1=0x02; //Activamos Pin Change PCINT9 (PC1)
}

//RUTINA DE SERVICIO PCINT1_vect
//El programa va a entrar en esta rutina de servicio cada vez que haya un cambio de
nivel alto a bajo, o viceversa, en el PIN PC1. Esto sucederá cuando se pulse el botón.
Lo que tiene que hacer el programa cuando se pulsa el botón es levantar un flag que
avise al programa principal que se tiene que hacer una calibración del nivel 0 (variable
calibracio). Además, vamos a añadir una variable control para reducir el efecto de los
rebotes del pulsador, aunque en este caso no tendrían un efecto determinante en el
programa.

ISR(PCINT1_vect)
{
    if (control == 0)
    {
        calibracio = 1;
        control = 1;
    }
}

//RUTINA DE SERVICIO TIMER1_COMPA_vect
//El programa va a entrar en esta rutina de servicio cada 3,2 ms. La función de esta ISR
será multiplexar la posición del display y mandarle la codificación del número a mostrar.
Por ejemplo, si tenemos seleccionada la posición 2 (0x02), cuando entre otra vez a la
ISR la variable selec_display pasará de valer 0x02 a 0x04 (con el << 1, que mueve un bit
a la izquierda).

ISR(TIMER1_COMPA_vect){

    selec_display = selec_display << 1;
    if (selec_display == 16) selec_display = 1; // Cuando el selec_display llega a 16
(0x10), que correspondería a una posición 5, que no tenemos, reiniciamos la variable
para que apunte a la primera posición

    PORTB = ~selec_display; // Si tenemos seleccionada la posición 2, el selec_display
valdrá 0x02. Tenemos que poner a 0 el PB1 (posición 2) y a 1 el resto. Lo logramos
invirtiendo este 0x02.
    switch (selec_display)
    {
        case D1:
            mostrarNum(c4); // Mostrar centésimos (posición 1, la de más a la derecha)
            break;
        case D2:
            mostrarNum(c3); // Mostrar décimos (posición 2, la segunda por la derecha)
            break;
        case D3:
            mostrarNum(c2); // Mostrar unidades (posición 3, la tercera por la derecha)
```

```

        break;
        case D4:
            mostrarNum(c1); // Mostrar decenas (posición 4, la de más a la izquierda)
            break;
    }
}

//RUTINA DE SERVICIO TIMER2_COMPA_vect
//El programa va a entrar en esta rutina de servicio cada 1 ms. Será la encargada de
calcular un valor medio de las medidas de capacidad realizadas. Además, va a descomponer
el valor de capacidad en decenas, unidades, décimos y centésimos.

ISR (TIMER2_COMPA_vect)
{
    capacitat_acum = capacitat + capacitat_acum;
    i++;
    if (i == 99) // Ejemplo con capacidad 12,12
    {
        capacitat_final = capacitat_acum/100;
        capacitat_acum = 0;
        i = 0;
        float b = 10;
        c1 = capacitat_final/b; // c1 = 12,12/10 = 1
        float resta = (float)c1*b; // resta = 1*10 = 10
        c2 = capacitat_final-resta; // c2 = 12,12 - 10 = 2
        b = b*10; // b =100
        float decimal = b*(capacitat_final-(int)capacitat_final); // decimal =
100*(12,12-12) = 12
        b = 10;
        c3 = decimal/b; // c3 = 12/10 = 1
        resta = (float)c3*b; // resta = 1*10 = 10
        c4 = decimal-resta; // c4 = 12 - 10 = 2
    }
}

//RUTINA PRINCIPAL DEL PROGRAMA

int main(void)
{
    PRR=PRR&0xfe; // Activamos ADC
    DDRC = 0x10; // PC4 como salida, el resto PCx como entrada
    DDRB = 0x0F; // PBx como salida
    DDRD = 0xFF; // PDx como salida
    ADMUX=0x40; // Canal PC0 y Vcc con condensador externo (incluido en la explained
mini)
    ADCSRA=0x82; // Preescaler 4 y ADC habilitado
    ini_PORTC_int();
    ini_timer1_int();
    ini_timer2_int();
    PORTD = 0x00; // Inicializamos a 0 PORTD
    sei();
    while (1)
    {
        if (calibracio == 0) // Si no hay solicitud de calibración
        {
            DDRC = DDRC & 0xfe; // PC0 como entrada
            PORTC = PORTC | 0x10; // Configuramos a 1 PC4
            ADCSRA = ADCSRA | 0x40; // Iniciamos conversión
            while ((ADCSRA & 0x10) == 0); // Espera mientras no termina la
conversión

            int resultat = ADC;
            PORTC = PORTC & 0xef; // Configuramos a 0 PC4
            DDRC = DDRC | 0x01; // PC0 como salida

```

```
        cap_parasita = (float)resultat * 30 / (float)(1023 - resultat)-
cap_parasita;
        if (control == 1) control = 0; // Si se pulsado el botón
anteriormente, bajamos la variable de control de los rebotes del pulsador
    }
    if (calibracio == 1) //Si hay solicitud de calibración
    {
        calibracio = 0; // Ya se ha atendido la solicitud de calibración,
por lo que bajamos la variable que lo indica
        DDRC = DDRC & 0xfe; // PC0 como entrada
        PORTC = PORTC | 0x10; // Configuramos a 1 PC4
        ADCSRA = ADCSRA | 0x40; // Iniciamos conversión
        while ((ADCSRA & 0x10) == 0); // Espera mientras no termina la
conversión

        int resultat_paras = ADC;
        PORTC = PORTC & 0xef; // Configuramos a 0 PC4
        DDRC = DDRC | 0x01; // PC0 como salida
        cap_parasita = (float)resultat_paras * 30 / (float)(1023 -
resultat_paras);
    }
}
```

```
//DEFINICIÓN DE LA FUNCIÓN mostrarNum
```

```
//Le pasamos a la función el número que queremos mostrar en la posición que corresponde,
gestionada por el timer1. Dependiendo de cuál sea el número, le asignaremos una
codificación u otra, para iluminar los segmentos correspondientes a ese número
```

```
void mostrarNum(char numero)
```

```
{
    switch (numero){
        case 0:
            PORTD = N0;
            break;
        case 1:
            PORTD = N1;
            break;
        case 2:
            PORTD = N2;
            break;
        case 3:
            PORTD = N3;
            break;
        case 4:
            PORTD = N4;
            break;
        case 5:
            PORTD = N5;
            break;
        case 6:
            PORTD = N6;
            break;
        case 7:
            PORTD = N7;
            break;
        case 8:
            PORTD = N8;
            break;
        case 9:
            PORTD = N9;
            break;
    }
}
```

