

The Triangulation of Titling Data in Non-Linear Gaussian Fashion via ρ Series^{*}

John Doe
Magic Department,^{**} Richard Miles University
Richard Row, L^AT_EX Academy

2017
December

1. Introducción

El objetivo de este trabajo consiste en describir la curva de respuesta de un diodo y la característica de un elemento integrado. Para ello utilizamos una placa de sonido de computadora como generador de ondas y como placa de adquisición, alimentando los diversos circuitos con su salida de audio y registrando la señal deseada con su entrada. En la sección 2 describimos el procedimiento experimental utilizado para realizar dicha caracterización, particularizando en 2.1 sobre la implementación del control de la placa de sonido mediante una biblioteca programada en Python y en 2.2 sobre la caracterización de la placa de audio como generador de ondas y placa de adquisición. A continuación dedicamos la sección 3 a la caracterización de los componentes elegidos, donde en 3.1 mostramos y discutimos los resultados obtenidos en la medición de la respuesta del diodo, mientras que en 3.2 presentamos la propuesta de trabajo para la caracterización de un componente integrado.

2. Método

2.1. Biblioteca utilizada

El control de la placa de sonido fue llevado a cabo a partir de programar una biblioteca propia en lenguaje Python que incluya las funciones orientadas al usuario final necesarias para utilizar el dispositivo como generador

^{*}No procrastination

^{**}I am no longer a member of this department

de ondas y placa de adquisición [1]. Para ello nos basamos en la biblioteca *PyAudio* [2], la cual consiste en una implementación de alto nivel y en Python de la API multiplataforma provista en C/C++ por *PortAudio* [3].

La comunicación con la placa de sonido se basa en el paradigma de *callbacks* de entrada y salida, que consiste en funciones que se ejecutan en bucles concurrentes y realizan respectivamente la lectura y escritura de los *buffers* que el sistema operativo dispone para el dispositivo para grabar y reproducir audio. En la implementación de nuestra librería, cada función *callback* cuenta con un *buffer* circular propio tipo FIFO (*first input, first output*) que alojan la señal que se quiere emitir (en el caso del *callback* de salida) y la señal que se está registrando (en el caso del *callback* de entrada). En cada iteración de ambos bucles se transfiere un paquete de datos de tamaño fijo y configurable que se guarda en el correspondiente *buffer* de forma cíclica, permitiendo una ejecución continua similar a la obtenida en un generador de funciones y un osciloscopio, donde las interrupciones en la ejecución debido a problemas de latencia pueden ser evitadas ajustando debidamente los tamaños del paquete y del *buffer*.

La biblioteca además incluye las funciones necesarias para calibrar la entrada y las salidas a unidades de tensión, así como para el multiplexado y demultiplexado de señales en orden de poder trabajar con dos canales (modo estéreo de grabación y reproducción). En la sección 2.2 describiremos los análisis que nos permitieron desarrollar estas funciones. Asimismo se incluye una función para generar señales de salida con formas de onda típicas y *scripts* que permiten realizar barridos de parámetros de dicha señal (frecuencia, amplitud, etc.).

2.2. Caracterización placa de sonido

En el presente trabajo utilizamos la placa de sonido *Realtek "MODELO"* para generación y adquisición de señales. Para ello utilizamos el puerto de salida analógica estéreo principal y el puerto de entrada analógica estéreo *line-in*, respectivamente.

Al adquirir señales estéreo con la placa de sonido, esta multiplexa las señales de los dos canales en un mismo paquete de datos. En la **Figura 1** se muestra el paquete de datos obtenido al ingresar una señal senoidal por uno de los canales de entrada mientras el otro canal se encuentra conectado a GND. Para poder separar las dos señales adquiridas en modo estéreo se implementó una función de demultiplexado que permite manipular por separado las señales de ambos canales de entrada. Del mismo modo que con la señal de entrada, cuando se desea utilizar ambos canales de la salida estéreo se debe proveer a la placa de sonido un paquete de datos con las dos señales multiplexadas. Para facilitar la manipulación de esos datos también implementamos la función de multiplexado respectiva.

