

ESCUELA POLITÉCNICA NACIONAL

INGENIERÍA ELÉCTRICA Y ELECTRÓNICA

PROCESAMIENTO DIGITAL DE SEÑALES

ALUMNO: Bryan Sangacha, Alejandra Silva

SEMESTRE: 2020-B

PARALELO: GR1

FECHA: 2020-12-23

PROFESOR:

DR. ROBIN ÁLVAREZ

TEMA: Generación y adquisición de ondas mediante hardware – Arduino (Parte 1)

**DEBER**

Tratando de vencer las limitaciones del puerto de audio, especialmente en las frecuencias bajas (menores a 100 Hz), y ya que muchísimos experimentos emplean frecuencias en este rango, se emplea el Arduino tanto para generación como para adquisición y análisis.

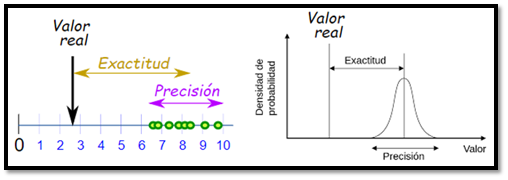
**PARTE 1: GENERADOR BÁSICO CON ARDUINO Y MEDICIÓN CON AUDIO TESTER**

Comparar los siguientes cuatro tipos de formas de generación para las ondas básicas (sinusoidal, triangular, cuadrada y diente de sierra):

1. Con funciones directas de Arduino, por ejemplo la instrucción sin, si es que las hay.
2. Con librería externa del arduino.
3. Con metodología aprendida en este documento.
4. Proponga un nuevo método.

Compare exactitud (realice una tabla para cuatro frecuencias diferentes: 500Hz, 750 Hz, 1000 Hz y la máxima que usted consiga). No se olvide que, en la metodología aprendida, la Fmax conseguida dependía del (número de muestras / período) que se considere. Grabe un video corto con los resultados.

OBSERVACIÓN: La exactitud es la cercanía de una medida al valor real, mientras que la precisión es el grado de cercanía de los valores de varias medidas en un punto.

****

**Materiales para utilizar**

* Multímetro
* Arduino DUE
* Pelador
* Cortador
* Cables para proto
* Tarjeta de audio
* Potenciómetro de 100K
* Jack

**Imagen que contiene tabla, interior, artículos, computadora

Descripción generada automáticamente**

Fig. 1. Materiales.

1. **Genera ración con funciones directas de Arduino**

Es este apartado solo se va a visualizar al función para seno ya que para las demás funciones no tienen una función ya establecida en Arduino.

**Seno**

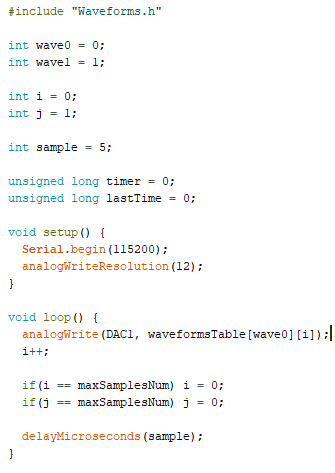
****

Fig. 2. función seno en Arduino.

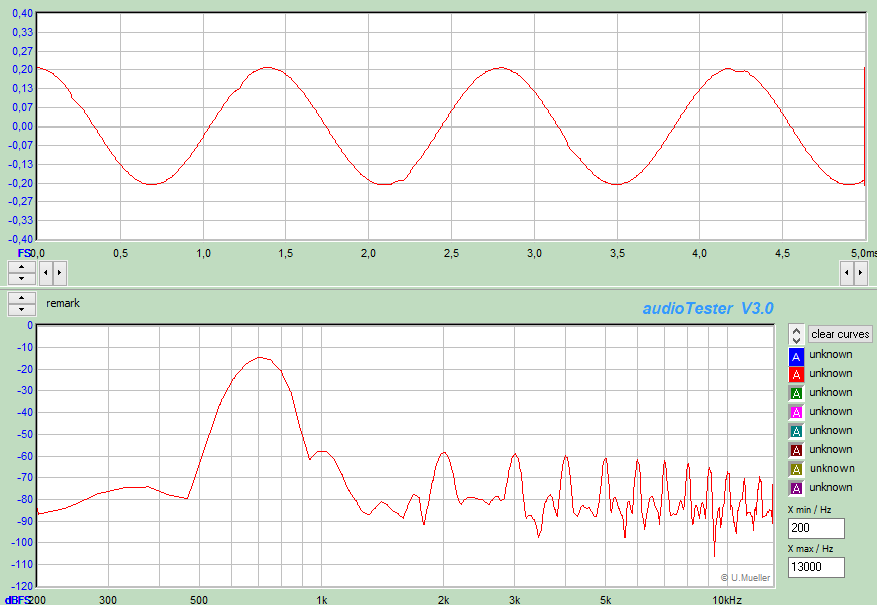


Fig. 3. Señal Seno.

1. **Generación con librería externa del arduino.**

En este apartado se utilizó códigos ya implementados que venían en Arduino como se muestra en la siguiente figura.

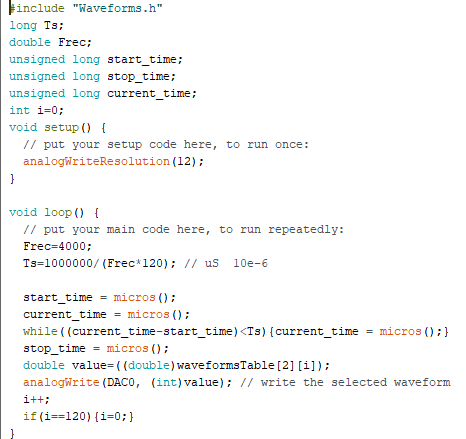


Fig. 4. Código con Librería.

Este código es de mucha ayuda ya que para poder implementar las demás señales como son triangular, diente de sierra y cuadrada solo cambia un parámetro como se indica a continuación.



Este indica va desde 0 hasta 3 donde:

* 0 hace referencia a la señal sinusoidal
* 1 hace referencia a la señal triangular
* 2 hace referencia a la señal diente de sierra
* 3 hace referencia a la señal cuadrada.

El siguiente paso es verificar y cargar el programaba para que pueda ser ejecutado y visualizado mediante Audiotester.

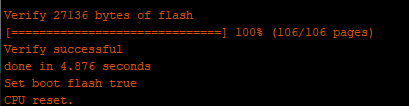


Fig. 5. Verificación del código en Arduino.

El siguiente paso es la visualización de las señales en el audiotestes.

