

Informe II – FFT e IFFT

Dylan Ferney Vasquez Rojas
1401597
Comunicaciones Digitales

Resumen— En esta práctica se analizó el espectro en frecuencia de distintas señales periódicas mediante la Transformada Rápida de Fourier (FFT) y se reconstruyeron en el dominio del tiempo utilizando la IFFT en MATLAB. A partir de archivos CSV con datos espectrales, se evidenció cómo los diferentes armónicos influyen en la forma de la señal. La actividad permitió comprender la relación entre los dominios temporal y frecuencial, y la importancia del procesamiento digital en el análisis de señales.

Abstract— In this practice, the frequency spectrum of different periodic signals was analyzed using the Fast Fourier Transform (FFT) and reconstructed in the time domain using the IFFT in MATLAB. CSV files with spectral data demonstrate how different harmonics influence the shape of the signal. The activity provided an understanding of the relationship between the time and frequency domains and the importance of digital processing in signal analysis.

I. INTRODUCCIÓN

El análisis de señales periódicas mediante la descomposición en series de Fourier constituye una herramienta fundamental en el estudio de sistemas electrónicos y de telecomunicaciones. Esta técnica permite representar cualquier señal periódica como la suma de componentes senoidales (armónicos), cada uno con una frecuencia, amplitud y fase específicas. Esta representación facilita el análisis del comportamiento de señales complejas, especialmente en el dominio de la frecuencia.

En este informe se estudian diferentes formas de onda, tales como señales senoidales, cuadradas, triangulares y pulsos, desde un enfoque teórico y experimental. A través del cálculo de los coeficientes de Fourier, se determinan las magnitudes de los armónicos que componen cada señal. Posteriormente, se realiza una comparación entre los valores teóricos y los obtenidos experimentalmente mediante instrumentos de medición, evaluando las discrepancias y su posible origen.

DESARROLLO DE LA PRÁCTICA

La presente práctica tuvo como objetivo analizar el comportamiento espectral de señales periódicas a través de la descomposición en armónicos mediante series de Fourier y Transformada Rápida de Fourier (FFT). Se trabajó con señales senoidal, cuadrada, triangular y de pulso, generadas en laboratorio y capturadas con un osciloscopio digital Tektronix, empleando la función MATH/FFT para su visualización en el dominio de la frecuencia.

Se activó la función FFT en el osciloscopio para cada señal. Posteriormente, se almacenaron los datos espectrales en archivos .CSV y se abrieron en Bloc de notas, MATLAB y Python para examinar su estructura. Además de los datos de frecuencia y magnitud, los archivos incluyen encabezados con información del canal, tasa de muestreo (1 024 muestras: número total de puntos capturados de la señal, 4.93213367 μ s (microsegundos): tiempo entre cada muestra.), fecha y configuración de adquisición, lo cual permitió entender el contexto de la medición.

II. Captura y exploración de datos

Se activó la función FFT en el osciloscopio para cada señal. Posteriormente, se almacenaron los datos espectrales en archivos .CSV y se abrieron en Bloc de notas, MATLAB y Python para examinar su estructura. Además de los datos de frecuencia y magnitud, los archivos incluyen encabezados con información del canal, tasa de muestreo, fecha y configuración de adquisición, lo cual permitió entender el contexto de la medición.

Para cada tipo de señal, se tabularon los primeros armónicos experimentales (frecuencia y magnitud), medidos directamente desde la visualización FFT del osciloscopio.

Señal Seno 1						
Armonico	An (Teorico)	An (Exp)	Magnitud (Exp) dB	Magnitud (Teorica) dB	Frecuencia	
1	1.077	1.072	0.61	0	1000	

Ilustración. 1. Tabla de la Señal Seno 1.

Señal Seno 2						
Armonico	An	An (Exp)	Magnitud (Exp) dB	Magnitud (Teorica) dB	Frecuencia	
0	1.027	1.026	0.23	0	0	
1	1.077	1.075	0.63	0	1000	

Ilustración. 2. Tabla de la Señal Seno 2.

Señal Triangular 1						
Armonico	An	An (Exp)	Magnitud (Exp) dB	Magnitud (Teorica) dB	Frecuencia	% De Error Magnitud
1	1.21	1.16	1.3	1.69	1000	23.08
3	0.13	0.097	-20.2	-17.38	3000	16.23
5	0.048	0.033	-29.4	-26.26	5000	11.96
7	0.024	0.017	-35	-32.1	7000	9.03
9	0.015	0.0093	-40.6	-36.47	9000	11.32
11	0.01	0.007	-42.2	-39.95	11000	5.63

Ilustración. 3. Tabla de la Señal Triangular 1.

Señal Triangular 2						
Armonico	An	An (Exp)	Magnitud (Exp) dB	Magnitud (Teorica) dB	Frecuencia	% De Error Magnitud
0	1.49	1.48	3.43	3.52	0	2.6
1	1.21	1.18	1.47	1.69	1000	13.0
3	0.13	0.097	-20.2	-17.38	3000	16.2
5	0.048	0.033	-29.4	-26.26	5000	12.0
7	0.024	0.018	-35	-32.1	7000	9.0
9	0.015	0.0093	-40.6	-36.47	9000	11.3
11	0.01	0.0077	-42.2	-39.95	11000	5.6

Ilustración. 4. Tabla de la Señal Triangular 2.

Señal Cuadrada 1						
Armonico	An	An (Exp)	Magnitud (Exp) dB	Magnitud (Teórica) dB	Frecuencia	% De Error Magnitud
1	1.9	1.32	4.2	5.62	1000	25.27
3	0.63	0.56	-4.97	-3.92	3000	26.79
5	0.38	0.27	-11.4	-8.35	5000	36.53
7	0.27	0.19	-14.2	-11.28	7000	25.89
9	0.21	0.15	-16.6	-13.46	9000	23.33
11	0.17	0.12	-18.2	-15.2	11000	19.74

Ilustración. 5. Tabla de la Señal Cuadrada 1.

Señal Cuadrada 2						
Armonico	An	An (Exp)	Magnitud (Exp) dB	Magnitud (Teórica) dB	Frecuencia	% De Error Magnitud
0	1.25	1.16	1.33	1.94	0	31.44
1	1.9	1.32	5.1	5.62	1000	9.25
3	0.63	0.56	-3.97	-3.92	3000	1.28
5	0.38	0.27	-11.4	-8.35	5000	36.53
7	0.27	0.19	-14.2	-11.28	7000	25.89
9	0.21	0.15	-16.6	-13.46	9000	23.33
11	0.17	0.12	-18.2	-15.2	11000	19.74

Ilustración. 6. Tabla de la Señal Cuadrada 2.

Señal Pulso 1						
Armonico	An	An (Exp)	Magnitud (Exp) dB	Magnitud (Teórica) dB	Frecuencia	% De Error Magnitud
1	0.95	1.16	1.3	-0.44	1000	19.5
3	-0.31	0.44	-6.99	-9.95	3000	29.7
5	0.19	0.28	-11	-14.42	5000	23.7
7	-0.13	0.19	-14.2	-17.72	7000	19.9
9	0.1	0.15	-16.2	-20	9000	19.0
11	-0.08	0.12	-18.2	-21.93	11000	17.0

Ilustración. 7. Tabla de la Señal Pulso 1.

Señal Pulso 2						
Armonico	An	An (Exp)	Magnitud (Exp) dB	Magnitud (Teórica) dB	Frecuencia	% De Error Magnitud
0	0.3	0.34	-9.19	-10.46	0	12.14
1	0.56	0.59	-4.47	-5.04	1000	11.31
2	0.45	0.51	-5.79	-6.94	2000	16.57
3	0.3	0.42	-7.39	-10.46	3000	29.35
4	0.14	0.18	-14.6	-17.08	4000	14.52
5	0	0.018	-34.6	-	5000	-
6	-0.093	-0.14	-16.6	-20.63	6000	19.54
7	-0.12	-0.18	-14.6	-18.42	7000	20.72
8	-0.11	-0.16	-15.8	-19.17	8000	17.59
9	-0.062	-0.089	-21	-24.15	9000	13.05

Ilustración. 8. Tabla de la Señal Pulso 2.

Señal Pulso 3						
Armonico	An	An (Exp)	Magnitud (Exp) dB	Magnitud (Teórica) dB	Frecuencia	% De Error Magnitud
0	0.45	0.4	-7.79	-6.93	0	12.41
1	0.77	0.8	-1.93	-2.27	1000	14.98
2	0.45	0.51	-5.79	-6.94	2000	16.52
3	0.098	0.117	-18.6	-20.18	3000	7.81
4	-0.14	-0.22	-13	-17.08	4000	23.88
5	-0.19	-0.26	-11.4	-14.42	5000	20.97
6	-0.093	-0.12	-18.2	-20.63	6000	11.78
7	-0.042	-0.071	-23	-27.54	7000	16.47
8	0.113	0.15	-16.2	-18.94	8000	14.46
9	0.085	0.12	-17.8	-21.41	9000	16.87

Ilustración. 9. Tabla de la Señal Pulso 3.

Señal Pulso 4						
Armonico	An	An (Exp)	Magnitud (Exp) dB	Magnitud (Teórica) dB	Frecuencia	% De Error Magnitud
0	1.2	1.12	1.01	1.58	0	36.22
1	0.56	0.61	-4.3	-5.04	1000	14.62
2	-0.45	-0.64	-3.79	-6.94	2000	45.36
3	0.3	0.44	-6.99	-10.46	3000	33.16
4	-0.14	-0.16	-15.4	-17.08	4000	9.82
5	0	0.03	-29	-	5000	-
6	0.093	0.15	-16.2	-20.63	6000	21.47
7	-0.12	-0.19	-14.2	-18.42	7000	22.89
8	0.11	0.16	-15.8	-19.17	8000	17.59
9	-0.062	-0.09	-20.2	-24.15	9000	16.36

Ilustración. 10. Tabla de la Señal Pulso 4.

III. Series de Fourier – Desarrollo teórico

Se calcularon los coeficientes A_n de las series de Fourier para diferentes tipos de señales, usando las siguientes fórmulas generales:

Formula usada para la señal cuadrada:

$$A_n = \frac{4 \cdot V}{n \cdot \pi}$$

Formula usada para la señal triangular :

$$A_n = \frac{8 \cdot V}{(n \cdot \pi)^2}$$

Formula usada para la señal triangular :

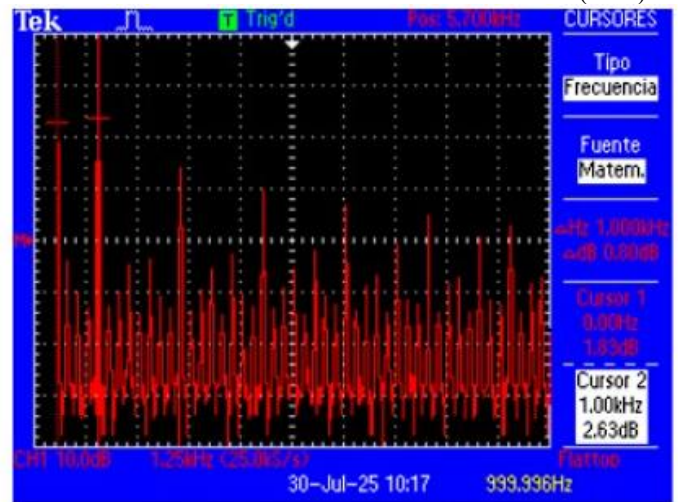
$$A_n = 2 \cdot A \cdot D \cdot \text{sinc}(n \cdot \pi \cdot D)$$

Para comparar las magnitudes con los valores teóricos se utilizó la fórmula:

$$\text{Magnitud} = 10^{\frac{\text{dB}}{20}}$$

Esto permite reconstruir la señal a partir de los armónicos medidos experimentalmente.

Luego se calculó experimentalmente la magnitud y frecuencia de cada una de las señales, para así también comprobar los armónicos como se ven en las ilustraciones anteriores (1-10)



TDS 2012C - 12:32:14 a. m. 30/07/2025
Ilustración. 11. Señal Cuadrada 2 en el osciloscopio.

Y así con cada una de las señales, estos datos se registraron en las tabulaciones, ya antes vistas desde la ilustración 1 a la 11.

Luego de determinar los coeficientes armónicos A_n mediante las fórmulas teóricas correspondientes a cada tipo de señal periódica, se elaboró un programa en MATLAB que permite reconstruir la señal original sumando una cantidad determinada de armónicos.

El principio fundamental detrás de esta reconstrucción es la serie de Fourier en su forma seno-coseno o compleja. La reconstrucción se realiza mediante la suma de los primeros N armónicos, y permite visualizar cómo se aproxima progresivamente la señal original.

Con base en los valores de amplitud obtenidos para los primeros armónicos teóricos, se desarrolló un código en MATLAB para reconstruir la señal periódica mediante la suma de los primeros cinco armónicos (1° , 5° , 7° , 9° y 11°), utilizando únicamente términos senoidales. En este caso, al tratarse de una señal puramente senoidal sin armónicos superiores, únicamente el armónico fundamental presenta una magnitud

significativa. La reconstrucción se efectuó mediante la siguiente ecuación:

$$A_1 \cdot \sin(2\pi ft) + A_5 \cdot \sin(2\pi 5 ft) + A_7 \cdot \sin(2\pi 7 ft) + A_9 \cdot \sin(2\pi 9 ft) + A_{11} \cdot \sin(2\pi 11 ft)$$

Y así se elaboró el siguiente código:

```
T = 0.1;
f = 1/T;

A1 = 1.077;
A5 = 0;
A7 = 0;
A9 = 0;
A11 = 0;
DC = 0;

t = 0:T/100:2*T;

V1 = A1 * sin(2 * pi * f * t);
V5 = A5 * sin(2 * pi * 5 * f * t);
V7 = A7 * sin(2 * pi * 7 * f * t);
V9 = A9 * sin(2 * pi * 9 * f * t);
V11 = A11 * sin(2 * pi * 11 * f * t);

VT = V1 + V5 + V7 + V9 + V11 + DC;

plot(t, VT);
title('Señal Reconstruida a partir de Armónicos Teóricos');
xlabel('Tiempo (s)');
ylabel('Amplitud');
grid on;
```

Así se visualizó la señal:

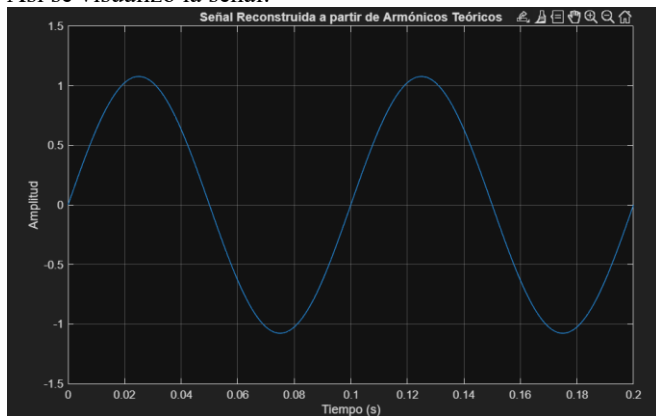


Ilustración. 12. Señal 1 Seno A Partir De Los Armónicos Teóricos.

La señal reconstruida a partir de los armónicos teóricos muestra un comportamiento suave y limpio. Esto se debe a la precisión de los cálculos realizados para los coeficientes de la serie de Fourier, así como al número limitado de armónicos considerados. La forma de onda obtenida se asemeja a la señal original medida, lo que valida la corrección del modelo teórico implementado en MATLAB.

Al comparar la señal reconstruida a partir de los armónicos teóricos con la señal medida experimentalmente, se observa

una ligera distorsión en la forma de onda reconstruida. Esta diferencia se debe principalmente a la precisión de los cálculos de los coeficientes de Fourier y a la cantidad limitada de armónicos considerados en la reconstrucción. Al utilizar solo los primeros armónicos, se pierde parte de la información de alta frecuencia presente en la señal real, lo que produce una forma menos exacta. No obstante, la señal reconstruida mantiene las características principales, validando el uso de la serie de Fourier para modelar señales periódicas.

Se reconstruyó una señal senoidal con componente DC a partir de armónicos teóricos. Solo se consideró el primer armónico con amplitud 1.077 y un componente DC de 1.027; los demás armónicos fueron nulos.

