EITF75 Systems and Signals

Summary of Course

Fredrik Rusek

EITF75 Systems and Signals

Sampling and reconstruction

EITF75 Systems and Signals

A/D and D/A

Signal $\xrightarrow{x(t)}$ sampling $\xrightarrow{x(n)}$

$X_a(F)$ $\qquad$ $X(f)$

Key step is to understand what X(f) looks like in terms of $X_a(F)$

$$X(f) = F_s \sum_{k=-\infty}^{\infty} X_a((f-k)F_s)$$

If sampling is to sparse, there is aliasing.

We find X(f) by the "folding technique"

EITF75 Systems and Signals

Example: Folding

Signal

F_s = 4000 Hz

Folding

Step 1: Identify $F_s/2$

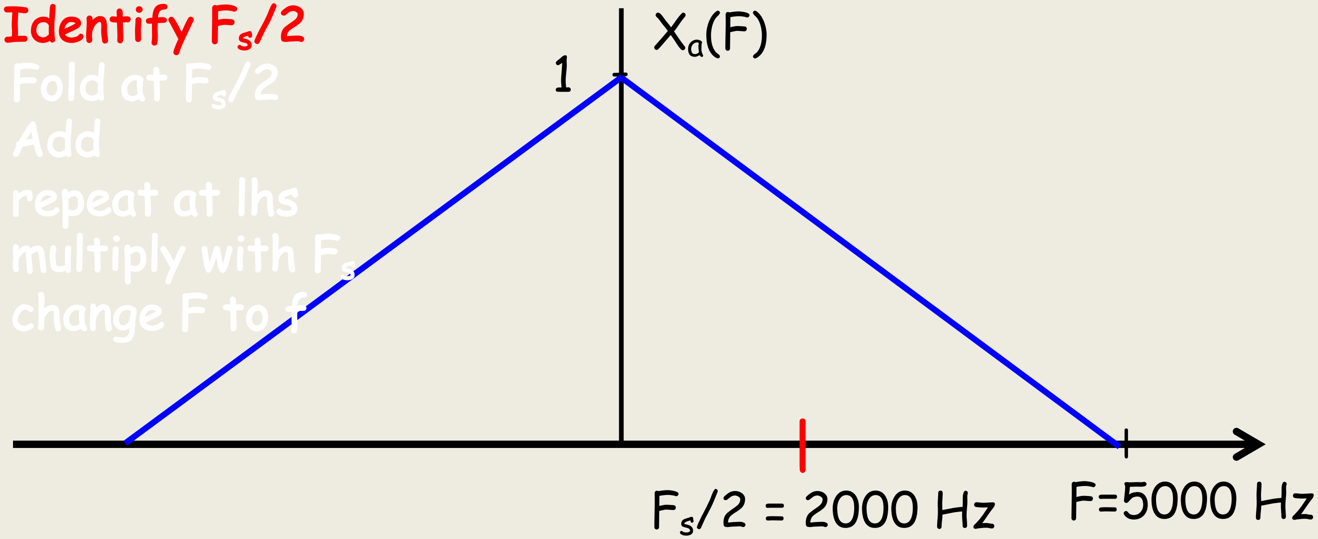
Step 2: Fold at $F_s/2$

Step 3: Add

Step 4: repeat at lhs

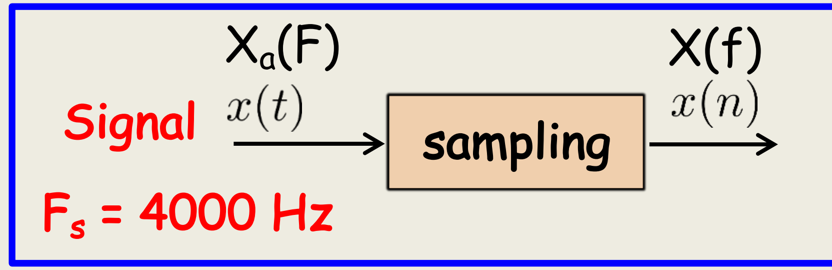
Step 5: multiply with F_s

Step 6: change F to f



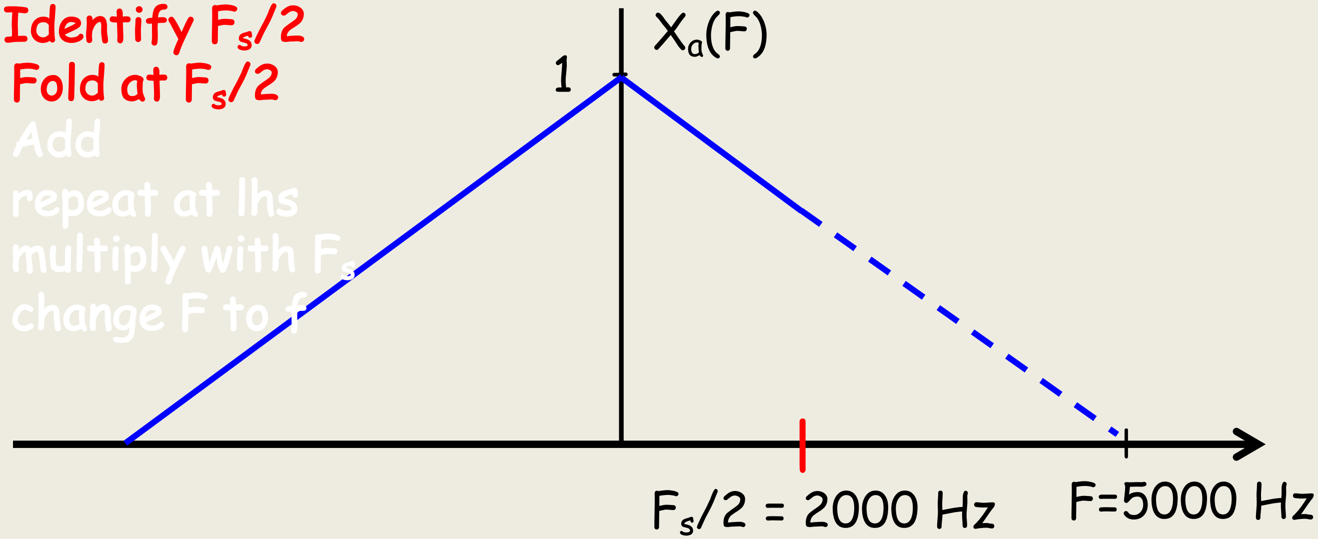
EITF75 Systems and Signals

Example: Folding



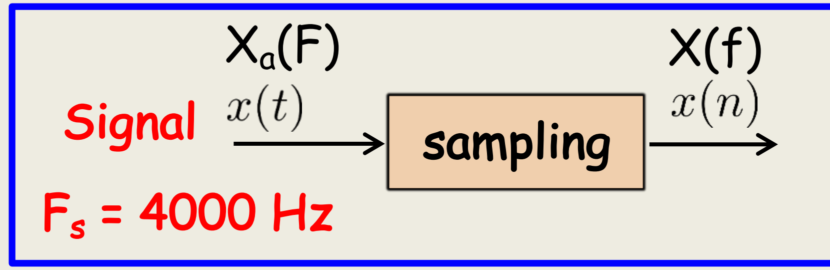
Folding

Step 1: Identify $F_s/2$
Step 2: Fold at $F_s/2$
Step 3: Add
Step 4: repeat at lhs
Step 5: multiply with F_s
Step 6: change F to f