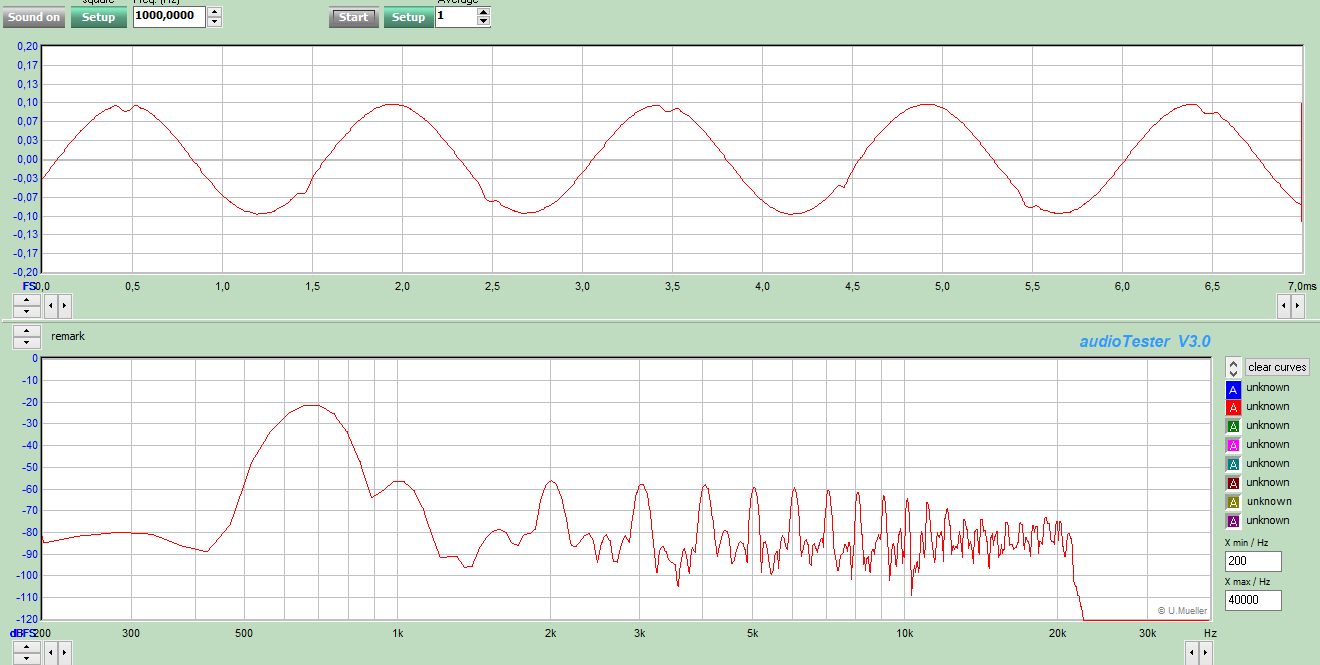


Fig. 6. Función seno.

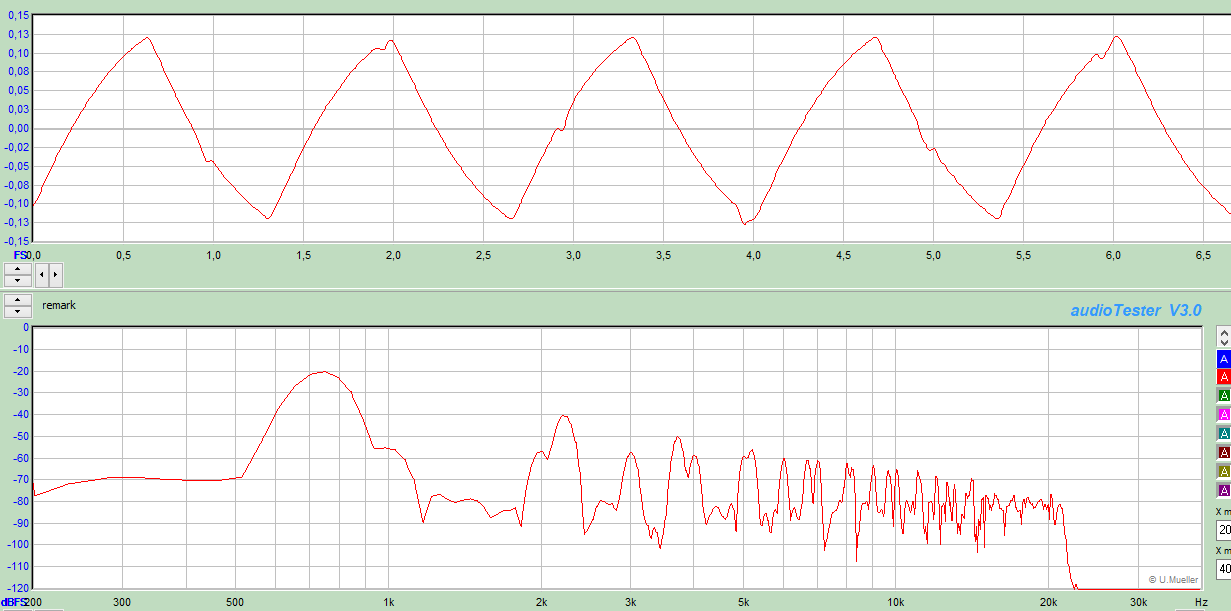


Fig. 7. Función triangular.

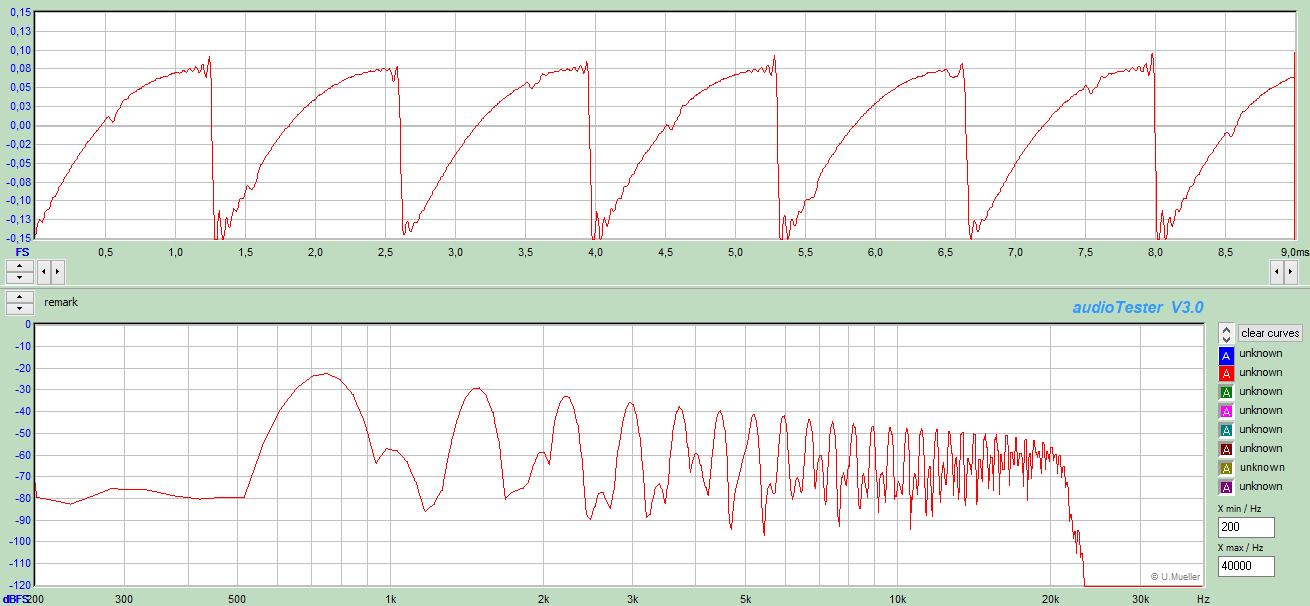


Fig. 8. Función diente de sierra.