El gráfico muestra una señal limpia, desplazada verticalmente por el valor DC. La precisión de los cálculos evita distorsiones, representando fielmente la forma esperada.

```
T = 0.1;
f = 1/T;
```

```
A1 = 1.077;
A5 = 0;
A7 = 0;
A9 = 0;
A11 = 0;
```

```
DC = 1.027;
```

```
t = 0:T/100:2*T;
```

```
V1 = A1 * sin(2 * pi * f * t);
V5 = A5 * sin(2 * pi * 5 * f * t);
V7 = A7 * sin(2 * pi * 7 * f * t);
V9 = A9 * sin(2 * pi * 9 * f * t);
V11 = A11 * sin(2 * pi * 11 * f * t);
```

```
VT = V1 + V5 + V7 + V9 + V11 + DC;
```

```
plot(t, VT);
title('Señal Reconstruida a partir de Armónicos Teóricos');
xlabel('Tiempo (s)');
ylabel('Amplitud');
grid on;
```

La señal se visualiza así:

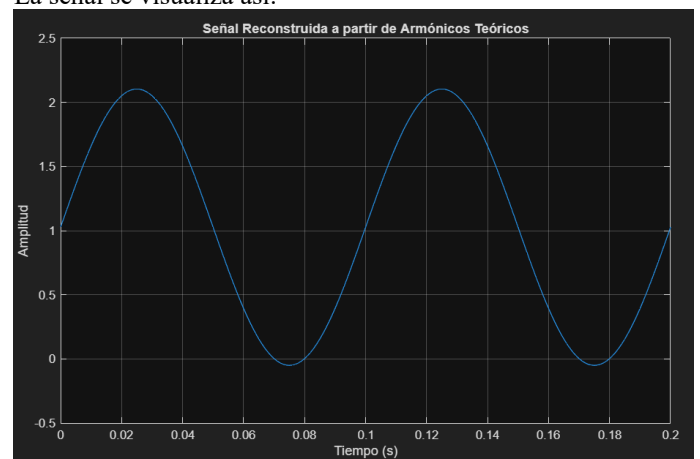


Ilustración. 13. Señal 2 Seno A Partir De Los Armónicos Teóricos.

Se reconstruyó la primera señal triangular empleando únicamente armónicos impares con amplitudes calculadas teóricamente:

```
T = 0.1;
f = 1/T;
```

```
A1 = 1.21;
A3 = 0.13;
A5 = 0.048;
A7 = 0.024;
A9 = 0.015;
A11 = 0.01;
```

```
t = 0:T/100:2*T;
```

```
V1 = A1 * cos(2 * pi * f * t);
V3 = A3 * cos(2 * pi * 3 * f * t);
V5 = A5 * cos(2 * pi * 5 * f * t);
V7 = A7 * cos(2 * pi * 7 * f * t);
V9 = A9 * cos(2 * pi * 9 * f * t);
V11 = A11 * cos(2 * pi * 11 * f * t);
```

```
VT = V1 + V3 + V5 + V7 + V9 + V11;
```

```
figure;
plot(t, VT);
title('Señal Triangular Reconstruida a partir de Armónicos Teóricos');
xlabel('Tiempo (s)');
ylabel('Amplitud');
grid on;
```

Dando como resultado la siguiente gráfica:

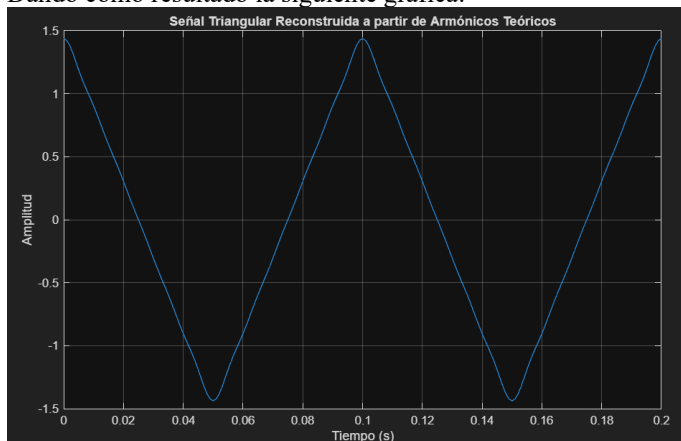


Ilustración. 14. Señal 1 Triangular A Partir De Los Armónicos Teóricos

Luego se realizó el código de la siguiente señal triangular:

```
T = 0.1;
f = 1/T;
```

```
A0 = 1.49;
A1 = 1.21;
A3 = 0.13;
A5 = 0.048;
```

```
A7 = 0.024;
A9 = 0.015;
A11 = 0.01;
```

```
t = 0:T/100:2*T;
```

```
V0 = A0 * ones(size(t));
V1 = A1 * cos(2 * pi * f * t);
V3 = A3 * cos(2 * pi * 3 * f * t);
V5 = A5 * cos(2 * pi * 5 * f * t);
V7 = A7 * cos(2 * pi * 7 * f * t);
V9 = A9 * cos(2 * pi * 9 * f * t);
V11 = A11 * cos(2 * pi * 11 * f * t);
```

```
VT = V0 + V1 + V3 + V5 + V7 + V9 + V11;
```

```
figure;
plot(t, VT);
title('Señal Reconstruida a partir de Armónicos Teóricos');
xlabel('Tiempo (s)');
ylabel('Amplitud');
grid on;
```

Obteniendo la siguiente gráfica:

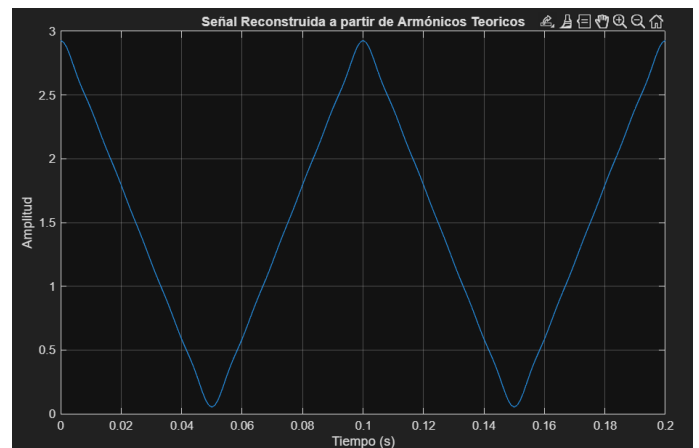


Ilustración. 15. Señal 1 Triangular A Partir De Los Armónicos Teóricos

Luego se procedió a realizar el código de la primera señal cuadrada:

```
T = 0.1;
f = 1/T;
```

```
A1 = 1.9;
A3 = 0.63;
A5 = 0.38;
A7 = 0.27;
A9 = 0.21;
A11 = 0.17;
```

```
t = 0:T/100:2*T;
```

```
V1 = A1 * sin(2 * pi * f * t);
V3 = A3 * sin(2 * pi * 3 * f * t);
```

```

V5 = A5 * sin(2 * pi * 5 * f * t);
V7 = A7 * sin(2 * pi * 7 * f * t);
V9 = A9 * sin(2 * pi * 9 * f * t);
V11 = A11 * sin(2 * pi * 11 * f * t);

V_square = V1 + V3 + V5 + V7 + V9 + V11;

offset = 0;
V_square_offset = V_square + offset;

figure;
plot(t, V_square_offset);
title('Señal Cuadrada Reconstruida con Armónicos');
xlabel('Tiempo (s)');
ylabel('Amplitud');
grid on;

```

Obteniendo como resultante la siguiente señal:

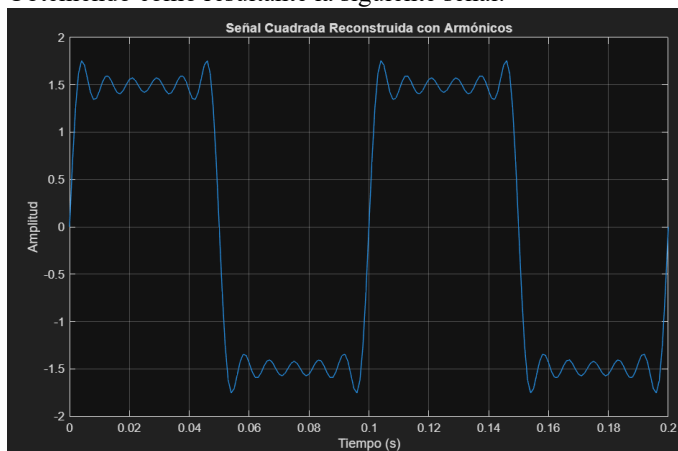


Ilustración. 16. Señal 1 Cuadrada A Partir De Los Armónicos Teóricos

La señal cuadrada reconstruida muestra las características típicas de este tipo de onda, con transiciones abruptas entre niveles altos y bajos. Sin embargo, debido a la precisión limitada en los cálculos de los coeficientes armónicos, se pueden observar ligeras distorsiones y ondulaciones, especialmente en los flancos.

Luego se prosiguió a realizar la segunda señal cuadrada.

```

% Parámetros de la señal
T = 0.1;      % Período (s)
f = 1 / T;    % Frecuencia fundamental (Hz)

% Amplitudes de los armónicos (incluye componente DC)
A0 = 1.25;    % Componente DC
A1 = 1.9;     % Armónico 1
A3 = 0.63;    % Armónico 3
A5 = 0.38;    % Armónico 5
A7 = 0.27;    % Armónico 7
A9 = 0.21;    % Armónico 9
A11 = 0.17;   % Armónico 11

```

```

% Vector de tiempo: 2 períodos con 100 muestras por período
t = linspace(0, 2*T, 2*100 + 1);

```

```

% Componentes de la señal

```

```

V0 = A0 * ones(size(t));
V1 = A1 * sin(2 * pi * 1 * f * t);
V3 = A3 * sin(2 * pi * 3 * f * t);
V5 = A5 * sin(2 * pi * 5 * f * t);
V7 = A7 * sin(2 * pi * 7 * f * t);
V9 = A9 * sin(2 * pi * 9 * f * t);
V11 = A11 * sin(2 * pi * 11 * f * t);

% Reconstrucción de la señal cuadrada
V_square = V0 + V1 + V3 + V5 + V7 + V9 + V11;

% Gráfica
figure;
plot(t, V_square, 'LineWidth', 1.2);
title('Señal Cuadrada Reconstruida con Armónicos y Componente DC');
xlabel('Tiempo (s)');
ylabel('Amplitud');
grid on;

```

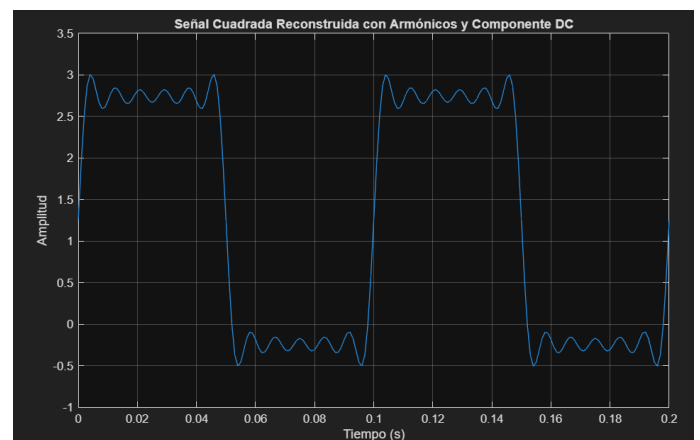


Ilustración. 17. Señal 2 Cuadrada A Partir De Los Armónicos Teóricos

Luego trabajamos la señal de pulso, siendo la primera la siguiente:

```

T = 0.1;
f = 1/T;

A1 = 0.95;
A3 = -0.31;
A5 = 0.19;
A7 = -0.13;
A9 = 0.10;
A11 = -0.08;

t = 0:T/100:2*T;

V1 = A1 * cos(2 * pi * f * t);
V3 = A3 * cos(2 * pi * 3 * f * t);
V5 = A5 * cos(2 * pi * 5 * f * t);
V7 = A7 * cos(2 * pi * 7 * f * t);
V9 = A9 * cos(2 * pi * 9 * f * t);
V11 = A11 * cos(2 * pi * 11 * f * t);

V_pulse = V1 + V3 + V5 + V7 + V9 + V11;

figure;

```

```

plot(t, V_pulse);
title('Señal de Pulso Reconstruida (Duty Cycle 50%)');
xlabel('Tiempo (s)');
ylabel('Amplitud');
grid on;

```

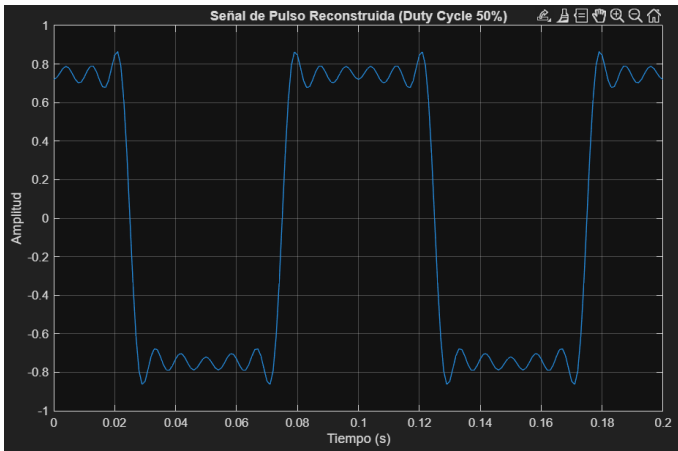


Ilustración. 18. Señal 1 Pulso A Partir De Los Armónicos Teóricos

Ahora graficaremos la señal 2 de pulso:

$T = 0.1;$

$f = 1/T;$

$A_0 = 0.3;$

$A_1 = 0.56;$

$A_2 = 0.45;$

$A_3 = 0.3;$

$A_4 = 0.14;$

$A_5 = 0;$

$A_6 = -0.093;$

$A_7 = -0.12;$

$A_8 = -0.11;$

$A_9 = -0.062;$

$t = 0:T/100:2*T;$

$V_0 = A_0;$

$V_1 = A_1 * \cos(2 * \pi * f * t);$

$V_2 = A_2 * \cos(2 * \pi * 2 * f * t);$

$V_3 = A_3 * \cos(2 * \pi * 3 * f * t);$

$V_4 = A_4 * \cos(2 * \pi * 4 * f * t);$

$V_5 = A_5 * \cos(2 * \pi * 5 * f * t);$

$V_6 = A_6 * \cos(2 * \pi * 6 * f * t);$

$V_7 = A_7 * \cos(2 * \pi * 7 * f * t);$

$V_8 = A_8 * \cos(2 * \pi * 8 * f * t);$

$V_9 = A_9 * \cos(2 * \pi * 9 * f * t);$

$V_pulse_20 = V_0 + V_1 + V_2 + V_3 + V_4 + V_5 + V_6 + V_7 + V_8 + V_9;$

```

figure;
plot(t, V_pulse_20);
title('Señal de Pulso Reconstruida con Duty Cycle 20%');
xlabel('Tiempo (s)');
ylabel('Amplitud');
grid on;

```

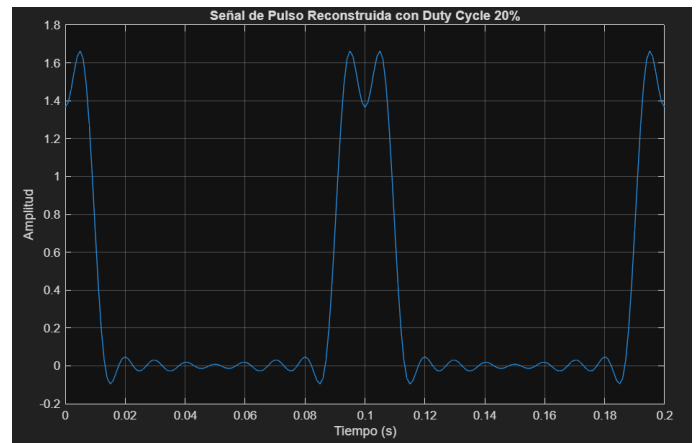


Ilustración. 19. Señal 2 Pulso A Partir De Los Armónicos Teóricos

Ahora seguiremos con la señal del pulso 3:

$T = 0.1;$

$f = 1/T;$

$A_0 = 0.45;$

$A_1 = 0.77;$

$A_2 = 0.45;$

$A_3 = 0.098;$

$A_4 = -0.14;$

$A_5 = -0.19;$

$A_6 = -0.093;$

$A_7 = -0.042;$

$A_8 = 0.113;$

$A_9 = 0.085;$

$t = 0:T/100:2*T;$

$V_0 = A_0;$

$V_1 = A_1 * \cos(2 * \pi * f * t);$

$V_2 = A_2 * \cos(2 * \pi * 2 * f * t);$

$V_3 = A_3 * \cos(2 * \pi * 3 * f * t);$

$V_4 = A_4 * \cos(2 * \pi * 4 * f * t);$

$V_5 = A_5 * \cos(2 * \pi * 5 * f * t);$

$V_6 = A_6 * \cos(2 * \pi * 6 * f * t);$

$V_7 = A_7 * \cos(2 * \pi * 7 * f * t);$

$V_8 = A_8 * \cos(2 * \pi * 8 * f * t);$

$V_9 = A_9 * \cos(2 * \pi * 9 * f * t);$

$V_pulse_30 = V_0 + V_1 + V_2 + V_3 + V_4 + V_5 + V_6 + V_7 + V_8 + V_9;$

```

figure;
plot(t, V_pulse_30);
title('Señal de Pulso Reconstruida con Duty Cycle 30%');
xlabel('Tiempo (s)');
ylabel('Amplitud');
grid on;