```
    if (selec_display == 4) // Si la posición en que se está mostrando el número es
    la tercera comenzando por la izquierda también tendremos que iluminar el punto
    (codificado como 0x80), para separar la parte entera de la parte decimal
    {
        PORTD = PORTD | 0x80;
    }
}
```

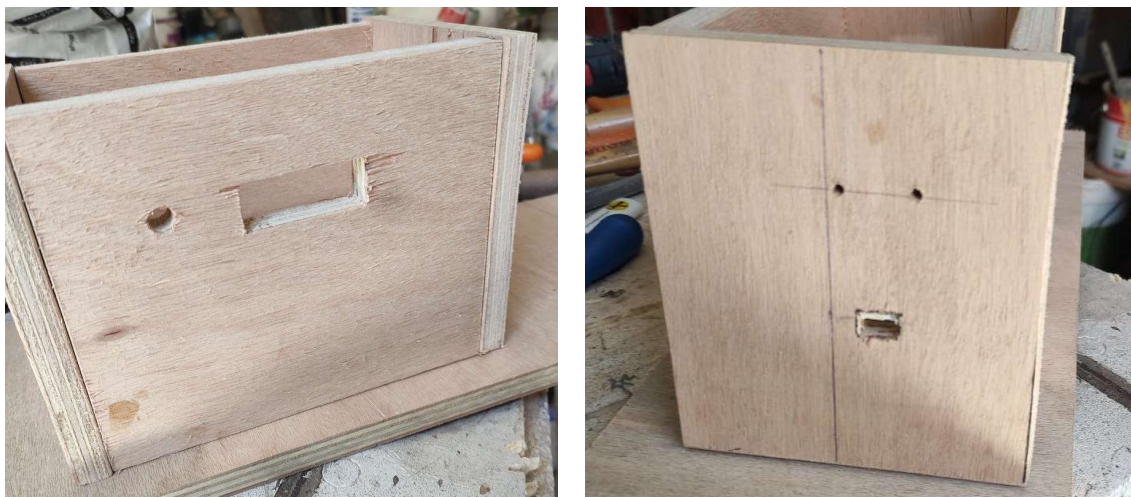
5.3. Construcción de la maqueta

En este subapartado explicaremos de forma breve el procedimiento que hemos seguido para integrar todos los elementos que conforman el prototipo.

La idea es, con madera, hacer una especie de caja de madera, que tendrá los agujeros necesarios (pantalla, botón, conexiones...). Dentro pondremos la placa del microcontrolador y realizaremos todas las conexiones requeridas.

Lo primero que hacemos es disponer las caras de la caja.

En la cara frontal realizamos los orificios necesarios para la pantalla y el botón. En una de las caras laterales pondremos la conexión de la placa a corriente y los orificios donde vamos a conectar las capacidades que queramos medir.



Figuras 38 y 39. Caras frontal y lateral, respectivamente, de la maqueta.

Dentro de la caja, para fijar la placa del microcontrolador, añadiremos una capa de poliespán en la base para que, aprovechando los pines que tiene soldados la placa, podamos clavarla.



Figura 40. Capa de poliespán en la base.

Otra cosa que tenemos que hacer es soldar las conexiones al botón. Con la pantalla podemos unir los pines con los de la placa con cables directamente. Asimismo, con el botón no podemos hacer lo mismo ya que los pines son muy pequeños.

Después de soldarlo, fijamos los cables con un tubo termorretráctil para facilitar, luego, la disposición dentro de la caja.

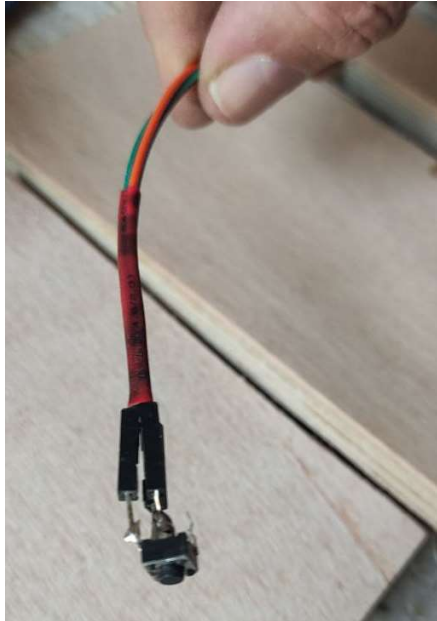


Figura 41. Botón con cables soldados y tubo termorretráctil.

Continuando con el proceso, le damos una primera capa de pintura a las cuatro caras y la base.



Figura 42. Proceso de pintura de las distintas caras.

Finalmente, cuando lo ensamblamos todo, obtenemos el siguiente prototipo.



Figura 43. Prototipo final de medidor de capacidades.

En el lateral tenemos la conexión a corriente de la placa del microcontrolador y los dos orificios, en los que podemos introducir los dos pines del condensador para poder medirlo.

En la cara frontal tenemos el display donde se muestra el valor de capacidad y el botón para poder establecer a 0 la capacidad medida, en el momento deseado.

Para comprobar de forma más precisa el funcionamiento del prototipo, se puede consultar el siguiente link:

<https://drive.google.com/file/d/1p8mCHnjIKwOr9EDtOG8lfEvrnCT-ao4K/view?usp=sharing>

6. Proceso de impresión

En este apartado vamos a ir explicando el proceso que hemos seguido a la hora de imprimir con la Dimatix DMP-2850 para lograr los objetivos finales, así como las operaciones previas al proceso de impresión que deben realizarse.

Hacer uso de este dispositivo es algo que nunca antes habíamos hecho, entonces tuvimos que pasar por un período de familiarización con el mismo y su entorno. Hemos tenido que hacer pruebas muy sencillas, para luego ir aumentando la complejidad de los patrones, y esto es lo que vamos a exponer en este apartado.

6.1. Operaciones previas a la impresión

En este apartado vamos a describir los procedimientos que deben realizarse antes de empezar a imprimir. Vamos a detallarlos para la tinta de plata.

6.1.1. Preparación del Kapton

Kapton es una película de poliimida desarrollada por la empresa DuPont a finales de la década de 1960 que se mantiene estable en un amplio rango de temperaturas, hasta 400 °C. El Kapton se utiliza, entre otras cosas, en electrónica flexible, impresión 3D, en máquinas de rayos X o como parte de mantas térmicas para instrumentos, satélites o naves espaciales.

Tenemos que hacerle un tratamiento al Kapton antes de empezar a imprimir sobre él. Los objetivos de este tratamiento son que quede limpio de polvo y partículas, por muy pequeñas que sean, y que la superficie resulte lo más óptima posible para que las gotas de tinta queden adheridas allí donde nosotros hemos diseñado.

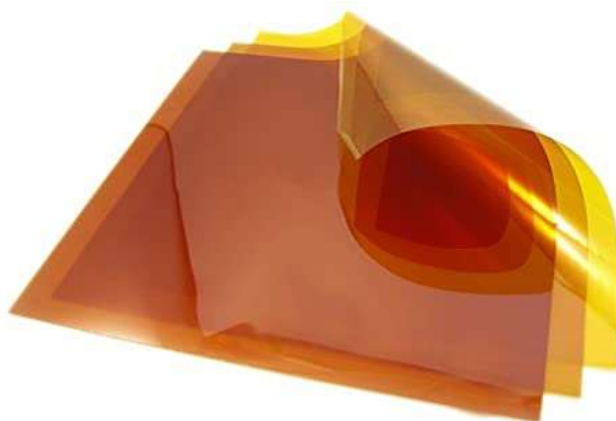


Figura 44. Láminas de Kapton.

Lo primero que hay que hacer es darle un baño de acetona durante 10 minutos y después darle otro baño de etanol, durante otros 10 minutos. Con esto tratamos de dotar al Kapton de más adherencia en la superficie y también intentamos eliminar partículas de polvo y otras materias que puedan afectar negativamente a las impresiones que vayamos a realizar.

Después de esto, introducimos la lámina o láminas de Kapton a la máquina de plasma, que básicamente añade moléculas de oxígeno a dicha lámina para facilitar la adherencia de los materiales de los que esté hecha la tinta.

En las figuras 45 y 46, mostraremos dos imágenes, una de la máquina y otra del Kapton dentro de la máquina, en la que podemos apreciar, ligeramente, como el Kapton tiene una pequeña capa de polvo, cosa que dificulta luego el proceso de impresión.