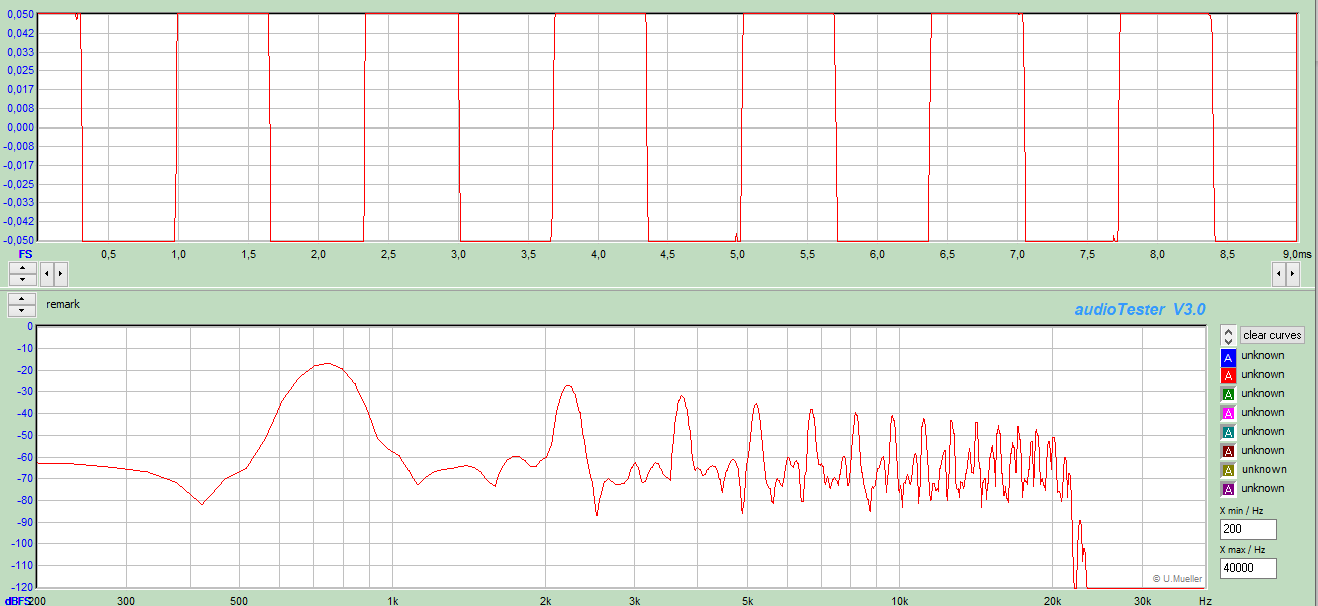


Fig. 9. Función cuadrada.

Analizando las frecuencias de las ondas generadas se puede observar que en la señal cuadrada cumple que se tengan todas las frecuencias pares como se indica en la figura, en la señal senoidal tal vez se pierden algunas frecuencia ya que no se hace uso del método por muestras que es mas exacto, lo mismo pasa en al triangular y finalmente, diente de sierra posee casi todos los armónicos como se sabe, se observa que existe un poco de inexactitud en tiempo esto debido a la tarjeta de audio.

1. **Generación con metodología aprendida en este documento**

En este apartado primero se va a generar los códigos en Matlab para determinar cada una de las muestras a utilizar, empezamos con la señal seno

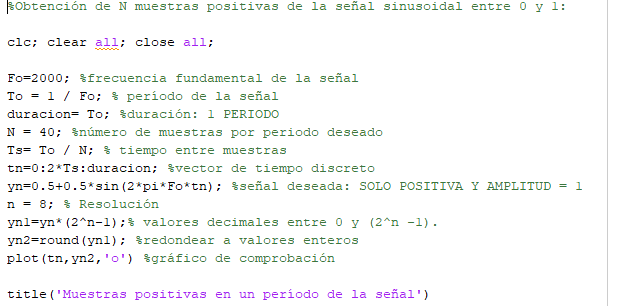


Fig. 10. Código por muestreo.



Fig. 11. Muestras para la función seno en Matlab.

Siguiente paso se crea el código en Arduino mediante un vector que este con estas muestras que se creó en Matlab, como se indica en la figura.

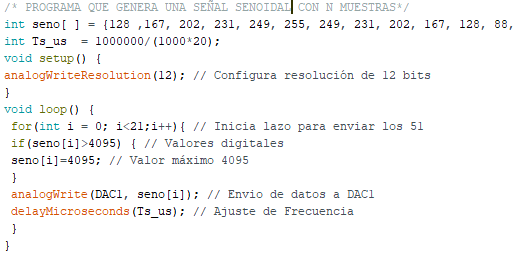


Fig. 12. Código en Arduino.

Finalmente se visualiza en el audiotester

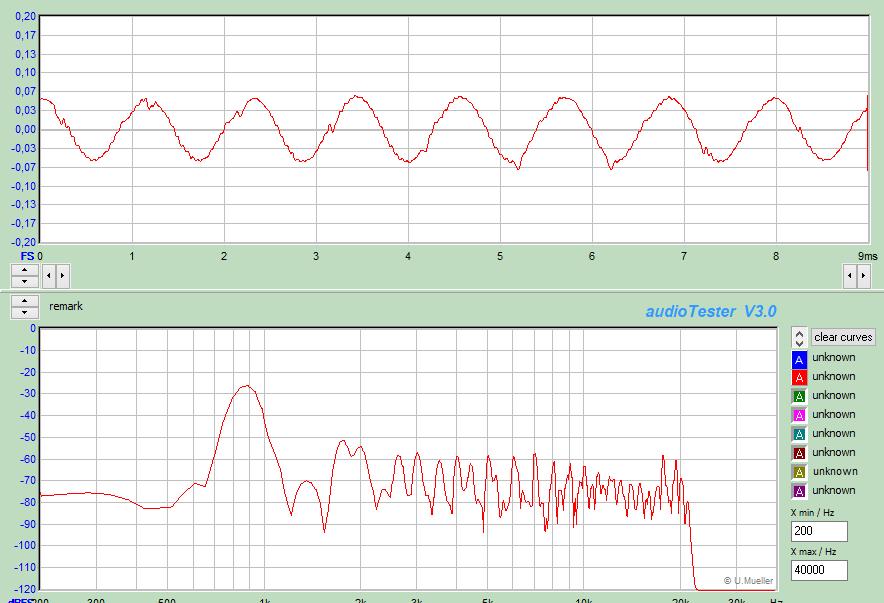


Fig. 13.Señal Senoidal.