```

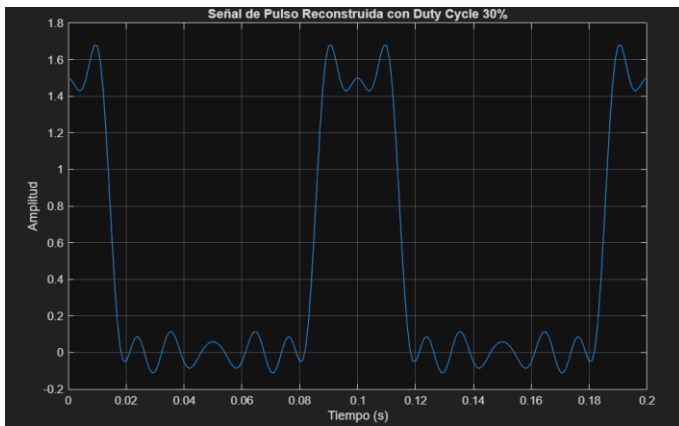



Ilustración. 20. Señal 3 Pulso A Partir De Los Armónicos Teóricos

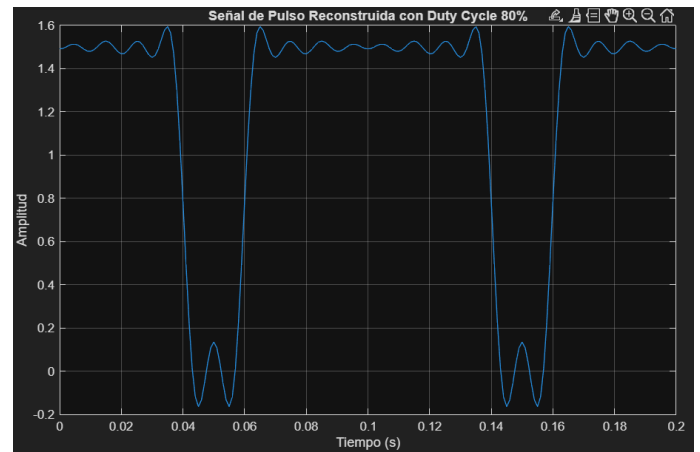


Ilustración. 21. Señal 4 Pulso A Partir De Los Armónicos Teóricos

Y para finalizar se realizó la señal de pulso 4:

$$T = 0.1;$$

$$f = 1/T;$$

$$A_0 = 1.2;$$

$$A_1 = 0.56;$$

$$A_2 = -0.45;$$

$$A_3 = 0.3;$$

$$A_4 = -0.14;$$

$$A_5 = 0;$$

$$A_6 = 0.093;$$

$$A_7 = -0.12;$$

$$A_8 = 0.11;$$

$$A_9 = -0.062;$$

$$t = 0:T/100:2*T;$$

$$V_0 = A_0;$$

$$V_1 = A_1 * \cos(2 * \pi * f * t);$$

$$V_2 = A_2 * \cos(2 * \pi * 2 * f * t);$$

$$V_3 = A_3 * \cos(2 * \pi * 3 * f * t);$$

$$V_4 = A_4 * \cos(2 * \pi * 4 * f * t);$$

$$V_5 = A_5 * \cos(2 * \pi * 5 * f * t);$$

$$V_6 = A_6 * \cos(2 * \pi * 6 * f * t);$$

$$V_7 = A_7 * \cos(2 * \pi * 7 * f * t);$$

$$V_8 = A_8 * \cos(2 * \pi * 8 * f * t);$$

$$V_9 = A_9 * \cos(2 * \pi * 9 * f * t);$$

$$V_pulse_80 = V_0 + V_1 + V_2 + V_3 + V_4 + V_5 + V_6 + V_7 + V_8 + V_9;$$

figure;

plot(t, V_pulse_80);

title('Señal de Pulso Reconstruida con Duty Cycle 80%');

xlabel('Tiempo (s)');

ylabel('Amplitud');

grid on;

Obteniendo así la siguiente gráfica:

Posteriormente, se desarrolló un programa en MATLAB con el objetivo de visualizar las gráficas correspondientes a las señales previamente analizadas. Estas señales no fueron generadas por simulación, sino que fueron obtenidas experimentalmente mediante el software OpenChoice Desktop.

Los datos fueron exportados desde el osciloscopio en formato .csv (valores separados por comas), lo cual permitió su posterior tratamiento en MATLAB. A partir de estos archivos, se representaron gráficamente las señales reales para su comparación con las reconstrucciones teóricas obtenidas mediante la serie de Fourier.

Esta comparación es útil para validar la precisión del análisis y comprender el comportamiento real de las señales en el dominio del tiempo y frecuencia.

Primero se graficaron todas las señales en el dominio del tiempo.

Empezando por la señal 1 del seno, se usaron los siguientes comandos en el apartado de Command Window.

```
t=S1ST.x_0_00286;
```

```
x=S1ST.x1_18;
```

```
plot(t,x)
```

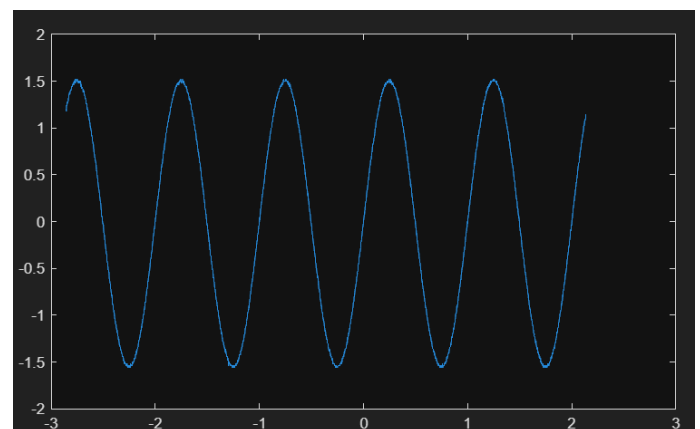


Ilustración. 22. Señal 1 Seno A Partir Del CSV

Señal 2 del seno, se usaron los siguientes comandos en el apartado de Command Window.

```
t=S2ST.x_0_0025;  
x=S2ST.x0_88;  
plot(t,x)
```

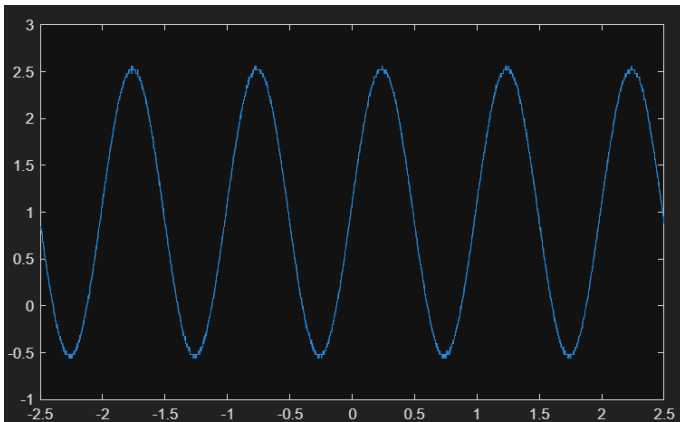


Ilustración. 23. Señal 2 Seno A Partir Del CSV

Señal 1 Triangular, se usaron los siguientes comandos en el apartado de Command Window.

```
t=S1TT.x_0_0025;  
x=S1TT.x_0_08;  
plot(t,x)
```

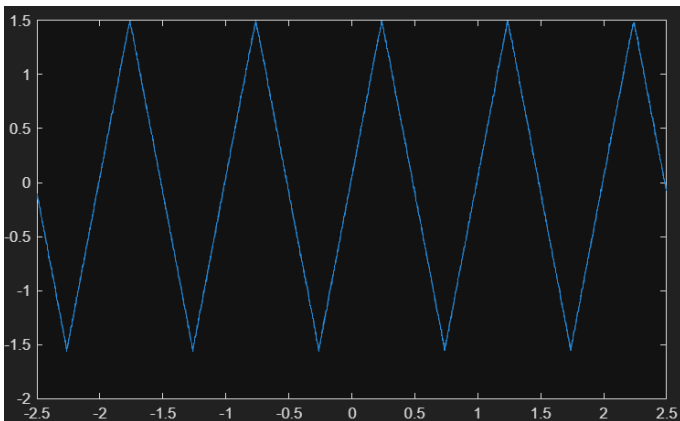


Ilustración. 24. Señal 1 Triangular A Partir Del CSV

Señal 2 Triangular, se usaron los siguientes comandos en el apartado de Command Window.

```
t=S2TT.x_0_0025;  
x=S2TT.x1_36;  
plot(t,x)
```

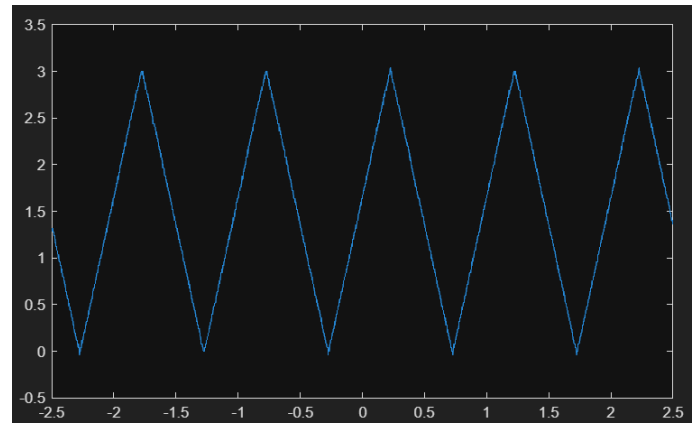


Ilustración. 25. Señal 2 Triangular A Partir Del CSV

Señal 1 Cuadrado, se usaron los siguientes comandos en el apartado de Command Window.

```
t=S1CT.x_0_0025;  
x=S1CT.x1_5;  
plot(t,x)
```

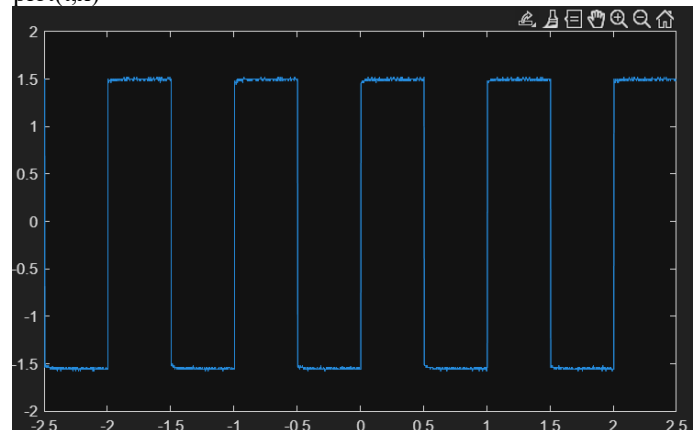


Ilustración. 26. Señal 1 Cuadrada A Partir Del CSV

Señal 2 Cuadrado, se usaron los siguientes comandos en el apartado de Command Window.

```
t=S2CT.x_0_0025;  
x=S2CT.x2_8;  
plot(t,x)
```

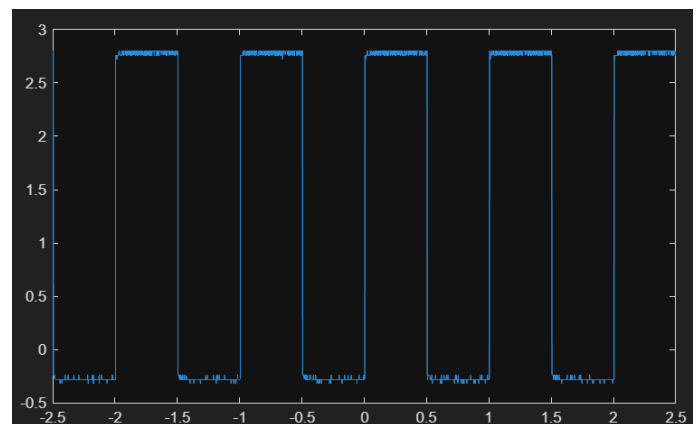


Ilustración. 27. Señal 1 Cuadrada A Partir Del CSV

Señal 1 Pulso, se usaron los siguientes comandos en el apartado de Command Window.

```
t=S1PT.x_0_0025;
x=S1PT.x1_48;
plot(t,x)
```

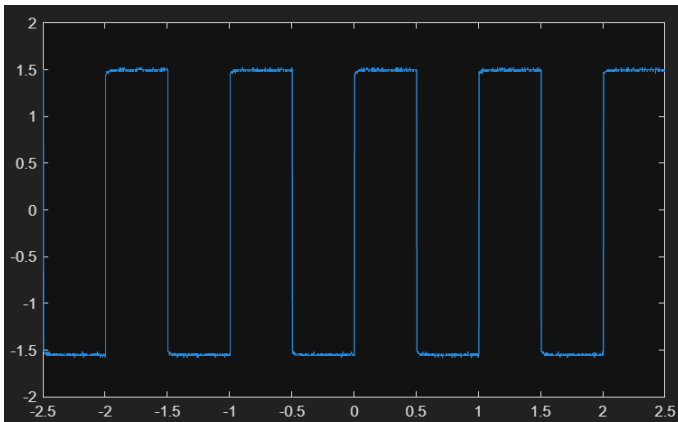


Ilustración. 28. Señal 1 Pulso A Partir Del CSV

Señal 2 Pulso, se usaron los siguientes comandos en el apartado de Command Window.

```
t=S2PT.x_0_0025;
x=S2PT.x_1_56;
plot(t,x)
```

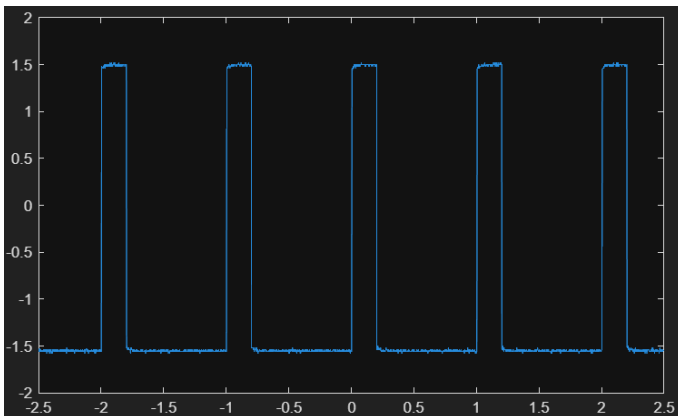


Ilustración. 29. Señal 1 Pulso A Partir Del CSV

Señal 3 Pulso, se usaron los siguientes comandos en el apartado de Command Window.

```
t=S3PT.x_0_0025;
x=S3PT.x_1_54;
plot(t,x)
```

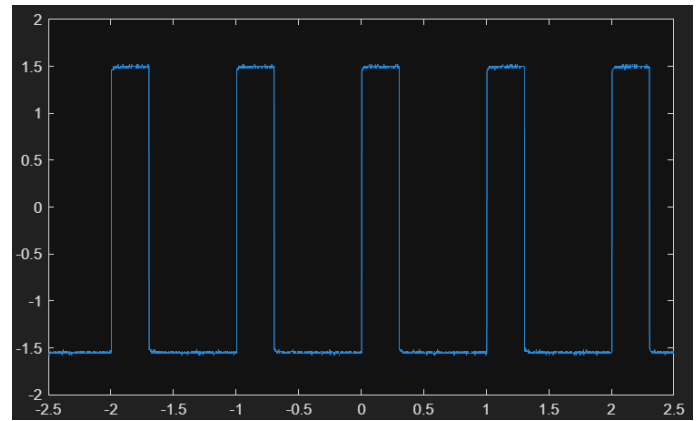


Ilustración.30. Señal 3 Pulso A Partir Del CSV

Señal 4 Pulso, se usaron los siguientes comandos en el apartado de Command Window.

```
t=S4PT.x_0_0025;
x=S4PT.x1_5;
plot(t,x)
```

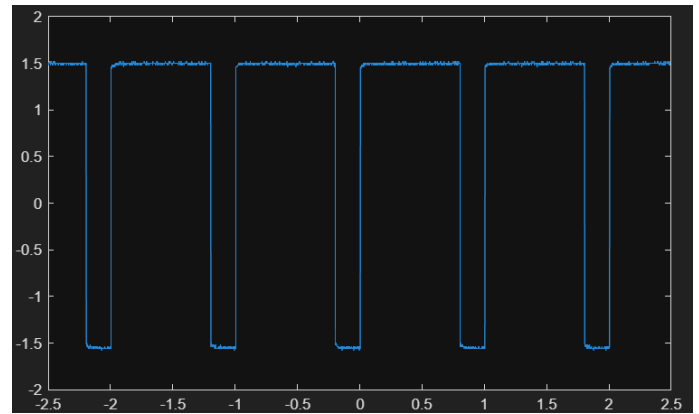


Ilustración.31. Señal 3 Pulso A Partir Del CSV

Ahora se procedió a graficar las señales en el dominio de la frecuencia, empezando nuevamente por la señal 1 seno, se usaron los siguientes comandos en el apartado de Command Window:

```
f=S1SF.x0;
x=S1SF.x_35_3897;
plot(f,x)
```