Figura 45. Máquina de plasma.

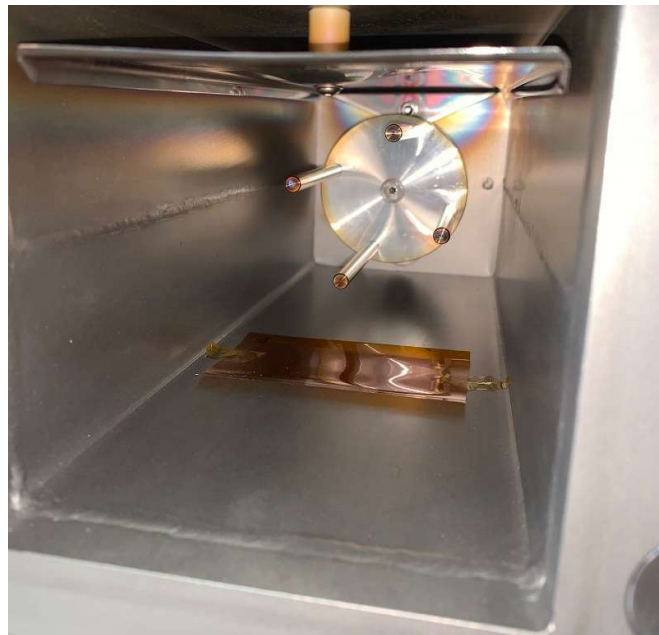


Figura 46. Kapton dispuesto dentro de la máquina de plasma.

Una vez ya hemos seguido estos breves pasos, tenemos que retirar el Kapton de la máquina de plasma, e introducirlo en recipientes herméticos (en este caso placas de Petri).

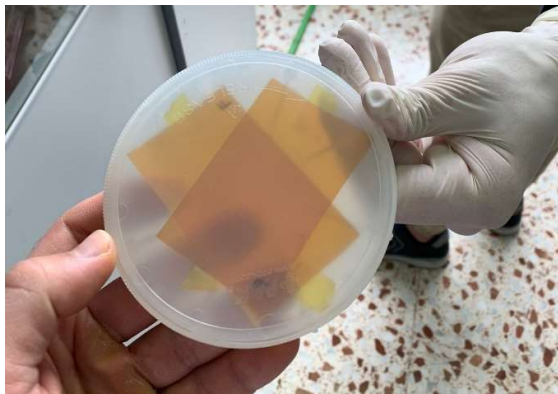


Figura 47. Kapton aislado dentro de placa de Petri.

6.1.2. Preparación de la tinta de plata

El primer paso que se tiene que hacer para asegurar que la tinta está en condiciones óptimas para ser usada en el proceso de impresión es someterla a sonicación mediante una máquina de ultrasonidos.



Figura 48. Bote de tinta sometido a sonicación.

Este proceso debe durar unos 10 – 15 minutos.

Una vez removida la tinta, con una jeringa, la extraemos del bote. Entonces tenemos que poner, en la salida de la jeringa, un filtro, y un inyector. Con esto ya podemos introducir el inyector en el cartucho y empezar a llenarlo. Podemos apreciarlo en la figura 49.



Figura 49. Llenado de un cartucho de tinta.

Por último, recubrimos el cartucho con Parafilm para que, sobre todo, los electrodos queden protegidos.

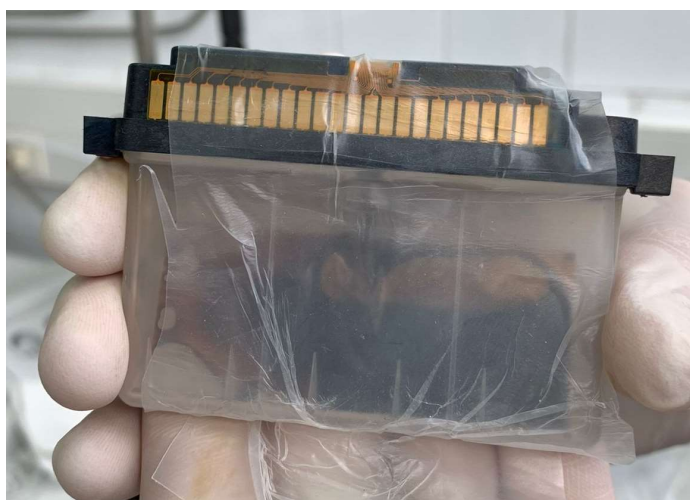


Figura 50. Cartucho recubierto con Parafilm.

6.2. Primeras pruebas

Las primeras semanas estuvimos aprendiendo el funcionamiento de la impresora e imprimiendo los primeros diseños con tinta negra para no empezar a desperdiciar tinta de plata.

En la figura 51 adjuntamos una imagen donde se ve cómo colocamos la tinta en la impresora:

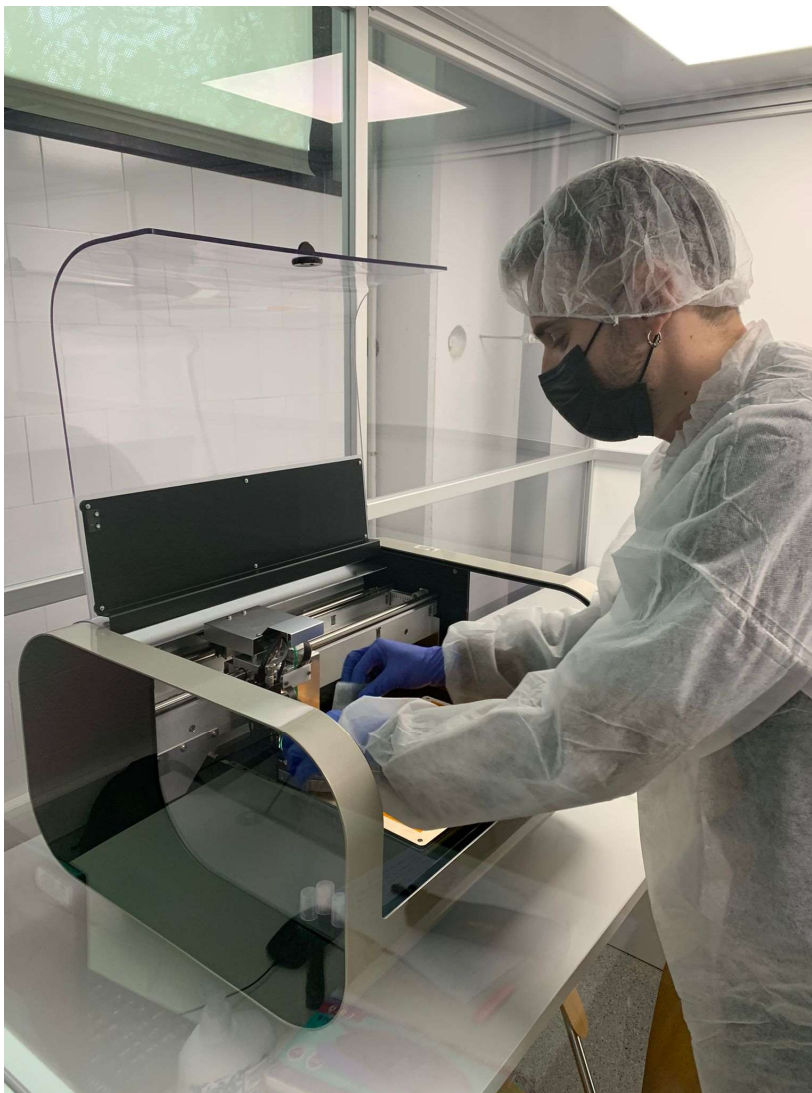


Figura 51. Colocación de tinta a la impresora.

El primer objetivo que nos fijamos fue imprimir dos líneas paralelas. Los diseños los hicimos con el Paint, con la mínima resolución de parrilla que permite.

La idea era mandarle imprimir dos diseños complementarios hechos por píxeles que tuvieran el mismo punto de origen.

Si los juntáramos en el mismo dibujo de Paint quedaría tal y como se muestra en la figura 52.

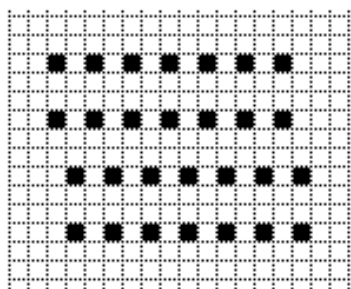


Figura 52. Patrón líneas paralelas.

Primero mandamos imprimir las dos primeras líneas paralelas y luego las segundas con el mismo punto de origen, de forma que, teóricamente, al finalizar las dos impresiones, tendríamos que tener dos líneas paralelas.

En la figura 53 podemos comprobar el resultado.

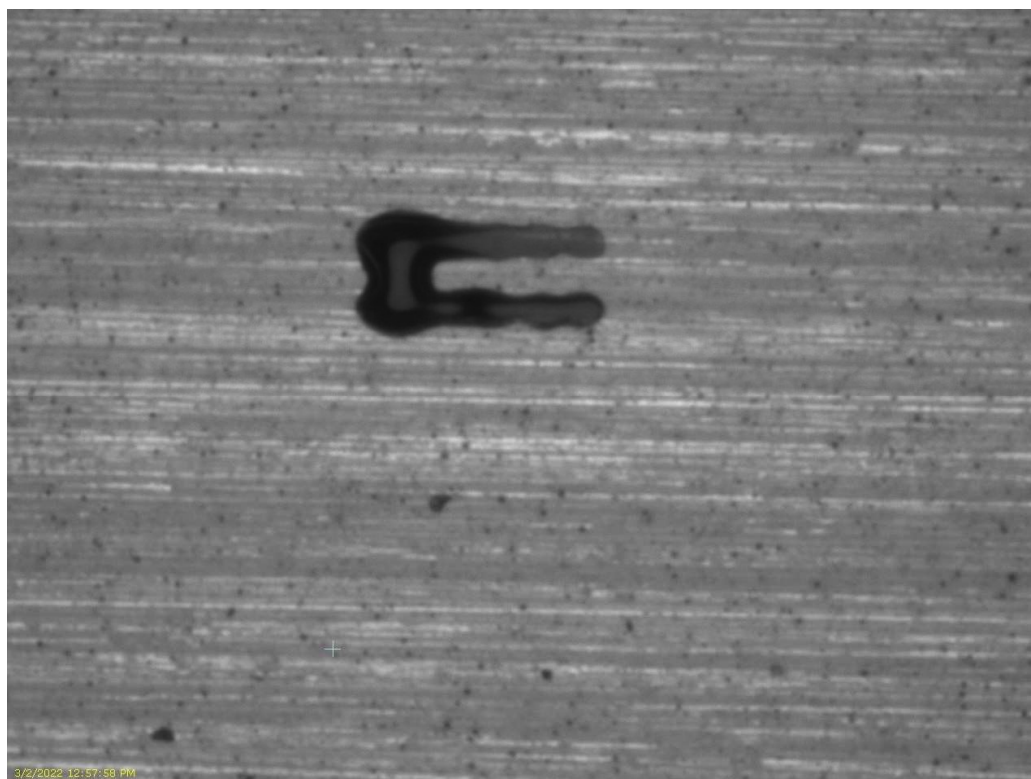


Figura 53. Resultado líneas paralelas (tinta negra).

Detectamos que la impresora junta las primeras tres gotas de cada línea, por esto se ve dos gotas mucho más grandes que las otras. Estas gotas más grandes coinciden con las tres primeras de cada línea.

Entonces lo que hacemos es mejorar el diseño poniendo unas primeras gotas al inicio que se junten, pero que sean ajenas a las dos líneas paralelas.

Lo veremos en la siguiente página, en la figura 54, con los patrones complementarios en la misma captura.

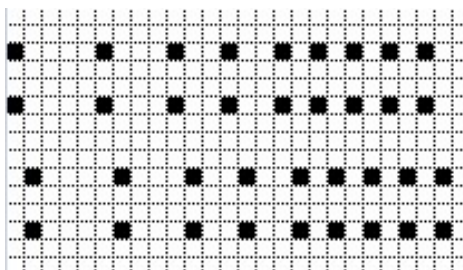


Figura 54. Patrón líneas paralelas mejorado.

Veamos el resultado de la impresión de las dos primeras líneas paralelas, la parte de arriba del patrón, en la figura 55.

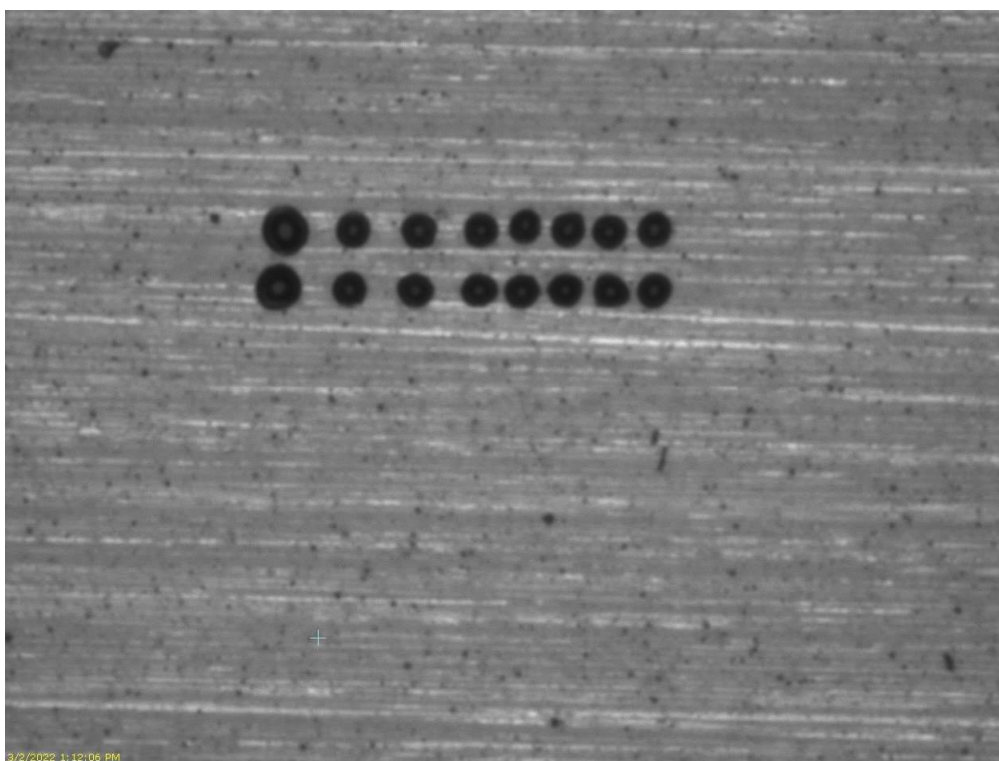


Figura 55. Resultado primera impresión líneas paralelas mejoradas (tinta negra).

Podemos observar cómo los cinco puntos de las líneas que nos interesan están intactos y que los primeros (en este caso 2 porque están más separados) se han vuelto a juntar.

Tenemos que mencionar que hemos configurado un "Drop Spacing", es decir, una distancia entre píxel y píxel (puntos en que la impresora es capaz de tirar una gota) de $30\ \mu\text{m}$. Luego haremos el análisis de la distancia aproximada que nos ha salido entre gota y gota.

Ahora, en la siguiente página, en la figura 56, vamos a adjuntar la captura de cómo queda el diseño finalmente, después de haber añadido el segundo patrón complementario.

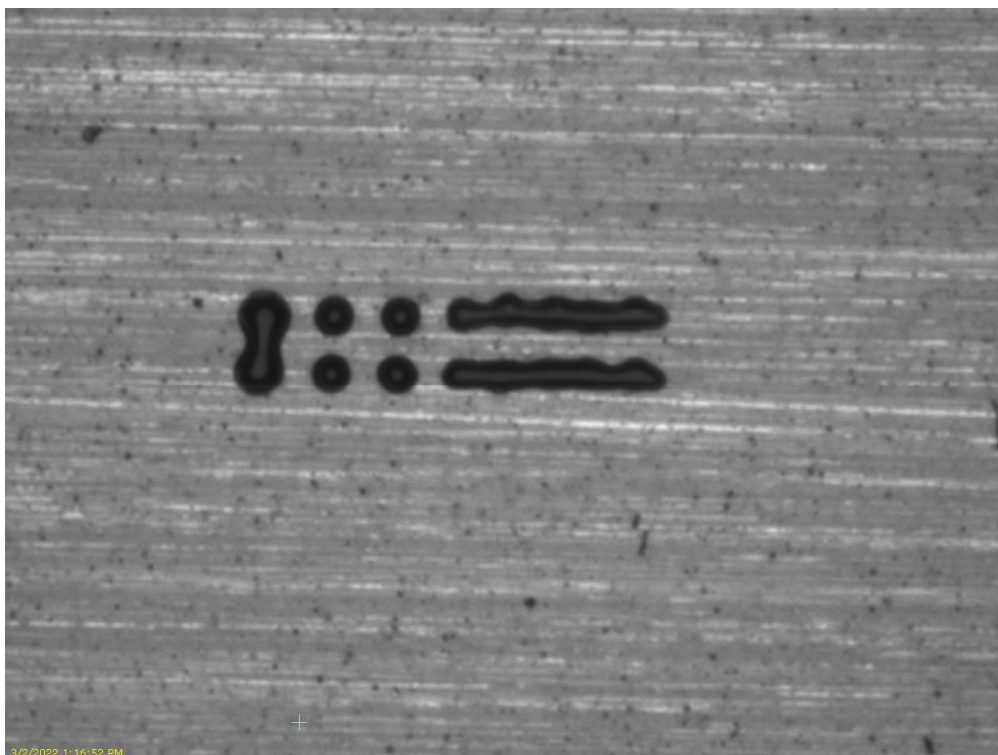


Figura 56. Resultado impresión definitiva líneas paralelas mejoradas (tinta negra).

Podemos ver como hemos conseguido trazar dos líneas paralelas que, en el futuro, con la tinta de plata, pueden ser dos pistas conductoras. Las gotas que aparecen a la izquierda las tendremos que añadir en todas las líneas de futuros diseños. Esto garantizará que lo que queremos que imprima de forma fiel quede bien porque ya habrá juntado las primeras gotas de la línea, que no formarán parte útil de nuestro diseño.

Ahora lo que haremos es calcular la distancia real que hay entre gotas de tinta negra con una separación entre píxeles de $30\ \mu\text{m}$. Con píxeles nos referimos a todos los puntos donde la impresora tiene la capacidad de tirar una gota.

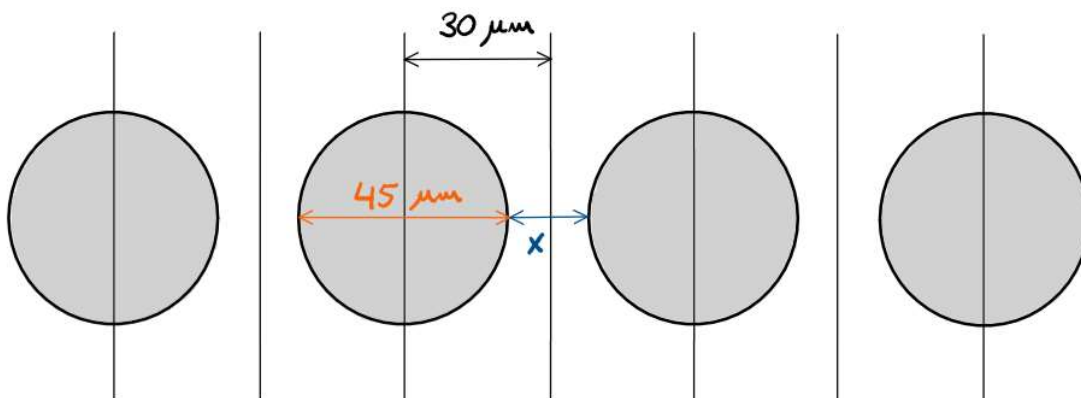


Figura 57. Dibujo de gotas impresas con distancias y medidas.