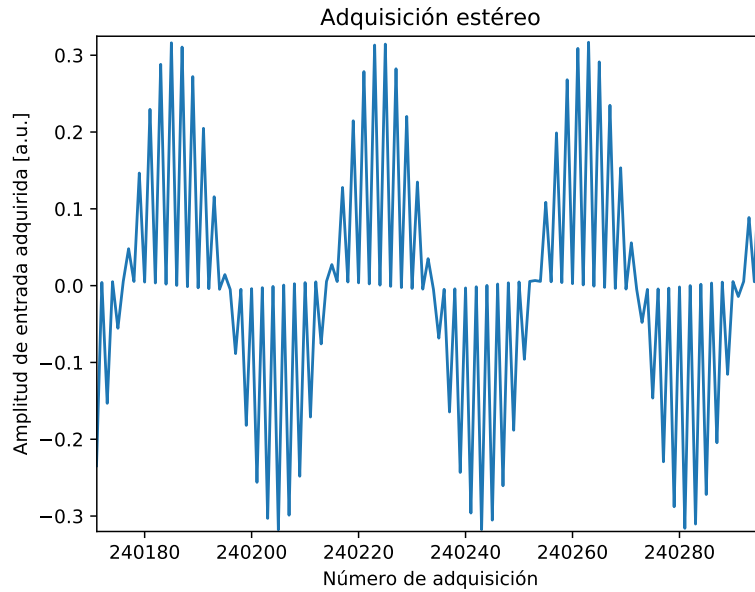


Figura 1: Paquete de datos obtenidos al leer la señal de entrada estereo cuando un canal se encuentra alimentado por una señal senoidal mientras el otro canal se encuentra conectado a GND.

Los paquetes de datos que se reciben y se envían a la placa de sonido contienen la información de la amplitud de las señal traducida a una unidad de cuentas arbitraria. Para poder establecer la correlación de esta unidad arbitraria con un valor de tensión realizamos un proceso de calibración tanto

de la entrada como de la salida de la placa de sonido. Para ello generamos con la placa de sonido señales sinusoidales de distintas amplitudes, estas señales fueron medidas con un osciloscopio externo y a su vez fueron ingresadas como entrada a la placa de sonido. Se registraron los valores de las unidades arbitrarias y su valores de tensión respectivo con el objetivo de ajustar una curva de calibración tanto de los canales de entrada como de los canales de salida de la placa. En las **Figuras 2 y 3** se muestran las curvas de calibración obtenidas. Los datos medidos fueron ajustados linealmente obteniéndose las siguientes expresiones,

$$\text{Tensión de salida} = 1.620 \text{ V} \times \text{Cuentas}$$

$$\text{Tensión de entrada} = 2.555 \text{ V} \times \text{Cuentas}$$

Utilizando las curvas de calibración obtenidas generamos funciones adicionales que permiten convertir una señal que contiene la información en cuentas de unidad arbitraria en valores de tensión y viceversa.

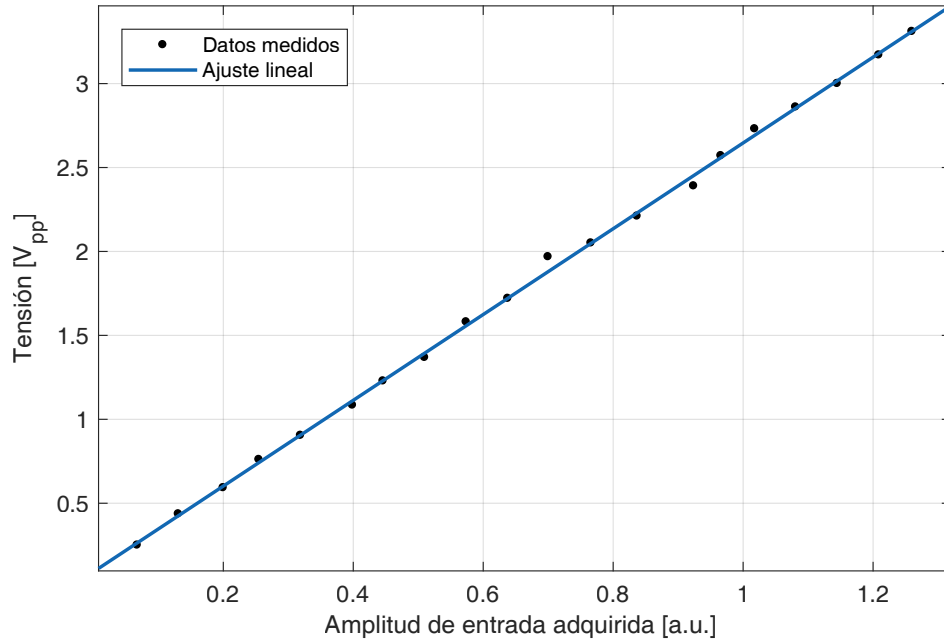
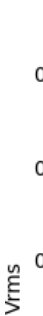


Figura 2: Curva de calibración de la señal de entrada de la placa de sonido.

