The Descrete Fourier Transform

Scope and Background Reading

This session introduces the z-transform which is used in the analysis of discrete time systems. As for the Fourier and Laplace transforms, we present the definition, define the properties and give some applications of the use of the z-transform in the analysis of signals that are represented as sequences and systems represented by difference equations.

The material in this presentation and notes is based on Chapter 10 of Steven T. Karris, Signals and Systems: with Matlab Computation and Simulink Modelling, 5th Edition (http://site.ebrary.com/lib/swansea/docDetail.action?docID=10547416) from the **Required Reading List**. Additional coverage is to be found in Chapter 12 of Benoit Boulet, Fundamentals of Signals and Systems (http://site.ebrary.com/lib/swansea/docDetail.action?docID=10228195) from the **Recommended Reading List**.

Agenda

- The discrete time fourier transform
- Even and Odd Properties of the DFT
- Common Properties and Theorems of the DFT
- Sampling Theorem, Windows, and the Picket Fence Effect

Introduction

- Fourier series: periodic and continuous time function leads to a non-periodic discrete frequency function.
- Fourier transform: non-periodic and continuous function leads to a non-periodic continuous frequency function.
- z and inverse z transforms produce a periodic and continuous frequency function, since they are evaluated on the unit circle.

Note

Frequency spectrum of a discrete time function $f[n]$ is obtained from its z-transform by substituting $z = e^{sT} = e^{j\omega T}$ as we saw from the mapping of the s-plane to the z-plane. This is continuous as there are an infinite number of points in the interval 0 to 2π; and it is periodic because for any point ωT there is an equivalent point $\omega T + 2N\pi$ later.

In practice, to compute the spectrum for a discrete time (DT) system, we only compute a finite number of equally spaced points.

In this session, we will see that a periodic and discrete time function results in a periodic and discrete frequency function.

For conveniemce we summarize these facts in a table:

Topic	Time Function	Frequency Function
Fourier Series	Continuous, Periodic	Discete, Non-Periodic
Fourier Transform	Continuous, Non-Periodic	Continuous, Non-Periodic
Z Transform	Discrete, Non-Periodic	Continuous, Periodic
Discrete Fourier Transform	Discrete, Periodic	Discrete, Periodic

Notation

In the following we shall denote a DT signal as $x[n]$ and its disctete frequency function as $X[m]$.

Z-Transform

Recall that

$$F(z) = \mathcal{Z}f[n] = \sum_{n=0}^{\infty} f[n]z^{-n}.$$

The value of this function on the unit circle in the z-plave will be

$$F(e^{j\omega T}) = \sum_{n=0}^{\infty} f[n]e^{-jn\omega T}.$$

This is an infinite sum so to compute it, we need to truncate it. Let's assume that instead of an infinite number of points, we have N, equally distributed around the unit circle, then the truncated version will be:

$$X[m] = \sum_{n=0}^{N-1} x[n]e^{-j2\pi \frac{mn}{N}}$$

where

$$\omega = \left(\frac{2\pi}{NT}\right) m$$

and $m = 0, 1, 2, \ldots, N-1$.

We refer to this equation as the N-point Discrete-time Fourier Transform (DFT) of $x[n]$.

The inverse DFT is defined as

$$x[n] = \frac{1}{N}\sum_{m=0}^{N-1} X[m]e^{j2\pi \frac{mn}{N}}$$

for $n = 0, 1, 2, \ldots, N-1$.

Note the symmetry of the DFT and the Inverse DFT!

In general, the DFT is complex, and thus it can be expressed as

$$X[m] = \mathfrak{R}\left\{X[m]\right\} + \mathfrak{I}\left\{X[m]\right\}$$

for $m = 0, 1, 2, \ldots, N-1$.