Los 45 μm son los resultantes de medir con el propio programa.

Para calcular la x tenemos que tener en cuenta que la mitad será la resultante de restar los 30 μm menos la mitad de los 45 μm .

$$\frac{x}{2} = 30 - \frac{45}{2} \quad (9)$$
$$x = 15 \mu\text{m}$$

Una vez hemos conseguido el trazado de dos líneas paralelas lo que intentamos es aumentar la superficie del futuro capacitor.

Para ello, lo que hacemos es disponer dos electrodos verticales. De cada electrodo irán saliendo líneas horizontales y paralelas, sin que lleguen a tocar al otro electrodo (interdigitados). A continuación, en la figura 58, adjuntaremos una figura con el dibujo de lo que pretendemos hacer.

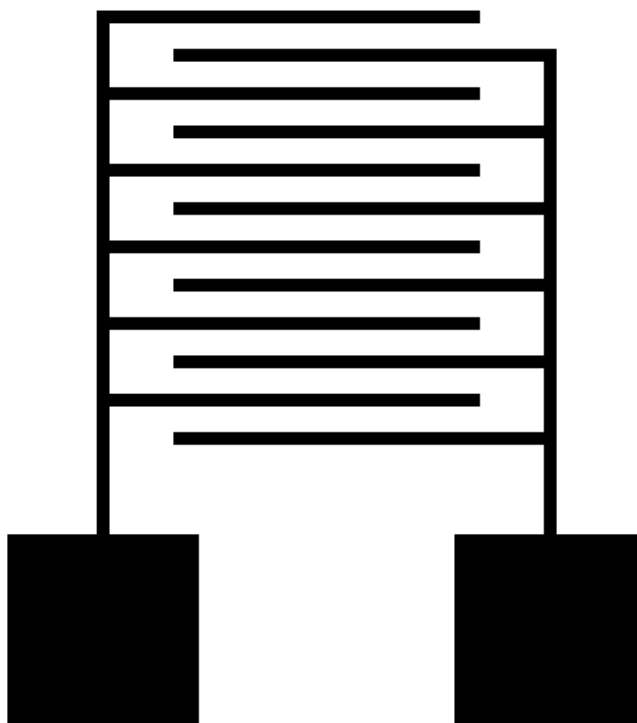


Figura 58. Dibujo de capacidad basada en interdigitados.

En las primeras pruebas que haremos con la impresora no pintaremos los electrodos para no desperdiciar mucha tinta. Así, pues, nos centraremos en lo que sería la capacidad propiamente dicha.

Como ya hicimos en las primeras pruebas de las líneas paralelas, haremos dos patrones en Paint, por puntos, y complementarios entre sí de tal forma que desplazando los puntos en el segundo diseño logremos que una vez impreso el primero y el segundo patrón las gotas se junten y formen las líneas.

En la siguiente página, en las figuras 59 y 60, adjuntaremos estos patrones.

Recordemos que al principio de cada línea hay que poner puntos para que la impresora tire las primeras gotas y el resto del patrón salga bien.

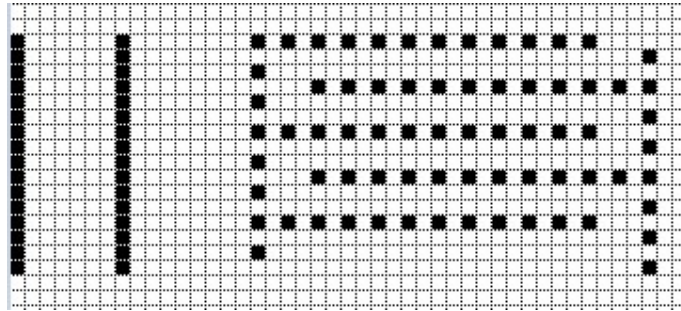


Figura 59. Patrón capacidad basada en interdigitados.

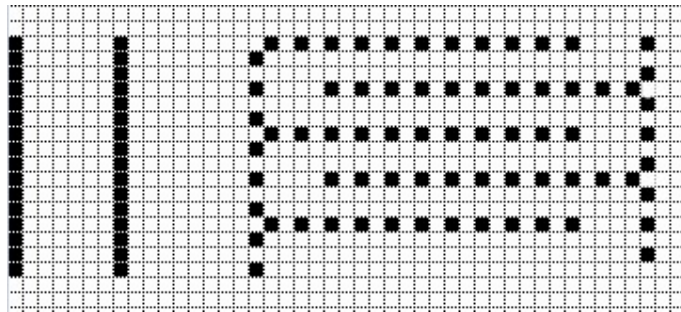


Figura 60. Patrón complementario capacidad basada en interdigitados.

El resultado de imprimir el primero de los patrones es el mostrado en la figura 61.

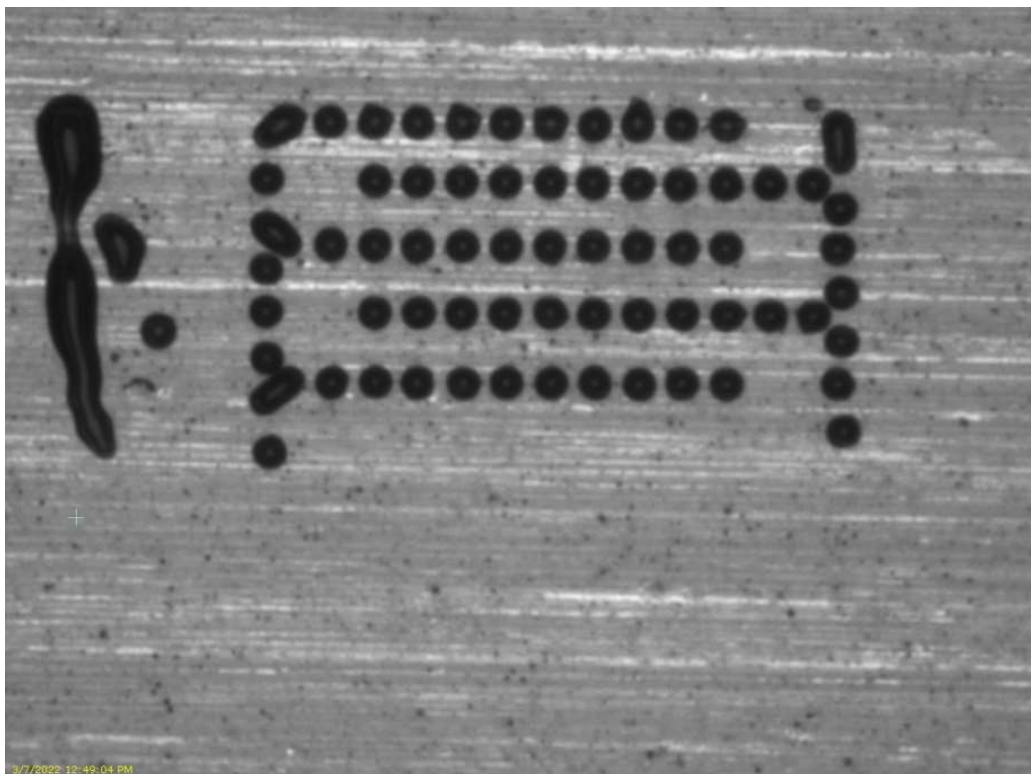


Figura 61. Resultado primera impresión interdigitado (tinta negra).

Después de imprimir el segundo patrón obtenemos lo mostrado en la figura 62.

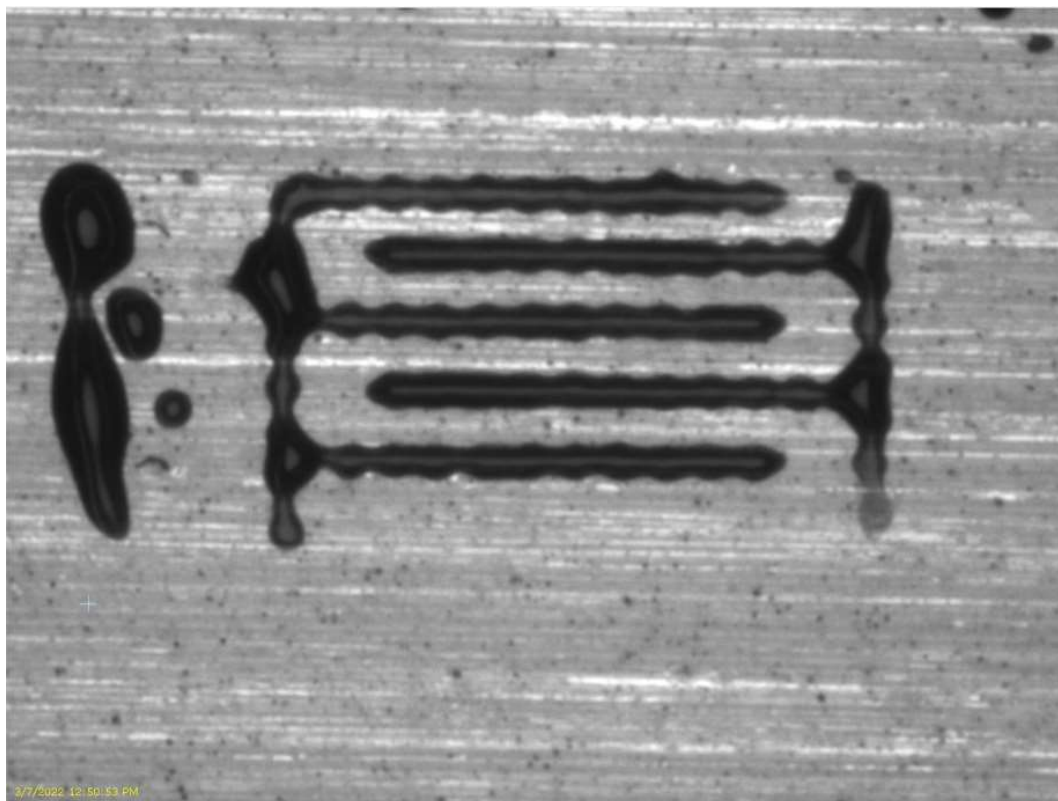


Figura 62. Resultado impresión definitiva interdigitado (tinta negra).

Podemos comprobar como el resultado es óptimo. Que en algunos lugares se formen pequeñas acumulaciones de tinta realmente no es muy importante, y menos cuando estamos hablando de μm . Lo importante es que no se junten las líneas paralelas que salen de los electrodos.

Asimismo, para intentar reducir las ondulaciones y las acumulaciones lo que hacemos es aumentar la temperatura a la que sometemos el sustrato para que así las gotas sequen más rápido y no tengan tanta opción a juntarse, amontonándose.

También hay el riesgo, si la temperatura fuera muy alta, que cuando vayamos a mirar qué ha impreso la Dimatix no veamos nada. Eso puede darse porque la gota se haya evaporado directamente, o porque se haya ido deslizando a otro punto del sustrato y ya no esté donde habíamos mandado imprimirla.

El objetivo, pues, es encontrar una temperatura ideal que nos mejore los resultados obtenidos cuando imprimimos a temperatura ambiente pero que no sea tan alta que nos evapore las gotas directamente o que no se nos vayan a otras zonas del Kapton.

Decidimos probar de poner la temperatura a 50 $^{\circ}\text{C}$ y volvemos a imprimir los patrones anteriores.

En la siguiente página, en la figura 63, podemos ver el resultado.

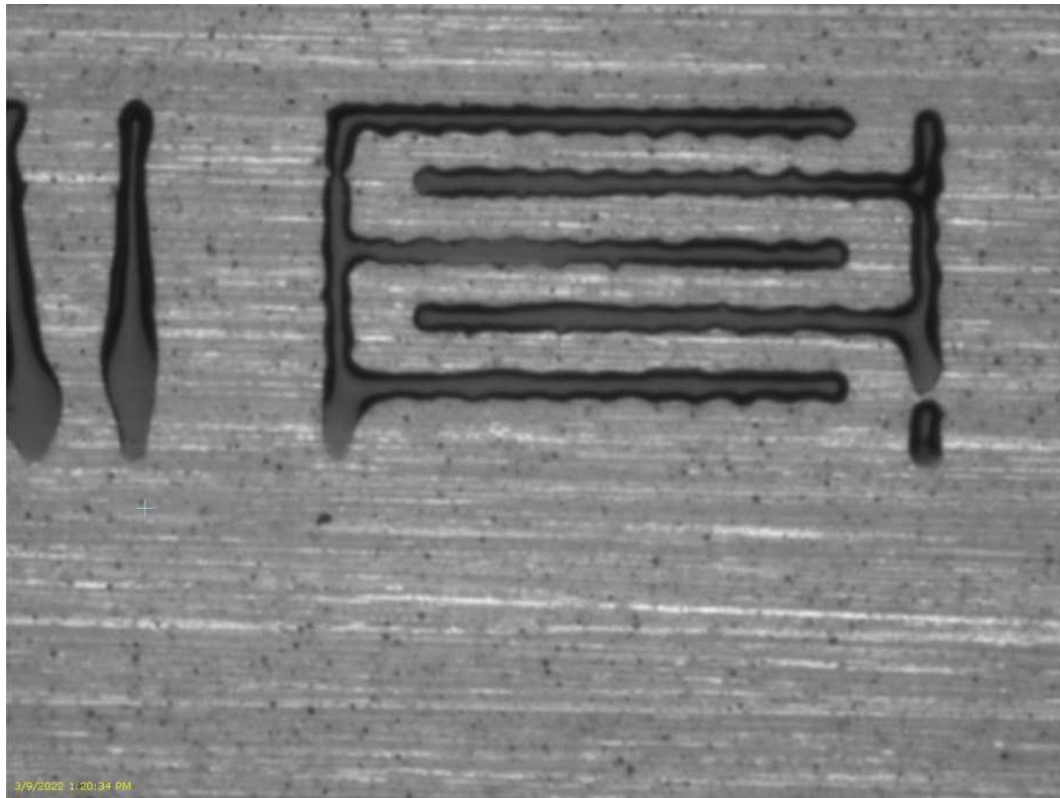


Figura 63. Resultado impresión definitiva interdigitado con más temperatura (tinta negra).

Podemos comprobar como las acumulaciones se han reducido muchísimo y la tinta queda mucho más repartida. También, las líneas quedan más finas.

En el electrodo de la derecha vemos que hay una separación. Esto puede deberse a una impureza en el Kapton. Si no fuera así, podríamos solucionarlo imprimiendo un patrón que solo contenga el píxel que falta por rellenar.

6.3. Pruebas con tinta de plata

Una vez hemos alcanzado un cierto dominio sobre el funcionamiento de la impresora, sustituimos la tinta negra por tinta de plata.

Empezamos imprimiendo el patrón de las líneas paralelas para determinar qué "Drop Spacing" tenemos que configurar, ya que la gota de plata no tendrá las mismas dimensiones que la de tinta negra.

Después de algunas pruebas, llegamos a la conclusión que con una separación entre gotas de 46 μm podemos obtener buenos resultados.

Mandamos a imprimir el siguiente patrón, que se muestra en la figura 64.



Figura 64. Primer patrón (tinta de plata).

Una vez impresa solo una capa obtenemos lo que se muestra en la figura 65.

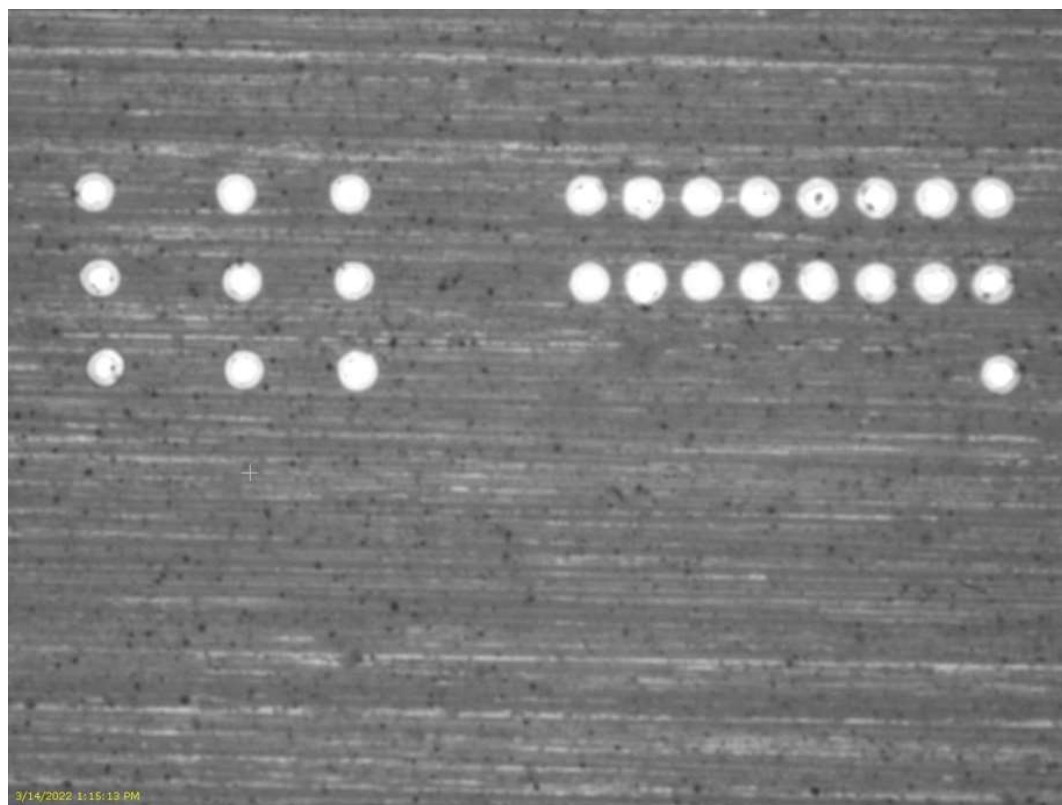


Figura 65. Impresión primer patrón (tinta de plata).

Comprobamos como el resultado es bastante óptimo. Las gotas son muy similares entre ellas, y las distancias son idóneas para poder poner otra capa complementaria y poder formar pistas conductivas.