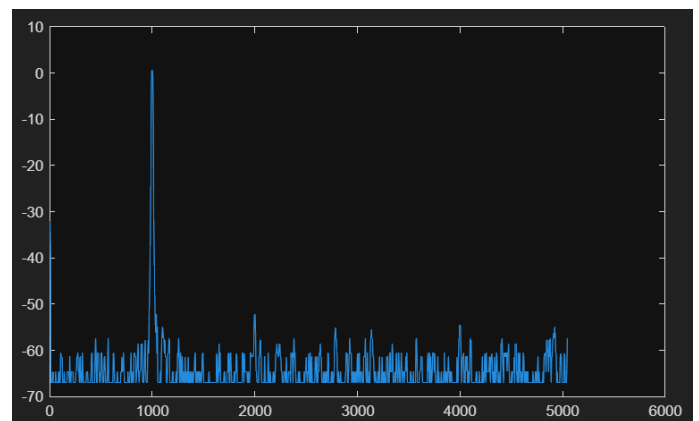


Ilustración.32. Señal 1 Seno A Partir Del CSV

Señal 2 Seno, se usaron los siguientes comandos en el apartado de Command Window.

```
f=S2SF.x0;  
x=S2SF.x_35_3897;  
plot(f,x)
```

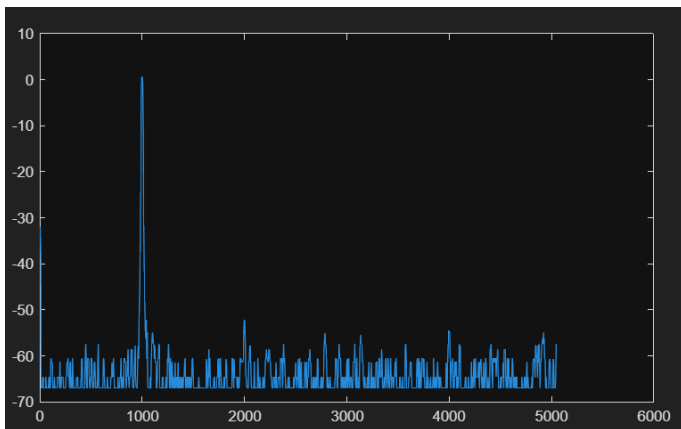


Ilustración.33. Señal 2 Seno A Partir Del CSV

Señal 1 Triangular, se usaron los siguientes comandos en el apartado de Command Window.

```
f=S1TF.x0;  
x=S1TF.x_35_3897;  
plot(f,x)
```

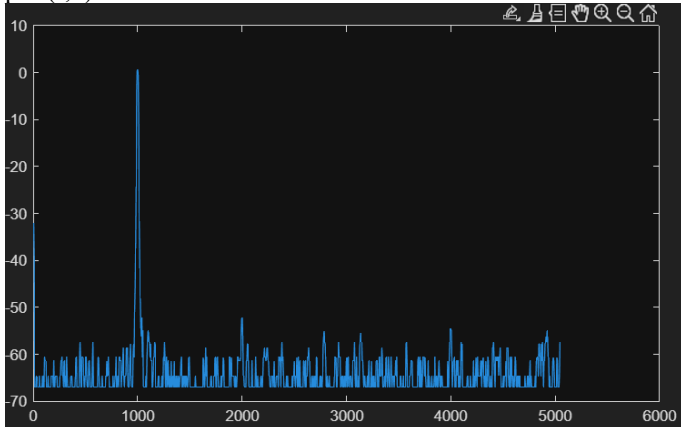


Ilustración.34. Señal 1 Triangular A Partir Del CSV

Señal 2 Triangular, se usaron los siguientes comandos en el apartado de Command Window.

```
F=S2TF.x_16_9691;  
T=S2TF.x0;  
plot(f,x)
```

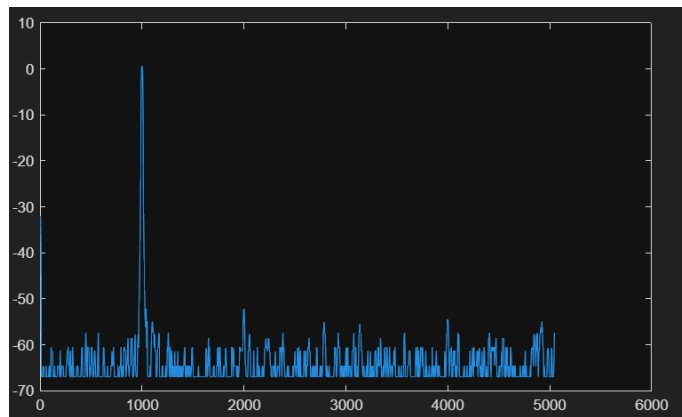


Ilustración.35. Señal 2 Triangular A Partir Del CSV

Señal 1 Cuadrada, se usaron los siguientes comandos en el apartado de Command Window.

```
c=S1CF.x_30_1691;  
b=S1CF.x0;  
plot(b,c)
```

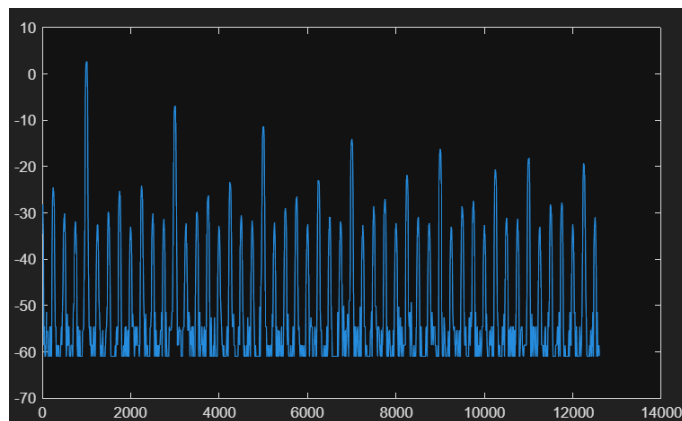


Ilustración.36. Señal 1 Cuadrada A Partir Del CSV

Señal 2 Cuadrada, se usaron los siguientes comandos en el apartado de Command Window.

```
h=S2CF.x0;  
n=S2CF.x1_8309;  
plot(h,c)
```

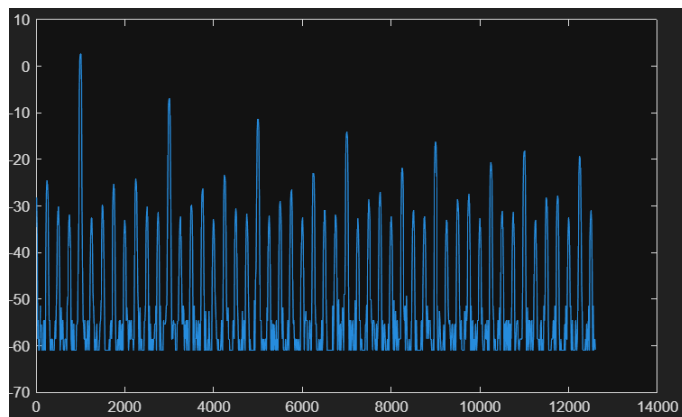


Ilustración.37. Señal 2 Cuadrada A Partir Del CSV

Señal 1 Pulso, se usaron los siguientes comandos en el apartado de Command Window.

```
p=S1PF.x0;
k=S1PF.x_40_1897;
plot(p,k)
```

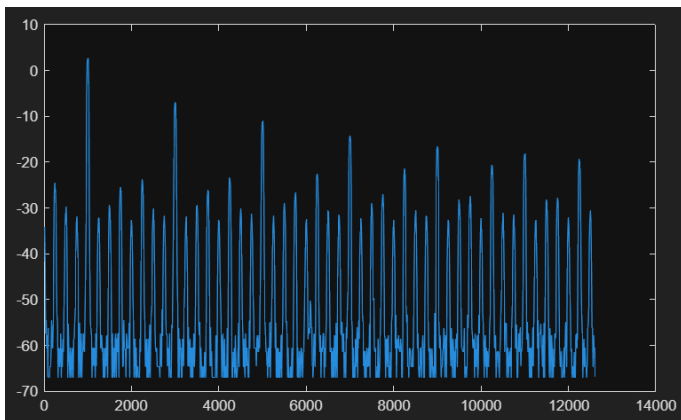


Ilustración.38. Señal 1 Pulso A Partir Del CSV

Señal 2 Pulso, se usaron los siguientes comandos en el apartado de Command Window.

```
z=S2PF.x0;
a=S2PF.x_1_7897;
plot(z,a)
```

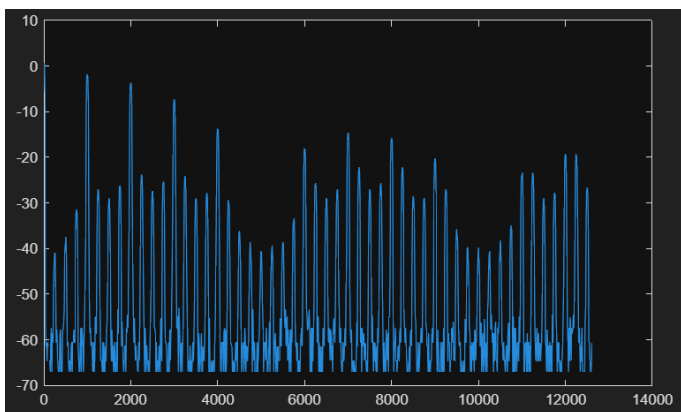


Ilustración.39. Señal 2 Pulso A Partir Del CSV

Señal 3 Pulso, se usaron los siguientes comandos en el apartado de Command Window.

```
u=S3PF.x0;
l=S3PF.x_8_1897;
plot(u,l)
```

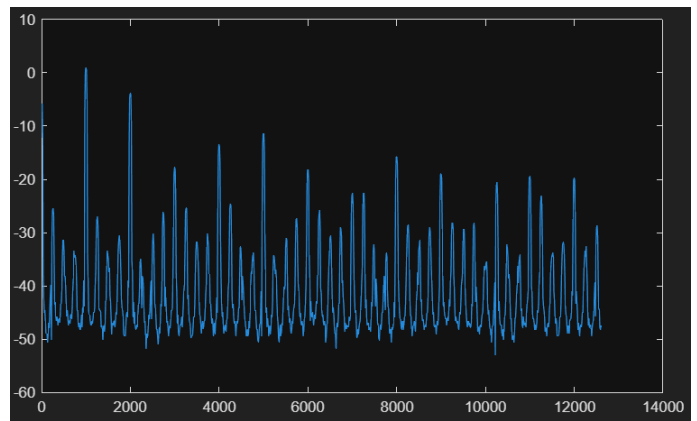


Ilustración.40. Señal 3 Pulso A Partir Del CSV

Señal 4 Pulso, se usaron los siguientes comandos en el apartado de Command Window.

```
r=S4PF.x0;
j=S4PF.x1_0103;
plot(r,j)
```

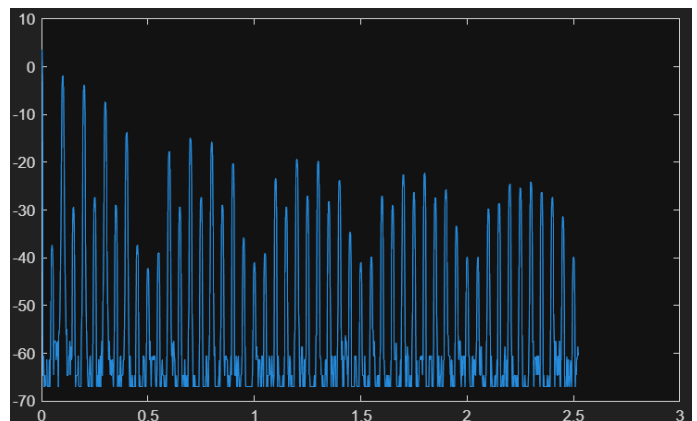


Ilustración.41. Señal 4 Pulso A Partir Del CSV

A continuación, se procede a la visualización de cada una de las señales experimentales capturadas mediante el osciloscopio.

