

Diseño de un sistema para control de un piezo-actuador vibratorio

Carlos Iván Alvarado Tovar

Instituto Tecnológico de Celaya
mkivan_at@yahoo.com

Sergio Iván Urbina Rodríguez

Instituto Tecnológico de Celaya

Sofía Magdalena Vega Díaz

Instituto Tecnológico de Celaya

Luis Alejandro Alcaráz Caracheo

Instituto Tecnológico de Celaya

Ramón Rodríguez Castro

Instituto Tecnológico de Celaya

Resumen

El proyecto consiste en la elaboración de un prototipo de un sistema embebido encargado de controlar un piezoeléctrico para su uso en la Deposición Química de Vapor, un fenómeno en un proceso químico utilizado para producir productos de alta pureza y de alto rendimiento de materiales sólidos.

Este sistema cuenta con las ventajas de tener un control completamente digital, ya que el sistema utilizado habitualmente, es analógico, y requiere habilidad para poder calibrar el aparato cada vez que se va a utilizar, así como un costo elevado. Con este prototipo, se plantea una alternativa con un costo fraccionario al producto original, así como una

interfaz fácil de operar por el usuario, para que la funcionalidad del sistema no dependa de la habilidad del usuario para calibrar dicha máquina.

Las tecnologías utilizadas para este proyecto fueron Raspberry Pi, así como circuitos integrados de potencia para la amplificación de la señal que va hacia el piezoeléctrico.

Palabra Clave: Deposición química de vapor, generador de funciones, piezoeléctrico, sistema embebido, vibración.

1. Introducción

La Deposición Química de Vapor (ver fig.1) o CVD (de sus siglas en inglés Chemical Vapor Deposition) es un proceso químico utilizado para elaborar productos de alta pureza y de alto rendimiento de materiales sólidos. El proceso se utiliza a menudo en la industria de semiconductores para producir películas delgadas formadas por nanotubos [1].

Los Nanotubos fueron descubiertos en Japón por S, Iijima en 1991, son una forma alotrópica del carbono, como el diamante, el grafito o los fulleneros. Están constituidos por redes hexagonales de carbón curvadas y cerradas, formando tubos nanométricos con una serie de propiedades fascinantes que fundamentan el interés que han despertado numerosas aplicaciones tecnológicas. Es, además, la primer sustancia conocida por la humanidad capaz de sustentar indefinidamente su propio peso, condición necesaria para poder construir un ascensor espacial.

En un proceso CVD estándar el sustrato (oblea) se expone a uno o más precursores volátiles, que reaccionan o se descomponen en la superficie del sustrato para producir el depósito deseado. Con frecuencia, también se producen subproductos volátiles, que son eliminados por medio de un flujo de gas que pasa a través de la cámara de reacción [2].