Since

$$e^{-j2\pi\frac{mn}{N}} = \cos\left(\frac{2\pi mn}{N}\right) + j\sin\left(\frac{2\pi mn}{N}\right)$$

the DFT can be expresssed as

$$X[m] = \sum_{n=0}^{N-1} x[n]e^{-j2\pi\frac{mn}{N}} = \sum_{n=0}^{N-1} x[n]\cos\left(\frac{2\pi mn}{N}\right) + j\sum_{n=0}^{N-1} x[n]\sin\left(\frac{2\pi mn}{N}\right).$$

For $n = 0$ this reduces to

$$X[m] = x[0].$$

Then the real part of $X[m]$ is

$$\mathfrak{R}\left\{X[m]\right\} = x[0] + \sum_{n=1}^{N-1} x[n]\cos\left(\frac{2\pi mn}{N}\right) \quad \text{for} \quad m = 0, 1, 2, \ldots, N-1$$

and the imaginary part is

$$\mathfrak{I}\left\{X[m]\right\} = -\sum_{n=1}^{N-1} x[n]\cos\left(\frac{2\pi mn}{N}\right) \quad \text{for} \quad m = 0, 1, 2, \ldots, N-1$$

.

Note that the summations are from 1 to $N-1$ because $n = 0$ is covered in the real term, and as $x[0]$ is real, it is zero in the corresponding imaginary term.

Example 1

A discrete time signal is defined by the sequence $x[0] = 1$, $x[1] = 2$, $x[2] = 2$, $x[3] = 1$, and $x[n] = 0$ for all other values of n. Compute the frequency components $X[m]$.

Solution 1

- Compute the N point DFT for $\Re\left\{X[m]\right\}$.

- Compute the four point DFT for $\Im\left\{X[m]\right\}$.

- Add these together to find $X[m]$.

Example 2

Use the inverse DFT to compute the discrete-time sequence $x[n]$ from $X[m]$.

Solution 2

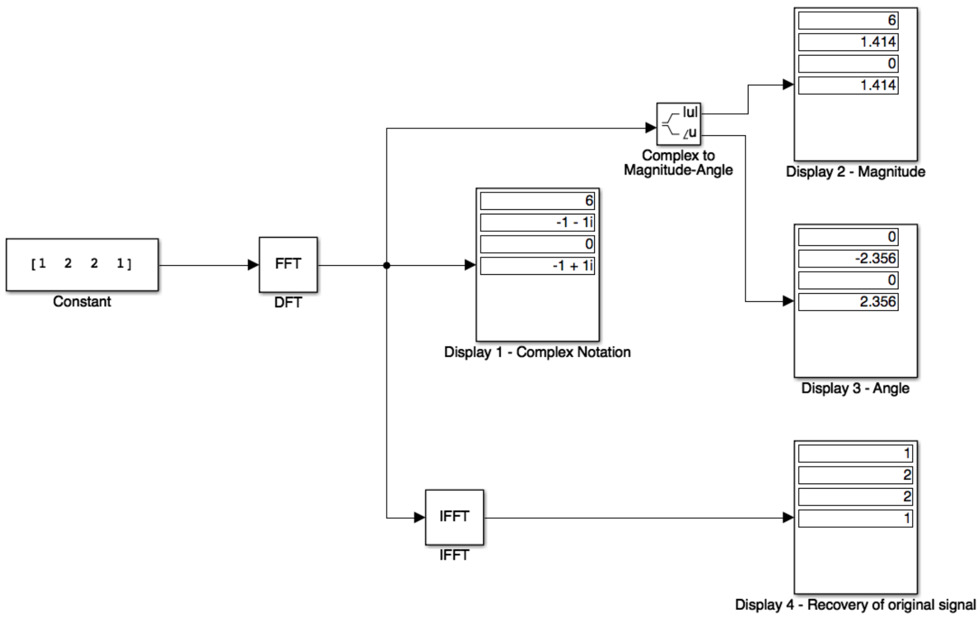
- Write down the expression $x[n]$ in terms of $X[m]$.

- Compute $x[0]$ from this result.