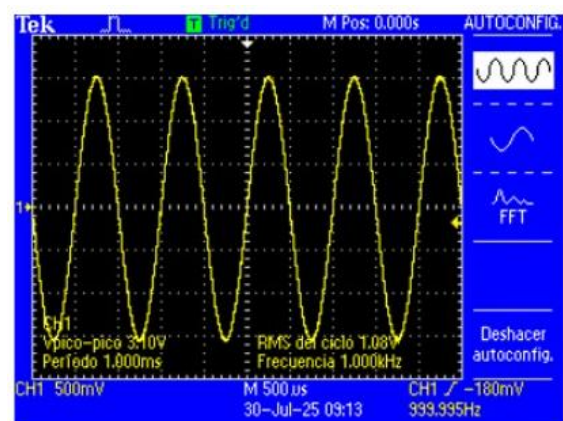


Ilustración.42. Señal 1 Seno En Osciloscopio Dominio Tiempo

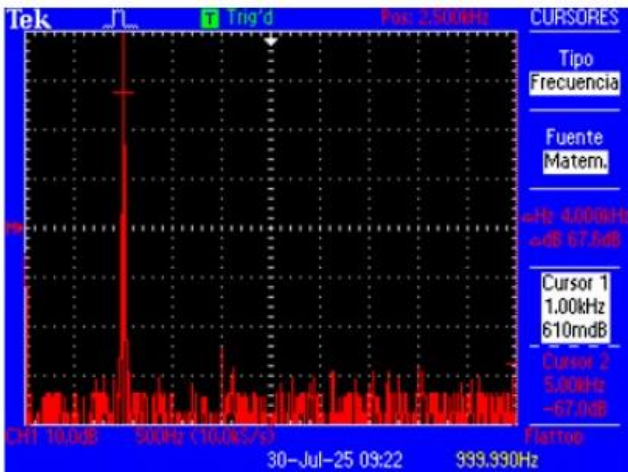


Ilustración.43. Señal 1 Seno En Osciloscopio Dominio Frecuencia

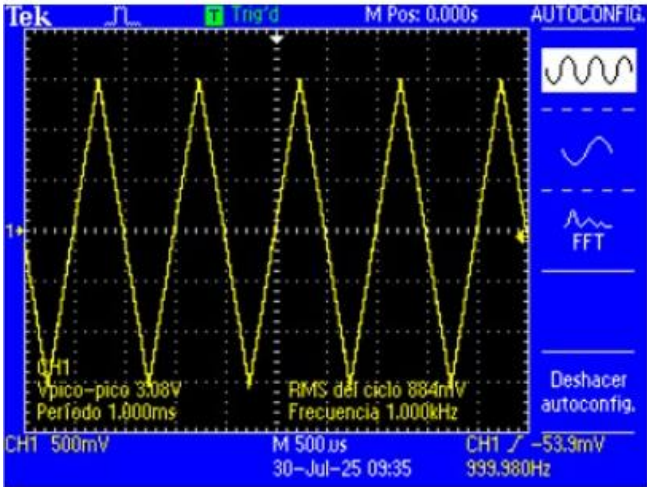


Ilustración.46. Señal 1 Triangular En Osciloscopio Dominio Tiempo

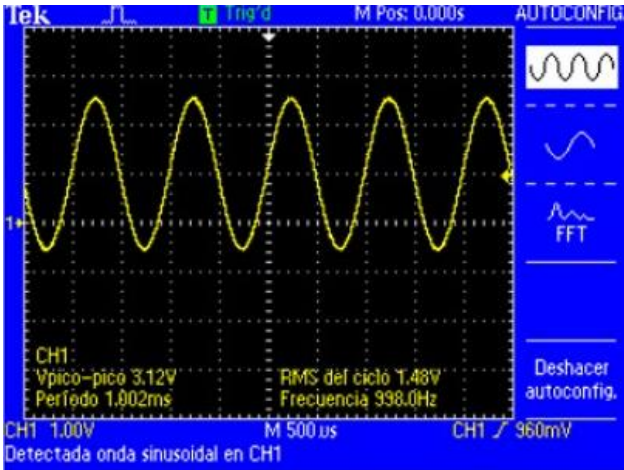


Ilustración.44. Señal 2 Seno En Osciloscopio Dominio Tiempo

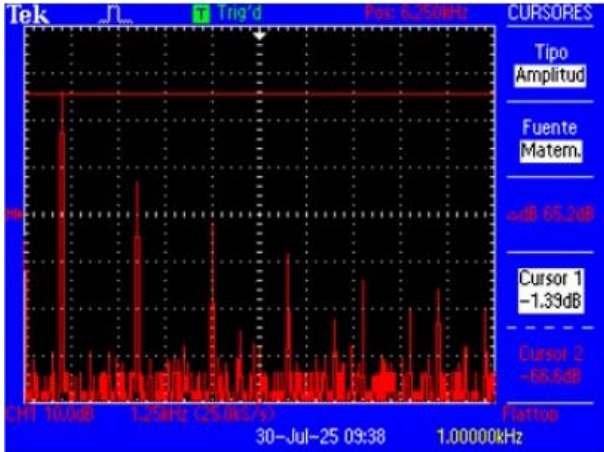


Ilustración.47. Señal 1 Triangular En Osciloscopio Dominio Frecuencia

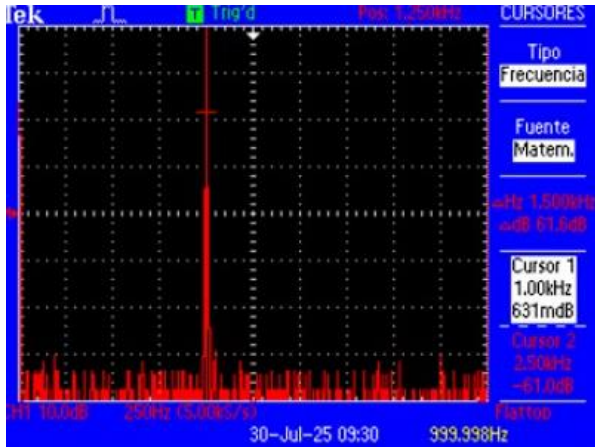


Ilustración.45. Señal 2 Seno En Osciloscopio Dominio Frecuencia

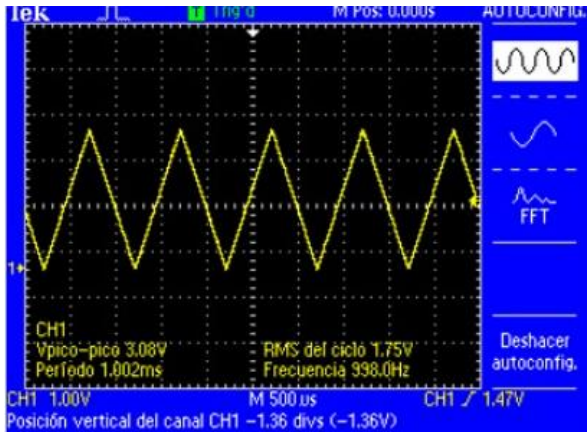


Ilustración.48. Señal 2 Triangular En Osciloscopio Dominio Tiempo

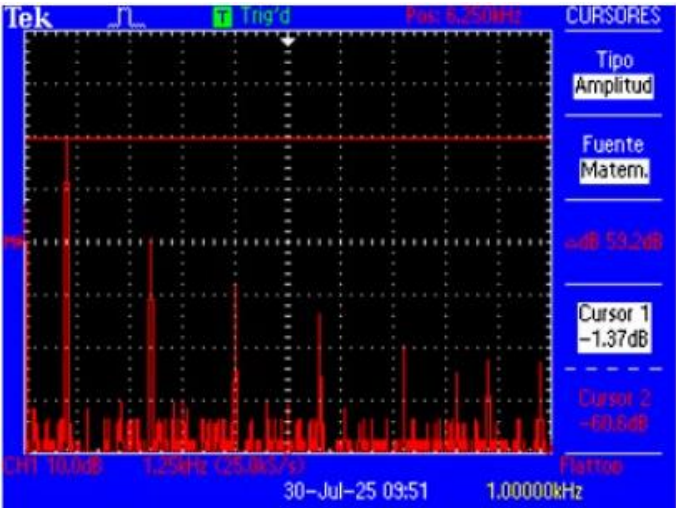


Ilustración.49. Señal 2 Triangular En Osciloscopio Dominio Frecuencia

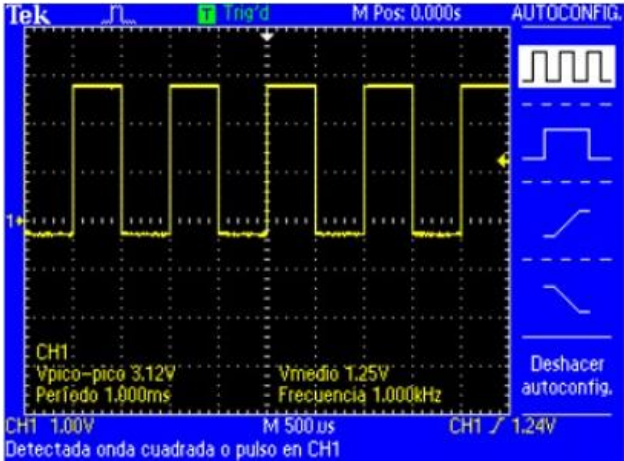


Ilustración.52. Señal 2 Cuadrada En Osciloscopio Dominio Tiempo

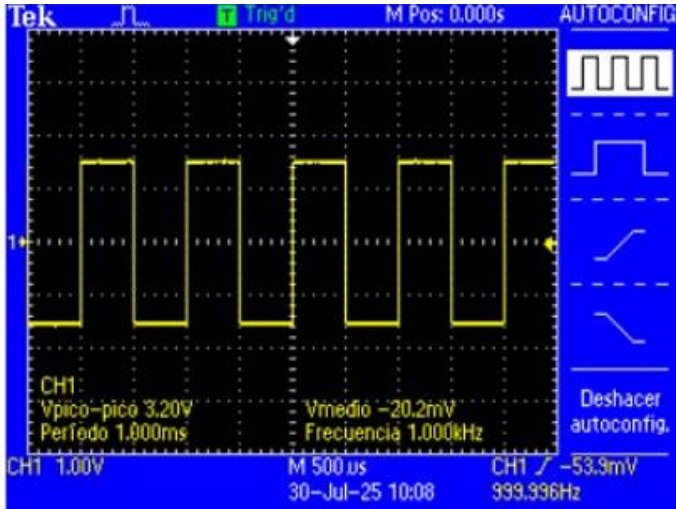


Ilustración.50. Señal 1 Cuadrada En Osciloscopio Dominio Tiempo

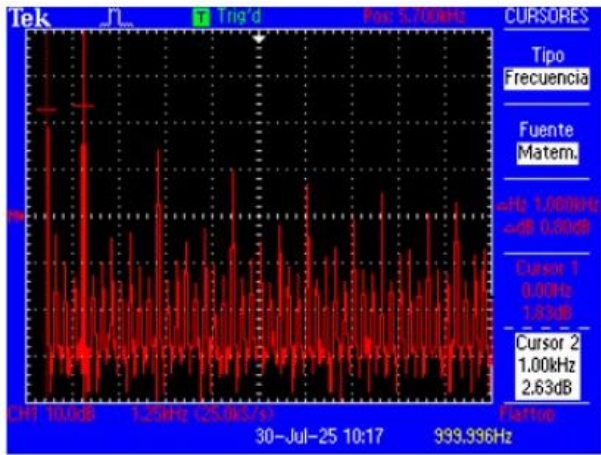


Ilustración.53. Señal 2 Cuadrada En Osciloscopio Dominio Frecuencia

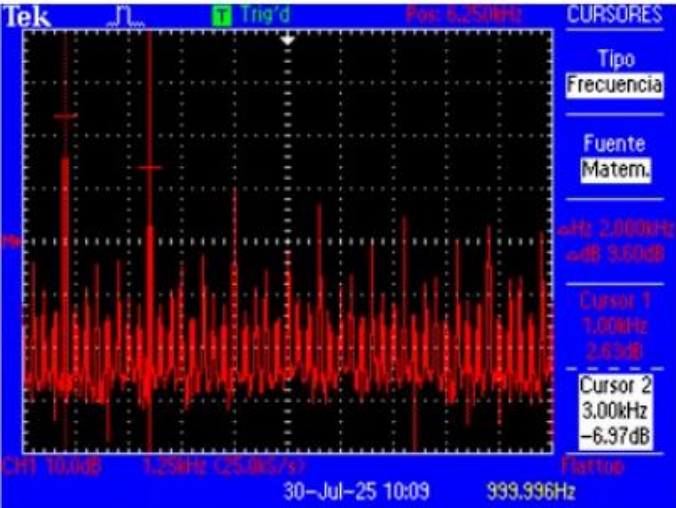


Ilustración.51. Señal 1 Cuadrada En Osciloscopio Dominio Frecuencia

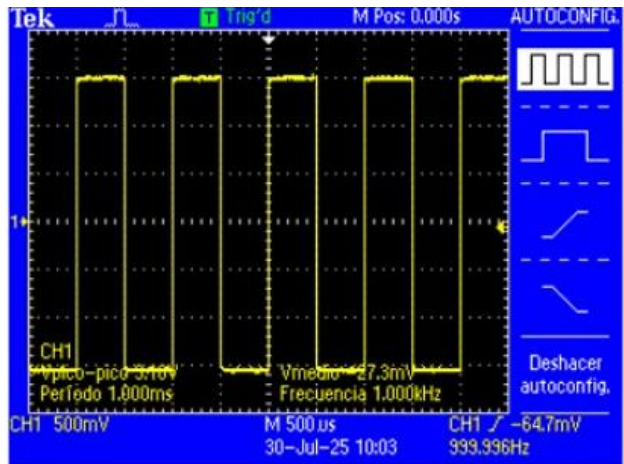


Ilustración.54. Señal 1 Pulso En Osciloscopio Dominio Tiempo

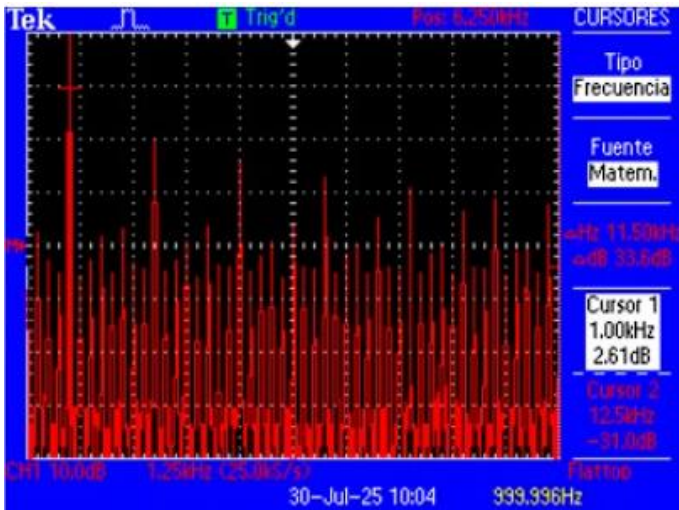


Ilustración.55. Señal 1 Pulso En Osciloscopio Dominio Frecuencia

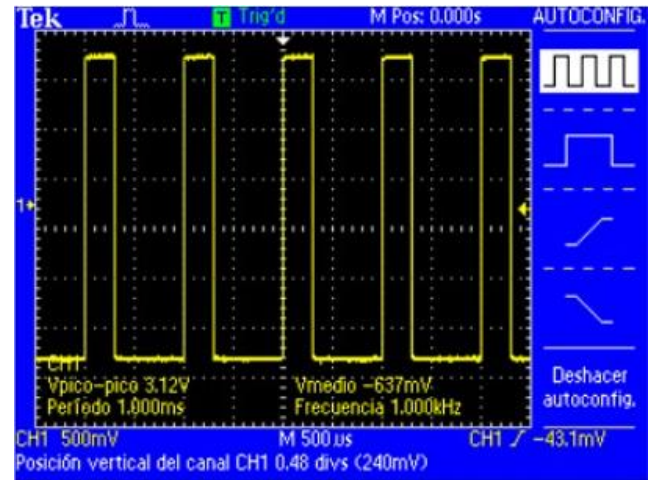


Ilustración.58. Señal 3 Pulso En Osciloscopio Dominio Tiempo

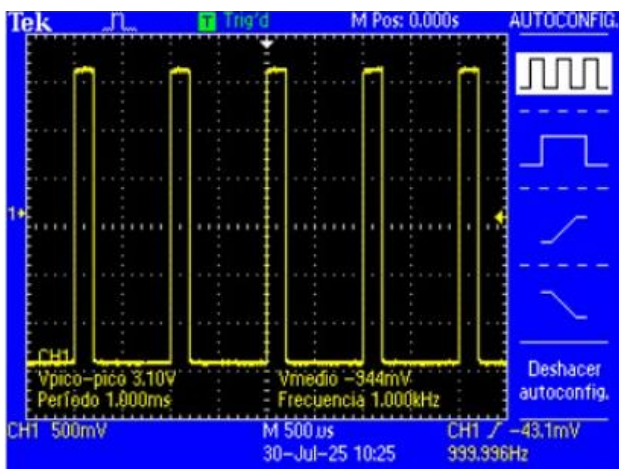


Ilustración.56. Señal 2 Pulso En Osciloscopio Dominio Tiempo

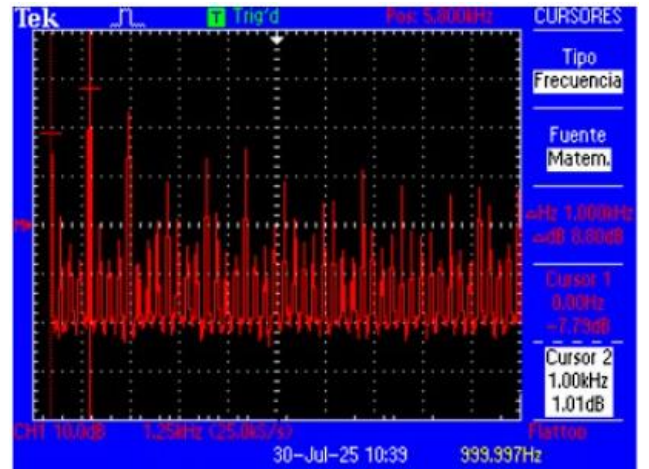


Ilustración.59. Señal 3 Pulso En Osciloscopio Dominio Frecuencia

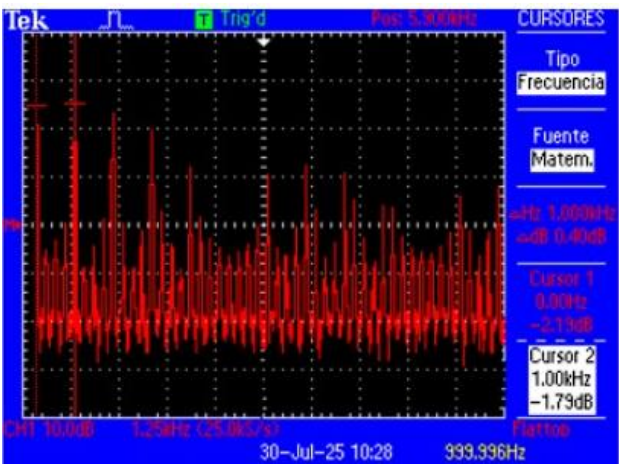


Ilustración.57. Señal 2 Pulso En Osciloscopio Dominio Frecuencia

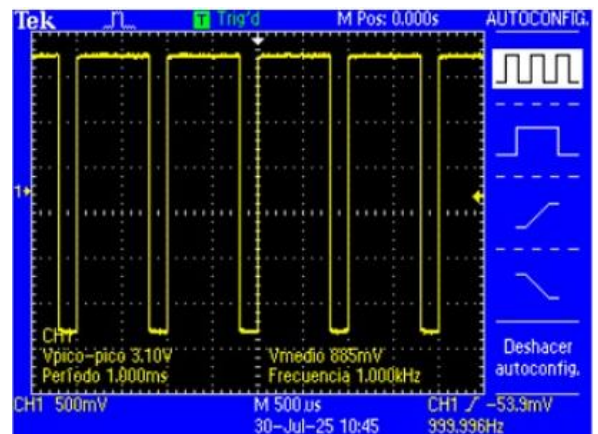


Ilustración.60. Señal 4 Pulso En Osciloscopio Dominio Tiempo

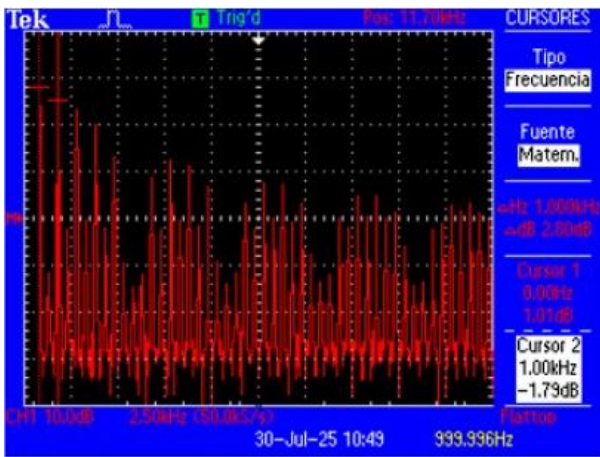


Ilustración.61. Señal 4 Pulso En Osciloscopio Dominio Frecuencia

Posteriormente, se graficaron las señales reconstruidas a partir de los armónicos obtenidos experimentalmente. Estas reconstrucciones fueron comparadas con las señales generadas a partir de los armónicos teóricos, con el fin de evaluar el grado de concordancia entre ambos resultados y validar el modelo de análisis de Fourier aplicado.