La configuración que hemos decidido acabar aplicando, después de ir modificando parámetros para comprobar cuando obteníamos las mejores impresiones, la mostramos en la figura 66.

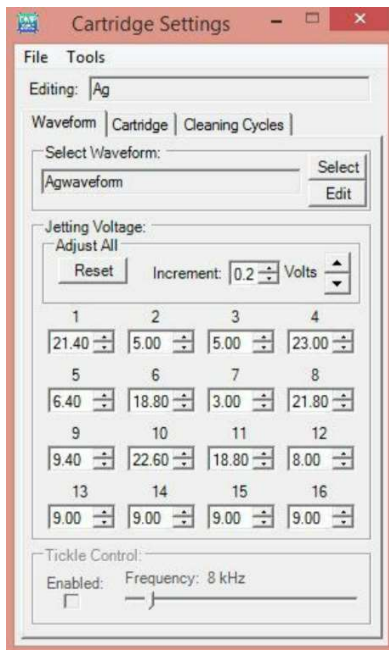


Figura 66. Configuración de tensión de los nozzles (imprimimos con el 8).

Además, la forma de onda que configuramos es la mostrada en la figura 67.

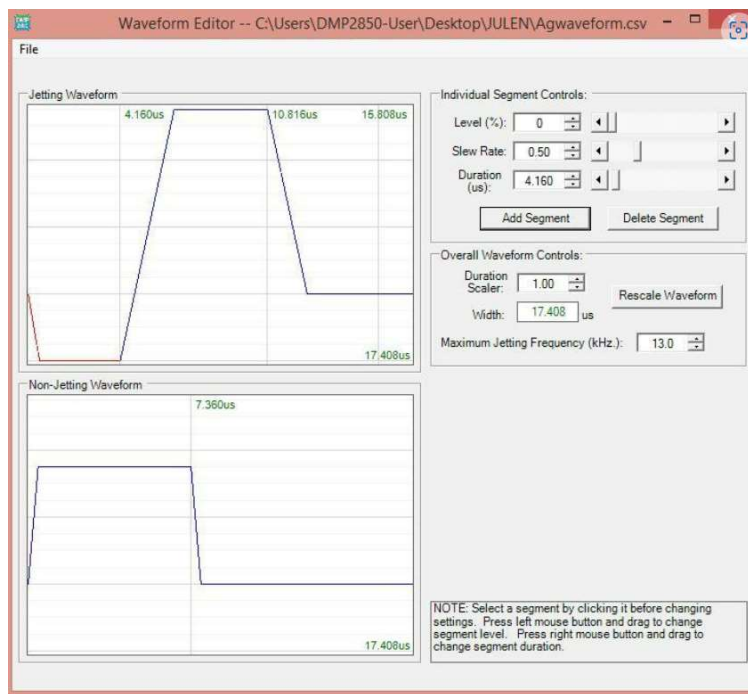


Figura 67. Forma de onda de la impresión con tinta de plata.

Una vez ya tenemos configurados los "Cartridge Settings", procedemos a imprimir el patrón de lo que en un futuro será una capacidad basada en interdigitados, de igual forma que hicimos con la tinta negra.

El resultado, después de las dos capas complementarias, se muestra en la figura 68.

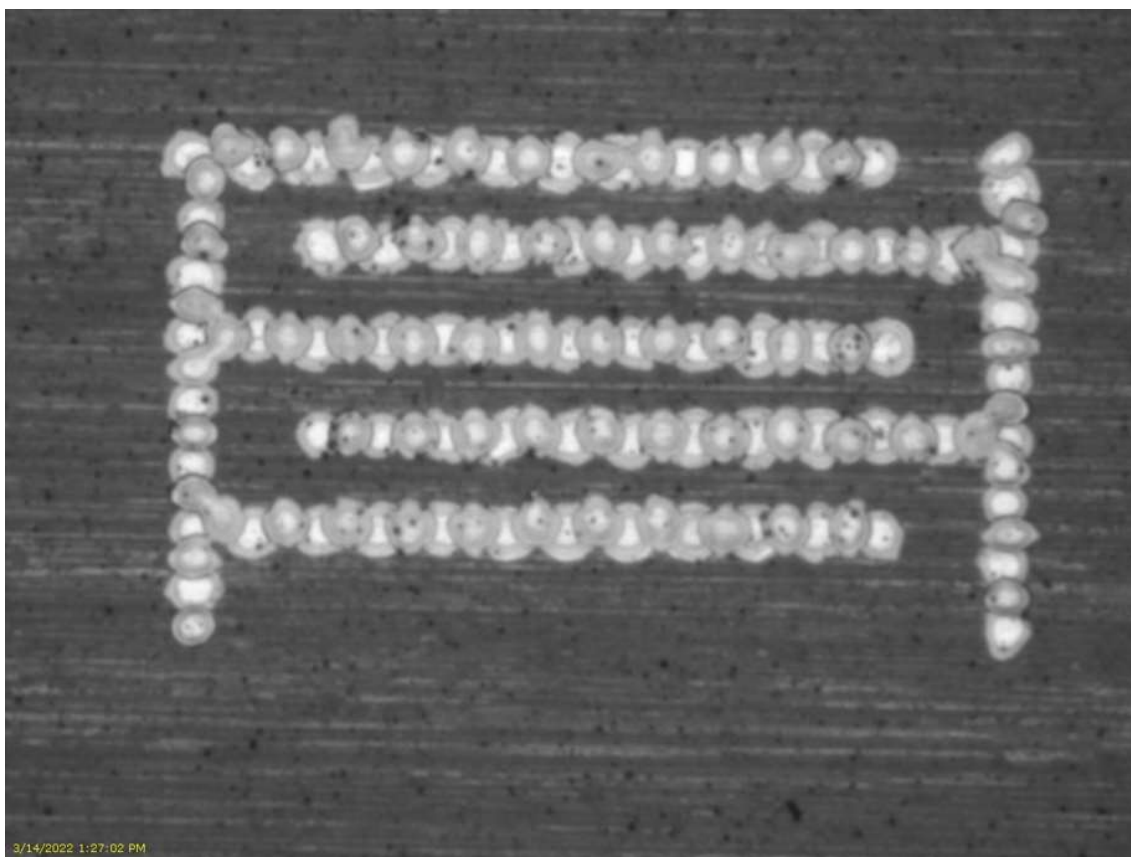


Figura 68. Impresión interdigitado (tinta de plata).

A diferencia de la tinta negra, con la plata vemos que las gotas no tienen mucha tendencia a mezclarse formando acumulaciones de tinta y zonas mucho más vacías. En esta ocasión, las gotas van superponiéndose entre ellas formando lo que más adelante serán pistas conductoras, pero no actúan de la misma forma que la tinta negra, la cual cosa es positiva porque tiene un comportamiento mucho más predecible y controlable.

Viendo que el diseño por patrones complementarios sale bien, lo siguiente que probamos es intentar imprimir un solo patrón continuo, sin píxeles en blanco. El objetivo es que se nos reduzca la distancia entre líneas paralelas, que hace que la capacidad sea más grande como más pequeña es esta distancia.

También reforzamos los electrodos verticales para asegurarnos que no hay ningún circuito abierto.

En la página siguiente, en la figura 69, mostramos el diseño hecho con Paint.

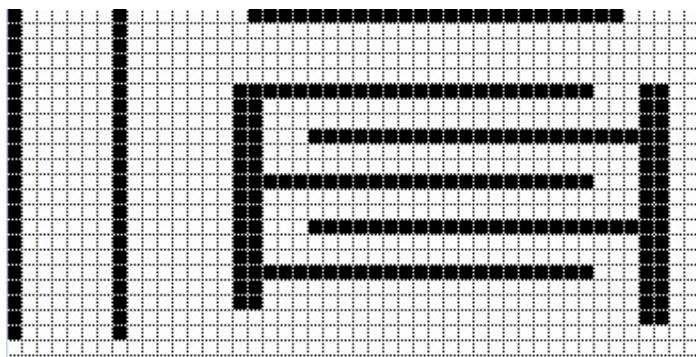


Figura 69. Patrón interdigitado continuo (tinta de plata).

Así, pues, la impresión resultante es la mostrada en la figura 70.

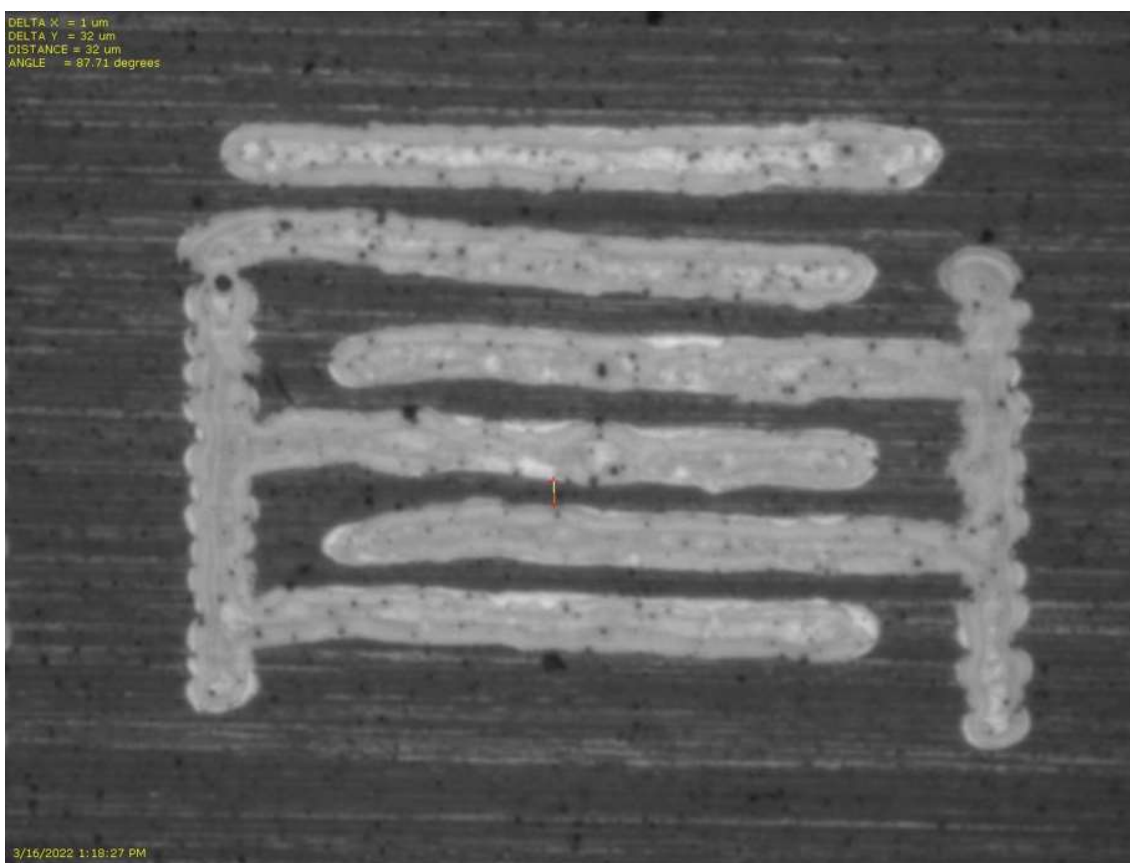


Figura 70. Impresión interdigitado continuo (tinta de plata).

El resultado no es tan visual ni tan paralelo como con los patrones complementarios, pero vemos que la distancia entre conductores se reduce. La que hemos medido en la pantalla hace 32 μm .

Llegados a este punto, en que las primeras pruebas con tinta de plata ya fueron realizadas de forma satisfactoria, se acabó el bote de tinta de plata.

Se tuvo que pedir más tinta de plata, que tenía que llegar en 48 horas. Asimismo, pero, se acabó demorando unas tres semanas, llegando una semana y media antes de tener que entregar la memoria del proyecto.

En el siguiente apartado, adjuntaremos los resultados finales de las impresiones con tinta de plata, es decir, con los patrones de lo que tendrían que ser los futuros sensores capacitivos.

6.4. Impresiones definitivas con tinta de plata

En este apartado lo que pretendemos conseguir es implementar a "gran escala" las pruebas que hemos ido realizando en dimensiones más reducidas.

El primer patrón que intentamos imprimir es el mostrado en la figura 71.

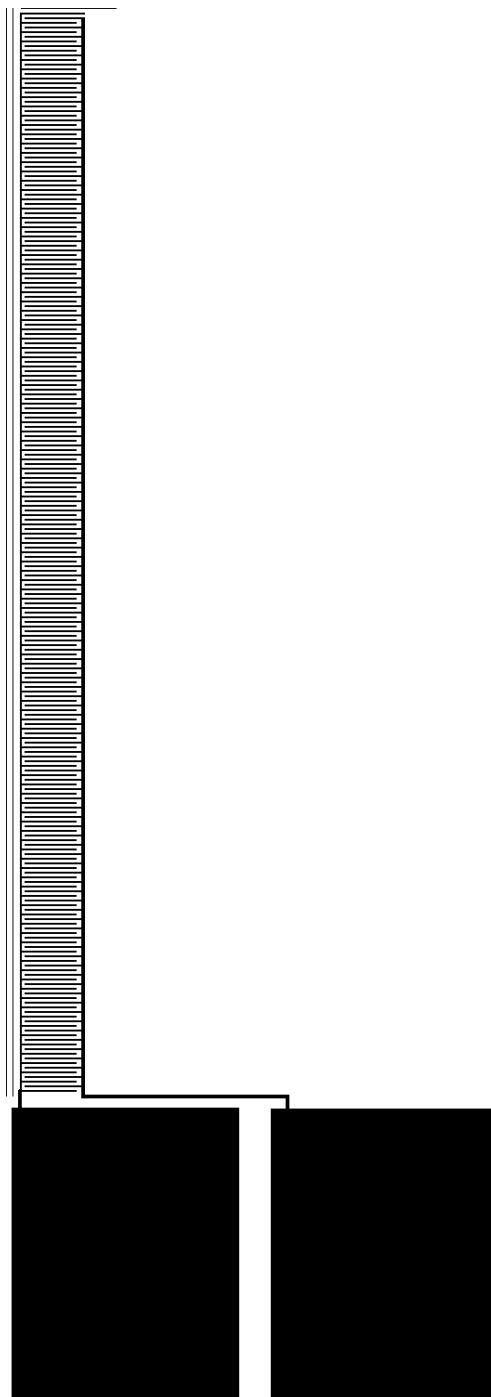


Figura 71. Patrón interdigitado continuo vertical (tinta de plata).

Podemos comprobar como hemos diseñado unos paths suficientemente grandes como para poder llegar a soldar un pequeño cable que haga de enlace con el instrumento de medida.

Realizando varias pruebas experimentales determinamos que para la tinta de plata vamos a imprimir con un drop space (distancia entre centro y centro de una gota) de $46 \mu\text{m}$, como ya hemos comentado anteriormente. Es la separación idónea para que las gotas queden juntas, con un buen grosor, pero que en principio no lleguen a juntarse con las de la línea paralela superior o inferior. Ya veremos más adelante que esta configuración, aunque parezca la idónea, no es condición suficiente para que no se junten líneas paralelas.

Entonces, sabiendo el drop space que usamos, podemos calcular qué dimensiones tendrá la capacidad que estamos intentando imprimir, teniendo en cuenta que un píxel es una gota y que entonces, tendremos que multiplicar la distancia entre centro y centro de la gota por el número de píxeles.

El patrón, sin considerar los paths, tiene un tamaño en horizontal de 70 píxeles, lo que se traduce en 3,22 mm.

Recordemos la fórmula de la capacidad y hagamos unos cálculos orientativos:

$$C = \epsilon_0 \cdot \epsilon_r \cdot \frac{A}{d} \quad (1)$$

Suponiendo que tenemos un grosor que involucra la superficie efectiva enfrentada de $50 \mu\text{m}$, esto supone un área enfrentada por placa (línea paralela) de:

$$A = 50 * 3220$$

$$A = 161 * 10^3 \mu\text{m}^2$$

Si entre dos líneas paralelas hay 3 píxeles de distancia, la variable d tomará un valor de $138 \mu\text{m}$.

La permitividad del vacío, ϵ_0 , ya sabemos cuál es, aproximadamente $8,85 \cdot 10^{-12} \text{ F/m}$. La permitividad relativa del dieléctrico, ϵ_r , es la del aire, de 1,0006.