Para poder caracterizar la respuesta en frecuencia de la placa de sonido realizamos un barrido en frecuencias generando señales sinusoidales de distintas frecuencias con la salida de la placa que fueron ingresada como entrada a la misma. Registramos la variación en la ganancia del sistema en función de la frecuencia, los resultados se muestran en las **Figuras 4 y 5** para dos frecuencias de muestreo distintas.



F



Fi

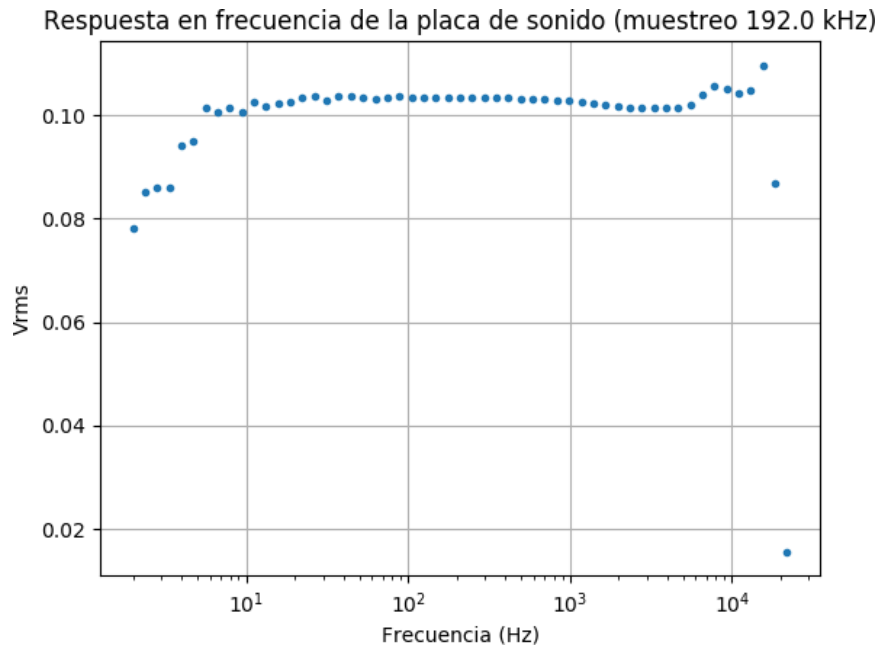


Figura 5: Variación de la ganancia en función de la frecuencia de la señal, para una frecuencia de muestreo configurada de 192 kHz

Conectamos ambos canales de la entrada estéreo a GND y registramos la señal obtenida, los resultados para cada canal se muestran en las **Figuras 6 y 7**. El valor del ruido RMS obtenido es de 277 μV .

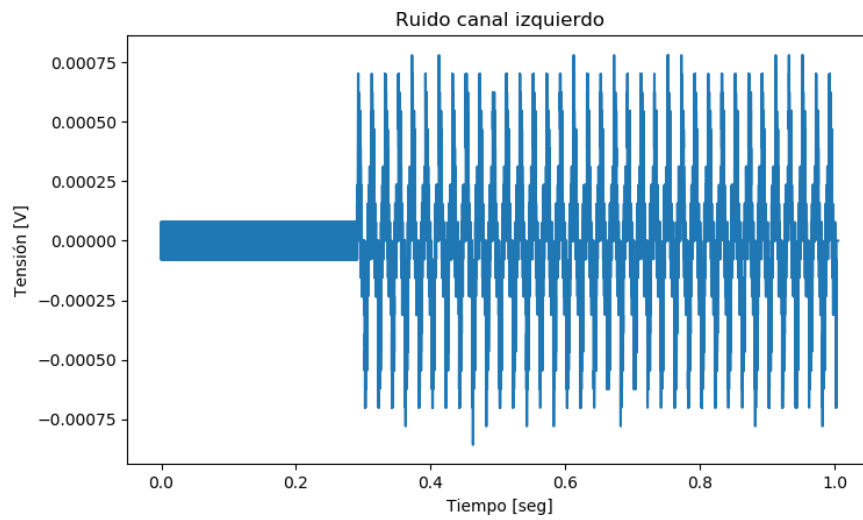


Figura 6: Señal de ruido correspondiente al canal izquierdo del puerto de entrada estéreo.

