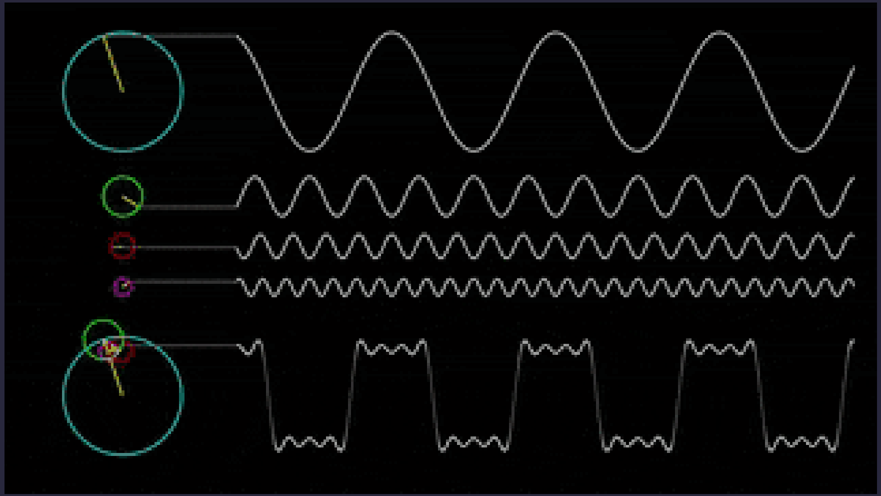
/TRANSFORMADA RÁPIDA DE FOURIER

Becerra Sipiran, Cledy Elizabeth
Oviedo Sivincha, Massiel
Villanueva Borda, Harold Alejandro



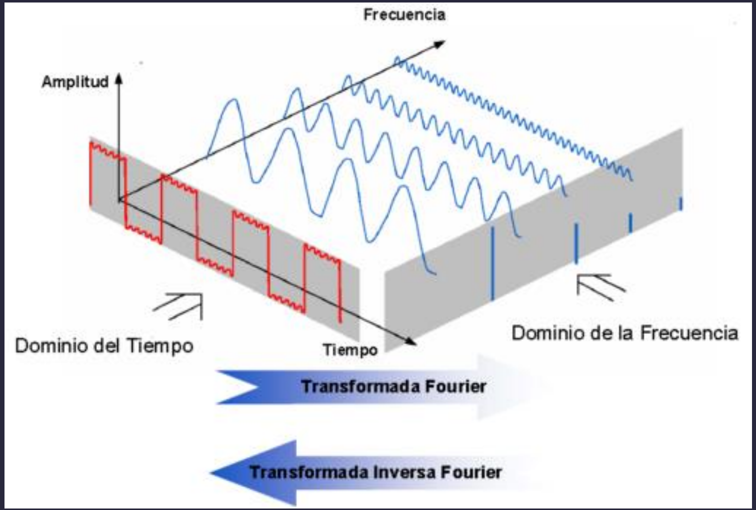
YOUR LOGO HERE

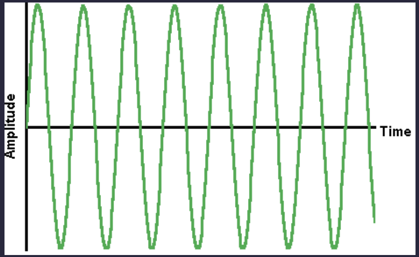
/ÍNDICE

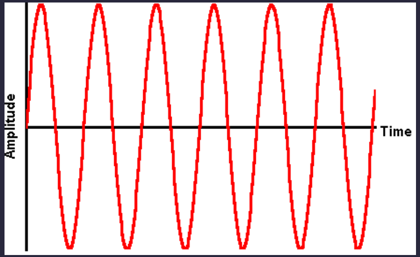
INDEX.HTML

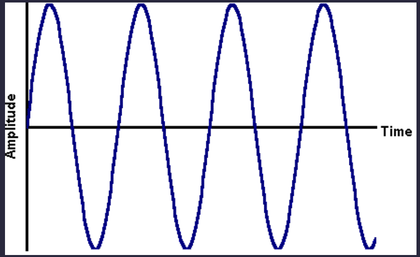
/01

/TRANSFORMADA DE FOURIER




Amplitude
Time


Amplitude
Time


Amplitude
Time

Condiciones de Dirichlet

$$\int_a^b |f(t)|dt < \infty$$

$$\int_{-\infty}^{\infty} |f(t)|dt < \infty$$

- Un número finito de discontinuidades finitas.
- Un número finito de máximos y un número finito de mínimos.