Se puede observar que en frecuencia posee un pico en la frecuencia fundamental mucho mas grande como era lo esperado.

Función diente de sierra

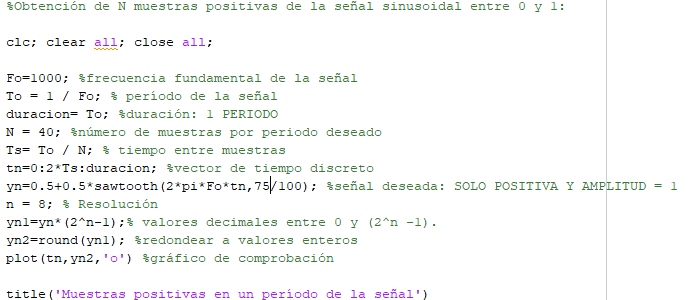


Fig. 14.Código por muestreo.



Fig. 15.Muestras para la función diente de sierra en Matlab.

Siguiente paso se crea el código en Arduino mediante un vector que este con estas muestras que se creó en Matlab, como se indica en la figura.

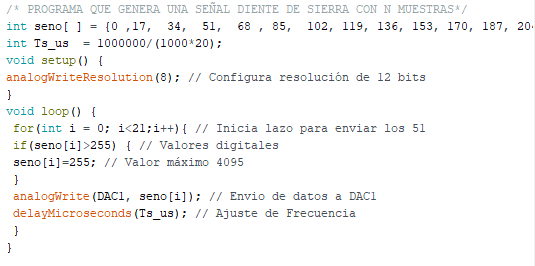


Fig. 16.Código en Arduino.

Finalmente se visualiza en el audiotester

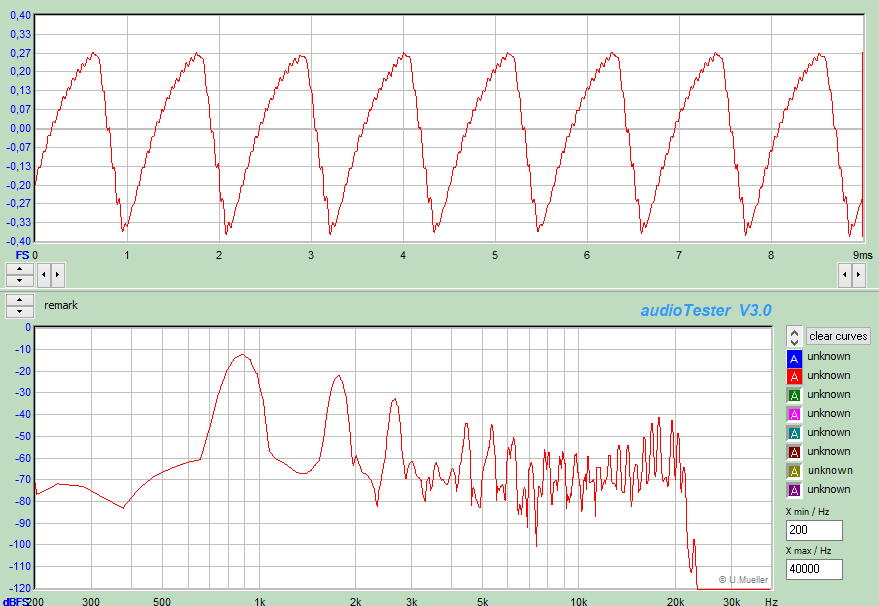


Fig. 17. Señal Diente de sierra.

Se puede observar que en frecuencia posee un pico en la frecuencia fundamental mucho mas grande como era lo esperado con todos los armónicos como se está señalando.

Función triangular

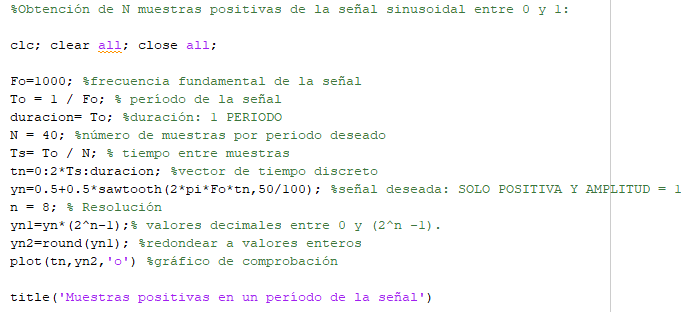


Fig. 18. Código de muestreo.



Fig. 19.Muestras para la función Triangular en Matlab.

Siguiente paso se crea el código en Arduino mediante un vector que este con estas muestras que se creó en Matlab, como se indica en la figura

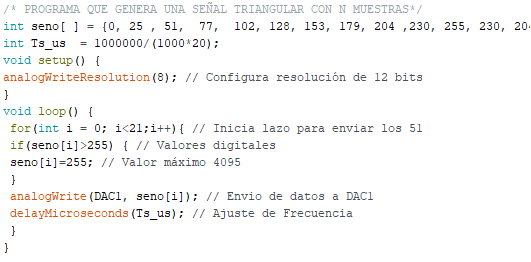


Fig. 20. Código en Arduino.

Finalmente se visualiza en el audiotester

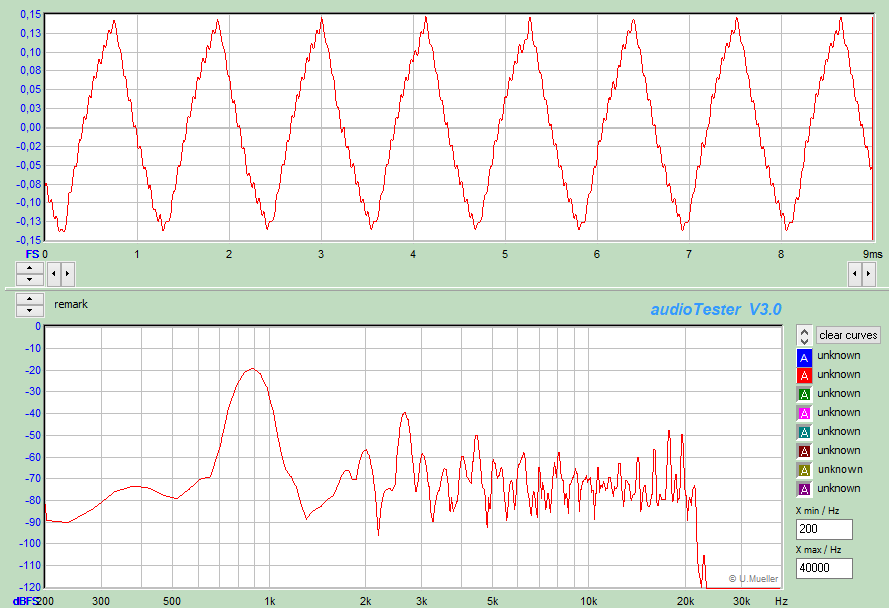


Fig. 21. Señal Triangular.