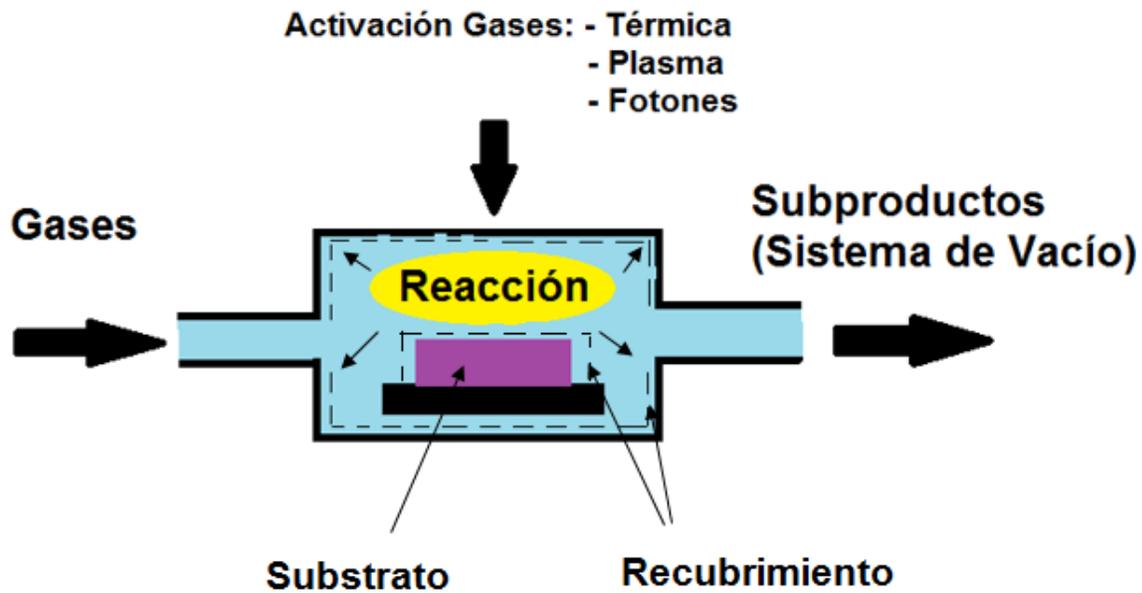


Fig.1.- Deposición Química en fase de vapor.

Un problema con estas máquinas para CVD (ver fig.2) es el control del mismo, ya que al realizar pruebas en la máquina se deben de ingresar los parámetros de frecuencia y amplitud de la onda para que el piezoeléctrico realice su trabajo, aquí el problema es al ingresar los datos, ya que se realiza de manera analógica por medio de perillas y se tiene cierta incertidumbre dada la naturaleza de un sistema analógico, ese valor depende de la precisión de la persona que esté manejando la máquina en ese momento.[3]

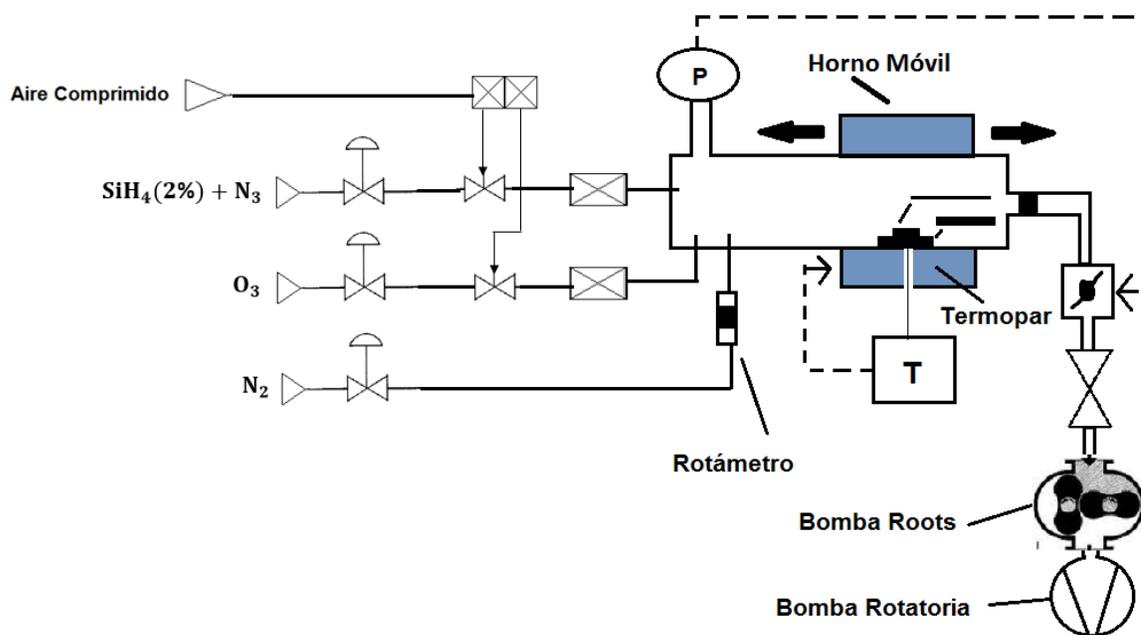


Fig. 2.- Esquema General de un sistema CVD.