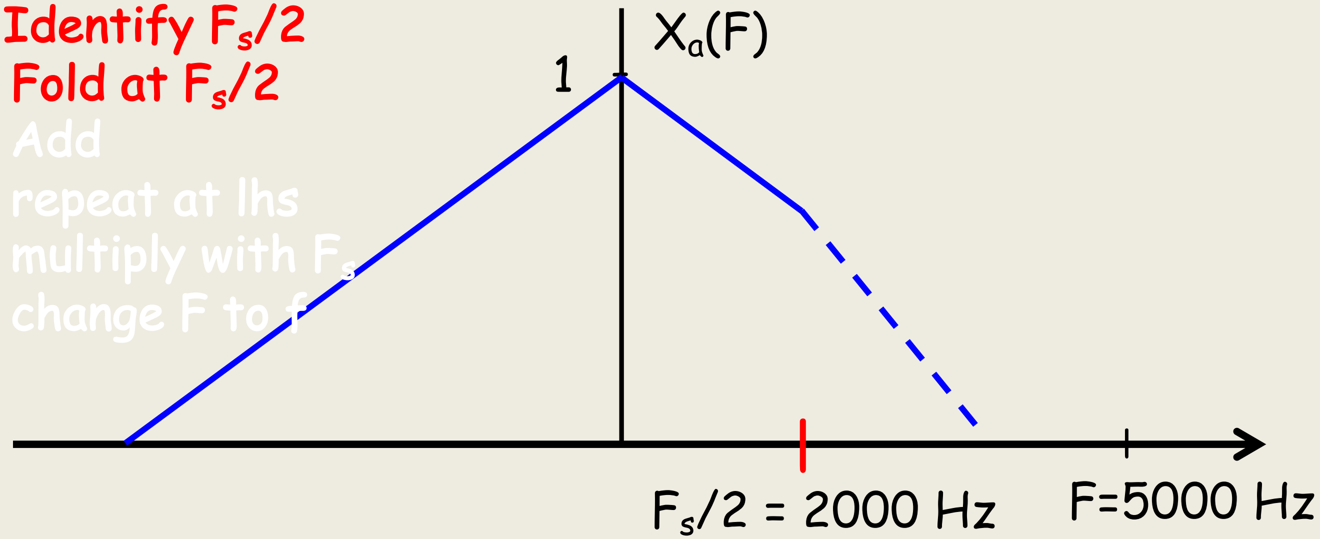
EITF75 Systems and Signals

Example: Folding



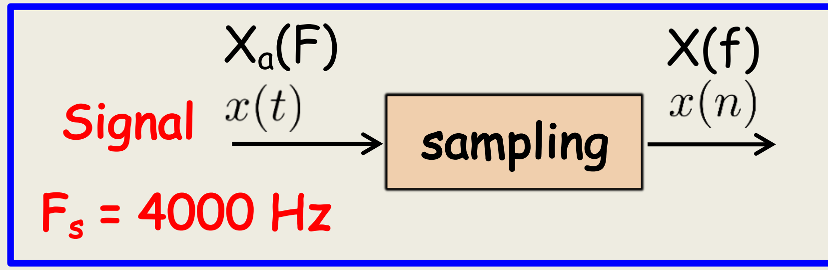
Folding

Step 1: Identify $F_s/2$
Step 2: Fold at $F_s/2$
Step 3: Add
Step 4: repeat at lhs
Step 5: multiply with F_s
Step 6: change F to f