Se puede observar que en frecuencia posee un pico en la frecuencia fundamental mucho mas grande como era lo esperado con todos los armónicos impares como se está señalando.

Función Cuadrada

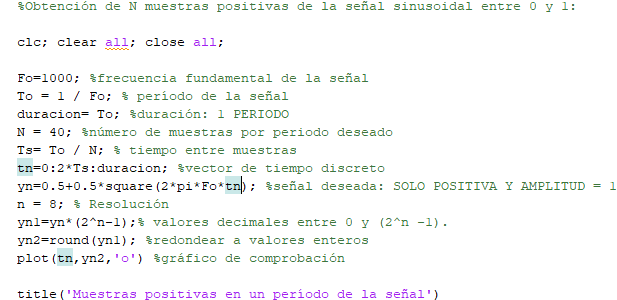


Fig. 22. Código de muestreo.



Fig. 23.Muestras para la función Cuadrada en Matlab.

Siguiente paso se crea el código en Arduino mediante un vector que este con estas muestras que se creó en Matlab, como se indica en la figura

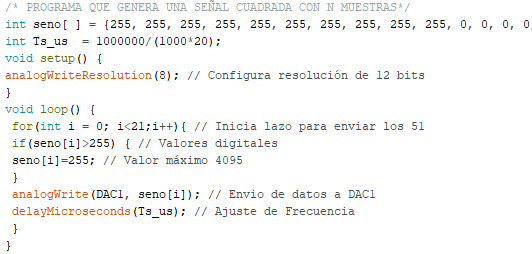


Fig. 24. Código en Arduino.

Finalmente se visualiza en el audiotester



Fig. 25. Señal Cuadrada.

1. **Generación con otro método**

**Función Senoidal:**

Mediante la funcionGenerator

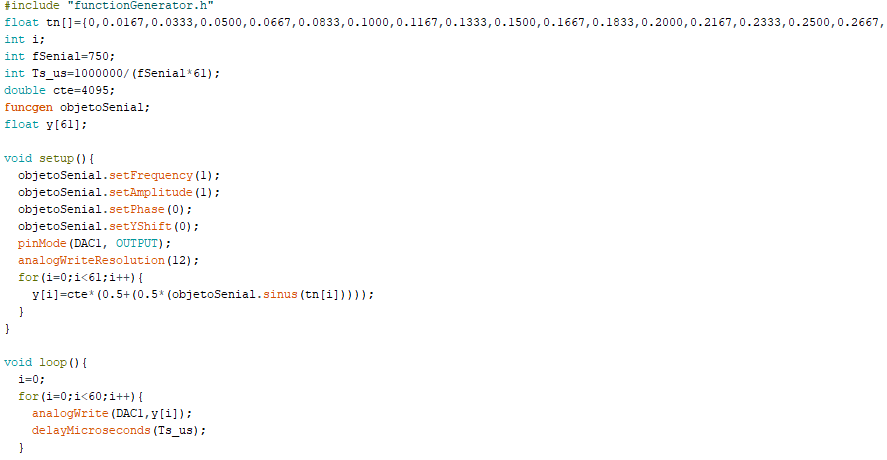


Fig. 26. Código en Arduino.

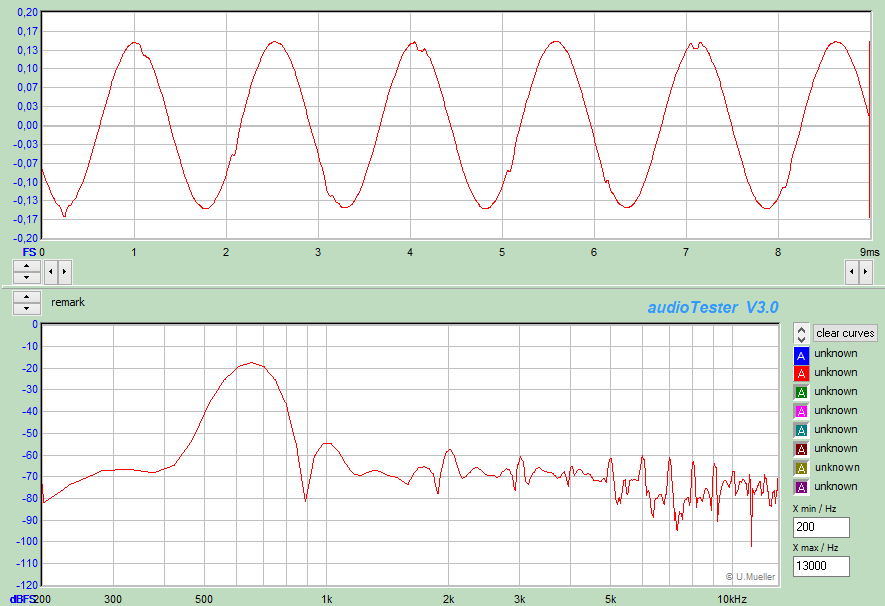
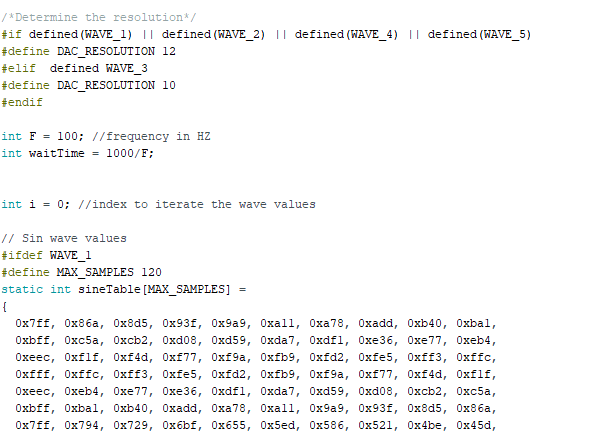
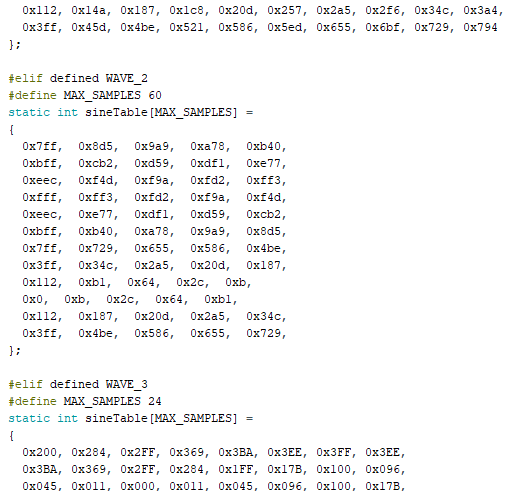