Fórmulas

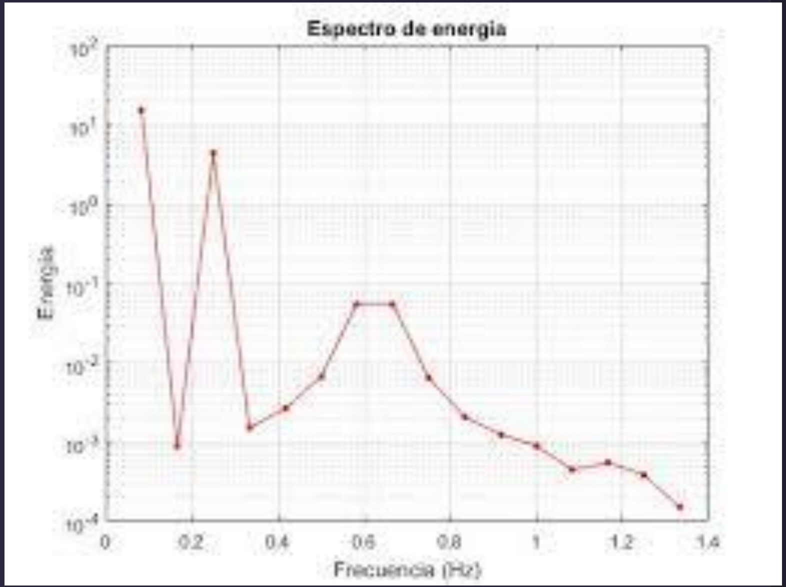
$$X(f) = \int_{-\infty}^{\infty} x(t) e^{-j2\pi f t} \, dt$$