EITF75 Systems and Signals

Example: Folding



Folding

Step 1: Identify $F_s/2$

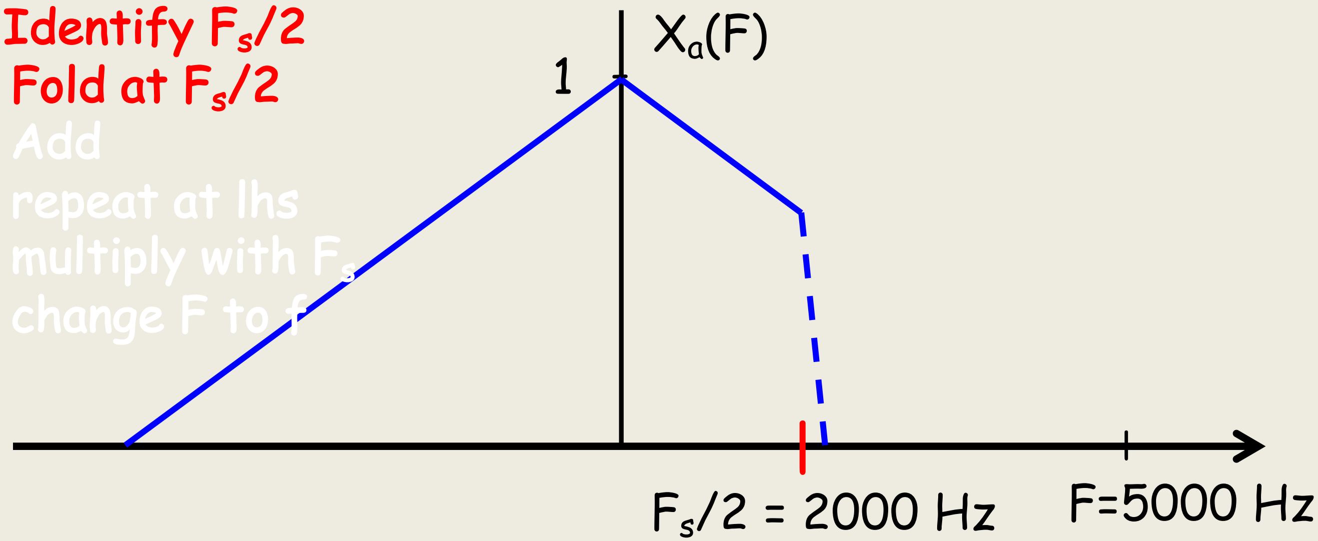
Step 2: Fold at $F_s/2$

Step 3: Add

Step 4: repeat at lhs

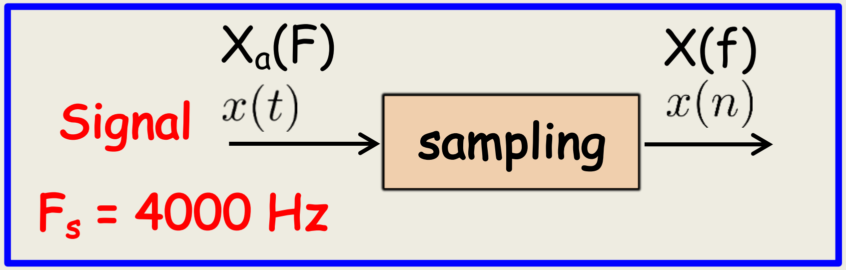
Step 5: multiply with F_s

Step 6: change F to f



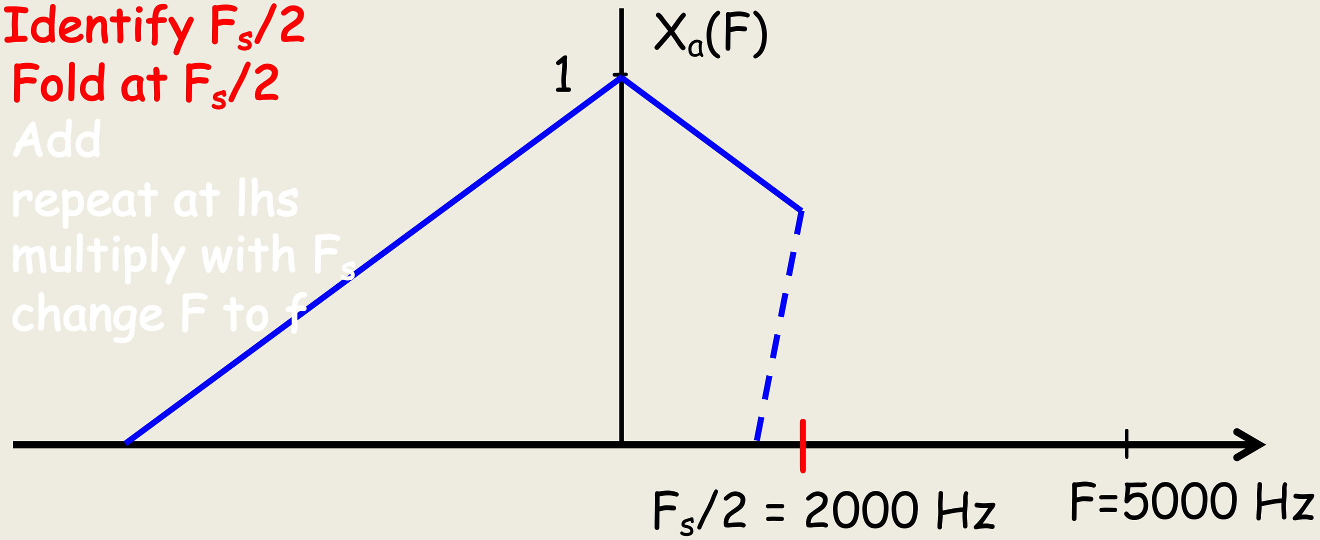
EITF75 Systems and Signals

Example: Folding



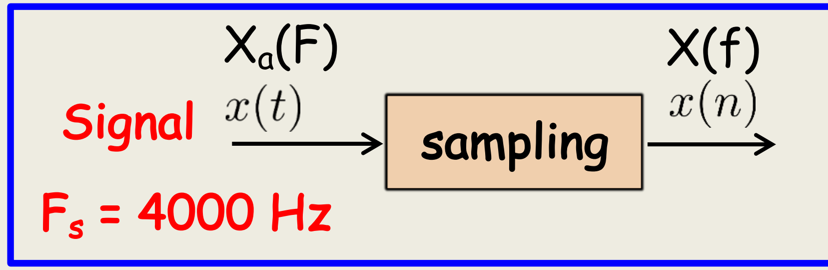
Folding

Step 1: Identify $F_s/2$
Step 2: Fold at $F_s/2$
Step 3: Add
Step 4: repeat at lhs
Step 5: multiply with F_s
Step 6: change F to f



EITF75 Systems and Signals

Example: Folding



Folding

Step 1: Identify $F_s/2$

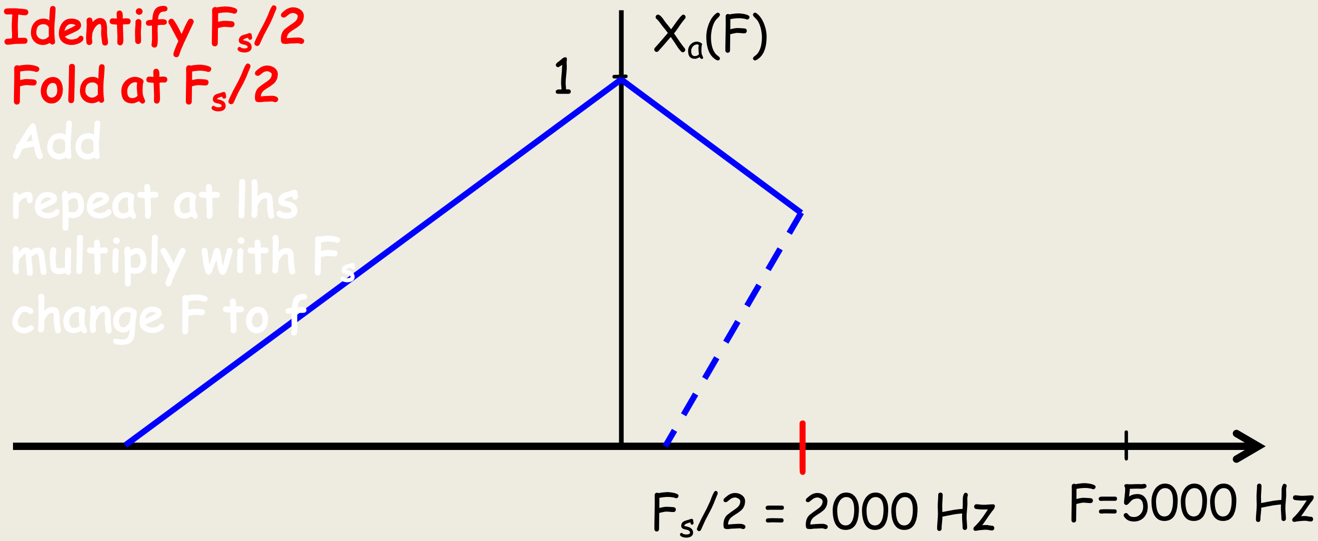
Step 2: Fold at $F_s/2$

Step 3: Add

Step 4: repeat at lhs

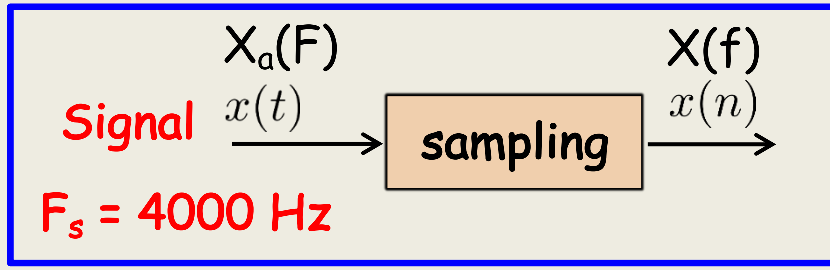
Step 5: multiply with F_s

Step 6: change F to f



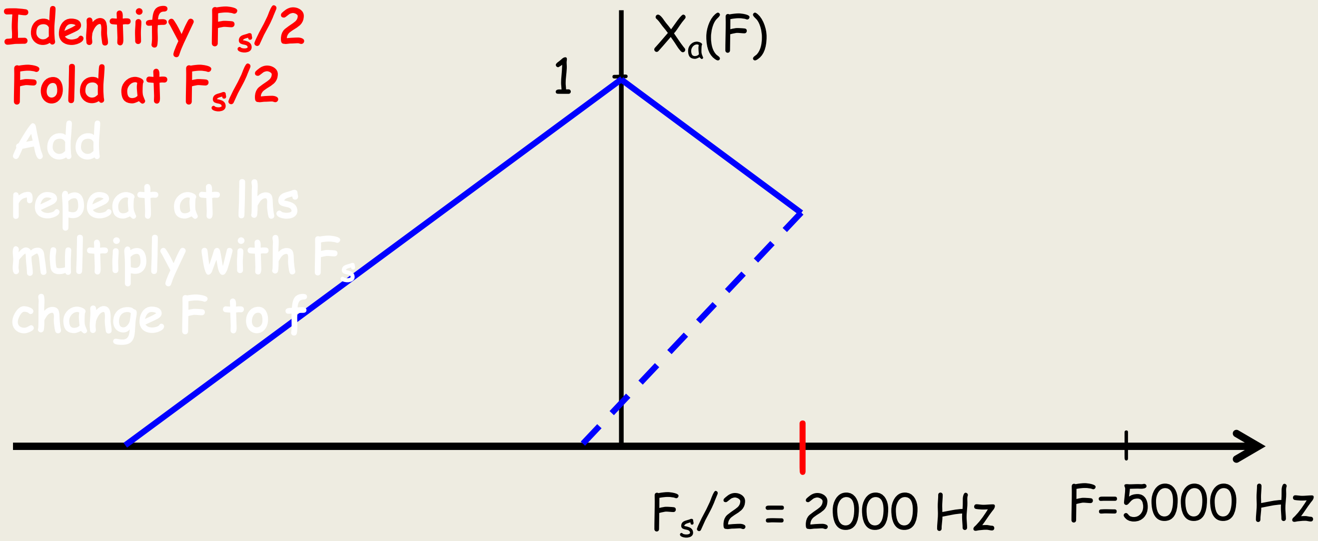
EITF75 Systems and Signals

Example: Folding



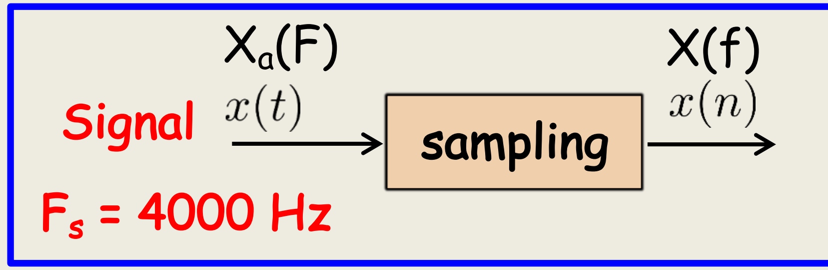
Folding

Step 1: Identify $F_s/2$
Step 2: Fold at $F_s/2$
Step 3: Add
Step 4: repeat at lhs
Step 5: multiply with F_s
Step 6: change F to f