El problema con este tipo de procesos analógicos es que los resultados tienden a variar utilizando los mismos parámetros, tanto de frecuencia como de amplitud, en diferentes máquinas, esto debido a la forma en que se ingresan los parámetros.

Una propuesta para atender la problemática anterior es desarrollar un dispositivo que consta de un generador de funciones para el control de la frecuencia y amplitud de una onda cuadrada que será enviada al piezo-actuador para producir una vibración y generar nanotubos de carbono. Dicho piezo-actuador produce una vibración la cual permite la creación de gotas pequeñas de ciertas sustancias, de las cuales salen los nanotubos de carbono antes mencionados, durante el proceso las gotas que se crean tienden a subir por el recipiente que contiene dicha sustancia y son recolectadas por la siguiente fase de la máquina de CVD. [4]

2. Desarrollo

La primera etapa del proyecto fue investigar el tipo de piezo-actuador que tiene la máquina de CVD, sin embargo este tipo de producto no se encuentra fácilmente en México, por lo que para este proyecto se decidió emplear un piezo-actuador contenido en el interior de los parlantes (Tweeters), ver fig.3.



Fig.3.- Tweeters, elementos con piezoeléctrico utilizados.

Una vez que se consiguió el piezo-actuador, se procedió a implementar una etapa de potencia para poder amplificar las señales del generador de funciones (ver fig.4).

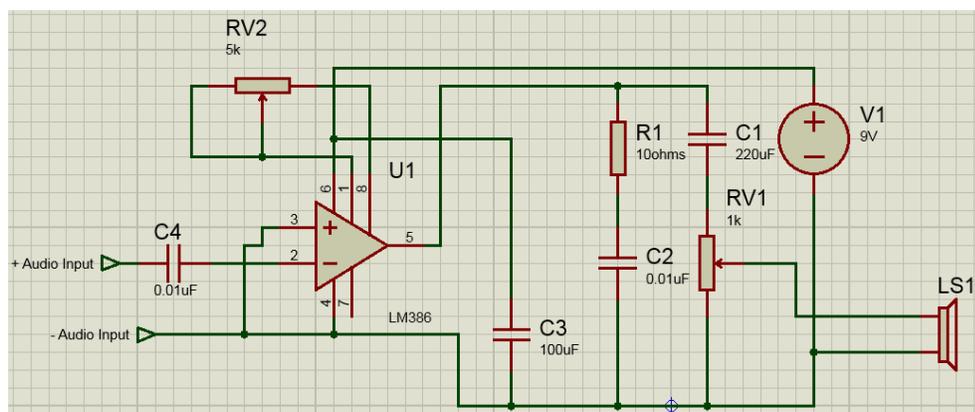


Fig.4.- Etapa de Potencia.

El circuito se pasó a una placa de circuito impreso (ver fig.5). Este cuenta con dos potenciómetros para regular ganancia y ciclo de trabajo, pero no es necesario que el usuario los manipule. Estos deben de ser calibrados únicamente al cambiar de piezoeléctrico, para que la onda se encuentre entre los rangos de voltaje que soporta, y sin que se sature.



Fig.5.- Circuito de potencia en placa.

Una vez construido el circuito, se prosiguió con la realización de la programación. A continuación, un resumen de las librerías y lenguajes utilizados, los cuales se pueden observar en la Tabla 1.

Raspberry Pi (ver fig.6), es una plataforma de hardware abierta, formada por un ordenador de placa reducida, con 1Ghz de CPU, 512 Mb de RAM, sin disco duro, alimentada por 5 voltios y un consumo menor a 3 Watts. Por sus prestaciones y precio, fue elegida para este proyecto.

Raspberry Pi, al ser un sistema de computadora completo, necesita un sistema operativo para poder funcionar. Decidimos utilizar Raspbian GNU/Linux como sistema operativo, dada su estabilidad y buen acoplamiento con Raspberry Pi, dos facciones importantes para el proyecto. Una imagen del sistema operativo fue colocada en una tarjeta SD para que pueda arrancar con ella [5].