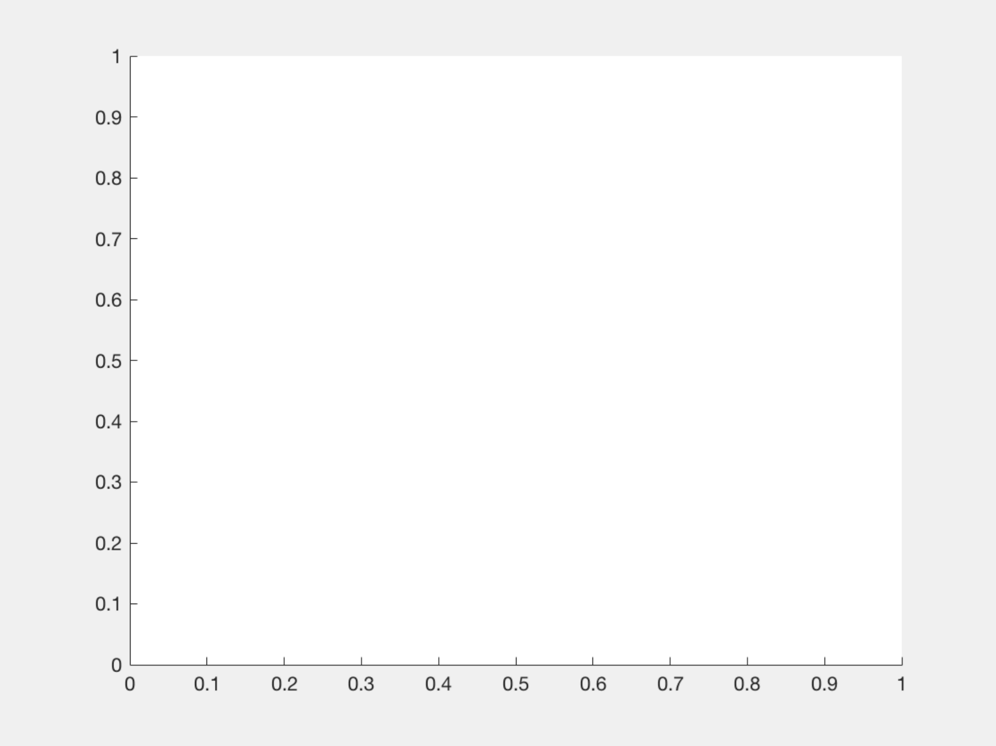
〉 La transformada de Fourier 〈

$$x(t) = \int_{-\infty}^{\infty} X(f) e^{j2\pi f t} \, dt$$

〉 Inversa de la transformada de Fourier 〈

/02 /TRANSFORMADA DISCRETA DE FOURIER




1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1

Condiciones del DFT

$$x[n] = 0, n < 0$$

$$x[n] = 0, n > N - 1$$

- La señal vale 0 para cualquier valor n negativo.
- Para cualquier valor de n superior al número de puntos a calcular menos 1

/Fórmulas

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-\frac{i2\pi}{N}kn}, k = 0, ..., N-1$$

〉 Para encontrar la DFT 〈

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X[k] e^{\frac{i2\pi}{N}kn}, n = 0, ..., N-1$$

〉 Para encontrar la Inversa de la DFT 〈

/EJERCICIO

Calcular la DFT de x[n] = {1, 2,0,-1}.
El tamaño del vector es: N = 4
Recordando que la fórmula para hallar una DFT es $X[k] = \sum_{n=0}^{N-1} x[n]e^{-\frac{i2\pi}{N}kn}$
Entonces

$$X[k] = \sum_{n=0}^{3} x[n]e^{-\frac{i2\pi}{4}kn}$$
$$X[k] = \sum_{n=0}^{3} x[n]e^{-\frac{i\pi}{2}kn}$$

Para $k = 0$:

$$X[0] = \sum_{n=0}^{3} x[n]e^{0}$$
$$X[0] = x[0] + x[1] + x[2] + x[3]$$
$$X[0] = 1 + 2 + 0 + (-1)$$
$$X[0] = 2$$

Para $k = 1$:

$$X[1] = \sum_{n=0}^{3} x[n]e^{-\frac{i\pi}{2}n}$$

Tener en cuenta que:

$$e^{-\frac{i\pi}{2}} = \cancelto{0}{\cos\frac{\pi}{2}} - i\cancelto{1}{\sin\frac{\pi}{2}} = -i$$

$$e^{-i\pi} = \cancelto{-1}{\cos\pi} - i\cancelto{0}{\operatorname{sen}\pi} = -1$$

$$e^{-\frac{i3\pi}{2}} = \cancelto{0}{\cos\frac{3\pi}{2}} - i\cancelto{-1}{\operatorname{sen}\frac{3\pi}{2}} = i$$

$$X[1] = x[0]e^0 + x[1]e^{-\frac{i\pi}{2}} + x[2]e^{-i\pi} + x[3]e^{-\frac{i3\pi}{2}}$$
$$X[1] = 1 + 2(-i) + 0 + (-1)(i)$$
$$X[1] = 1 - 3i$$

Para $k = 2$:

$$X[2] = \sum_{n=0}^{3} x[n]e^{-i\pi n}$$
$$X[2] = x[0]e^0 + x[1]\cancelto{-1}{e^{-i\pi}} + x[2]\cancelto{1}{e^{-i2\pi}} + x[3]\cancelto{-1}{e^{-i3\pi}}$$
$$X[2] = 1 + (2)(-1) + 0 + (-1)(-1)$$
$$X[2] = 0$$

Para $k = 3$:

$$X[3] = x[0]e^0 + x[1]\cancelto{i}{-\frac{3\pi}{2}} + x[3]\cancelto{-1}{-i3\pi} + x[3]\cancelto{-i}{-\frac{9\pi}{2}}$$
$$X[3] = 1 + 2i + 0 + (-1)(-i)$$
$$X[3] = 1 + 3i$$