EITF75 Systems and Signals

Example: Folding



Folding

Step 1: Identify $F_s/2$

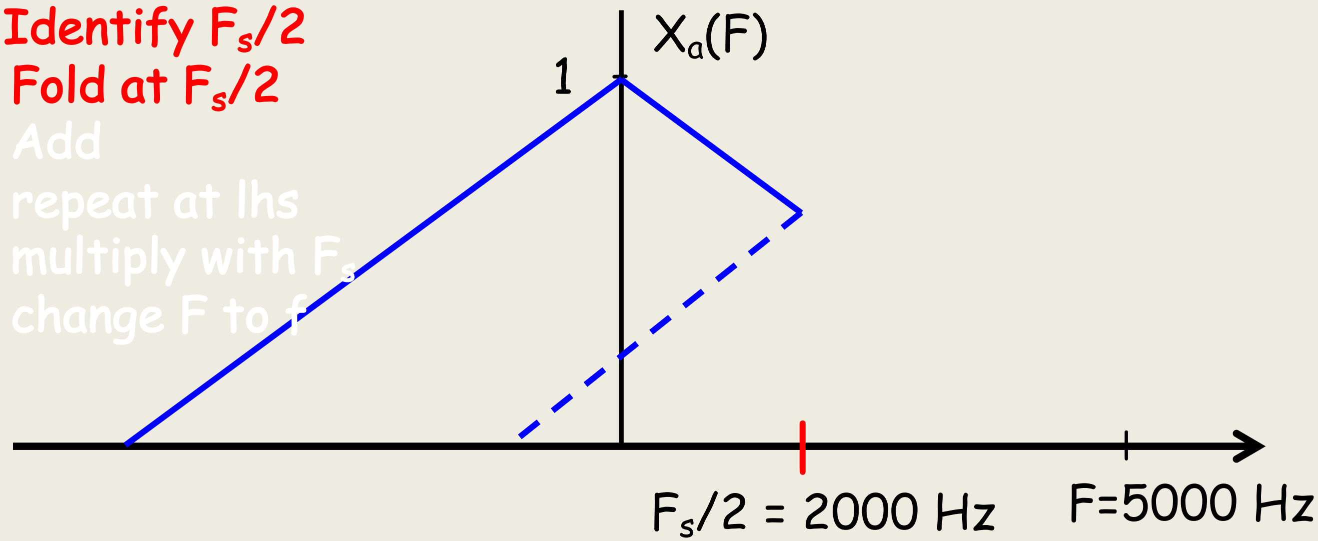
Step 2: Fold at $F_s/2$

Step 3: Add

Step 4: repeat at lhs

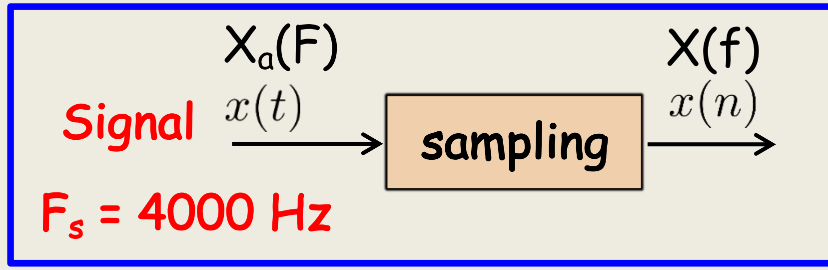
Step 5: multiply with F_s

Step 6: change F to f



EITF75 Systems and Signals

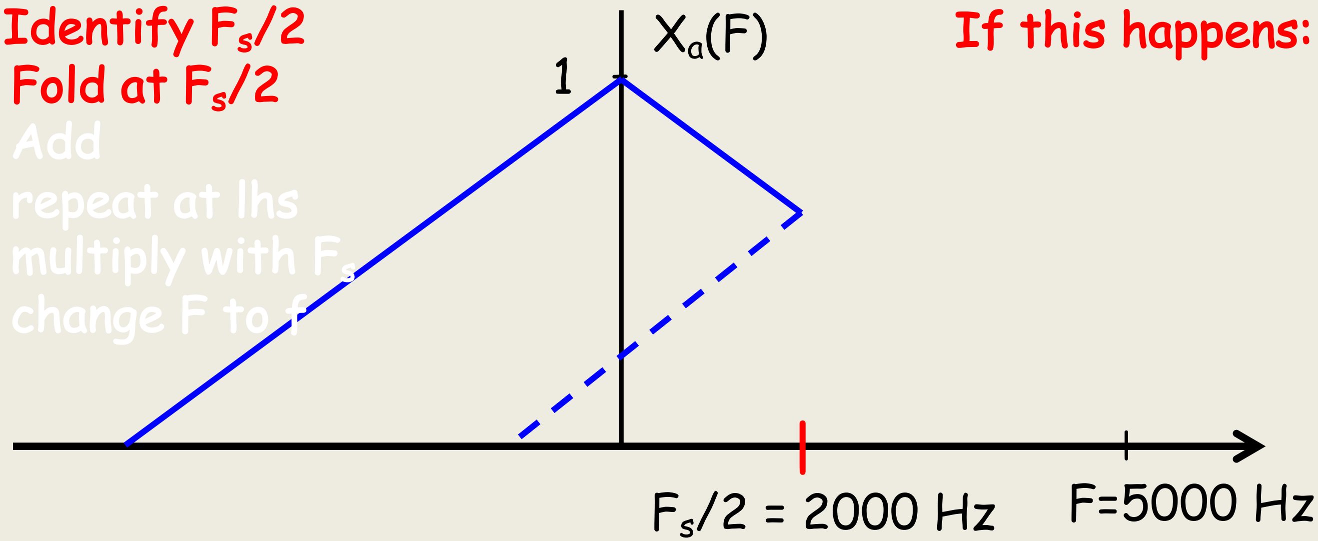
Example: Folding



Folding

Step 1: Identify $F_s/2$
Step 2: Fold at $F_s/2$
Step 3: Add
Step 4: repeat at lhs
Step 5: multiply with F_s
Step 6: change F to f

If this happens:



EITF75 Systems and Signals

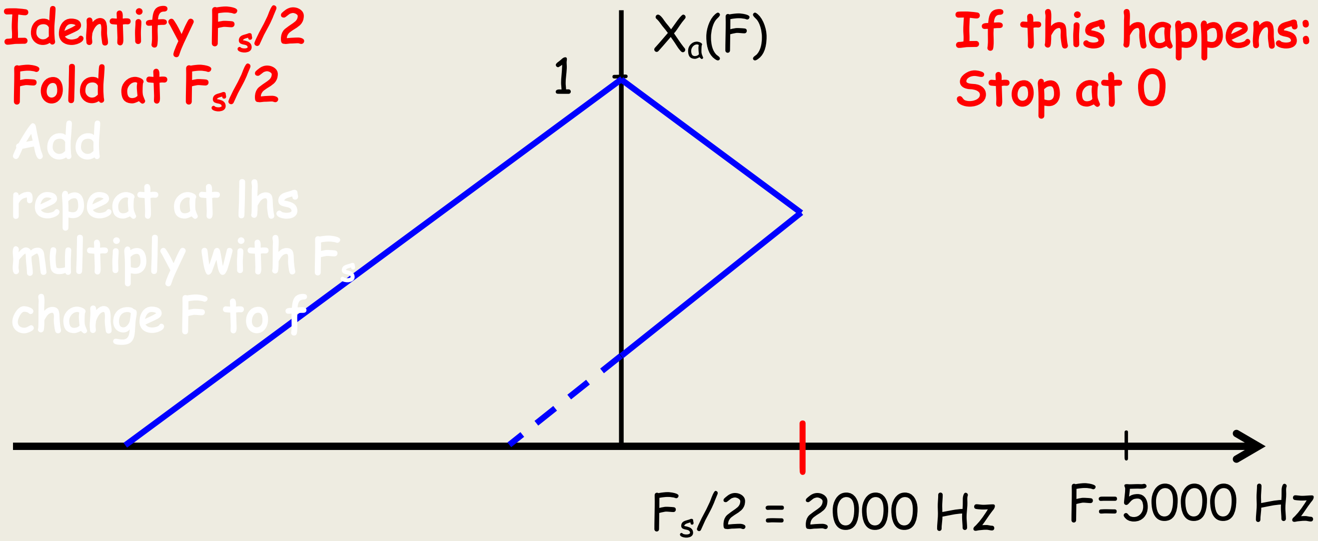
Example: Folding

Signal

F_s = 4000 Hz

Folding

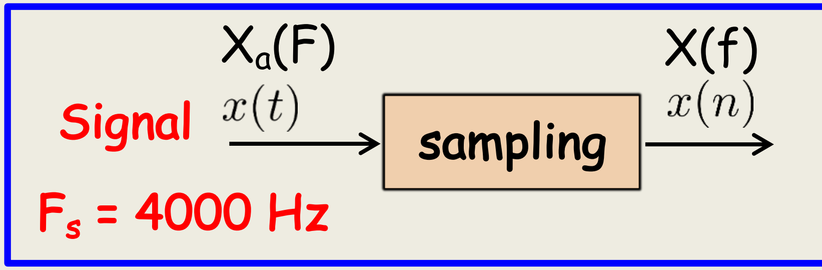
Step 1: Identify $F_s/2$
Step 2: Fold at $F_s/2$
Step 3: Add
Step 4: repeat at lhs
Step 5: multiply with F_s
Step 6: change F to f



If this happens:
Stop at 0

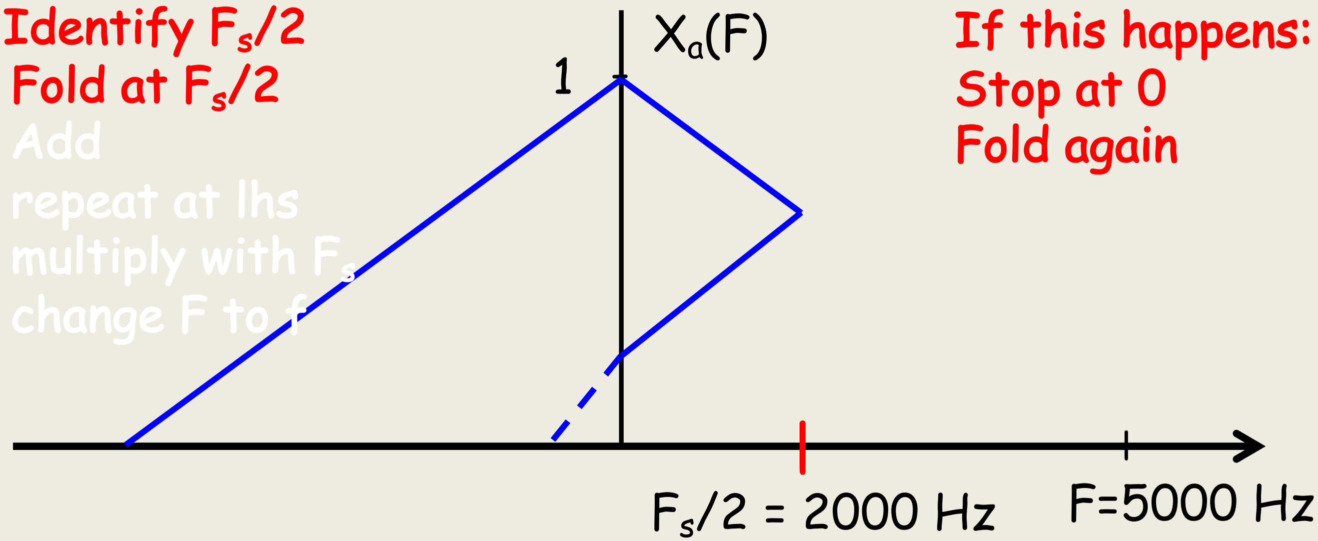
EITF75 Systems and Signals

Example: Folding



Folding

Step 1: Identify $F_s/2$
Step 2: Fold at $F_s/2$
Step 3: Add
Step 4: repeat at lhs
Step 5: multiply with F_s
Step 6: change F to f



If this happens:
Stop at 0
Fold again

EITF75 Systems and Signals

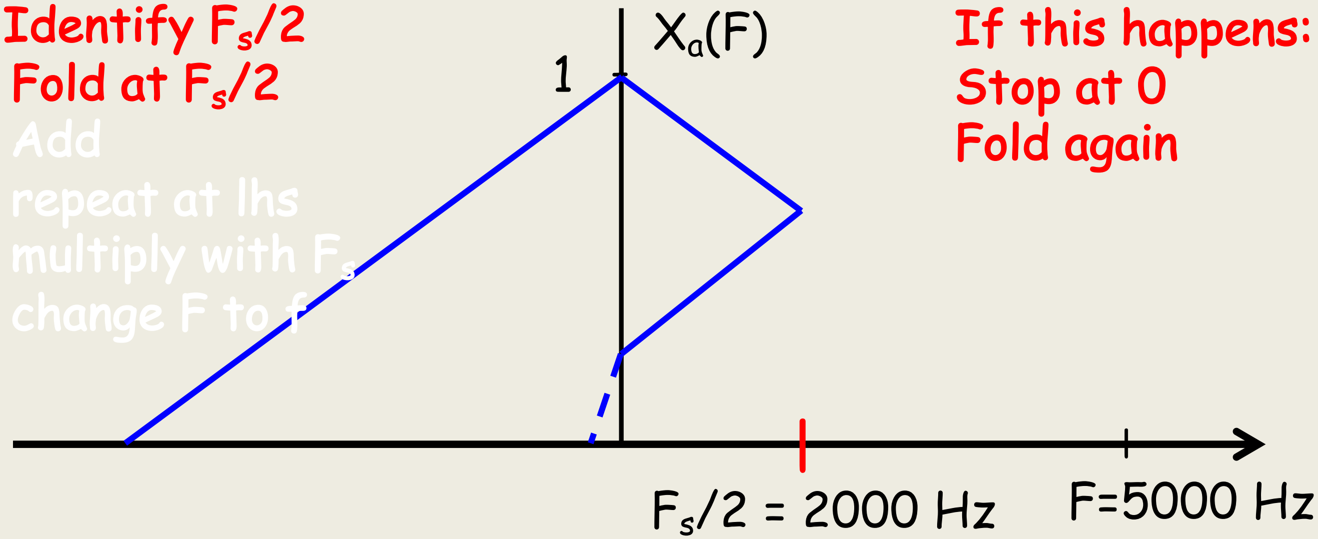
Example: Folding

Signal

F_s = 4000 Hz

Folding

Step 1: Identify $F_s/2$
Step 2: Fold at $F_s/2$
Step 3: Add
Step 4: repeat at lhs
Step 5: multiply with F_s
Step 6: change F to f