Parte	Lenguaje o Programa	Librería
Sistema Operativo	Rasphbian GNU/Linux	Negritas
Interfaz Gráfica	Phyton	Pygame SSUI
Driver para Teclado Matricial		GPIO
Generación de Funciones	C	Siggen
Plantillas para Interfaz	Phyton	YAML
Guardado de Configuraciones		Pyini

Tabla 1.- Lenguajes y librerías utilizados en el proyecto.



Fig. 6.- Raspberry Pi, Modelo B.

Los lenguajes utilizados para crear el programa que corre el sistema operativo, fueron C y Python. C fue elegido por su rapidez y funciones de bajo nivel, mientras que Python fue usado en las partes de alto nivel, como la interfaz gráfica, dada su similitud con el pseudocódigo y su cantidad de librerías disponibles.

La interfaz gráfica fue creada desde cero, ya que no se encontró un entorno completamente de este tipo. Alternativas como LabVIEW (del inglés Laboratory Virtual Instrumentation Engineering Workbench) no son compatibles con el hardware de forma autónoma, a su vez, el costo de la licencia aumentaría considerablemente el costo del producto final. [6] Es un sistema simple el cual permite crear pantallas que se mostrarán al usuario y permitirá insertar datos y conocer variables, similar a un Panel View. Ejemplos de las pantallas se pueden ver en la fig.7.



Fig. 7.- Ejemplos de la interfaz gráfica.

Como se puede observar, la interfaz consta de 4 partes, selección de onda, selección de frecuencia, ganancia (que viene a ser la amplitud), y la pantalla de inicio que cuenta con un temporizador del tiempo a mantener funcionando el sistema.

Cada una de estas opciones se guarda en una archivo .ini, por lo cual, al momento de reiniciar la máquina, no se pierden los valores guardados.

Cada una de las pantallas se escribió en un lenguaje de marcado llamado YAML (del inglés YAML Ain't Markup Language, definición recursiva). Un ejemplo de la plantilla para la pantalla se puede apreciar a continuación en la figura 8.

El control de la interfaz se puede realizar mediante un teclado matricial (fig.9), pero también puede utilizarse un teclado convencional. Este se conectó a las entradas generales de la Raspberry (GPIO, del inglés Genral Puropose Input / Output).



Fig. 9.- Ejemplo de control mediante teclado matricial.

```
%YAML 1.2
---
YAML: YAML Ain't Markup Language

What It Is: YAML is a human friendly data serialization
            standard for all programming languages.

YAML Resources:
  YAML 1.2 (3rd Edition): http://yaml.org/spec/1.2/spec.html
  YAML 1.1 (2nd Edition): http://yaml.org/spec/1.1/
  YAML 1.0 (1st Edition): http://yaml.org/spec/1.0/
  YAML Issues Page: https://github.com/yaml/yaml/issues
  YAML Mailing List: yaml-core@lists.sourceforge.net
  YAML IRC Channel: "#yaml on irc.freenode.net"
  YAML Cookbook (Ruby): http://yaml4r.sourceforge.net/cookbook/ (local)
  YAML Reference Parser: http://yaml.org/ypaste/

Projects:
  C/C++ Libraries:
  - libyaml # "C" Fast YAML 1.1
  - Syck # (dated) "C" YAML 1.0
  - yaml-cpp # C++ YAML 1.2 implementation
  Ruby:
  - psych # libyaml wrapper (in Ruby core for 1.9.2)
  - RbYaml # YAML 1.1 (PyYaml Port)
  - yaml4r # YAML 1.0, standard library syck binding
  Python:
  - PyYaml # YAML 1.1, pure python and libyaml binding
  - PySyck # YAML 1.0, syck binding
```

Fig. 8.- Ejemplo del lenguaje YAML para pantallas.

La generación de funciones se realizó en C, junto con la librería siggen, se hizo un programa ejecutable por la línea de comandos el cual puede ser manipulado, la ganancia de la onda (traducida en amplitud de la onda), y la frecuencia de la misma.

Dado que la onda es reproducida por la tarjeta de sonido, la frecuencia disponible varía de 18 Hz a 22 khz. El sistema original trabaja habitualmente a 15khz.