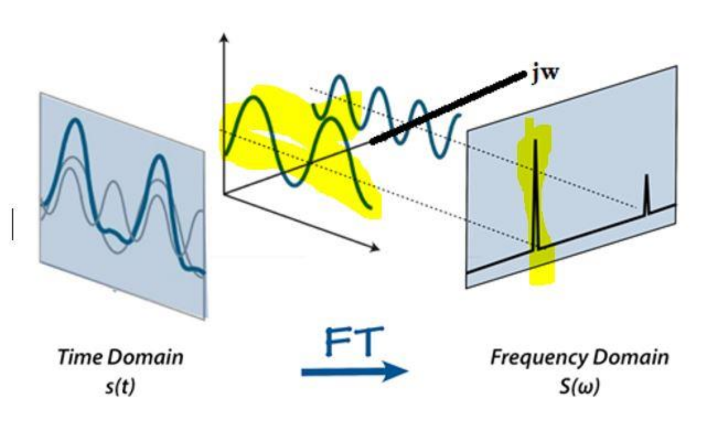
La DFT es periódica, al mostrearse esta señal analógica y periódica resulta un conjunto de muestras que también se repiten, pero ¿cuándo lo hacen?

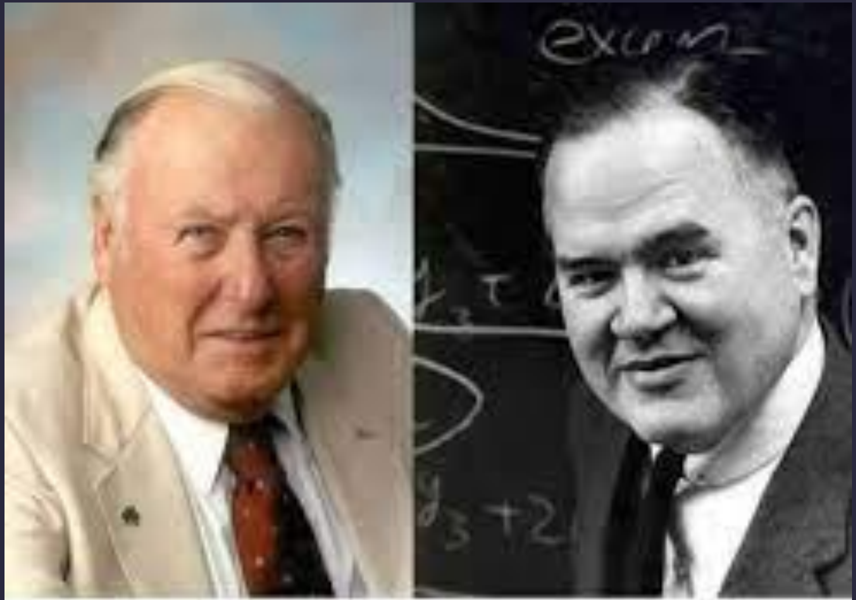
Supongamos:

$$X[4] = \sum_{n=0}^{3} x[n]e^{-i2\pi n}$$
$$X[4] = X[0]\cancelto{1}{e^0} + x[1]\cancelto{1}{e^{-i2\pi}} + x[4]\cancelto{1}{-i4\pi} + x[3]\cancelto{1}{-i6\pi}$$
$$X[4] = 1 + 2 + 0 + (-1)$$
$$X[4] = 2$$

Entonces X[4] = X[2] y se repite cada 4 muestras. ¿Tendrá alguna relación con la señal X[n] que es 4? La respuesta es sí, ya que, x[k] es periódica con periodo N, donde N define el número de muestras de x[n].

/03 /TRANSFORMADA RÁPIDA DE FOURIER





James William Cooley (1926-) John Wilder Tukey (1915-2000)

Fórmulas

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-\frac{i2\pi}{N}kn}, k = 0, ..., N-1$$

〉 DFT para Algoritmo FFT diezmado en tiempo 〈

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X[k] e^{\frac{i2\pi}{N}kn}, n = 0, ..., N-1$$

〉Inversa de la DFT para el FFT diezmado en la frecuencia〈

Algoritmo FFT diezmado en tiempo (base 2, esto es cuando N =2n)

Sea la DFT de x[n]:

$$X[k] = \sum_{r=0}^{N-1} x[n] w_N^{kr}, k = 0, 1, 2, ..., N-1$$

donde $w_N = e^{-\frac{2\pi}{N}}$

Pares

$$X[k] = \sum_{r=0}^{\frac{N}{2}-1} x[2r] w_N^{2rk} + \sum_{r=0}^{\frac{N}{2}-1} x[2r+1] w_N^{(2r+1)k}$$