If this happens:
Stop at 0
Fold again

EITF75 Systems and Signals

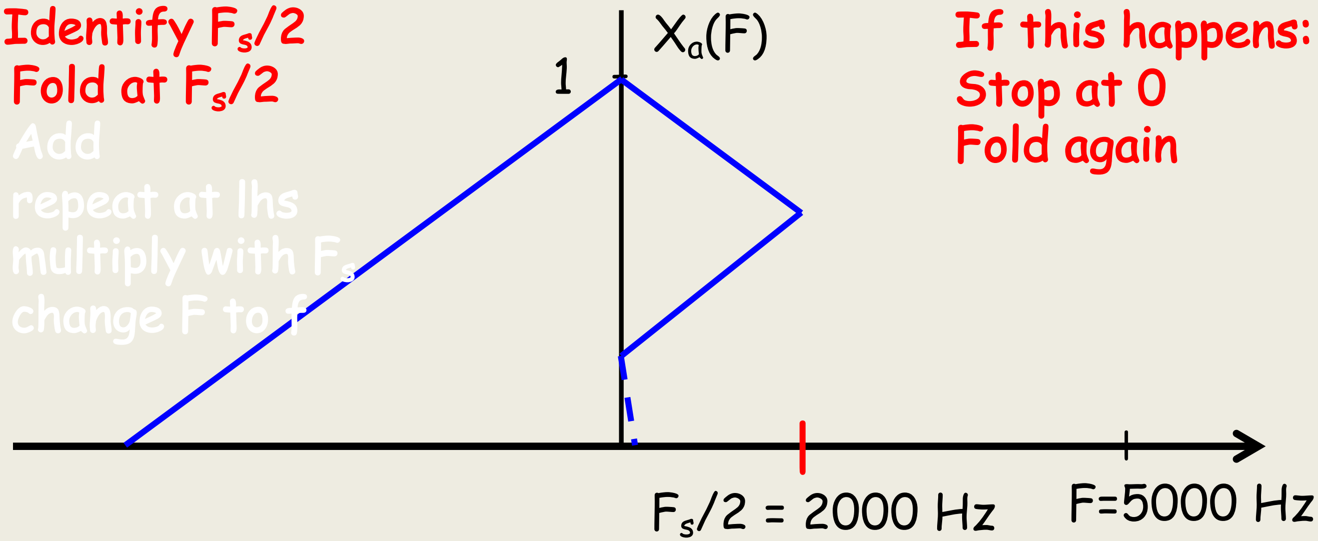
Example: Folding

Signal

F_s = 4000 Hz

Folding

Step 1: Identify $F_s/2$
Step 2: Fold at $F_s/2$
Step 3: Add
Step 4: repeat at lhs
Step 5: multiply with F_s
Step 6: change F to f



If this happens:
Stop at 0
Fold again

EITF75 Systems and Signals

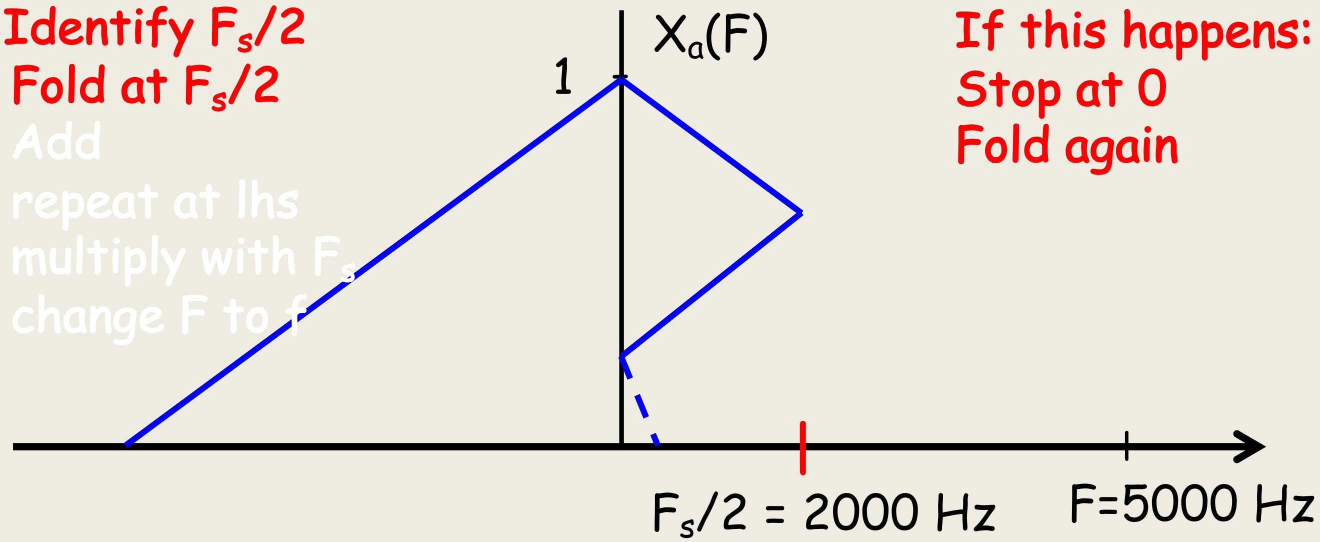
Example: Folding

Signal

F_s = 4000 Hz

Folding

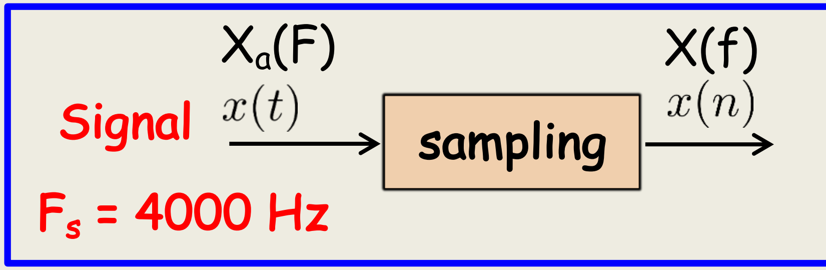
Step 1: Identify $F_s/2$
Step 2: Fold at $F_s/2$
Step 3: Add
Step 4: repeat at lhs
Step 5: multiply with F_s
Step 6: change F to f