Las onda generadas son: cuadrada, seno, triangular y diente de sierra.

La onda senoidal (fig.10), fue la que se obtuvo un mejor comportamiento, debido a que es la más utilizada en los sistemas de audio, por lo tanto, es la más adecuada a la realidad.

Con esta, el piezoeléctrico hacía una vibración suave, difícil de apreciar.

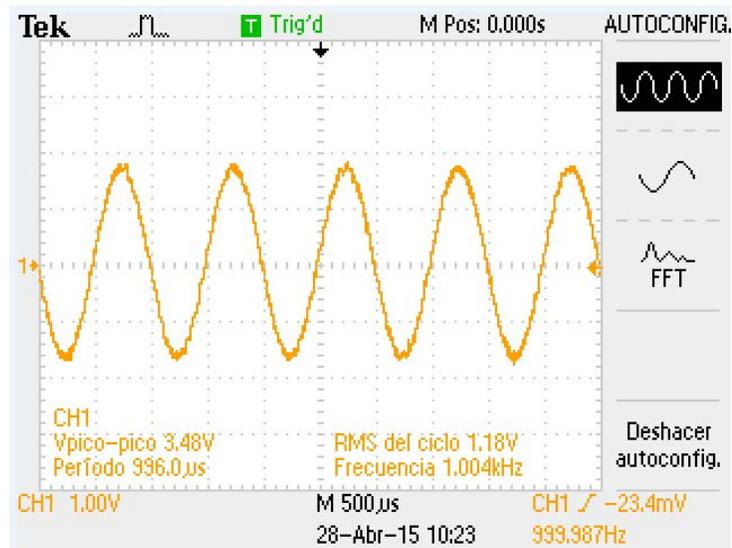


Fig. 10.- Onda senoidal generada a 1 khz.

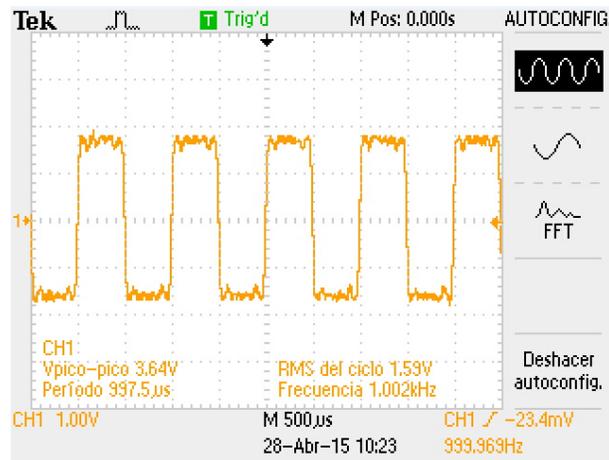


Fig. 11.- Onda cuadrada generada a 1 Khz.

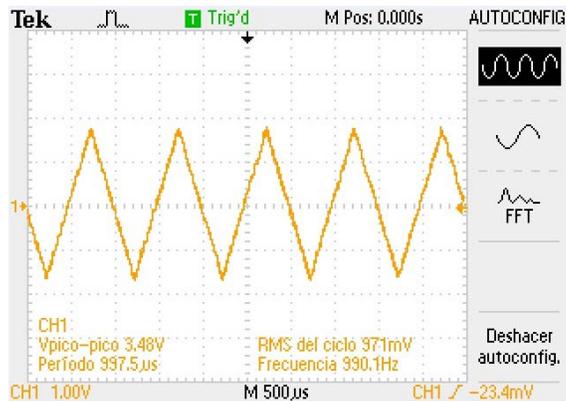


Fig.12.- Onda triangular generada a 1 Khz.

Con la onda cuadrada (fig.11), el sistema tenía una respuesta con menos fidelidad, sin embargo, el sistema mostraba una excelente respuesta en cuanto a la vibración. Aquí se puede apreciar que el sistema tiene problemas para realizar ondas completamente cuadradas, aunque esto varía de sistema a sistema. La onda triangular (fig.12) se aprecia de forma casi perfecta, y al momento de conectarle el piezoeléctrico, se obtenía una vibración menos pronunciada que la cuadrada, pero aun así apreciable.