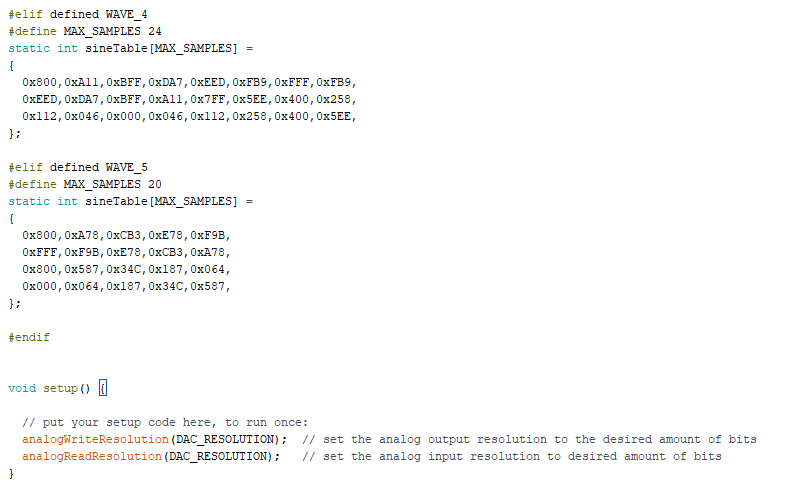
Fig. 27. Función Senoidal.

**Función cuadrada**

Método de resolución con la cual se crean mayor numero de muestras para cada resolución, pero lo innovador es que estas muestras están en hexadecimal, código encontrado en internet







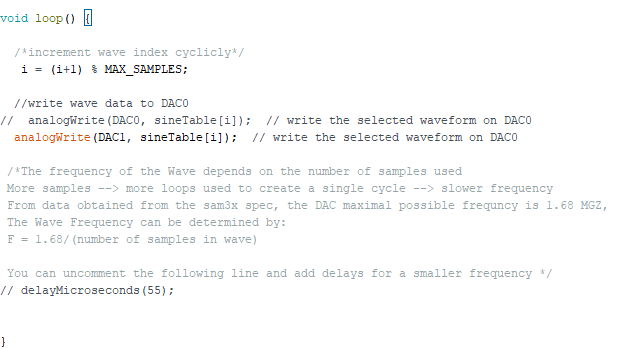




Fig. 28. Función Cuadrada.

En la gráfica se observa que a partir de los datos obtenido se tiene que la frecuencia máxima es 1.68GHz, así como en al del documento se calcula la frecuencia de onda se puede determinar mediante:

F = 1,68 / (número de muestras en onda)

**Exactitud**

**Para 750(Hz)**

**Senoidal**

**Imagen que contiene Gráfico

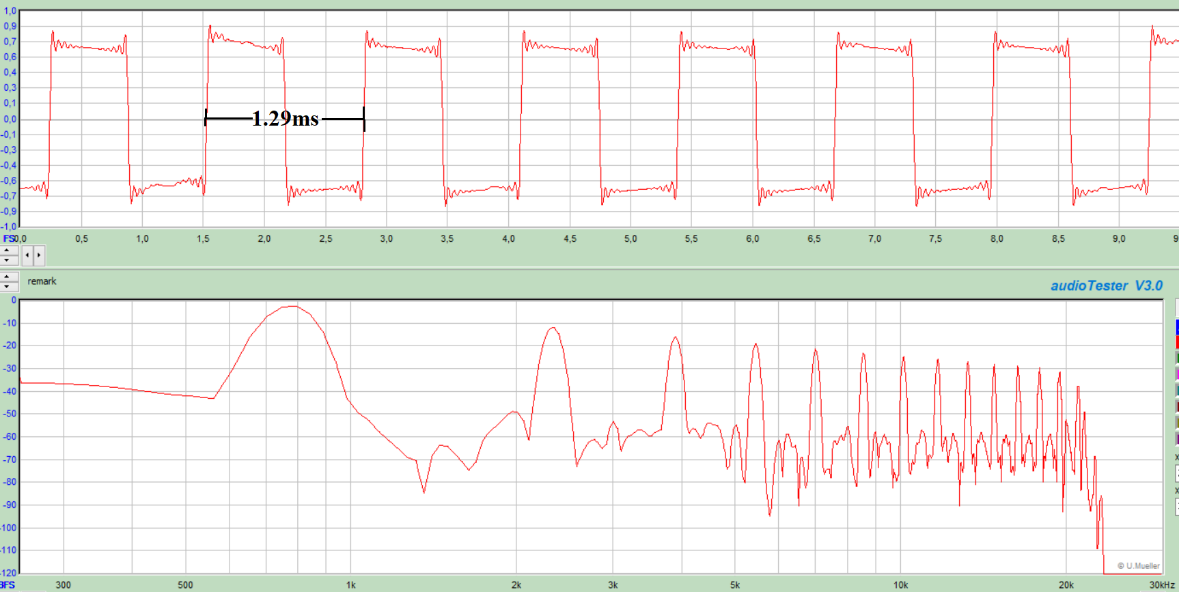
Descripción generada automáticamente**

**Triangular**

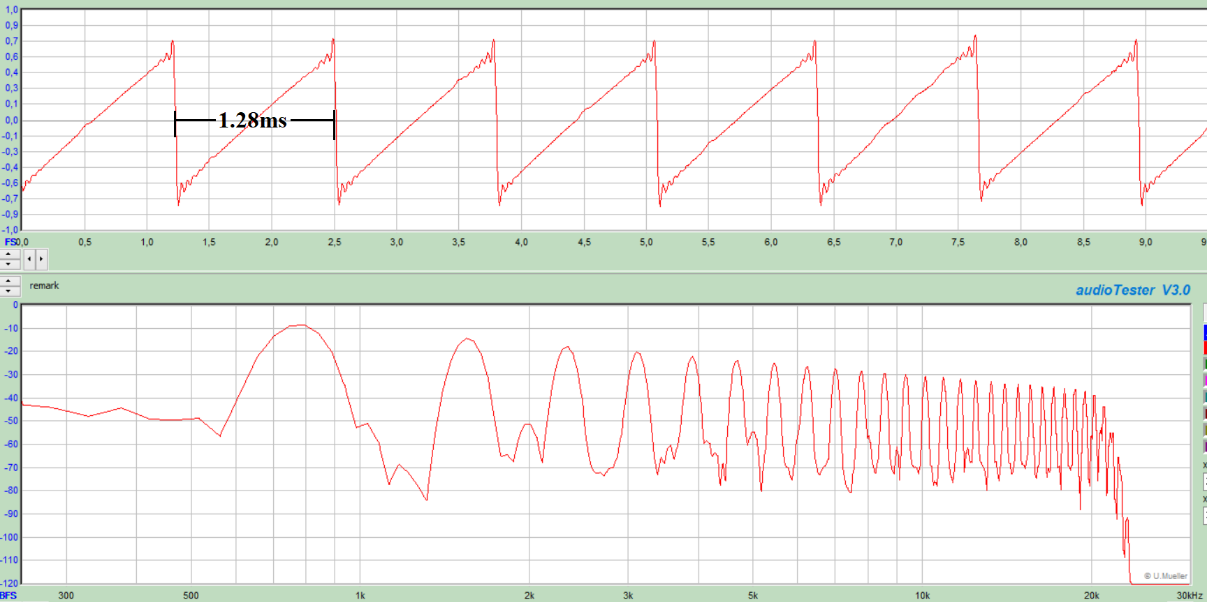
**Imagen que contiene Interfaz de usuario gráfica