If this happens:
Stop at 0
Fold again

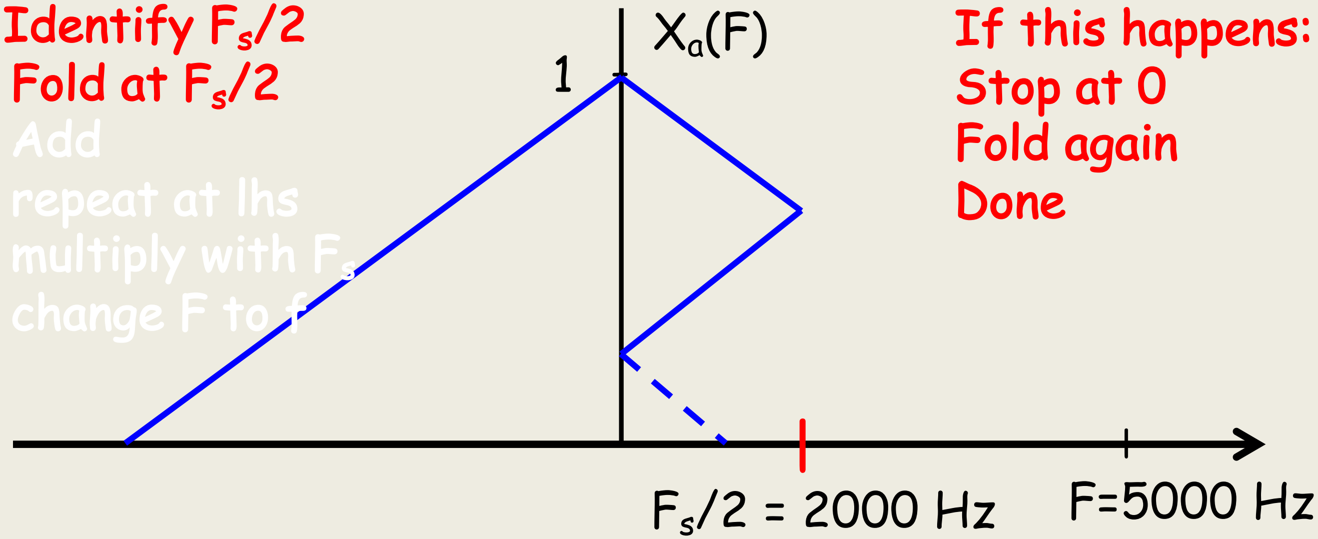
EITF75 Systems and Signals

Example: Folding



Folding

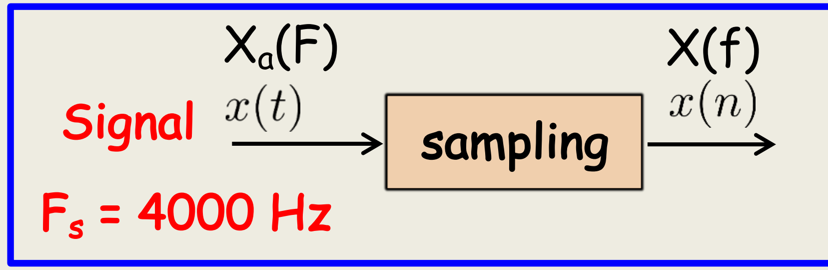
Step 1: Identify $F_s/2$
Step 2: Fold at $F_s/2$
Step 3: Add
Step 4: repeat at lhs
Step 5: multiply with F_s
Step 6: change F to f



If this happens:
Stop at 0
Fold again
Done

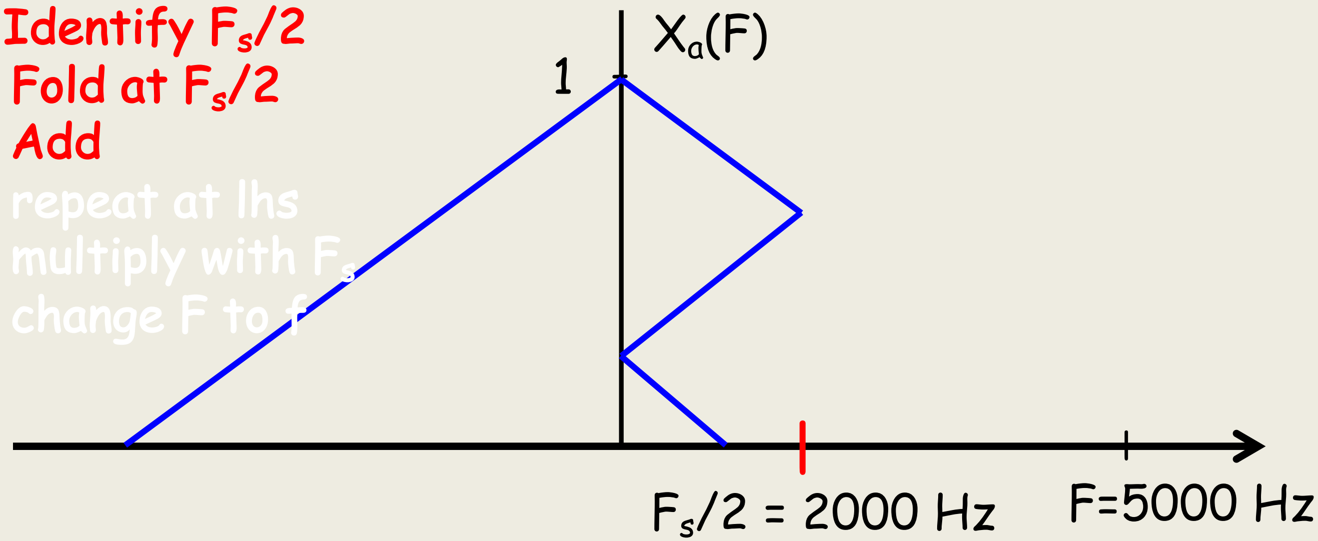
EITF75 Systems and Signals

Example: Folding

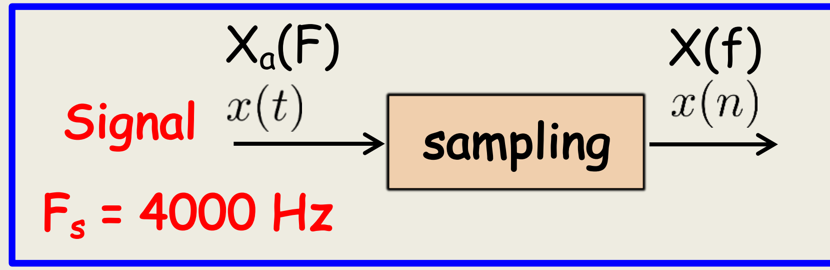


Folding

Step 1: Identify $F_s/2$
Step 2: Fold at $F_s/2$
Step 3: Add
Step 4: repeat at lhs
Step 5: multiply with F_s
Step 6: change F to f

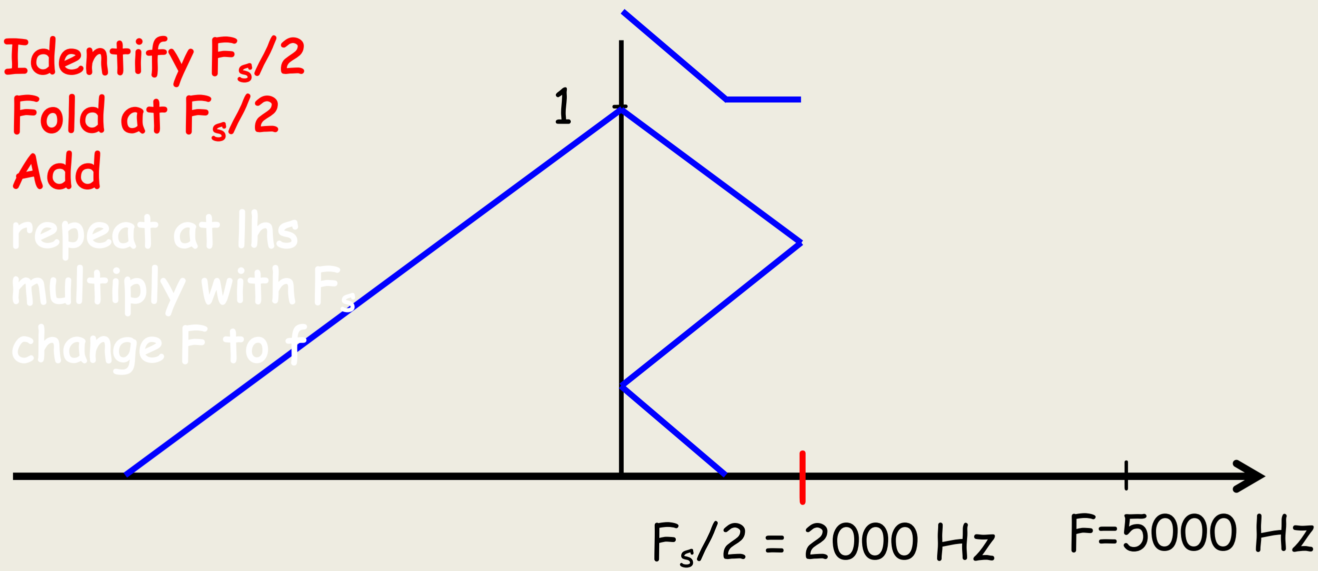


Example: Folding



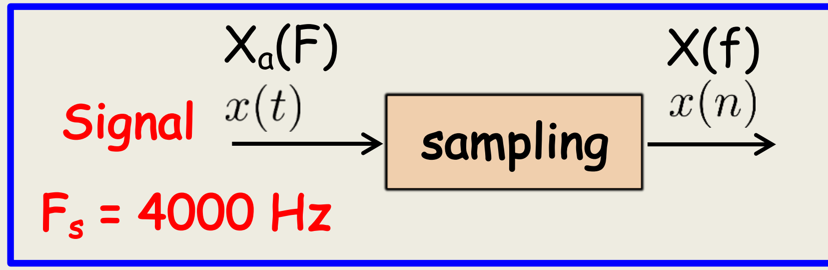
Folding

Step 1: Identify $F_s/2$
Step 2: Fold at $F_s/2$
Step 3: Add
Step 4: repeat at lhs
Step 5: multiply with F_s
Step 6: change F to f