La onda de diente de sierra (fig.13) tiene un comportamiento bastante similar a la triangular, con el único cambio de un sonido más grave que la triangular, pero similar en cuanto a comportamiento de la onda y apreciación de la vibración.

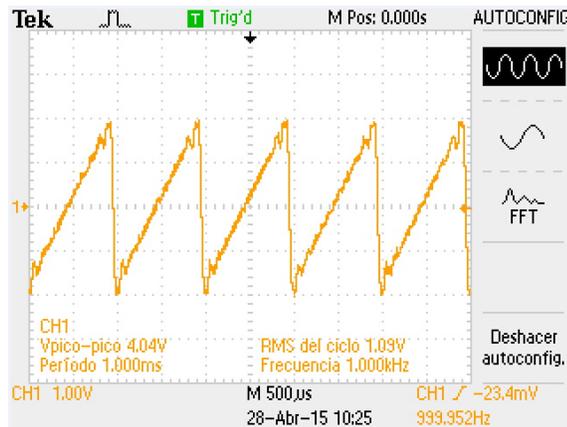


Fig. 13.- Onda diente de sierra generada a 1 khz.

3. Resultados

El prototipo creado (fig.14) logró controlar el piezoeléctrico de tal forma que su vibración se podía apreciar en el sonido que generaba, variando según la frecuencia y el tipo de onda.

La forma de manejar la interfaz y colocar los parámetros de funcionamiento fue intuitiva y sencilla de utilizar, lo que representa una ventaja para cualquier tipo de usuario.



Fig. 14.- Prototipo completo.

5. Conclusiones

A pesar de que se realizaron pruebas con diferentes tipos de ondas, se llegó a la conclusión de que la onda que mejor hace actuar al piezoeléctrico es la onda cuadrada, esto debido a los cambios bruscos que esta realiza, los cuales hacen que el piezoeléctrico realice un movimiento de sube y baja de manera rápida lo cual permite la creación de la vibración.

Un sistema como estos también podría incluir otras mejoras como lo pueden ser una pantalla touch-screen, o un control de lazo cerrado. Hasta ahora, no se utilizó el lazo cerrado para máquinas de CVD, pero podría ser una buena idea para la investigación.

Para futuros trabajos un sistema touch-screen podría ser implementado con facilidad, dadas las bondades del lenguaje Python, pero por el momento no se contaba con el hardware para poder llevarlo a cabo.

6. Referencias

- [1] K. Tadanaga, A. Yamaguchi, A. Hayashi, M. Tatsumisago, J. Mosa, M. Aparicio. "Preparation of $\text{Li}_4\text{Ti}_5\text{O}_{12}$ electrode thin films by a mist CVD process with aqueous precursor solution", *Journal of Asian Ceramic Societies*, Volumen 3, Marzo 2015, Páginas 88-91.
- [2] S. Fay, L. Feitknecht, R. Schluchter, U. Kroll, E. Vallat-Sauvain, & Shah, "Process and their effect in improving performances of amorphous and microcrystalline silicon solar cells", *Solar Energy Materials and Solar Cells*, 90(18), 2006, 2960-2967.
- [3] 2012. USO DE LA TECNOLOGÍA CVD <http://www.iit.upcomillas.es/pfc/resumenes/4fc683d39589e.pdf>., Mayo 2015.
- [4] A. Mystkowski, A. Piotr Koszennik, "Mu-Synthesis robust control of 3D bar structure vibration using piezo-stack actuators", *Mechanical Systems and Signal Processing*, 29 Agosto del 2015, Páginas 1-10.
- [5] Brock, J. Dean, F. Rebecca and E. Cameron, "Changing the world with a Raspberry Pi.". *Journal of Computing Sciences in Colleges*, 29(2), 152-153.
- [6] 2015. LabVIEW for Arduino versus LabVIEW for Raspberry Pi. <http://www.tsxperts.com/labview-arduino-versus-labview-raspberry-pi-round-1/>. Mayo 2015.