Se empezó por el código de la primera señal seno:

```
T = 0.1;
f = 1/T;
```

```
A1 = 1.072;
A5 = 0;
A7 = 0;
A9 = 0;
A11 = 0;
```

```
DC = 0;
```

```
t = 0:T/100:2*T;
```

```
V1 = A1 * sin(2 * pi * f * t);
V5 = A5 * sin(2 * pi * 5 * f * t);
V7 = A7 * sin(2 * pi * 7 * f * t);
V9 = A9 * sin(2 * pi * 9 * f * t);
V11 = A11 * sin(2 * pi * 11 * f * t);
```

```
VT = V1 + V5 + V7 + V9 + V11 + DC;
```

```
figure;
plot(t, VT);
title('Señal Reconstruida a partir de Armónicos Exp');
xlabel('Tiempo (s)');
ylabel('Amplitud');
grid on;
```

Obteniendo como resultado la siguiente gráfica:



Ilustración.62. Señal 1 Seno Reconstruida Armónicos Experimentales

Luego con la señal 2 del seno:

```
T = 0.1;
f = 1/T;
```

```
A1 = 1.075;
A5 = 0;
A7 = 0;
A9 = 0;
A11 = 0;
DC = 1.026;
t = 0:T/100:2*T;
```

```
V1 = A1 * sin(2 * pi * f * t);
V5 = A5 * sin(2 * pi * 5 * f * t);
V7 = A7 * sin(2 * pi * 7 * f * t);
V9 = A9 * sin(2 * pi * 9 * f * t);
V11 = A11 * sin(2 * pi * 11 * f * t);
```

```
VT = V1 + V5 + V7 + V9 + V11 + DC;
figure;
plot(t, VT);
title('Señal Reconstruida a partir de Armónicos Exp');
xlabel('Tiempo (s)');
ylabel('Amplitud');
grid on;
```

Obteniendo la siguiente señal:



Ilustración.63. Señal 2 Seno Reconstruida Armónicos Experimentales

Ahora se realizará lo mismo con la primera señal triangular:

```
T = 0.1;
f = 1/T;

A1 = 1.16;
A3 = 0.097;
A5 = 0.033;
A7 = 0.017;
A9 = 0.0093;
A11 = 0.007;

t = 0:T/100:2*T;

V1 = A1 * cos(2 * pi * f * t);
V3 = A3 * cos(2 * pi * 3 * f * t);
V5 = A5 * cos(2 * pi * 5 * f * t);
V7 = A7 * cos(2 * pi * 7 * f * t);
V9 = A9 * cos(2 * pi * 9 * f * t);
V11 = A11 * cos(2 * pi * 11 * f * t);

VT = V1 + V3 + V5 + V7 + V9 + V11;

figure;
plot(t, VT);
title('Señal Triangular Reconstruida a partir de Armónicos
Exp');
xlabel('Tiempo (s)');
ylabel('Amplitud');
grid on;
```

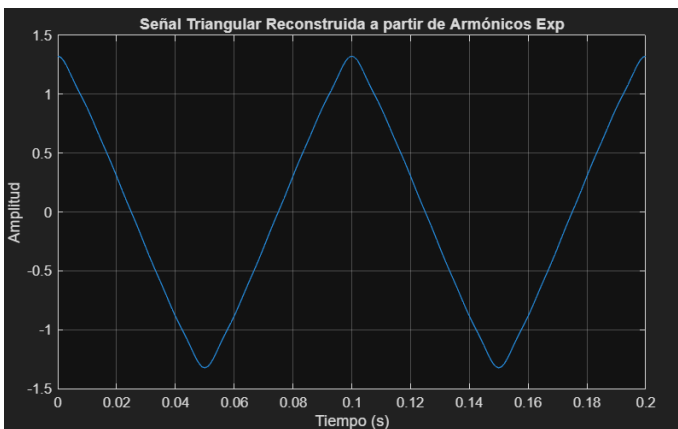


Ilustración.64. Señal 1 Triangular Reconstruida Armónicos Experimentales

Luego con la señal 2 de la triangular:

```
T = 0.1;
f = 1/T;

A0 = 1.48;
A1 = 1.18;
A3 = 0.097;
A5 = 0.033;
A7 = 0.018;
A9 = 0.0093;
A11 = 0.0077;
```

```
t = 0:T/100:2*T;

V0 = A0 * ones(size(t));
V1 = A1 * cos(2 * pi * f * t);
V3 = A3 * cos(2 * pi * 3 * f * t);
V5 = A5 * cos(2 * pi * 5 * f * t);
V7 = A7 * cos(2 * pi * 7 * f * t);
V9 = A9 * cos(2 * pi * 9 * f * t);
V11 = A11 * cos(2 * pi * 11 * f * t);

VT = V0 + V1 + V3 + V5 + V7 + V9 + V11;

figure;
plot(t, VT);
title('Señal Reconstruida a partir de Armónicos Exp');
xlabel('Tiempo (s)');
ylabel('Amplitud');
grid on;
```

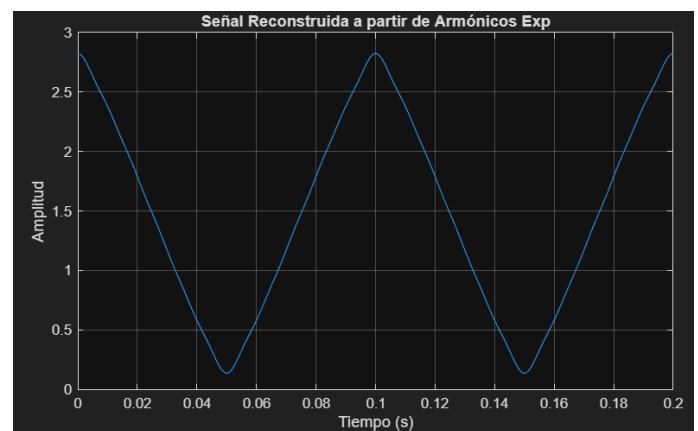


Ilustración.65. Señal 1 Triangular Reconstruida Armónicos Experimentales

Ahora se realizará lo mismo con la primera señal cuadrada:

```
T = 0.1;
f = 1/T;

A1 = 1.32;
A3 = 0.56;
A5 = 0.27;
A7 = 0.19;
A9 = 0.12;
A11 = 0.12;

t = 0:T/100:2*T;
V1 = A1 * sin(2 * pi * f * t);
V3 = A3 * sin(2 * pi * 3 * f * t);
V5 = A5 * sin(2 * pi * 5 * f * t);
V7 = A7 * sin(2 * pi * 7 * f * t);
V9 = A9 * sin(2 * pi * 9 * f * t);
V11 = A11 * sin(2 * pi * 11 * f * t);

V_square = V0 + V1 + V3 + V5 + V7 + V9 + V11;

figure;
plot(t, V_square);
```

```

title('Señal Cuadrada Reconstruida con Armónicos y
Componente DC');
xlabel('Tiempo (s)');
ylabel('Amplitud');
grid on;

```

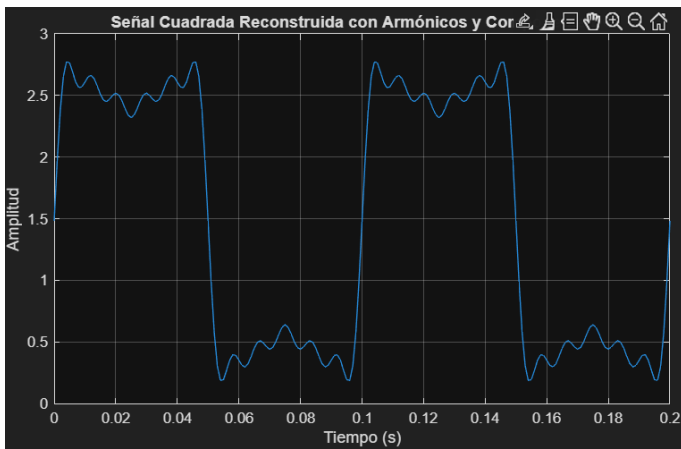


Ilustración.66. Señal 1 Cuadrada Reconstruida Armónicos Experimentales

Luego con la señal 2 de la Cuadrada:

```

T = 0.1;
f = 1/T;

A0 = 1.16;
A1 = 1.32;
A3 = 0.56;
A5 = 0.27;
A7 = 0.19;
A9 = 0.15;
A11 = 0.12;

t = 0:T/100:2*T;

V0 = A0 * ones(size(t));
V1 = A1 * sin(2 * pi * f * t);
V3 = A3 * sin(2 * pi * 3 * f * t);
V5 = A5 * sin(2 * pi * 5 * f * t);
V7 = A7 * sin(2 * pi * 7 * f * t);
V9 = A9 * sin(2 * pi * 9 * f * t);
V11 = A11 * sin(2 * pi * 11 * f * t);

V_square = V0 + V1 + V3 + V5 + V7 + V9 + V11;

figure;
plot(t, V_square);
title('Señal Cuadrada Reconstruida con Armónicos y
Componente DC');
xlabel('Tiempo (s)');
ylabel('Amplitud');
grid on;

```

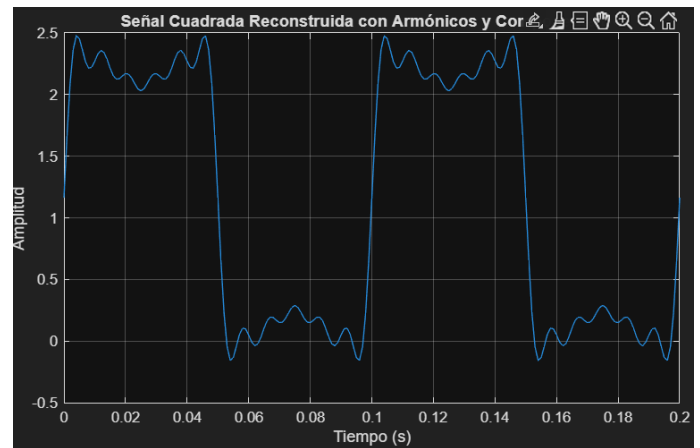


Ilustración.67. Señal 2 Cuadrada Reconstruida Armónicos Experimentales

Ahora se realizará los mismo con las señales de pulso, para la primera señal se tiene el siguiente codigo:

```

T = 0.1;
f = 1/T;

A1 = 1.16;
A3 = 0.44;
A5 = 0.28;
A7 = 0.19;
A9 = 0.15;
A11 = 0.15;

t = 0:T/100:2*T;

V1 = A1 * sin(2 * pi * f * t);
V3 = A3 * sin(2 * pi * 3 * f * t);
V5 = A5 * sin(2 * pi * 5 * f * t);
V7 = A7 * sin(2 * pi * 7 * f * t);
V9 = A9 * sin(2 * pi * 9 * f * t);
V11 = A11 * sin(2 * pi * 11 * f * t);

V_pulse = V1 + V3 + V5 + V7 + V9 + V11;
figure;
plot(t, V_pulse);
title('Señal de Pulso Reconstruida (Duty Cycle 50%)');
xlabel('Tiempo (s)');
ylabel('Amplitud');
grid on;

```

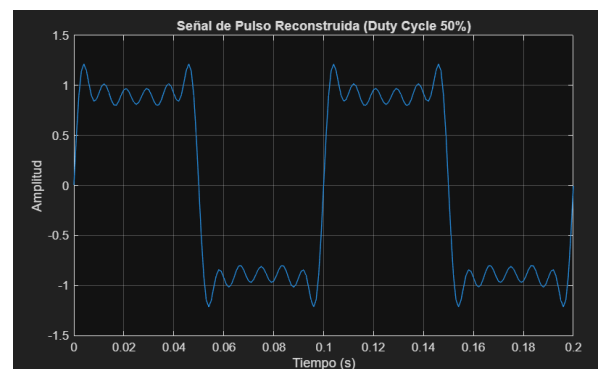


Ilustración.68. Señal 1 Pulso Reconstruida Armónicos Experimentales

Para la segunda señal del pulso se tiene el siguiente código:

```
T = 0.1;
f = 1/T;

A0 = 0.34;
A1 = 0.59;
A2 = 0.51;
A3 = 0.42;
A4 = 0.18;
A5 = 0.018;
A6 = -0.14;
A7 = -0.18;
A8 = -0.16;
A9 = -0.089;

t = 0:T/100:2*T;

V0 = A0;
V1 = A1 * cos(2 * pi * f * t);
V2 = A2 * cos(2 * pi * 2 * f * t);
V3 = A3 * cos(2 * pi * 3 * f * t);
V4 = A4 * cos(2 * pi * 4 * f * t);
V5 = A5 * cos(2 * pi * 5 * f * t);
V6 = A6 * cos(2 * pi * 6 * f * t);
V7 = A7 * cos(2 * pi * 7 * f * t);
V8 = A8 * cos(2 * pi * 8 * f * t);
V9 = A9 * cos(2 * pi * 9 * f * t);

V_pulse_20 = V0 + V1 + V2 + V3 + V4 + V5 + V6 + V7 +
V8 + V9;

figure;
plot(t, V_pulse_20);
title('Señal de Pulso Reconstruida con Duty Cycle 20%');
xlabel('Tiempo (s)');
ylabel('Amplitud');
grid on;
```

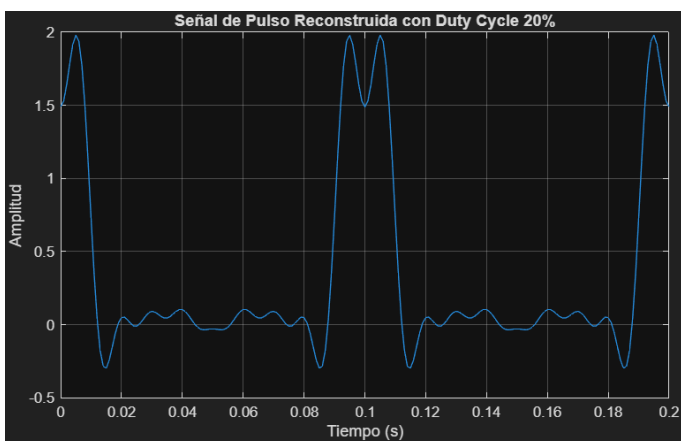


Ilustración.69. Señal 2 Pulso Reconstruida Armónicos Experimentales

Para la tercera señal del pulso se realizó el siguiente código:

```
T = 0.1;
f = 1/T;

A0 = 0.4;
A1 = 0.8;
A2 = 0.51;
A3 = 0.117;
A4 = -0.22;
A5 = -0.26;
A6 = -0.12;
A7 = -0.071;
A8 = 0.15;
A9 = 0.12;

t = 0:T/100:2*T;

V0 = A0;
V1 = A1 * cos(2 * pi * f * t);
V2 = A2 * cos(2 * pi * 2 * f * t);
V3 = A3 * cos(2 * pi * 3 * f * t);
V4 = A4 * cos(2 * pi * 4 * f * t);
V5 = A5 * cos(2 * pi * 5 * f * t);
V6 = A6 * cos(2 * pi * 6 * f * t);
V7 = A7 * cos(2 * pi * 7 * f * t);
V8 = A8 * cos(2 * pi * 8 * f * t);
V9 = A9 * cos(2 * pi * 9 * f * t);

V_pulse_30 = V0 + V1 + V2 + V3 + V4 + V5 + V6 + V7 +
V8 + V9;

figure;
plot(t, V_pulse_30);
title('Señal de Pulso Reconstruida con Duty Cycle 30%');
xlabel('Tiempo (s)');
ylabel('Amplitud');
grid on;
```

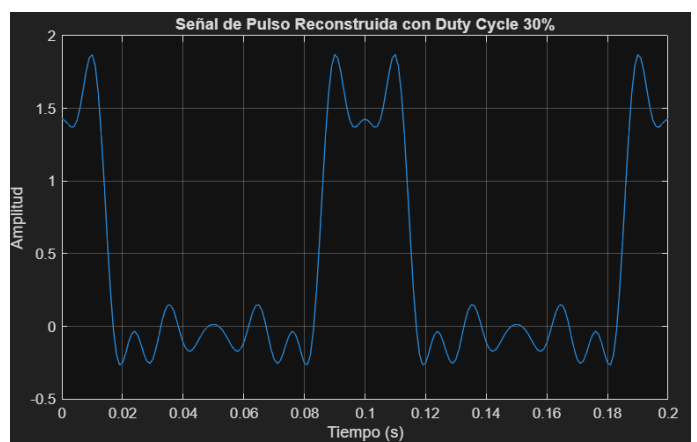


Ilustración.70. Señal 3 Pulso Reconstruida Armónicos Experimentales

Y para finalizar se realizó el siguiente código para la señal 4 del pulso:

```
T = 0.1;
```