- Repeat for $x[1]$, $x[2]$ and $x[3]$.

Simulink model of the DFT

See dft_ex10_1.slx (https://github.com/cpjobling/EG-247-Resources/blob/master/week10/matlab/dft_ex10_1.slx?raw=true)



Try inputting your student number.

MATLAB model of the DFT

Uses the DFT (https://github.com/cpjobling/EG-247-Resources/blob/master/week10/matlab/dft.m) and IDFT (https://github.com/cpjobling/EG-247-Resources/blob/master/week10/matlab/idft.m) functions.

See ex10_1.m (https://github.com/cpjobling/EG-247-Resources/blob/master/week10/matlab/ex10_1.m)

```
>> ex10_1

xn =

     1     2     2     1


Xm =

   6.0000 + 0.0000i  -1.0000 - 1.0000i   0.0000 - 0.0000i  -1.0000 + 1.000
0i


xn =

   1.0000 - 0.0000i   2.0000 - 0.0000i   2.0000 + 0.0000i   1.0000 + 0.000
0i
```

Again, try inputting your student number.

A useful compact notation

The term

$$e^{-j(2\pi)}/N$$

is a rotating vector where the range $0 <= \theta <= 2\pi$ is divided into $360/N$ equal segments.

It is convenient to represent this as W_N, that is

$$W_N = e^{-\frac{j2\pi}{N}}$$

and consequently,

$$W_N^{-1} = e^{\frac{j2\pi}{N}}.$$

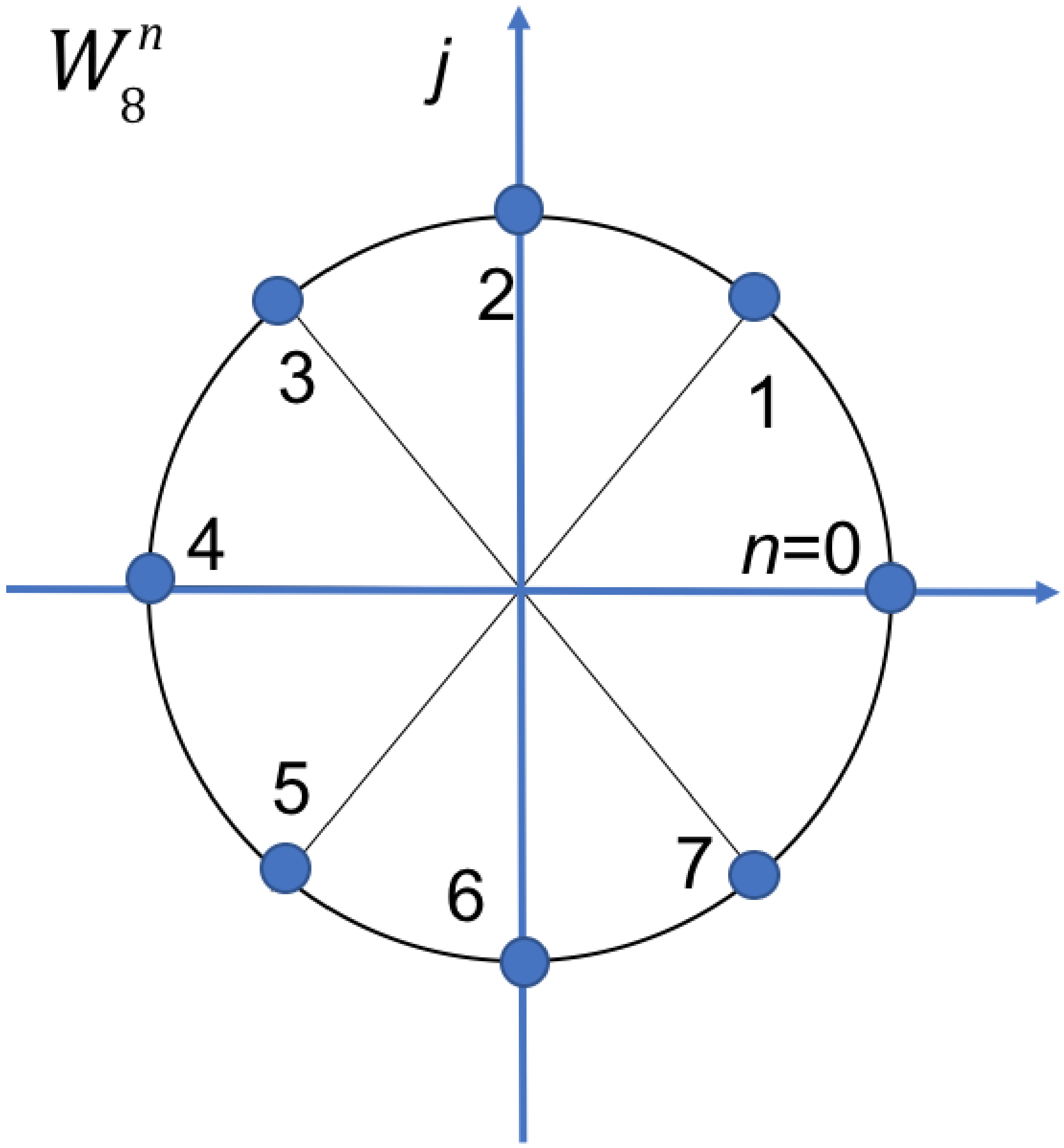
Example 3

Compute the complex numbers represented by the rotating vector W_8

Solution 3

- Rewrite W_8 in exponential form

- Visualize on unit circle



- Complete this table

n	θ	**Real**	**Imaginary**	W_8^n
0	0	1	0	1

Using this notation, the DFT and inverse DFT pairs are represented as:

$$X[m] = \sum_{n=0}^{N-1} x[n] W_N^{nm}$$

and

$$x[n] = \frac{1}{N} \sum_{n=0}^{N-1} x[n] W_N^{-nm}$$

MATLAB implementation of DFT