Con estos datos, podríamos hacer una aproximación rápida de la capacidad que tendremos entre dos líneas, o placas, paralelas. Luego, la vamos a multiplicar por cuántas de estas líneas paralelas tenemos.

$$C = 8,85 \cdot 10^{-12} \text{ F/m} \cdot 1,0006 \cdot \frac{161 * 10^{-9} \text{ m}^2}{138 * 10^{-6} \text{ m}}$$

$$C = 0,01 \text{ pF}$$

Tenemos, aproximadamente, 270 repeticiones de esta configuración de líneas paralelas. Así, pues, podemos aproximar de forma teórica, que si midiésemos esta configuración obtendríamos una capacidad cercana a 2,7 pF (siempre y cuando la suposición que el grosor que involucra la superficie efectiva enfrentada sea aproximadamente $50 \mu\text{m}$, ya que es una medida bastante complicada de obtener).

En la página siguiente, podremos comprobar los resultados obtenidos de imprimir este patrón.

Primero de todo, adjuntaremos una fotografía (figura 72) de lo que podemos ver con el ojo humano, y luego, fotografías más concretas (realizadas con la lente microscópica de la impresora) sobre algunas problemáticas que se han detectado.



Figura 72. Resultado interdigitado continuo vertical (tinta de plata).

Ya se puede detectar, a simple vista, cómo hay algunas zonas que presentan cortes (circuitos abiertos), y otras que presentan uniones entre líneas paralelas (cortocircuito entre los dos electrodos).

A la hora de realizar capacidades, que haya circuitos abiertos en las líneas paralelas, si no son muchos, tampoco es una problemática muy importante. Lo único que vamos a detectar es que la capacidad es menor de lo que debería. Si el corte se produce en los electrodos verticales, y sobre todo si este corte está cerca del path, el problema es mayor, ya que se quedan sin alimentación todas las líneas paralelas que están por encima del corte.

Por lo que a las uniones se refiere, sí que es mucho más crítico para el resultado final. Solo que se toquen dos líneas paralelas por un pequeño contacto, se produce un cortocircuito entre un electrodo y otro, por lo que no vamos a medir ninguna capacidad.

En las figuras 73 y 74 vamos a ver cómo hemos experimentado estas dos problemáticas en la impresión de este patrón, por lo que no obtenemos una capacidad funcional en este intento.

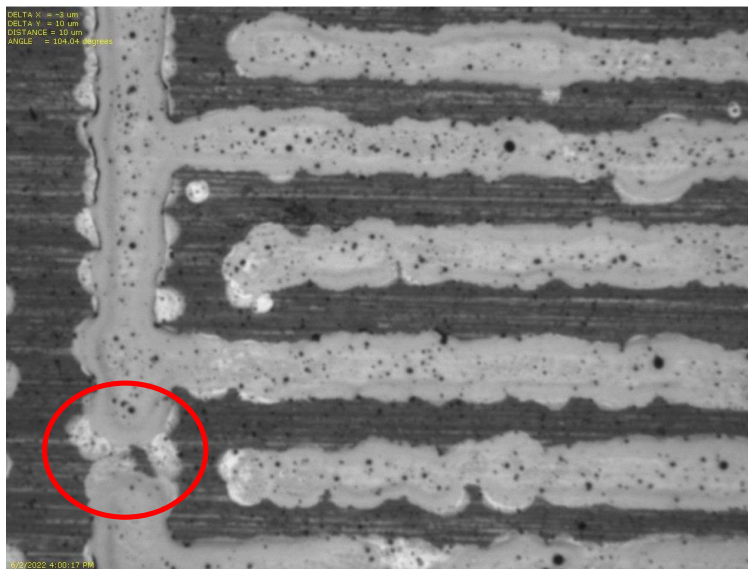


Figura 73. Corte interdigitado continuo vertical (tinta de plata).

En este caso, el corte está muy cerca del path. Podemos comprobar cómo hay pequeñas uniones que hacen que el corte no sea limpio del todo, pero cuando curemos la impresión no sabemos si estas pequeñas uniones van a conducir o no. Había más cortes como este.

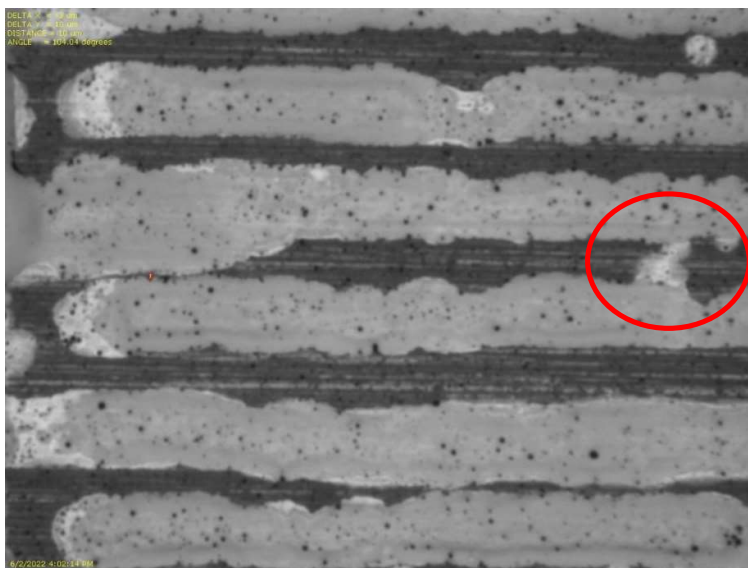


Figura 74. Unión interdigitado continuo vertical (tinta de plata).

Una pequeña gota puede dejar sin el funcionamiento esperado toda la impresión, por lo que ya hemos comentado del efecto que tenían las uniones entre electrodos distintos.

En general, lo que hemos detectado con la impresión de este patrón, es que conforme va avanzando para abajo, cada vez imprime peor y el cartucho se va calentando, cosa que influye de forma notable en el trabajo de la impresora.

Entonces, la idea pasa a ser hacer la capacidad más grande de forma horizontal y hacerla más pequeña de forma vertical.

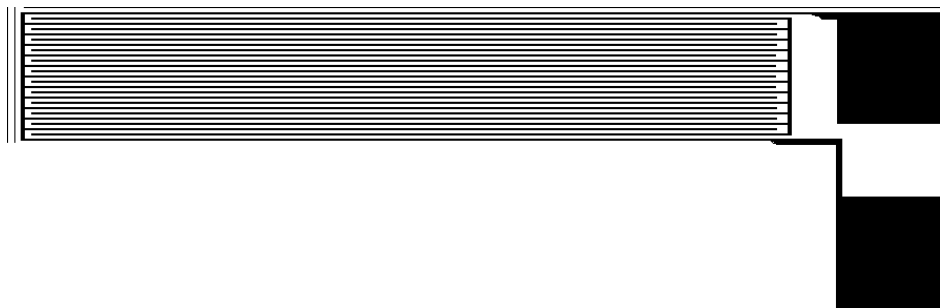


Figura 75. Patrón interdigitado continuo horizontal (tinta de plata).

El patrón, sin considerar los paths, tiene un tamaño en horizontal de 700 píxeles, lo que se traduce en 32,2 mm.

Recordemos la fórmula de la capacidad y hagamos unos cálculos orientativos:

$$C = \epsilon_0 \cdot \epsilon_r \cdot \frac{A}{d} \quad (1)$$

Suponiendo que tenemos un grosor que involucra la superficie efectiva enfrentada de 50 μm , esto supone un área enfrentada por placa (línea paralela) de:

$$A = 50 \cdot 32200$$

$$A = 161 \cdot 10^4 \mu\text{m}^2$$

Si entre dos líneas paralelas hay 3 píxeles de distancia, la variable d tomará un valor de 138 μm .

La permitividad del vacío, ϵ_0 , ya sabemos cuál es, aproximadamente $8,85 \cdot 10^{-12}$ F/m. La permitividad relativa del dieléctrico, ϵ_r , es la del aire, de 1,0006.

Con estos datos, podríamos hacer una aproximación rápida de la capacidad que tendremos entre dos líneas, o placas, paralelas. Luego, la vamos a multiplicar por cuántas de estas líneas paralelas tenemos.

$$C = 8,85 \cdot 10^{-12} \text{ F/m} \cdot 1,0006 \cdot \frac{161 \cdot 10^{-8} \text{ m}^2}{138 \cdot 10^{-6} \text{ m}}$$

$$C = 0,1 \text{ pF}$$

Tenemos, aproximadamente, 25 repeticiones de esta configuración de líneas paralelas. Así, pues, podemos aproximar de forma teórica, que si midiésemos esta configuración obtendríamos una capacidad cercana a 2,5 pF (siempre y cuando la suposición que el grosor que involucra la superficie efectiva enfrentada sea aproximadamente 50 μm).

En líneas generales, al realizar la impresión de este patrón parece que el resultado es más óptimo que el anterior, de hecho, casi no hay cortes ni uniones. Asimismo, recordemos que solo una unión puede inhabilitar el correcto funcionamiento del capacitor.