$f = 1/T$;

A0 = 1.12;
 A1 = 0.61;
 A2 = -0.64;
 A3 = 0.44;
 A4 = -0.16;
 A5 = 0.03;
 A6 = 0.15;
 A7 = -0.19;
 A8 = 0.16;
 A9 = -0.09;

$t = 0:T/100:2*T$;

V0 = A0;
 V1 = A1 * cos(2 * pi * f * t);
 V2 = A2 * cos(2 * pi * 2 * f * t);
 V3 = A3 * cos(2 * pi * 3 * f * t);
 V4 = A4 * cos(2 * pi * 4 * f * t);
 V5 = A5 * cos(2 * pi * 5 * f * t);
 V6 = A6 * cos(2 * pi * 6 * f * t);
 V7 = A7 * cos(2 * pi * 7 * f * t);
 V8 = A8 * cos(2 * pi * 8 * f * t);
 V9 = A9 * cos(2 * pi * 9 * f * t);

V_pulse_80 = V0 + V1 + V2 + V3 + V4 + V5 + V6 + V7 +
 V8 + V9;

figure;
 plot(t, V_pulse_80);
 title('Señal de Pulso Reconstruida con Duty Cycle 80%');
 xlabel('Tiempo (s)');
 ylabel('Amplitud');
 grid on;

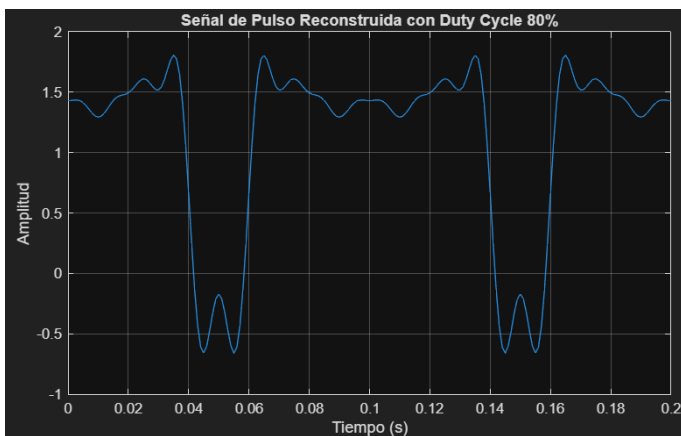


Ilustración.71. Señal 4 Pulso Reconstruida Armónicos Experimentales

Al realizar la comparación entre los armónicos teóricos y los experimentales, se observa una mayor precisión en los valores teóricos. Esto se debe a que los cálculos matemáticos permiten una estimación más exacta de las amplitudes armónicas, mientras que los valores experimentales están sujetos a errores de medición, ruido y limitaciones del equipo. Por lo tanto, la reconstrucción de la señal a partir de los armónicos teóricos

presenta una forma más ideal y cercana al modelo matemático original.

Posteriormente, se desarrolló un programa en MATLAB que permite obtener el espectro en frecuencia de las señales a través del cálculo de la Transformada Rápida de Fourier (FFT). Esta herramienta facilita la identificación de las componentes armónicas presentes en cada señal, así como su amplitud relativa, lo cual es fundamental para el análisis espectral y la validación de los resultados experimentales frente a los teóricos.

Para la primera señal seno se obtiene el siguiente código:

T = 0.1;
 $f = 1/T$;

A1 = 1.077;
 A5 = 0;
 A7 = 0;
 A9 = 0;
 A11 = 0;

DC = 0;

$t = 0:T/100:2*T$;

V1 = A1 * sin(2 * pi * f * t);
 V5 = A5 * sin(2 * pi * 5 * f * t);
 V7 = A7 * sin(2 * pi * 7 * f * t);
 V9 = A9 * sin(2 * pi * 9 * f * t);
 V11 = A11 * sin(2 * pi * 11 * f * t);

VT = V1 + V5 + V7 + V9 + V11 + DC;

figure;
 subplot(2,1,1);
 plot(t, VT);
 title('Señal Reconstruida a partir de Armónicos Teóricos');
 xlabel('Tiempo (s)');
 ylabel('Amplitud');
 grid on;

N = length(VT);
 $f_fft = fft(VT)$;

$f_rec = (0:N-1)*(1/T)/N$;

$mag_f_fft = abs(f_fft)/N$;

subplot(2,1,2);
 $f_half = floor(N/2)$;
 $plot(f_rec(1:f_half), mag_f_fft(1:f_half))$;
 title('Espectro de Frecuencia (FFT) de la Señal');
 xlabel('Frecuencia (Hz)');
 ylabel('Magnitud');
 grid on;

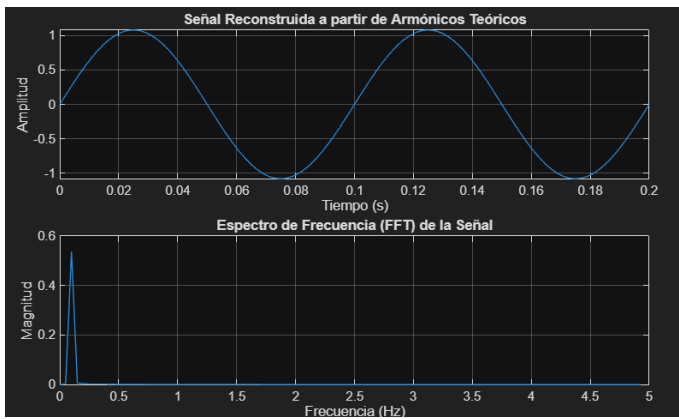


Ilustración.72. Señal 1 Seno Con FFT

Para la señal 2 del seno se obtiene el siguiente código:

```
T = 0.1;
f = 1/T;

A1 = 1.077;
A5 = 0;
A7 = 0;
A9 = 0;
A11 = 0;

DC = 1.027;

t = 0:T/100:2*T;

V1 = A1 * sin(2 * pi * f * t);
V5 = A5 * sin(2 * pi * 5 * f * t);
V7 = A7 * sin(2 * pi * 7 * f * t);
V9 = A9 * sin(2 * pi * 9 * f * t);
V11 = A11 * sin(2 * pi * 11 * f * t);

VT = V1 + V5 + V7 + V9 + V11 + DC;

figure;
subplot(2,1,1);
plot(t, VT);
title('Señal Reconstruida a partir de Armónicos Teóricos');
xlabel('Tiempo (s)');
ylabel('Amplitud');
grid on;

N = length(VT);
f_fft = fft(VT);

f_rec = (0:N-1)*(1/T)/N;

mag_f_fft = abs(f_fft)/N;

subplot(2,1,2);
f_half = floor(N/2);
plot(f_rec(1:f_half), mag_f_fft(1:f_half));
title('Espectro de Frecuencia (FFT) de la Señal');
xlabel('Frecuencia (Hz)');
```

```
ylabel('Magnitud');
grid on;
```

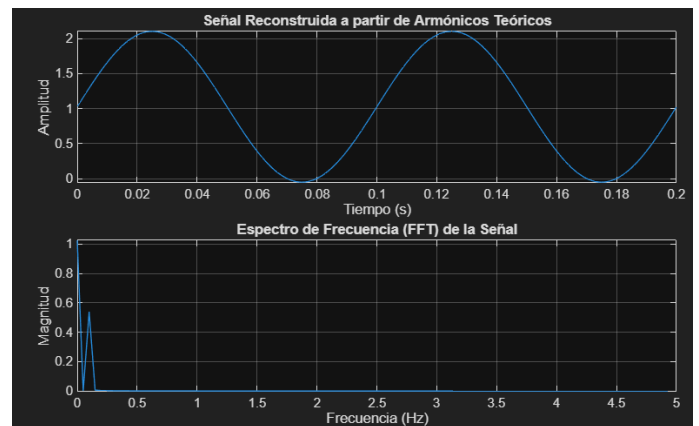


Ilustración.73. Señal 2 Seno Con FFT

Para la señal 1 Triangular se realizó el siguiente código:

```
T = 0.1;
f = 1/T;

A1 = 1.21;
A3 = 0.13;
A5 = 0.048;
A7 = 0.024;
A9 = 0.015;
A11 = 0.01;

t = 0:T/100:2*T;

V1 = A1 * cos(2 * pi * f * t);
V3 = A3 * cos(2 * pi * 3 * f * t);
V5 = A5 * cos(2 * pi * 5 * f * t);
V7 = A7 * cos(2 * pi * 7 * f * t);
V9 = A9 * cos(2 * pi * 9 * f * t);
V11 = A11 * cos(2 * pi * 11 * f * t);

VT = V1 + V3 + V5 + V7 + V9 + V11;

figure;
subplot(2,1,1);
plot(t, VT);
title('Señal Triangular Reconstruida a partir de Armónicos Teóricos');
xlabel('Tiempo (s)');
ylabel('Amplitud');
grid on;

N = length(VT);
f_fft = fft(VT);
f_rec = (0:N-1)*(1/T)/N;
mag_f_fft = abs(f_fft)/N;

subplot(2,1,2);
f_half = floor(N/2);
plot(f_rec(1:f_half), mag_f_fft(1:f_half));
title('Espectro de Frecuencia (FFT) de la Señal Triangular');
```

```

xlabel('Frecuencia (Hz)');
ylabel('Magnitud');
grid on;

```



Ilustración.73. Señal 1 Triangular Con FFT

Para la señal 2 triangular se realizó el siguiente código:

```

T = 0.1;
f = 1/T;

A0 = 1.49;
A1 = 1.21;
A3 = 0.13;
A5 = 0.048;
A7 = 0.024;
A9 = 0.015;
A11 = 0.01;

t = 0:T/100:2*T;

V0 = A0 * ones(size(t));
V1 = A1 * cos(2 * pi * f * t);
V3 = A3 * cos(2 * pi * 3 * f * t);
V5 = A5 * cos(2 * pi * 5 * f * t);
V7 = A7 * cos(2 * pi * 7 * f * t);
V9 = A9 * cos(2 * pi * 9 * f * t);
V11 = A11 * cos(2 * pi * 11 * f * t);

VT = V0 + V1 + V3 + V5 + V7 + V9 + V11;

figure;
subplot(2,1,1);
plot(t, VT);
title('Señal Reconstruida a partir de Armónicos Teóricos');
xlabel('Tiempo (s)');
ylabel('Amplitud');
grid on;

N = length(VT);
f_fft = fft(VT);
f_rec = (0:N-1)*(1/T)/N;
mag_f_fft = abs(f_fft)/N;

subplot(2,1,2);

```

```

f_half = floor(N/2);
plot(f_rec(1:f_half), mag_f_fft(1:f_half));
title('Espectro de Frecuencia (FFT) de la Señal');
xlabel('Frecuencia (Hz)');
ylabel('Magnitud');
grid on;

```

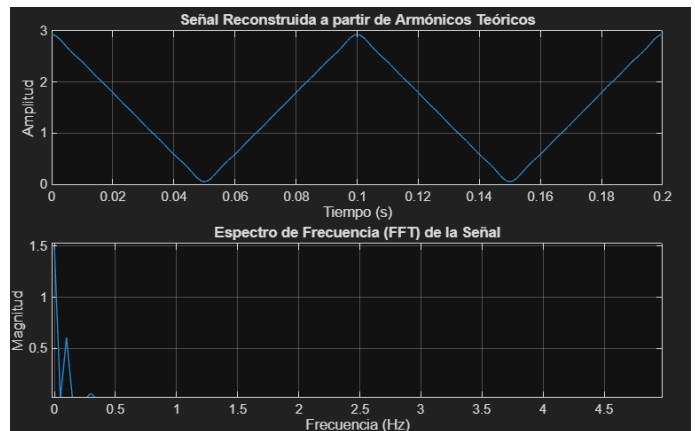


Ilustración.73. Señal 2 Triangular Con FFT

Para la primera señal cuadrada se realizó el siguiente código:

```

T = 0.1;
f = 1/T;

A1 = 1.9;
A3 = 0.63;
A5 = 0.38;
A7 = 0.27;
A9 = 0.21;
A11 = 0.17;

t = 0:T/100:2*T;

V1 = A1 * sin(2 * pi * f * t);
V3 = A3 * sin(2 * pi * 3 * f * t);
V5 = A5 * sin(2 * pi * 5 * f * t);
V7 = A7 * sin(2 * pi * 7 * f * t);
V9 = A9 * sin(2 * pi * 9 * f * t);
V11 = A11 * sin(2 * pi * 11 * f * t);

V_square = V1 + V3 + V5 + V7 + V9 + V11;

offset = 0;
V_square_offset = V_square + offset;

figure;
subplot(2,1,1);
plot(t, V_square_offset);
title('Señal Cuadrada Reconstruida con Armónicos');
xlabel('Tiempo (s)');
ylabel('Amplitud');
grid on;

N = length(V_square_offset);
f_fft = fft(V_square_offset);

```

```
f_rec = (0:N-1)*(1/T)/N;
mag_f_fft = abs(f_fft)/N;

subplot(2,1,2);
f_half = floor(N/2);
plot(f_rec(1:f_half), mag_f_fft(1:f_half));
title('Espectro de Frecuencia (FFT) de la Señal Cuadrada');
xlabel('Frecuencia (Hz)');
ylabel('Magnitud');
grid on;
```

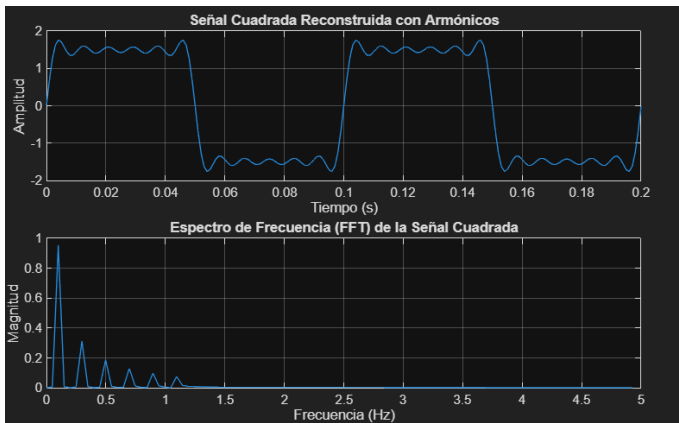


Ilustración.74. Señal 1 Cuadrada Con FFT

Para la segunda señal cuadrada se realizó el segundo código:

```
T = 0.1;
f = 1/T;

A0 = 1.25;
A1 = 1.9;
A3 = 0.63;
A5 = 0.38;
A7 = 0.27;
A9 = 0.21;
A11 = 0.17;

t = 0:T/100:2*T;

V0 = A0 * ones(size(t));
V1 = A1 * sin(2 * pi * f * t);
V3 = A3 * sin(2 * pi * 3 * f * t);
V5 = A5 * sin(2 * pi * 5 * f * t);
V7 = A7 * sin(2 * pi * 7 * f * t);
V9 = A9 * sin(2 * pi * 9 * f * t);
V11 = A11 * sin(2 * pi * 11 * f * t);

V_square = V0 + V1 + V3 + V5 + V7 + V9 + V11;

figure;
subplot(2,1,1);
plot(t, V_square);
title('Señal Cuadrada Reconstruida con Armónicos y Componente DC');
```

```
xlabel('Tiempo (s)');
ylabel('Amplitud');
grid on;
```

```
N = length(V_square);
f_fft = fft(V_square);
```

```
f_rec = (0:N-1)*(1/T)/N;
mag_f_fft = abs(f_fft)/N;
```

```
subplot(2,1,2);
f_half = floor(N/2);
plot(f_rec(1:f_half), mag_f_fft(1:f_half));
title('Espectro de Frecuencia (FFT) de la Señal Cuadrada');
xlabel('Frecuencia (Hz)');
ylabel('Magnitud');
grid on;
```