Using the W notation, it is very easy to write a function to implement the DFT. For example, consider dft.m (https://github.com/cpjobling/EG-247-Resources/blob/master/week10/matlab/dft.m):

```
function [ Xm ] = dft( xn, N )
% Computes Discrete Fourier Transform
% -----------------------------------
% [Xm]  = dft(xn, N)
% Xm = DFT coeff. array over 0 <= m <= N-1
% xn = N-point finite-duration sequence
%  N = length of DFT
%
n = [0:1:N-1];          % row vector for n
m = [0:1:N-1];          % row vector for m
WN = exp(-j*2*pi/N);    % Wn factor
nm = n'*m;              % creates an N by N matrix of nm values
WNnm = WN .^ nm;        % DFT matrix
Xm = xn * WNnm;         % row vector of DFT coefficients
```

Similarly for the inverse DFT idft.m (https://github.com/cpjobling/EG-247-Resources/blob/master/week10/matlab/idft.m):

```
function [ xn ] = idft( Xm, N )
% Computes Inverse Discrete Fourier Transform
% -------------------------------------------
% [xn]  = idft(Xm, N)
% xn = N-point sequence over 0 <= n <= N-1
% Xm = DFT coeff. array over 0 <= m <= N-1
%  N = length of DFT
%
n = [0:1:N-1];          % row vector for n
m = [0:1:N-1];          % row vector for m
WN = exp(-j*2*pi/N);    % Wn factor
nm = n'*m;              % creates an N by N matrix of nm values
WNnm = WN .^ (-nm);     % DFT matrix
xn = (Xm * WNnm)/N;     % row vector for IDFT values
```

Notes

In the remainder of these notes, the correspondence between $x[n]$ and $X[m]$ will be written

$$x[n] \Leftrightarrow X[m]$$

In example 2, we found that, although the DT sequence $x[n]$ was real, the discrete frequency (DF) sequence was complex. However, in most applications we are interested in the magnitude and phase of the DF, that is

$$|X[m]|$$

and

$$\angle X[m]$$

.