Impares

$$X[k] = \sum_{r=0}^{\frac{N}{2}-1} x[2r] (w_N^2)^{rk} + w_N^k \sum_{r=0}^{\frac{N}{2}-1} x[2r+1] (w_N^2)^{rk}$$

$$w_N^n = (e^{-\frac{i2\pi}{N}})^n = e^{-\frac{i2\pi n}{N}}$$

Se puede escribir como el recíproco de n:

$$w_N^n = e^{-\frac{\frac{i2\pi}{N}}{n}}$$

Entonces si n=2:

$$w_N^2 = e^{-\frac{2\pi}{\frac{N}{2}}} = w_{\frac{N}{2}}$$

Sustituimos $w_N^2 = w_{\frac{N}{2}}$

$$X[k] = \sum_{r=0}^{\frac{N}{2}-1} x[2r] w_{\frac{N}{2}}^{rk} + w_N^k \sum_{r=0}^{\frac{N}{2}-1} x[2r+1] w_{\frac{N}{2}}^{rk}$$

G[k]: corresponde a la sumatoria de pares.

H[k]: corresponde a la sumatoria de impares.

X[k]: corresponde a la suma de G[k] + H[k].

Finalmente obtenemos:

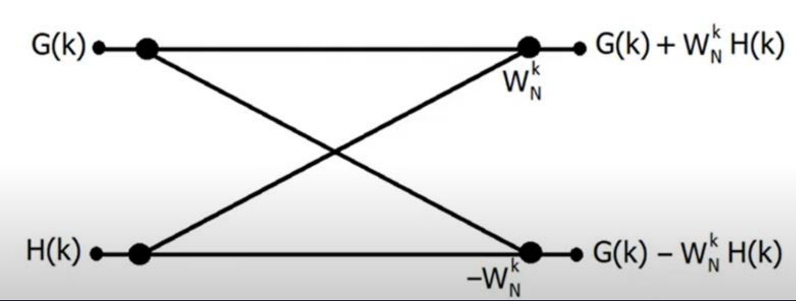
$$X[K] = G[k] + w_N^k H[k], k = 0, 1, 2, 3, ..., \frac{N-1}{2}$$

$$X[k] = G[k] + W_N^k H[k], 0 \leq k \leq \frac{N}{2} - 1$$

$$X[k] = G[k] - w_N^k H[k], \frac{N}{2} - 1 \leq k \leq N - 1$$

La señal X[k] se define como la asociación de 2 DFT más simples.

Esta pareja de ecuaciones puede representar gráficamente con un diagrama de flujos:

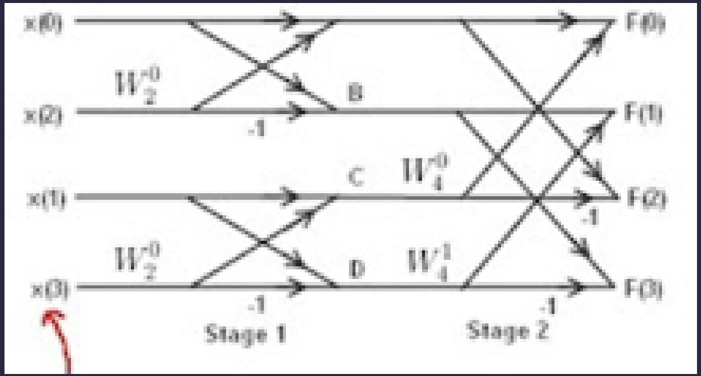


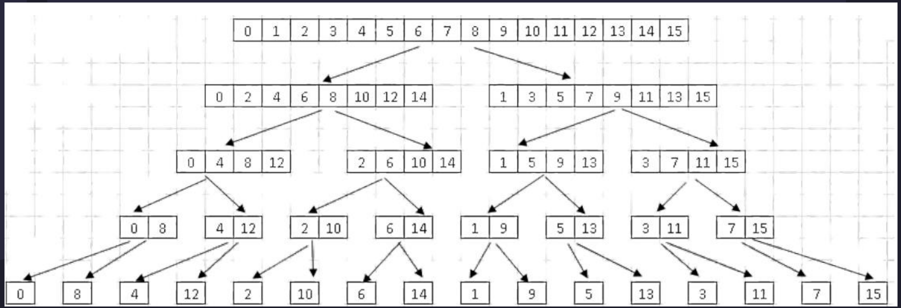
/Diagrama mariposa de la FFT

Por etapa, se requieren N/2 mariposas.
Hay log N etapas.
Se cumple $w^{\frac{N}{2}} = -1$ y por lo tanto,