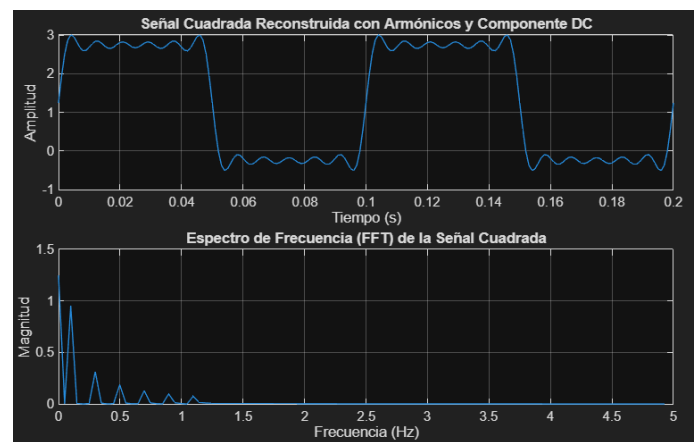


Ilustración.75. Señal 2 Cuadrada Con FFT

Para las señales del pulso se realizó el mismo procedimiento, con la primera señal se hizo este código:

```
T = 0.1;
f = 1/T;

A1 = 0.95;
A3 = -0.31;
A5 = 0.19;
A7 = -0.13;
A9 = 0.10;
A11 = -0.08;

t = 0:T/100:2*T;

V1 = A1 * cos(2 * pi * f * t);
V3 = A3 * cos(2 * pi * 3 * f * t);
V5 = A5 * cos(2 * pi * 5 * f * t);
V7 = A7 * cos(2 * pi * 7 * f * t);
V9 = A9 * cos(2 * pi * 9 * f * t);
V11 = A11 * cos(2 * pi * 11 * f * t);

V_pulse = V1 + V3 + V5 + V7 + V9 + V11;

figure;
```



```

subplot(2,1,1);
plot(t, V_pulse);
title('Señal de Pulso Reconstruida (Duty Cycle 50%)');
xlabel('Tiempo (s)');
ylabel('Amplitud');
grid on;

N = length(V_pulse);
f_fft = fft(V_pulse);
f_rec = (0:N-1)*(1/T)/N;
mag_f_fft = abs(f_fft)/N;

subplot(2,1,2);
f_half = floor(N/2);
plot(f_rec(1:f_half), mag_f_fft(1:f_half));
title('Espectro de Frecuencia (FFT) de la Señal de Pulso');
xlabel('Frecuencia (Hz)');
ylabel('Magnitud');
grid on;

```

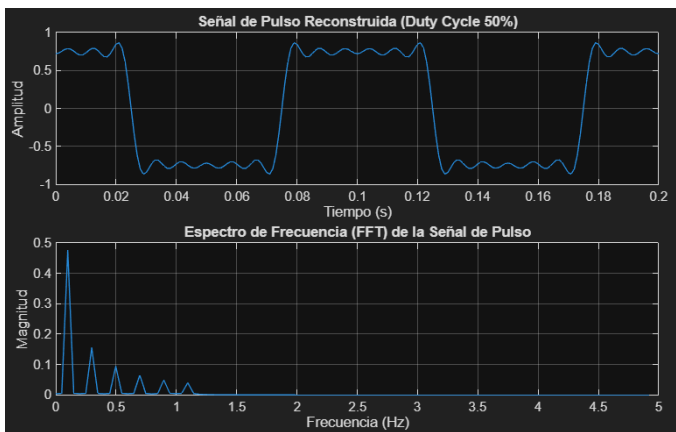


Ilustración.75. Señal 1 Pulso Con FFT

Para la segunda señal de pulso se realizó el siguiente código:

```

T = 0.1;
f = 1/T;

A0 = 0.3;
A1 = 0.56;
A2 = 0.45;
A3 = 0.3;
A4 = 0.14;
A5 = 0;
A6 = -0.093;
A7 = -0.12;
A8 = -0.11;
A9 = -0.062;

t = 0:T/100:2*T;

V0 = A0;
V1 = A1 * cos(2 * pi * f * t);
V2 = A2 * cos(2 * pi * 2 * f * t);
V3 = A3 * cos(2 * pi * 3 * f * t);
V4 = A4 * cos(2 * pi * 4 * f * t);

```

```

V5 = A5 * cos(2 * pi * 5 * f * t);
V6 = A6 * cos(2 * pi * 6 * f * t);
V7 = A7 * cos(2 * pi * 7 * f * t);
V8 = A8 * cos(2 * pi * 8 * f * t);
V9 = A9 * cos(2 * pi * 9 * f * t);

V_pulse_20 = V0 + V1 + V2 + V3 + V4 + V5 + V6 + V7 +
V8 + V9;

```

```

figure;
subplot(2,1,1);
plot(t, V_pulse_20);
title('Señal de Pulso Reconstruida con Duty Cycle 20%');
xlabel('Tiempo (s)');
ylabel('Amplitud');
grid on;

```

```

N = length(V_pulse_20);
f_fft = fft(V_pulse_20);
f_rec = (0:N-1)*(1/T)/N;
mag_f_fft = abs(f_fft)/N;

subplot(2,1,2);
f_half = floor(N/2);
plot(f_rec(1:f_half), mag_f_fft(1:f_half));
title('Espectro de Frecuencia (FFT) de la Señal de Pulso');
xlabel('Frecuencia (Hz)');
ylabel('Magnitud');
grid on;

```

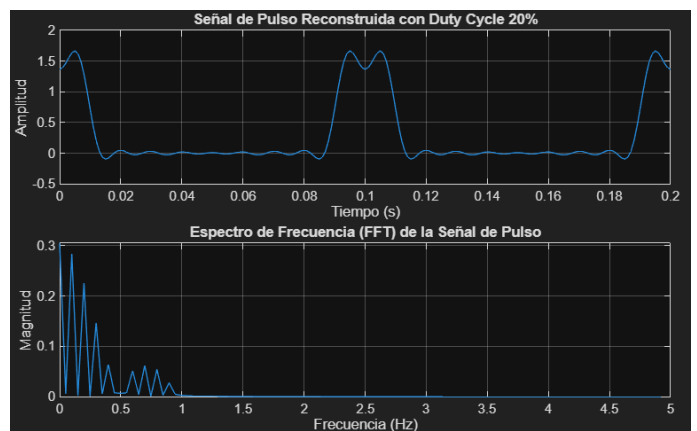


Ilustración.76. Señal 2 Pulso Con FFT

Para la tercera señal de pulso se realizó el siguiente código:

```

T = 0.1;
f = 1/T;

A0 = 0.45;
A1 = 0.77;
A2 = 0.45;
A3 = 0.098;
A4 = -0.14;
A5 = -0.19;
A6 = -0.093;
A7 = -0.042;

```

```

A8 = 0.113;
A9 = 0.085;

t = 0:T/100:2*T;

V0 = A0;
V1 = A1 * cos(2 * pi * f * t);
V2 = A2 * cos(2 * pi * 2 * f * t);
V3 = A3 * cos(2 * pi * 3 * f * t);
V4 = A4 * cos(2 * pi * 4 * f * t);
V5 = A5 * cos(2 * pi * 5 * f * t);
V6 = A6 * cos(2 * pi * 6 * f * t);
V7 = A7 * cos(2 * pi * 7 * f * t);
V8 = A8 * cos(2 * pi * 8 * f * t);
V9 = A9 * cos(2 * pi * 9 * f * t);

V_pulse_30 = V0 + V1 + V2 + V3 + V4 + V5 + V6 + V7 +
V8 + V9;

figure;
subplot(2,1,1);
plot(t, V_pulse_30);
title('Señal de Pulso Reconstruida con Duty Cycle 30%');
xlabel('Tiempo (s)');
ylabel('Amplitud');
grid on;

N = length(V_pulse_30);
f_fft = fft(V_pulse_30);
f_rec = (0:N-1)*(1/T)/N;
mag_f_fft = abs(f_fft)/N;

subplot(2,1,2);
f_half = floor(N/2);
plot(f_rec(1:f_half), mag_f_fft(1:f_half));
title('Espectro de Frecuencia (FFT) de la Señal de Pulso');
xlabel('Frecuencia (Hz)');
ylabel('Magnitud');
grid on;

```

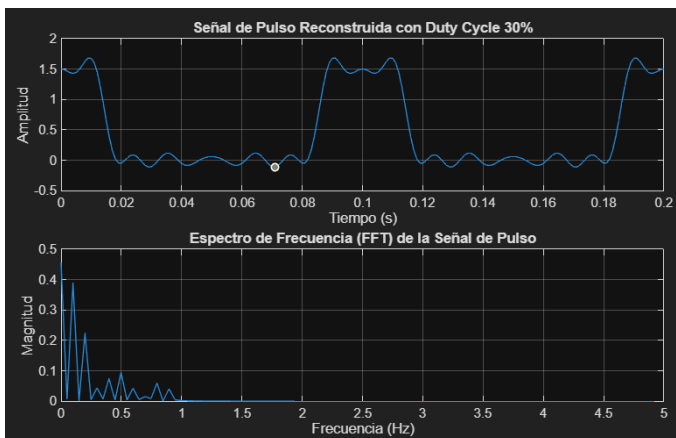


Ilustración.77. Señal 3 Pulso Con FFT

Para la tercera señal de pulso se realizó el siguiente código:

```

T = 0.1;
f = 1/T;
A0 = 1.2;
A1 = 0.56;
A2 = -0.45;
A3 = 0.3;
A4 = -0.14;
A5 = 0;
A6 = 0.093;
A7 = -0.12;
A8 = 0.11;
A9 = -0.062;
t = 0:T/100:2*T;

V0 = A0;
V1 = A1 * cos(2 * pi * f * t);
V2 = A2 * cos(2 * pi * 2 * f * t);
V3 = A3 * cos(2 * pi * 3 * f * t);
V4 = A4 * cos(2 * pi * 4 * f * t);
V5 = A5 * cos(2 * pi * 5 * f * t);
V6 = A6 * cos(2 * pi * 6 * f * t);
V7 = A7 * cos(2 * pi * 7 * f * t);
V8 = A8 * cos(2 * pi * 8 * f * t);
V9 = A9 * cos(2 * pi * 9 * f * t);

V_pulse_80 = V0 + V1 + V2 + V3 + V4 + V5 + V6 + V7 +
V8 + V9;

figure;
subplot(2,1,1);
plot(t, V_pulse_80);
title('Señal de Pulso Reconstruida con Duty Cycle 80%');
xlabel('Tiempo (s)');
ylabel('Amplitud');
grid on;

N = length(V_pulse_80);
f_fft = fft(V_pulse_80);
f_rec = (0:N-1)*(1/T)/N;
mag_f_fft = abs(f_fft)/N;

subplot(2,1,2);
f_half = floor(N/2);
plot(f_rec(1:f_half), mag_f_fft(1:f_half));
title('Espectro de Frecuencia (FFT) de la Señal de Pulso');
xlabel('Frecuencia (Hz)');
ylabel('Magnitud');
grid on;

```

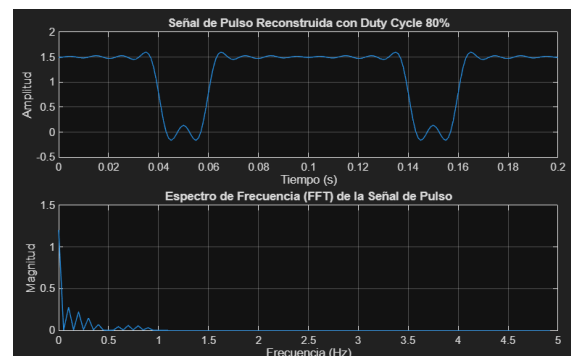


Ilustración.78. Señal 4 Pulso Con FFT

En el siguiente código, se generó una señal de pulso sintética (de 1000 Hz y 80% de ciclo de trabajo) y se analizó en el dominio de la frecuencia mediante la Transformada Rápida de Fourier (FFT). Esto permitió obtener su espectro de frecuencia, es decir, los componentes armónicos que la componen.

Luego, para volver al dominio del tiempo y comprobar si la señal original puede ser reconstruida fielmente, se aplicó la Transformada Inversa de Fourier (IFFT) sobre el espectro completo (complejo) obtenido con la FFT. Esta operación reconstruyó la forma de onda original a partir de sus componentes espectrales.

La IFFT devuelve una señal compleja, pero como la señal original es real, se tomó únicamente la parte real (`real(y_reconstruida)`) para graficarla. Así se confirmó que la forma de onda reconstruida coincide con la señal de pulso original.

```
% 1. Parámetros de la señal
f_pulso = 1000; % Frecuencia del pulso (Hz)
A = 1.5; % Amplitud pico (±1.5 V)
duty_cycle = 80; % Porcentaje
fs = 100000; % Frecuencia de muestreo (Hz)
duracion = 0.05; % Duración de la señal (s)
t = 0 : 1/fs : duracion - 1/fs; % Vector de tiempo

% 2. Generar señal de pulso manual (sin 'square')
T_pulso = 1/f_pulso; % Periodo del pulso
% Señal: +A durante el duty_cycle%, -A durante el resto
y = 2*(mod(t, T_pulso) < (duty_cycle/100)*T_pulso) - 1;
y = A * y; % Ajustar amplitud

% 3. Calcular FFT
L = length(y);
Y = fft(y);
P2 = abs(Y/L);
P1 = P2(1:L/2+1);
P1(2:end-1) = 2*P1(2:end-1);
f = fs*(0:(L/2))/L;

% 4. Graficar FFT
figure;
plot(f/1000, 20*log10(P1));
title('Espectro de Frecuencia de la Señal de Pulso (80% Duty Cycle)');
xlabel('Frecuencia (kHz)');
ylabel('Amplitud (dB)');
grid on;
xlim([0 15]);
hold on;

for n = 1:10
    armonico_f = n * f_pulso;
    if mod(n, 5) == 0
```

```
xline(armonico_f/1000, '--k', sprintf('Nulo en %.1f kHz',
    armonico_f/1000));
else
    xline(armonico_f/1000, '--r', sprintf('%dº Armónico (%.1f
    kHz)', n, armonico_f/1000));
end
end
hold off;
```

```
% 5. Reconstrucción con IFFT
y_reconstruida = ifft(Y);
```

```
% 6. Graficar señal reconstruida
figure;
plot(t, real(y_reconstruida), 'r', 'LineWidth', 1.5);
title('Señal de Pulso Reconstruida por IFFT');
xlabel('Tiempo (s)');
ylabel('Amplitud (V)');
grid on;
xlim([0 3*(1/f_pulso)]);
```

Nos da como resultado la siguiente gráfica:

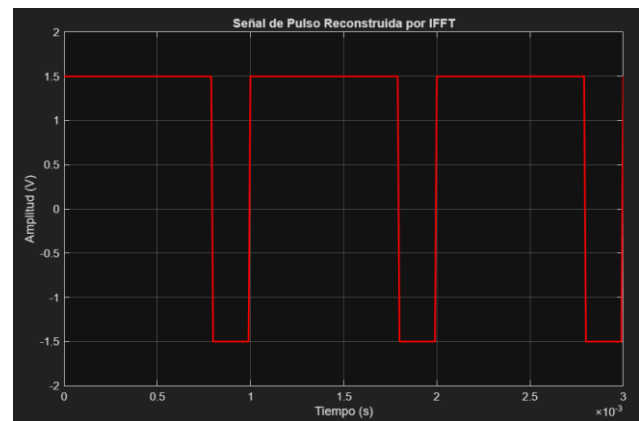


Ilustración.79. Señal 4 Pulso Reconstruida Con IFFT

CONCLUSIONES

[1] El uso adecuado del archivo CSV es esencial: la correcta organización de los datos (separación por comas, ausencia de encabezados innecesarios, y tipo de datos numéricos) influye directamente en la correcta aplicación de la IFFT en MATLAB.

[2] Los errores más comunes en el proceso incluyen: formato incorrecto del archivo CSV, presencia de encabezados o caracteres no numéricos, y uso inadecuado de funciones como `plot`, o `ifft`.

[3] La práctica desarrollada demostró la importancia de entender la correlación entre el dominio del tiempo y de la frecuencia, permitiendo comprobar que diferentes combinaciones de armónicos generan distintas formas de onda (pulso, cuadrada, triangular, etc.).

[4] La Transformada Inversa de Fourier (IFFT) es una herramienta fundamental para reconstruir señales en el dominio del tiempo a partir de su representación en frecuencia,

permitiendo visualizar cómo los componentes espectrales contribuyen a la forma de onda original.

REFERENCIAS

- [1] Formas de onda del oscilador: tipos y usos - PARTE I - Aulart. (s.f.). Aulart. <https://www.aulart.com/es/blog/formas-de-onda-del-oscilador-tipos-y-usos-parte-i/?srsltid=AfmBOorSAA8SeNNRPX7iYEcTFpdlAZIXHA3P8ruPpjltNls3zGxhTFBI> x
- [2] Armonicos. Explicación sencilla. Teoría y ejemplos prácticos. (s.f.). Aplicaciones informáticas para la ingeniería. Software de cálculo. <https://www.herramientasingeneria.com/tech/armonicos/armonicos.html>
- [3] “Google Colab”. Google Colab. Accedido el 27 de julio de 2025. [En línea]. Disponible: <https://colab.research.google.com/>
- [4] “MATLAB Online”. MathWorks - Maker of MATLAB and Simulink - MATLAB & Simulink. Accedido el 27 de julio de 2025. [En línea]. Disponible: <https://www.mathworks.com/products/matlab-online/limitations.html>