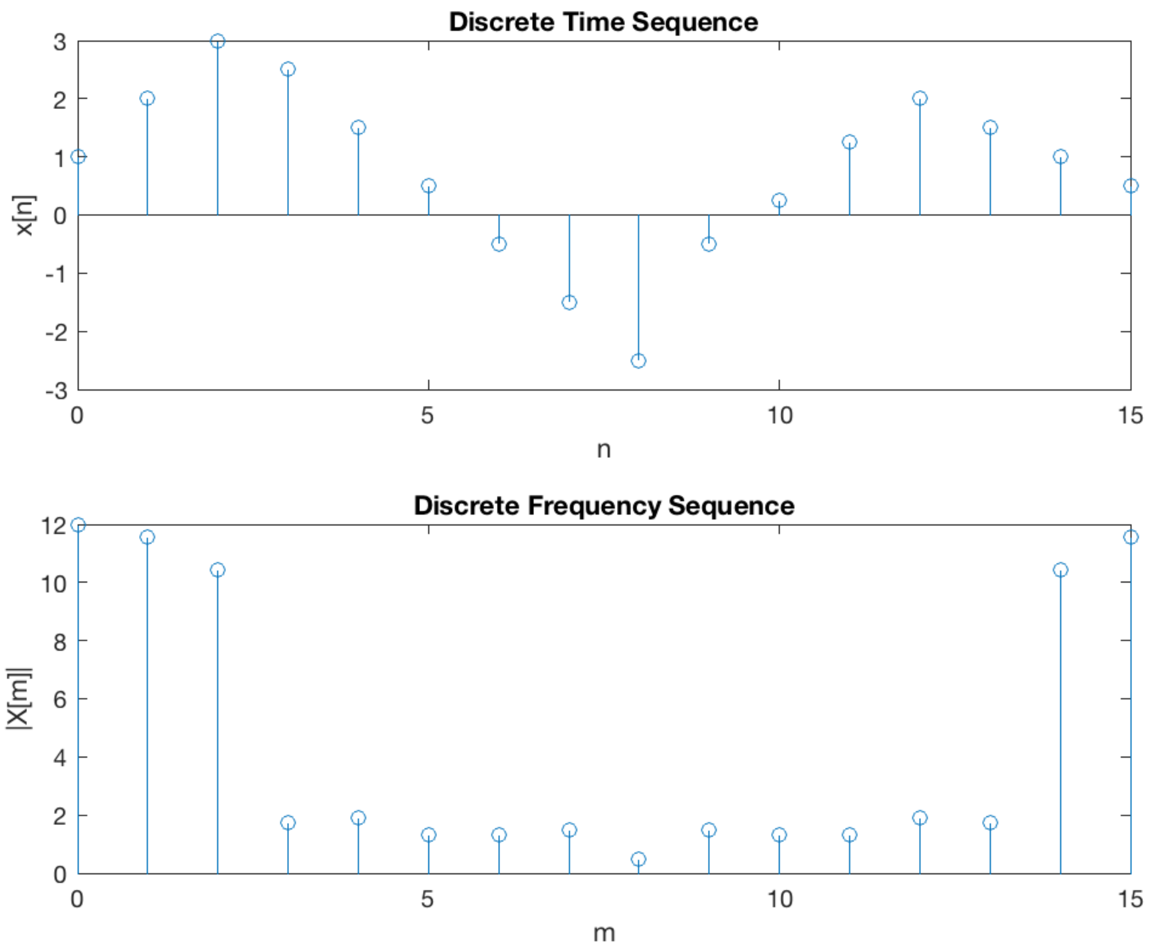
Example 4

Use MATLAB to compute the magnitude of the frequency components of the following DT function:

n	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
$x[n]$	1.0	1.5	2.0	2.3	2.7	3.0	3.4	4.1	4.7	4.2	3.5	3.6	3.2	2.9	2.5	1.8

We will compute this in class and make some comments afterwards.