$$w^{r+\frac{N}{2}} = w^r w^{\frac{N}{2}} = -w^r$$

Se reduce así en un factor 2 los productos complejos:





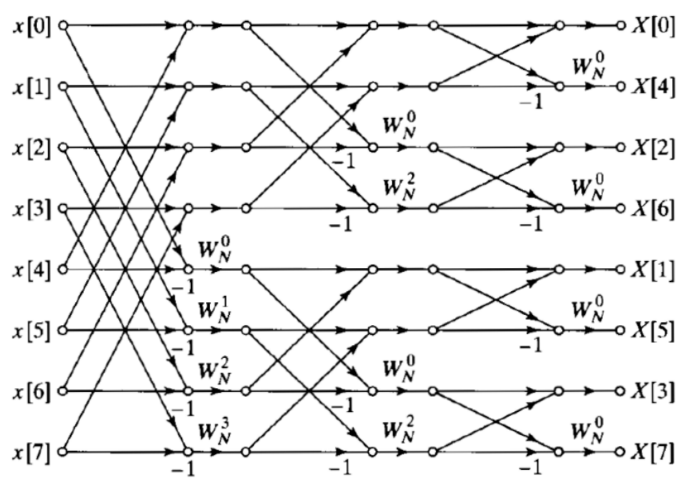
Sample numbers in normal order		Sample numbers after bit reversal	
Decimal	*Binary*	*Decimal*	*Binary*
0	0000	0	0000
1	0001	8	1000
2	0010	4	0100
3	0011	12	1100
4	0100	2	0010
5	0101	10	1010
6	0110	6	0110
7	0111	14	1110
8	1000	1	0001
9	1001	9	1001
10	1010	5	0101
11	1011	13	1101
12	1100	3	0011
13	1101	11	1011
14	1110	7	0111
15	1111	15	1111

/Bit inverso

Algoritmo FFT diezmado en Frecuencia

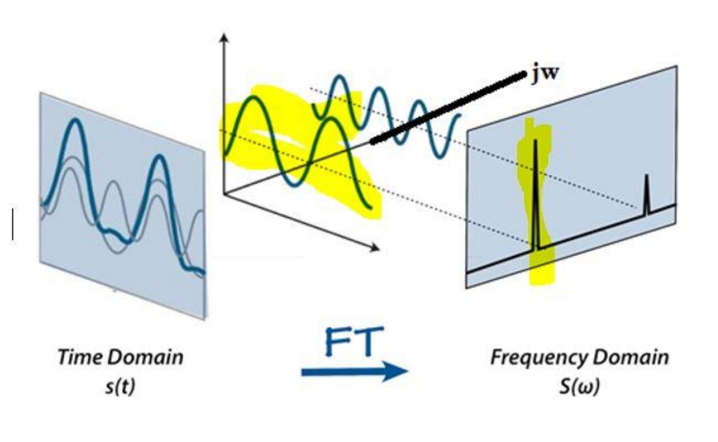
$$X[k] = \sum_{n=0}^{\frac{N}{2}-1} x[n]w_N^{kn} + \sum_{n=\frac{N}{2}}^{N-1} x[n]w_N^{kn}$$

$$= \sum_{n=0}^{\frac{N}{2}-1} x[n]w_N^{kn} + w_N^{\frac{Nk}{2}} \sum_{n=0}^{\frac{N}{2}-1} x\left[n+\frac{N}{2}\right] w_N^{kn}$$