Detectamos una unión justo en las dos últimas líneas paralelas, como adjuntamos en la figura 76.

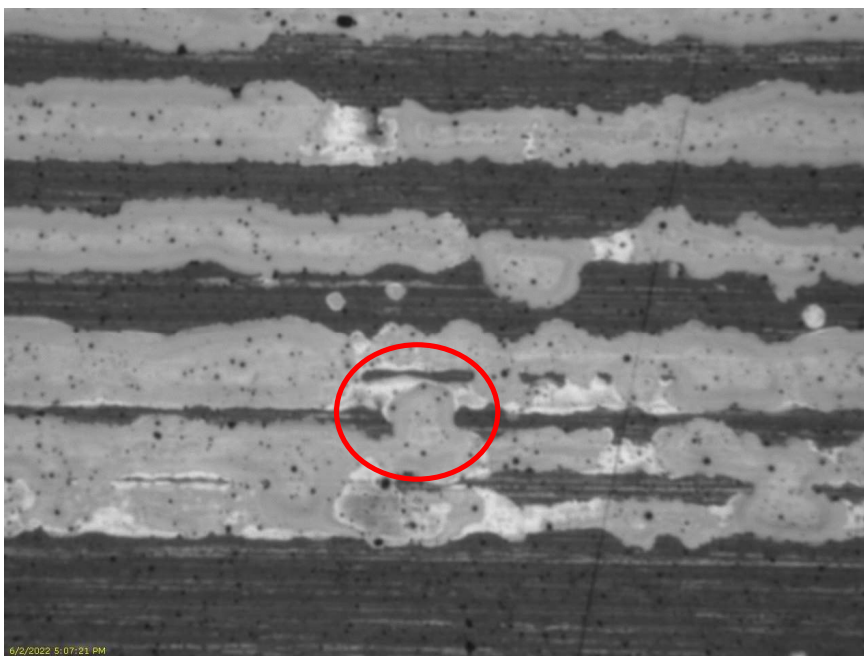


Figura 76. Patrón interdigitado continuo horizontal (tinta de plata).

Para poder comparar como comienza imprimiendo con como acaba, vamos a adjuntar, en la figura 77, una captura del inicio del diseño.

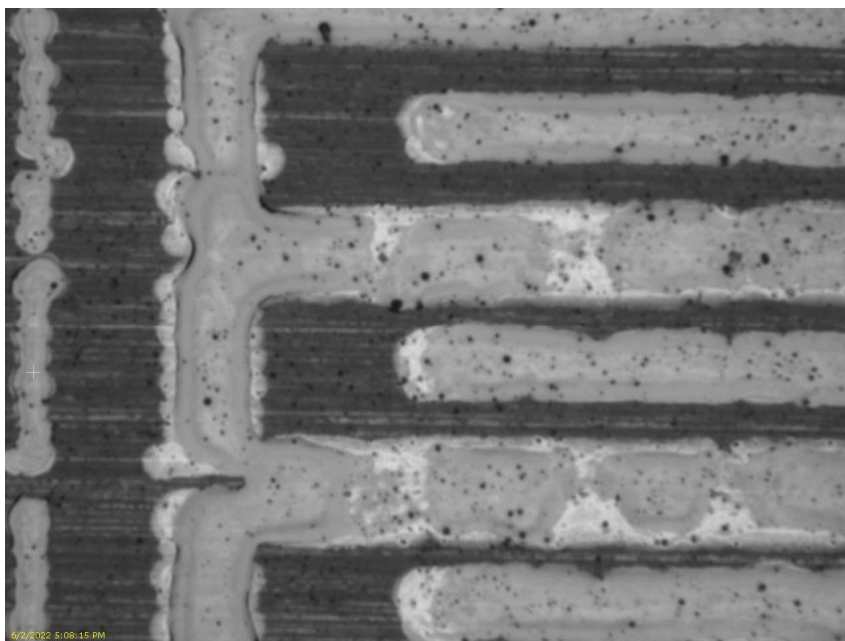


Figura 77. Inicio impresión interdigitado continuo horizontal (tinta de plata).

Que esta opción tampoco haya funcionado, obliga a repensar el diseño otra vez. Optamos por la opción de separar más las líneas paralelas e imprimir dos capas, para añadir grueso, aumentar el área, y poder hacer frente al aumento de d (distancia entre placas, o líneas, encaradas), que, al ser inversamente proporcional a la capacidad, hace que esta disminuya.

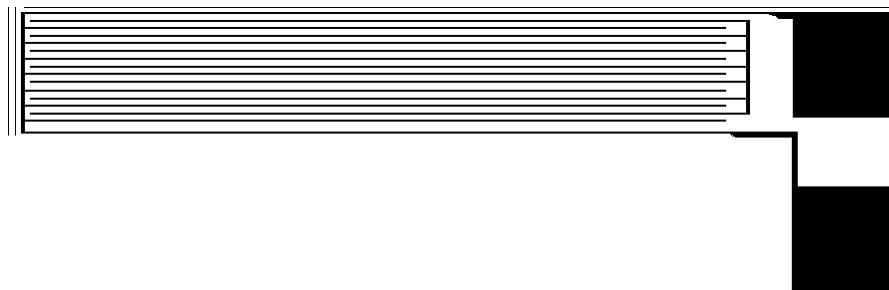


Figura 78. Patrón interdigitado continuo horizontal espaciado (tinta de plata).

La distancia d pasa a ser prácticamente el doble, pero el grosor también. Por este motivo, no prevemos cambios significativos respecto al valor de capacidad teórico calculado en el caso anterior.

A nivel de ojo humano, obtenemos lo mostrado en la figura 79.

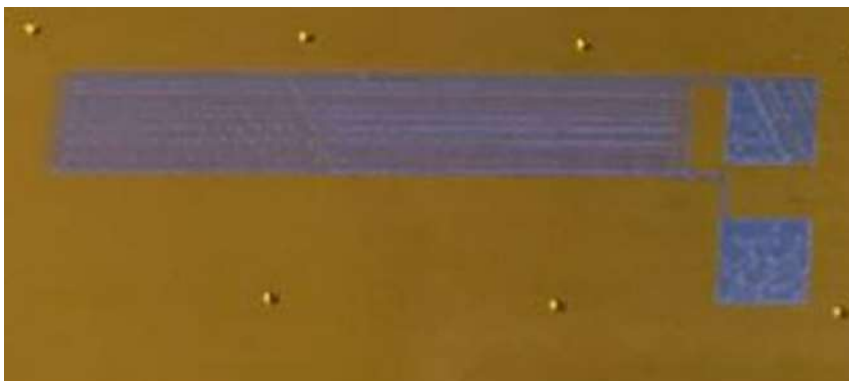


Figura 79. Resultado interdigitado continuo horizontal espaciado (tinta de plata).

Revisando a través de la lupa microscópica de la impresora no detectamos ninguna unión y hay muy pocos cortes.

El resultado es óptimo, a comparación de lo que hemos obtenido con los diseños anteriores.

En la siguiente página, en las figuras 80 y 81, vamos a adjuntar algunas capturas del "Fiducial Camera" para apoyar la hipótesis que el resultado en este caso es correcto.

Cabe remarcar que en todo este proceso de impresión hemos tenido muchos problemas con el Kapton. Los sustratos que hemos preparado nuevos son los que peor adherían, por eso, hemos acabado imprimiendo en Kapton antiguo, con el desgaste que este pudiera tener. Muchos problemas, sobre todo los de cortes, derivaban de imperfecciones, impurezas o suciedad presentes en el sustrato.

Que el resultado de la impresión sea el correcto no es condición suficiente para que se pueda medir capacidad. Hay dos grandes situaciones que pueden afectar a que, aunque la capacidad esté bien diseñada e incluso bien impresa, no podamos llegar a medir ningún valor. La primera es que no todo lo que veamos sea plata, y a la hora de hornearla para hacerla conductiva, nos queden circuitos abiertos. La segunda tiene que ver con el área de las caras encaradas. Nosotros hemos supuesto un grosor que involucra la superficie efectiva enfrentada, pero esta variable no la conocemos realmente y podría afectar de forma muy significativa en

la medida de la capacidad, incluso llegando a incrementar el valor que habíamos previsto de forma teórica.

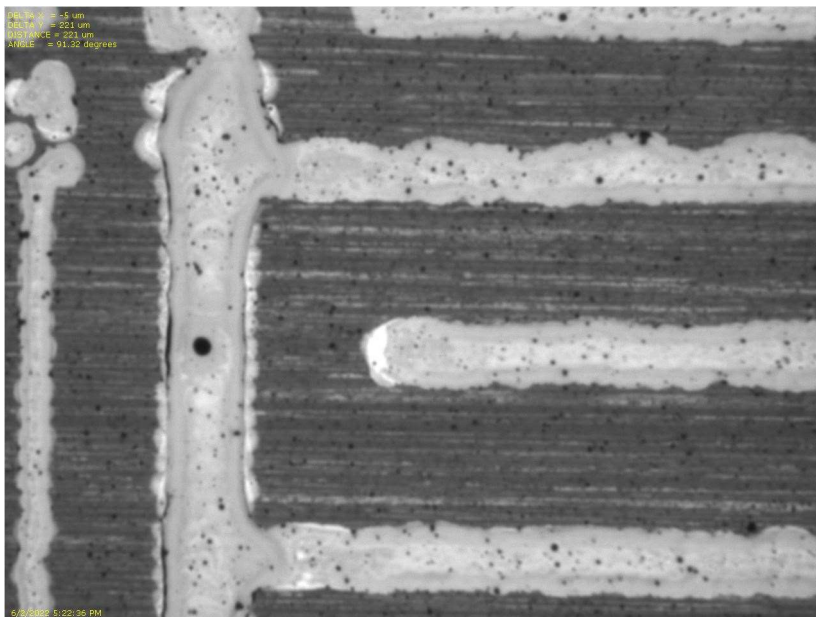


Figura 80. Patrón interdigitado continuo horizontal espaciado (1ª capa - tinta de plata).

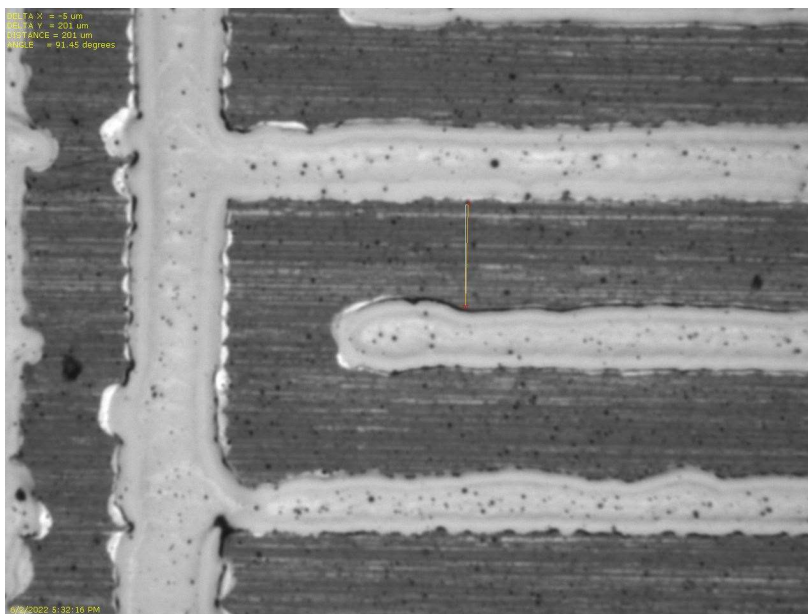


Figura 81. Patrón interdigitado continuo horizontal espaciado (2ª capa - tinta de plata).

Podemos concluir que es mejor imprimir diseños escalados horizontalmente, y no verticalmente. También, si dejamos menos de cuatro píxeles entre línea y línea corremos el riesgo de tener uniones en el tipo de diseños con los que hemos estado trabajando.

7. Mediciones y resultados

Antes de poder medir la capacidad resultante de alguno de los patrones que hemos impreso, tenemos que curarlos (introducir en un horno para que la tinta de plata se haga conductiva) y preparar los paths para que podamos conectar cable que luego vaya conectado al sistema de medición.

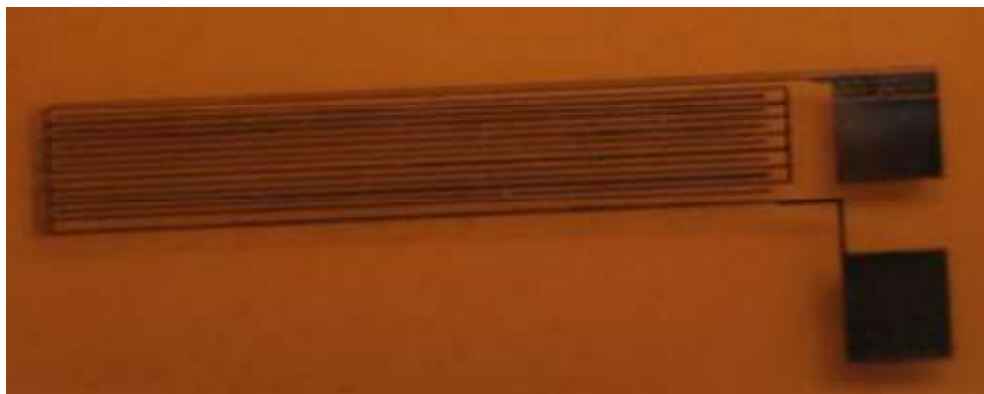
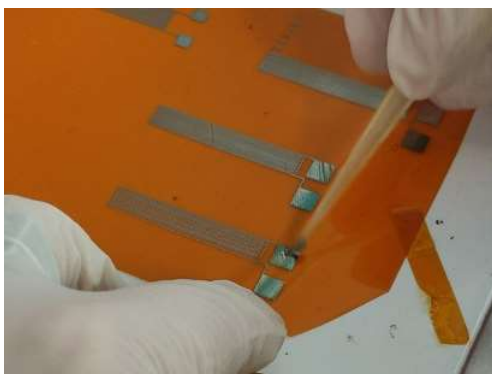


Figura 82. Resultado después del proceso de curado.

En los paths añadimos pasta de plata para disponer encima un hilo conductor.



Figuras 83 y 84. Adición de pasta de plata (izquierda) y colocación de hilo (derecha).

Una vez ya hemos acondicionado los paths, procedemos a realizar unas primeras mediciones en el laboratorio.



Figura 85. Primeras mediciones realizadas en el laboratorio.

En este momento detectamos que, por la influencia de elementos parásitos, la medida que arroja el instrumento medidor de capacidades, realizado por nosotros, oscila bastante, dentro de una horquilla de unos 2 pF.

Para remediar esta problemática lo que hacemos es modificar el fragmento de código que calculaba el promedio de 100 medidas. Vamos a configurarlo para que haga la media cada 1000 medidas, así se tendría que visualizar un valor mucho más estable por pantalla. A continuación, mostramos el fragmento de código, que corresponde a la rutina de servicio a la interrupción TIMER2_COMPA_vect, corregido:

```
//RUTINA DE SERVICIO TIMER2_COMPA_vect
//El programa va a entrar en esta rutina de servicio cada 1 ms. Será la encargada de
calcular un valor medio de las medidas de capacidad realizadas. Además, va a descomponer
el valor de capacidad en decenas, unidades, décimos y centésimos.

ISR (TIMER2_COMPA_vect)
{
    capacitat_acum = capacitat + capacitat_acum;
    i++;
    if (i == 999) // Ejemplo con capacidad 12,12
    {
        capacitat_final = capacitat_acum/1000;
        capacitat_acum = 0;
        i = 0;
        float b = 10;
        c1 = capacitat_final/b; // c1 = 12,12/10 = 1
        float resta = (float)c1*b; // resta = 1*10 = 10
        c2 = capacitat_final-resta; // c2 = 12,12 - 10 = 2
        b = b*10; // b =100
        float decimal = b*(capacitat_final-(int)capacitat_final); // decimal =
100*(12,12-12) = 12
        b = 10;
        c3 = decimal/b; // c3 = 12/10 = 1
        resta = (float)c3*b; // resta = 1*10 = 10
        c4 = decimal-resta; // c4 = 12 - 10 = 2
    }
}
```

El valor de capacidad que conseguimos medir está alrededor de 2 pF. Las capacidades parásitas tienen una influencia muy grande cuando tratamos de medir capacitores del orden de pocos pF, y aunque el ajuste a 0 que tenemos programado funciona muy satisfactoriamente, no nos soluciona la totalidad de la problemática ya que cualquier mínimo movimiento influye.

Que la medida resultante sea de 2 pF satisface casi totalmente la previsión teórica realizada. Recordemos que obtuvimos, después de realizar unos cálculos aproximados, 2,7 pF. En estos cálculos suposimos un grosor que involucra la superficie efectiva enfrentada de 50 μm , y parece que no nos desviamos mucho de la realidad.

Además, la capacidad cambia al acercar el dedo a ella, lo que demostraría su sensibilidad.

En el siguiente link podemos ver un vídeo donde se comprueba este suceso:

https://drive.google.com/file/d/1uJgrt1dNpk6VftnReu_xVYfrF9-7a3ws/view?usp=sharing.

8. Conclusiones y riesgos

Después de la realización de este proyecto podemos extraer las siguientes conclusiones, en las que también vienen implícitos algunos riesgos que asumimos al aceptar el proyecto, o que hemos ido descubriendo a medida que ha avanzado el mismo:

- Sobre medir capacidades del orden de los pF:
 - Es muy complicado encontrar un método que se base en circuitería analógica (amplificadores operacionales, resistencias, rectificadores y otros elementos) que no genere una cantidad de ruido tal que sea posible medir valores de capacidad de este orden.
 - Además, con este tipo de circuitos, como se han probado sobre Protoboard, a menudo estábamos midiendo capacidades parásitas más grandes que el propio valor del condensador incógnita.
 - Finalmente, pero, hemos conseguido aplicar y probar de forma experimental un método que es funcional para poder realizar algunas mediciones en este proyecto.

- Sobre la impresora, el proceso de impresión y las mediciones:
 - El hecho que no haya manual de uso de la impresora y que no haya ninguna guía de cómo hay que realizar las impresiones ha dificultado bastante el proceso. De cara a futuros usos de la impresora, en este proyecto se han añadido explicaciones básicas de cómo funciona el software y cuáles son las operaciones previas a imprimir, así como la configuración de la tinta de plata que hemos usado.
 - La preparación de Kapton no ha sido todo lo óptima que debería y esto ha dificultado bastante que pudiésemos obtener como resultado los patrones que teníamos diseñados, sin desviaciones.
 - Imprimir escalando el patrón horizontalmente da mejores resultados que hacerlo verticalmente en cuanto a tiempo y a rendimiento de la impresora (el cartucho se calienta menos, y, por ende, no introduce tantas desviaciones en la impresión respecto al patrón).
 - Hemos intentado ir al límite con la distancia entre placas (líneas) de los interdigitados para obtener el máximo de capacidad en la mínima superficie posible. Como no se han podido obtener resultados favorables con este tipo de configuraciones, hemos tenido que ampliar esta distancia. Igualmente, esta sigue siendo bastante pequeña, de unos 200 μm .
 - Finalmente, pero, hemos logrado imprimir un modelo de capacitor basado en interdigitados que es funcional. El valor que hemos obtenido de capacidad se asemeja bastante al que habíamos previsto de forma teórica. Además, la capacidad tiene cierta sensibilidad a estímulos tales como proximidad o contacto.

- Sobre el desarrollo del proyecto:

- La situación geopolítica y de "shortage" de componentes ha influido de forma notable en el proyecto. La tinta de planta se demoró hasta tal punto que llegó una semana y media antes de la entrega del proyecto.
- A pesar de esta circunstancia, la parte correspondiente a encontrar un sistema de caracterización para luego poder medir las capacidades resultantes, se realizó y documentó con suficiente antelación.
- Hemos podido aplicar conocimientos aprendidos en asignaturas muy diversas que se han cursado durante el grado: electrónica analógica, electrónica de potencia, instrumentación, teoría de circuitos, informática industrial o microcontroladores. Además, hemos experimentado y hemos tenido que investigar sobre algo que se nos presentaba como novedad, la impresora.

9. Webgrafía

9.1. Información preliminar

- EDUARD GL. 2013. "SENSORES CAPACITIVOS" [consultado el 11/11/2021]. Disponible en: [\(PDF\) Sensores Capacitivos | Eduard GL - Academia.edu](#).
- E.S. KOLOSOVAS. S.F. "1 Parámetros de un capacitor interdigitado" [consultado el 14/11/2021]. Disponible en: [1 Parámetros de un capacitor interdigitado. | Download Scientific Diagram \(researchgate.net\)](#).
- FUJIFILM. S.F. "Impresora de materiales Dimatix DMP-2850" [consultado el 20/11/2021]. Disponible en: [Impresora de materiales Dimatix DMP-2850 | Fujifilm \[Spain\]](#).
- WIKIPEDIA. S.F. "Kapton" [consultado el 27/02/2022]. Disponible en: <https://es.wikipedia.org/wiki/Kapton>.

9.2. Métodos para medir capacidades

- KERRY WONG. Octubre 16, 2010. "AVR LC Meter With Frequency Measurement" [consultado el 04/12/2022]. Disponible en: [AVR LC Meter With Frequency Measurement – Kerry D. Wong \(kerrywong.com\)](#).
- ELECTGPL. S.F. "Pico Capacimetro con Arduino y Display" [consultado el 11/12/2022]. Disponible en: [Electgpl Electrónica: Pico Capacimetro con Arduino y Display](#).
- ELECTRICALENGINEERING. S.F. "Reading very low capacitances with Arduino" [consultando el 12/12/2022]. Disponible en: [capacitor - Reading very low capacitances with Arduino - Electrical Engineering Stack Exchange](#).
- ELECTGPL. S.F. "Medidor de Bobinas/Inductores y Capacitores con Arduino y Display LCD" [consultado el 26/01/2022]. Disponible en: [\(1\) Medidor de Bobinas/Inductores y Capacitores con Arduino y Display LCD - YouTube](#).

9.3. Programa y maqueta final

- ELECTROSAURIO. Junio 27, 2019. "CONTADOR DEL 0 AL 999, SIN ARDUINO, ATMEGA328P" [consultado el 03/03/2022]. Disponible en: [\(1\) CONTADOR DEL 0 AL 999 🔥 SIN ARDUINO 🔥 ATMEGA328P - DISPLAY DE 7 SEGMENTOS - YouTube](#).
- LUIS LLAMAS. Septiembre 2, 2014. "LEER UN PULSADOR CON ARDUINO" [consultado el 04/03/2022]. Disponible en: [Leer un pulsador con Arduino \(luisllamas.es\)](#).

Todas las imágenes que aparecen en este proyecto están referenciadas en el pie de la figura, en alguna de las webs anteriores o son de creación propia.

10. Bibliografía

- Mikolajek, M., Reinheimer, T., Bohn, N., Kohler, C., Hofmann, M.J. & Binder, J.R. (2019). "Fabrication and Characterization of Fully Inkjet Printed Capacitors Based on Ceramic/Polymer Composite Dielectrics on Flexible Substrates". NatureSearch, Scientific Reports, [consultado el 05/01/2022]. Disponible en: <https://www.nature.com/articles/s41598-019-49639-3.pdf>.
- Brodeala, A., Bonea, A., Alcade, A., Mihailescu, B., Vasile, A. & Svasta, P. (2012). "Electrical Characterization of Ink-Jet Printed Organic Capacitors on Flexible Substrate". Proc. of SPIE Vol. 8411 84111B-1, [consultado el 05/01/2022]. Disponible en: https://www.researchgate.net/publication/245539291_Electrical_Characterization_of_Ink-Jet_Printed_Organic_Capacitors_on_Flexible_Substrate.
- Ibrahim, N., Akindoyo, J.O. & Mariatti, M. (2021). "Recent development in silver-based ink for flexible electronics". Journal of Science: Advanced Materials and Devices 7 (2022) 100395, [consultado el 02/02/2022]. Disponible en: <https://reader.elsevier.com/reader/sd/pii/S2468217921000770?token=056CB477ADD2C40053FA264BB706876866328654840295E5592C72D888972DF18C5C0DA3038B6EAC8AD46995F80D374C&originRegion=eu-west-1&originCreation=20220530173622>.
- Milena Kiliszkievicz *et al.* (2020). "Structural characterization of inkjet printed capacitor layers in various technological conditions". Soldering & Surface Mount Technology, Volume 32, Number 4, 235–240, [consultado el 02/02/2022]. Disponible en: <https://www.emerald.com/insight/content/doi/10.1108/SSMT-11-2019-0039/full/pdf?title=structural-characterization-of-inkjet-printed-capacitor-layers-in-various-technological-conditions>.
- Alvarado Pérez, Miriam (2020). "Development of Flexible Gas Sensors Based on Additive Fabrication Processes". Universitat Rovira i Virgili, [consultado el 21/02/2022]. Disponible en: <http://hdl.handle.net/10803/669439>.
- Zhige Tu *et al.* (2022). "Structural design of flexible interdigital capacitor based upon 3D printing and spraying process". Smart Mater. Struct. 31 045005, [consultado el 17/05/2022]. Disponible en: <https://iopscience.iop.org/article/10.1088/1361-665X/ac54574>.
- ATmega328P Xplained Mini User Guide.