Example: $x[n]$ and $X[m]$ for a N=16 point DFT (produced by script repeat.m (https://github.com/cpjobling/EG-247-Resources/blob/master/week10/matlab/repeat.m)).



Points to note:

- $X[0] = 12$ is the DC component of the DT sequence.
- After the $|X[8]| = 1.4872$ term, the magnitude of the frequency values for the range $9 <= m <= 15$ are the mirror image of the values for the range $0 <= m <= 7$.
- This is not a coincidence, in fact if $x[n]$ is an N-point *real discrete time function*, only $N/2$ of the *frequency components* of $|X[m]|$ *are unique*.

Even and Odd Properties of the DFT

The discrete time and discrete frequency functions are defined as *even* or *odd* in according to the following relations:

Even time function: $f[N-n] = f[n]$

Odd time function: $f[N-n] = -f[n]$

Even frequency function: $F[N-m] = F[m]$

Odd frequency function: $F[N-m] = -F[m]$

Even and odd properties of the DFT

Discrete time sequence $f[n]$	Discrete frequency sequence $F[m]$
Real	Complex Real part is Even Imaginary part is Odd
Raal and Even	Real and Even
Raal and Odd	Imaginary and Even
Imaginary	Complex Real part is Odd Imaginary part is Even
Imaginary and Even	Imaginary and Even
Imaginary and Odd	Real and Odd

It is not difficult to prove these by expanding

$$X[m] = \sum_{n=0}^{N-1} x[n] W_N^{mn}$$

into its real and imaginary parts using Euler's identity and considering the cosine (even) and sine (odd) terms that result.

Common Properties and Theorems of the DFT

We denote the DFT and inverse DFT using as follows:

$$X[m] = \mathcal{D}\left\{x[n]\right\}$$

and

$$x[n] = \mathcal{D}^{-1}\left\{X[m]\right\}$$

We then state the following useful properties. For proofs, see Karris, 10.3. Not examined.

Linearity

$$ax_1[n] + bx_2[n] + \cdots \Leftrightarrow aX_1[m] + bX_2[m] + \cdots$$

Time-shift

$$x[n-k] \Leftrightarrow W_n^{km} X[m]$$

Frequency shift

$$W_n^{-km} x[n] \Leftrightarrow X[m-k]$$

Time convolution

$$x[n] * h[n] \Leftrightarrow X[m]H[m]$$

Frequency convolution

$$x[n]y[n] \Leftrightarrow \frac{1}{N}\sum_{k=0}^{N-1} X[k]Y[m-k] \Leftrightarrow X[m] * Y[m]$$

Sampling Theorem, Windows, and the Picket Fence Effect

Sampling Theorem

The *sampling theorem* known as *Nyquist/Shannon's Sampling Theorem* (see wp>Nyquist/Shannon Sampling Theorem (https://en.wikipedia.org/wiki/Nyquist%E2%80%93Shannon_sampling_theorem)), states that *if a continuous time function, $f(t)$ is band-limited with its highest frequency component less that W, then $f(t)$ can be completely recovered from its sampled values, $f[n]$, f the sampling frequency is equal or greater than $2W$.

For example, say the highest frequency component in a signal is 18 kHz, this signal must be sampled at $2 \times 18 = 36$ kHz or higher so that it can be completely specified by its sampled values. If the sampled frequency remains the same, i.e., 36 kHz, and the highest frequency in the signal is increased, to say 25 kHz, this signal cannot be recovered by a Digital to Analogue Converter (DAC).

Since many real signals are not band limited, a typical digital signal processing system will include a low-pass filter, often called a *pre-sampling-filter* or simply a *pre-filter*, to ensure that the highest frequency signal allowed into the system will be equal or less than the sampling frequency so that the signal can be recovered. The highest frequency allowed in the system is referred to as the Nyquest frquency denoted as f_n.