Descripción generada automáticamente**

**Cuadrada**

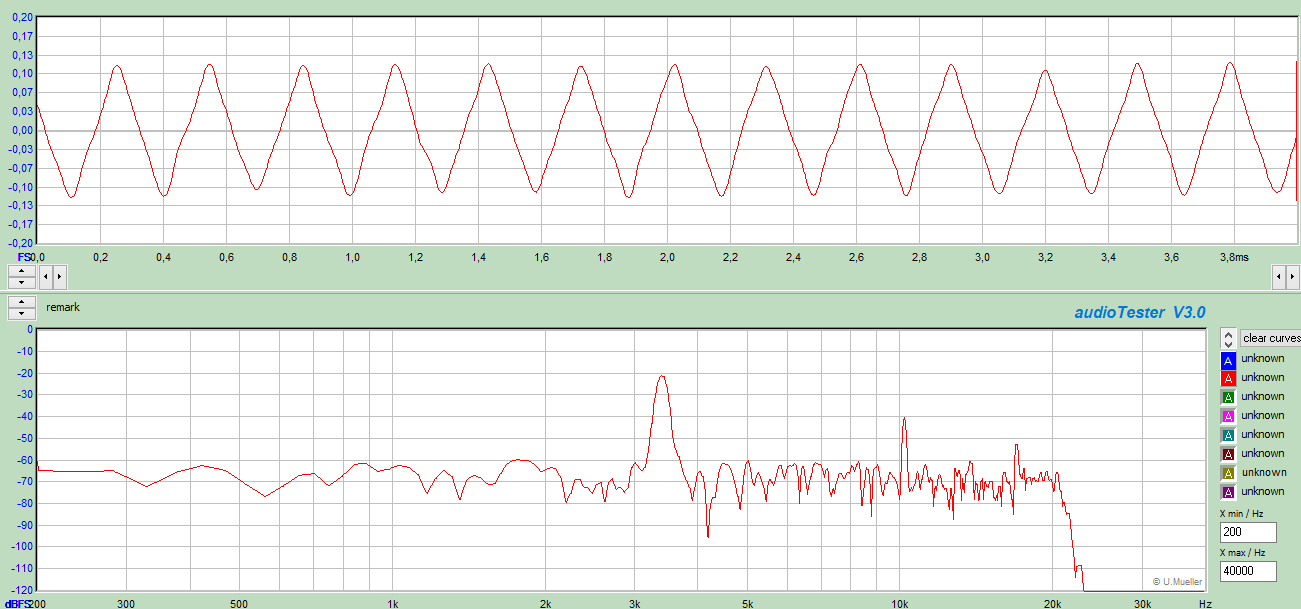


**Diente de sierra**

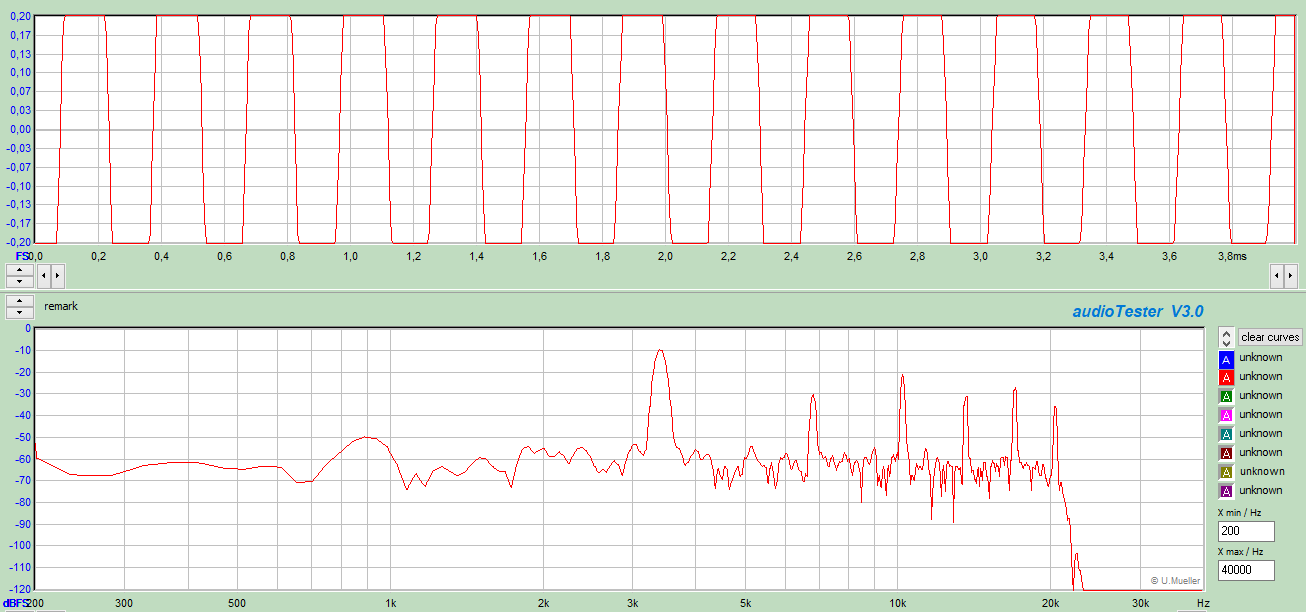


**Para 4(KHz)**

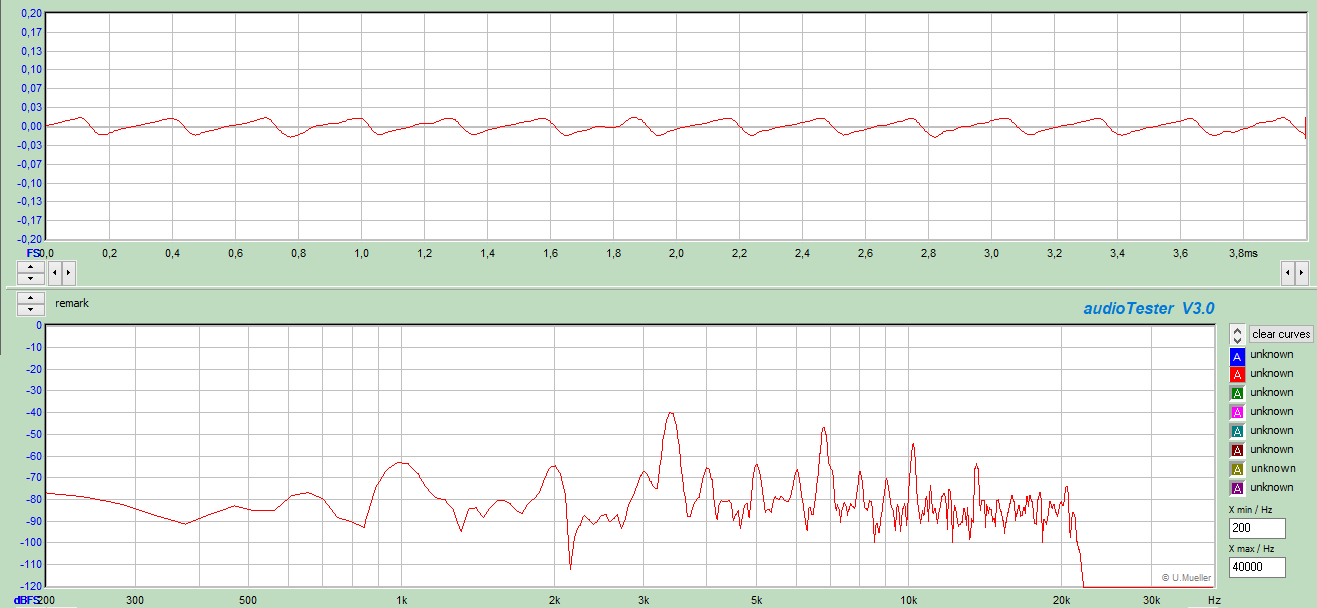
**Triangular**



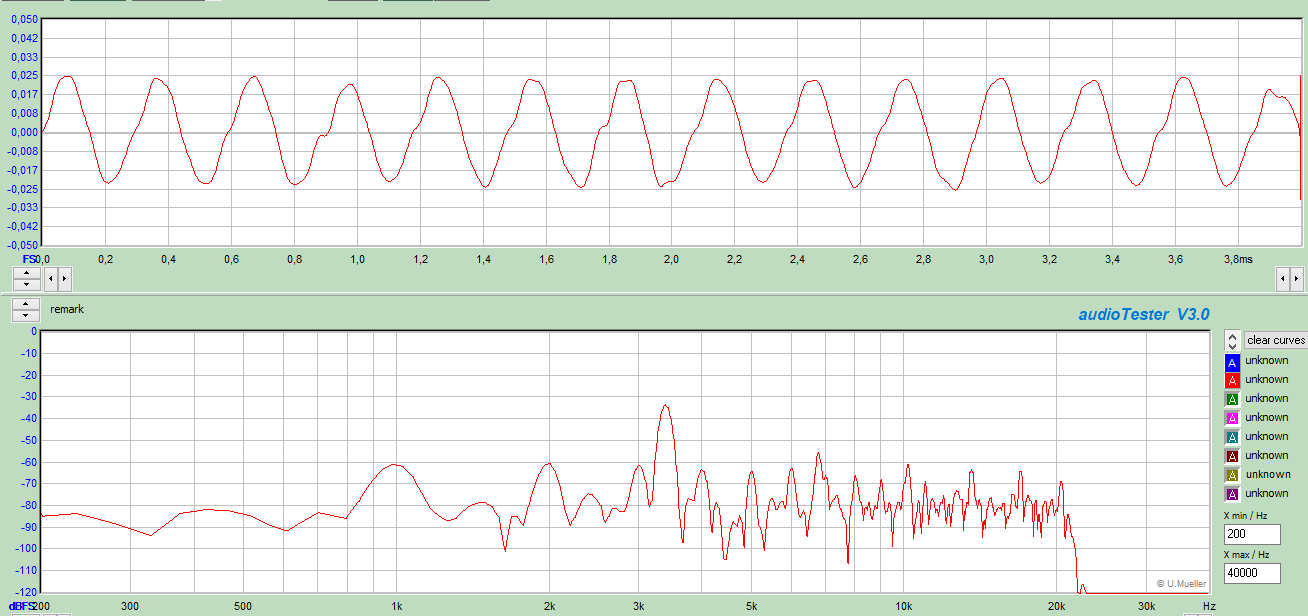
**Cuadrada**



**Diente de sierra**



**Senoidal**



|  |  |  |  |
| --- | --- | --- | --- |
| Tabla de error de exactitud | | | |
| Señal Senoidal | | | |
| Frecuencias  (Hz) | **Exactitud Real (ms)** | **Exactitud Simulada (ms)** | **Error (%)** |
| 750 | 1.33 | 1.1 | 17.2 |
| 1k | 1 | 1.28 | 28 |
| 4K | 0.25 | 0.25 | 0 |

|  |  |  |  |
| --- | --- | --- | --- |
| Tabla de error de exactitud | | | |
| Señal Triangular | | | |
| Frecuencias  (Hz) | **Exactitud Real (ms)** | **Exactitud Simulada (ms)** | **Error (%)** |
| 750 | 1.33 | 1.46 | 9.77 |
| 1k | 1 | 1 | 0 |
| 4K | 0.25 | 0.25 | 0 |

|  |  |  |  |
| --- | --- | --- | --- |
| Tabla de error de exactitud | | | |
| Señal Diente de sierra | | | |
| Frecuencias  (Hz) | **Exactitud Real (ms)** | **Exactitud Simulada (ms)** | **Error (%)** |
| 750 | 1.33 | 1.28 | 3.75 |
| 1k | 1 | 1 | 0 |
| 4K | 0.25 | 0.27 | 8 |

|  |  |  |  |
| --- | --- | --- | --- |
| Tabla de error de exactitud | | | |
| Señal Cuadrada | | | |
| Frecuencias  (Hz) | **Exactitud Real (ms)** | **Exactitud Simulada (ms)** | **Error (%)** |
| 750 | 1.33 | 1.29 | 3 |
| 1k | 1 | 1.08 | 8 |
| 4K | 0.25 | 0.33 | 32 |

Comentario el método mas efectivo es por las muestras especificado en el documento ya que al tener mayor cantidad de muestras se puede tener mas punto al dibujar y generar la onda lo que ayuda a disminuir el error y que la gráfica sea exacta