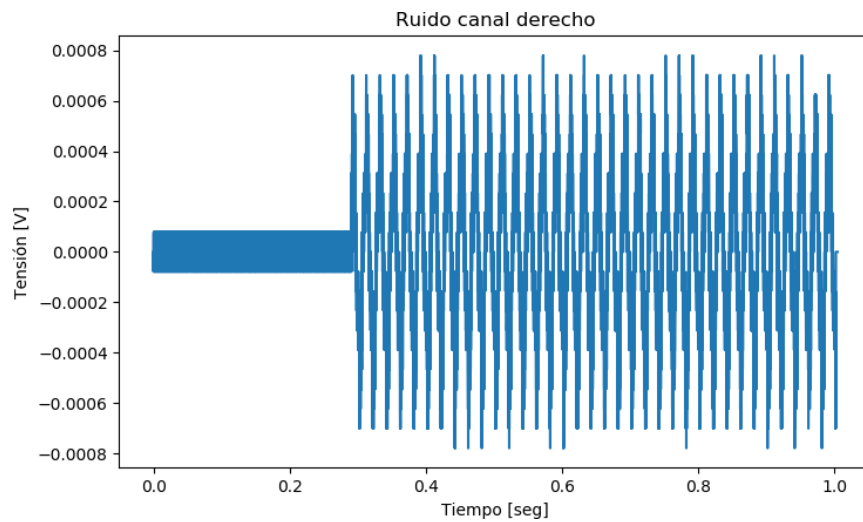


Figura 7: Señal de ruido correspondiente al canal derecho del puerto de entrada estéreo.

3. Resultados

3.1. Discreto

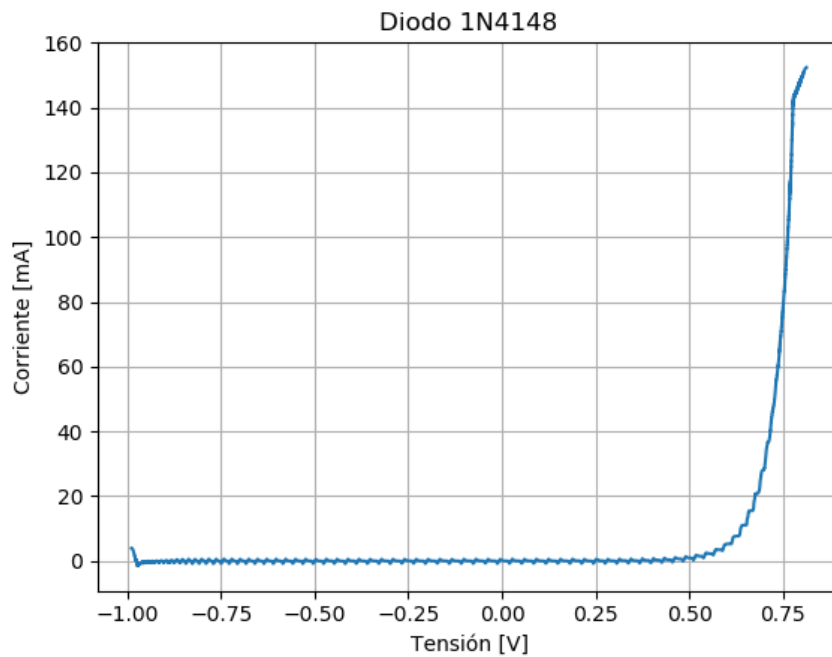


Figura 8

3.2. Integrado

Referencias

- [1] Repositorio en GitHub de la biblioteca propia utilizada en el trabajo, URL: <https://github.com/fotonicaOrg/placadeaudio>, actualizado el 19/09/2018. La biblioteca está programada en el archivo *PlacaAudio.py*.
- [2] Página web oficial de PyAudio, URL: <https://people.csail.mit.edu/hubert/pyaudio/>, accedido el 19/09/2018.
- [3] Página web oficial de PortAudio, URL: <http://www.portaudio.com/>, accedido el 19/09/2018.