If the signal is not band limited, or the sampling frequency is too low, the spectral components of the signal will overlap each other and this is called *aliasing*. To avoid aliasing, we must increase the sampling rate.

Windowing

A DT signal may have an infinite length; in this case it must be limited to a finite interval before it is sampled. We can terminate the signal at a defined number of terms by multiplying it by a *window function*. There are several window functions that are used in practice such as the *rectangular*, *triangular*, *Hanning*, *Hamming*, *Kaiser*, etc. Window functions, and there design, are outside the scope of this module, but are discussed in Appendix E of Karris.

All I will say here is that the window function must be carefully chosen to avoid the signal being terminated too abrubtly and causing *leakage* -- that is a spread of the spectrum outside the bounds imposed by the window.

Picket fence

A third problem introduced by the DFT is the fact that as the spectrum of the DFT is not continuous, important frequencies may fall between spectrum lines and therefore not be detected. This is called the picket fence effect, named after the white fences seen in the suburbs in US movies. A way round this is to padd the signal with zeros so that the effective period changes and therefore changes the locations of the spectral lines.

You should remember that the sampling theorem states under what conditions a signal may be recovered. It does not guarantee that all significant frequencies will be present in the sampled signal.

A summary of the important features of sampling and the DFT

- N is the number of samples in frequency.
- f_s sampling frequency, samples per seconds.
- T_t period of a periodic DT function.

- t_s interval between the N samples in time period T_t.
- f_f period of a periodic DF function.
- F_s interval between the N samples in frequency period T_f.

The relationships between these quantities are:

$$\begin{aligned} t_s &= \frac{T_t}{N} \\ f_s &= \frac{1}{t_s} \\ f_f &= \frac{F_f}{N} \\ t_s &= \frac{1}{F_f} \\ f_f &= \frac{1}{T_t} \end{aligned}$$

We will add these to the figure above in class.

Example 5

The period of a periodic DT function is 0.125 ms and it is sampled at 1024 equally spaced points. It is assumed that with this number of samples, the sampling theorem is satisfied and thus there will be no aliasing.

1. Compute the period of the frequency spectrum in kHz
2. Compute the interval between frequency components in kHz
3. Compute the sampling frequency f_s.
4. Compute the Nyquist frequency f_n.

Solution

To be done in class.

- Compute the period of the frequency spectrum in kHz

- Compute the interval between frequency components in kHz

- Compute the sampling frequency f_s.

- Compute the Nyquist frequency f_n.

Summary

- The discrete time fourier transform
- Even and Odd Properties of the DFT
- Common Properties and Theorems of the DFT
- Sampling Theorem, Windows, and the Picket Fence Effect

Next session

- The Fast Fourier Transform

(without the mathematics)

Answers

Example 1

$$\begin{aligned} X(0) &= 6 \\ X(1) &= -1 - j \\ X(2) &= 0 \\ X(3) &= -1 + j \end{aligned}$$

Example 3

$$W_8 = \left[1, \frac{1}{\sqrt{2}} + j\frac{1}{\sqrt{2}}, j, -\frac{1}{\sqrt{2}} + j\frac{1}{\sqrt{2}}, -1, -\frac{1}{\sqrt{2}} - j\frac{1}{\sqrt{2}}, -j, \frac{1}{\sqrt{2}} - j\frac{1}{\sqrt{2}}\right]$$

Example 4

1. 8192 kHz (8.2 Mhz)
2. 8 kHz
3. 8.2 Mhz
4. 4.1 Mhz

Homework

Try Exercise 1 and Exercise 2 in Karris 10.8 by hand.

For the exam, I wouldn't expect you to compute the whole sequence for a signal with more than 4 samples. However, you will need to be able to compute the DFT $x[n]$ and IDFT $X[m]$ of an 8-point sequence for any single value n or m.