/04

/ALGORITMO

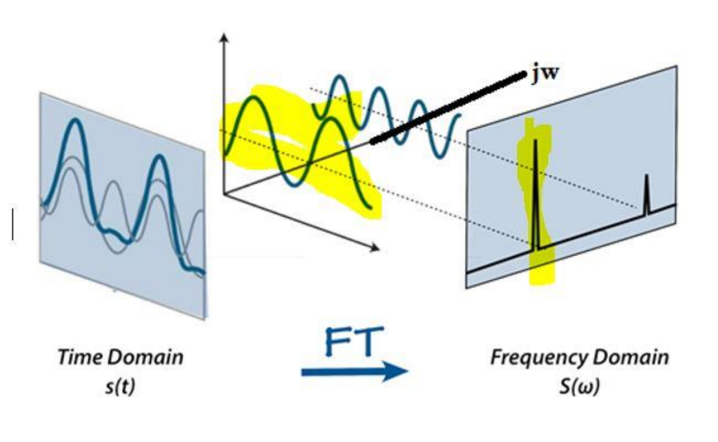


```
//TRANSFORMADA RÁPIDA DE FOURIER
void FFT_t ( arr & v ) {
    int n = v . size () ;
    if( n > 1) {
        arr odd ( n / 2) ;
        arr even ( n / 2) ;
        for (int i = 0; 2 * i < n ; ++ i ) {
            even [ i ] = v [2 * i ];
            odd [ i ] = v [2 * i + 1];
        }
        FFT_t ( even ) ;
        FFT_t ( odd ) ;
        double ang = -2 * PI / n ;
        x_n w (1.0) ;
        x_n wn ( cos ( ang ) , sin ( ang ) ) ;
        for (int k = 0; k < n / 2; ++ k ) {
            v [ k ] = even [ k ] + w * odd [ k ];
            v [ k + n / 2] = even [ k ] - w * odd [ k ];
            w *= wn ;
        }
    }
}
```

```
//TRANSFORMADA INVERSA RÁPIDA DE FOURIER
void IFFT_f ( arr & v ) {
    v = v . apply ( conj ) ;
    FFT_t ( v ) ;
    v = v . apply ( conj ) ;
    v /= v . size () ;
}
```

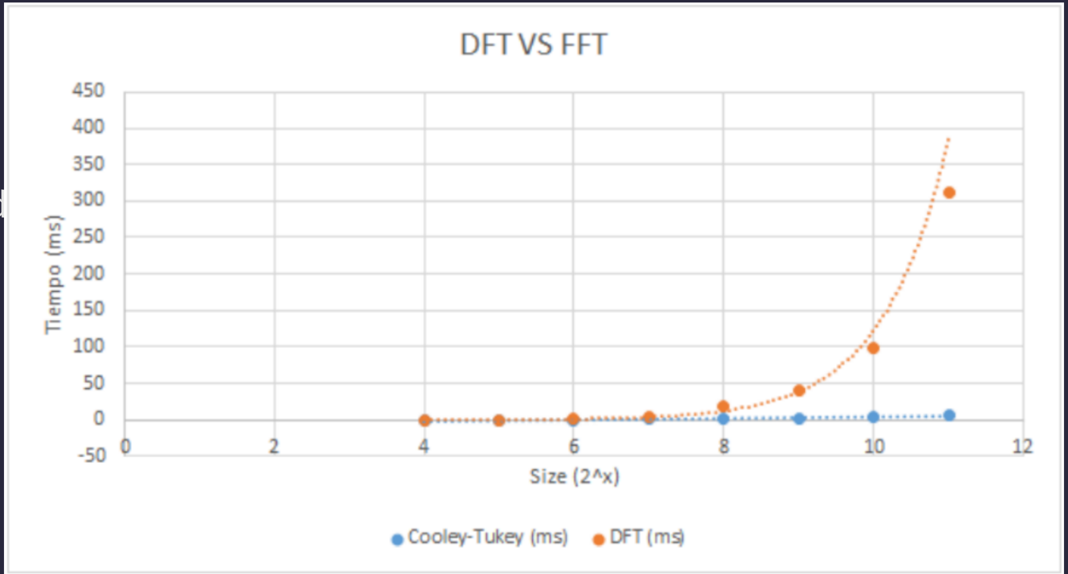
/05

/CONCLUSIONES



/RESULTADOS

d



2	1.594	40.086
2^{10}	4.012	99.645
2^{11}	7.369	311.895