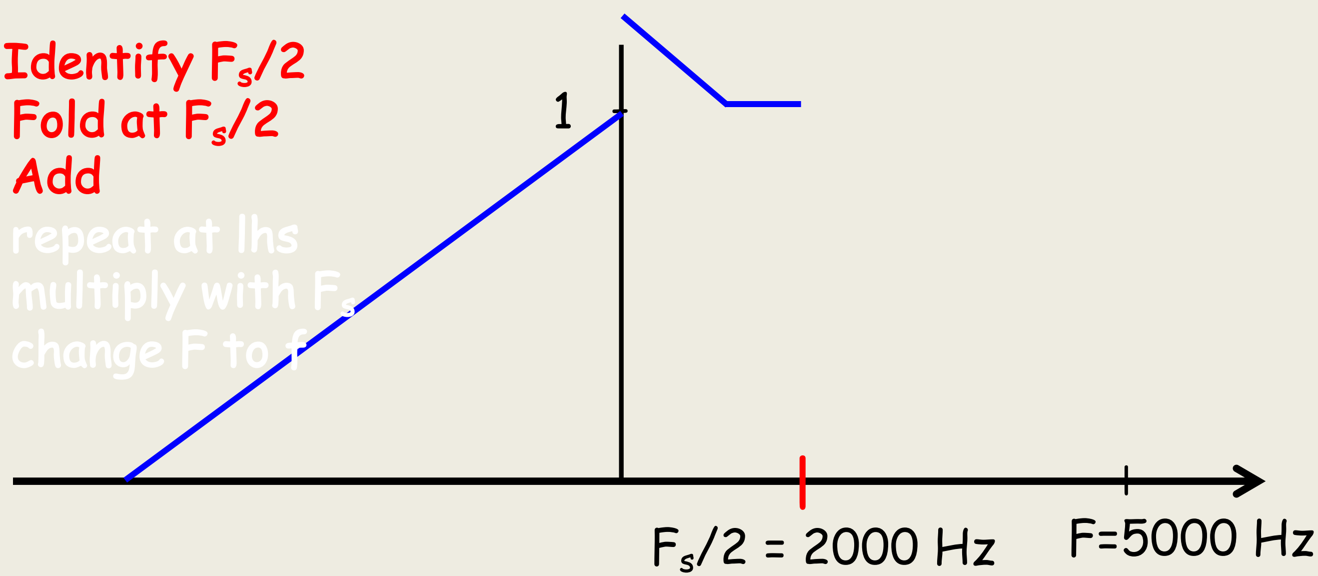
EITF75 Systems and Signals

Example: Folding



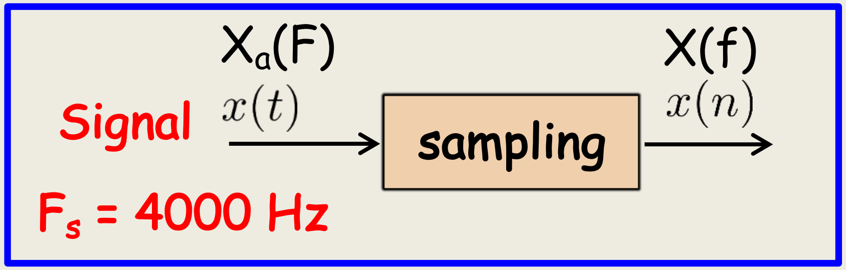
Folding

Step 1: Identify $F_s/2$
Step 2: Fold at $F_s/2$
Step 3: Add
Step 4: repeat at lhs
Step 5: multiply with F_s
Step 6: change F to f



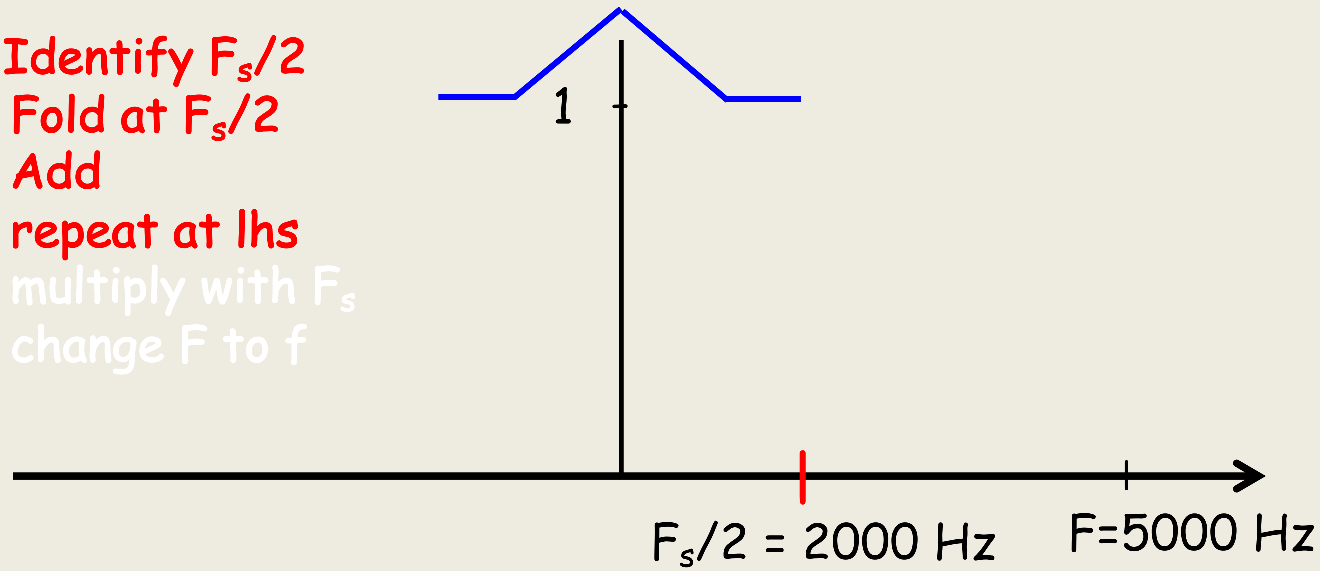
EITF75 Systems and Signals

Example: Folding



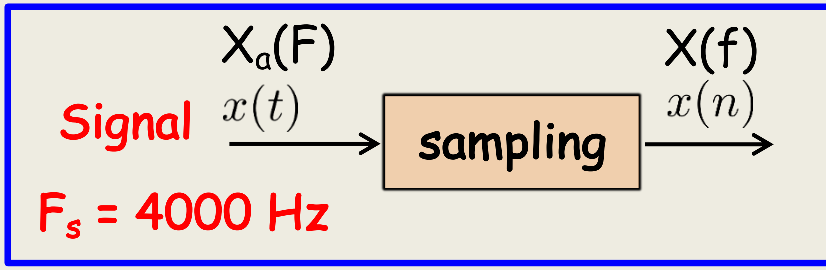
Folding

Step 1: Identify $F_s/2$
Step 2: Fold at $F_s/2$
Step 3: Add
Step 4: repeat at lhs
Step 5: multiply with F_s
Step 6: change F to f



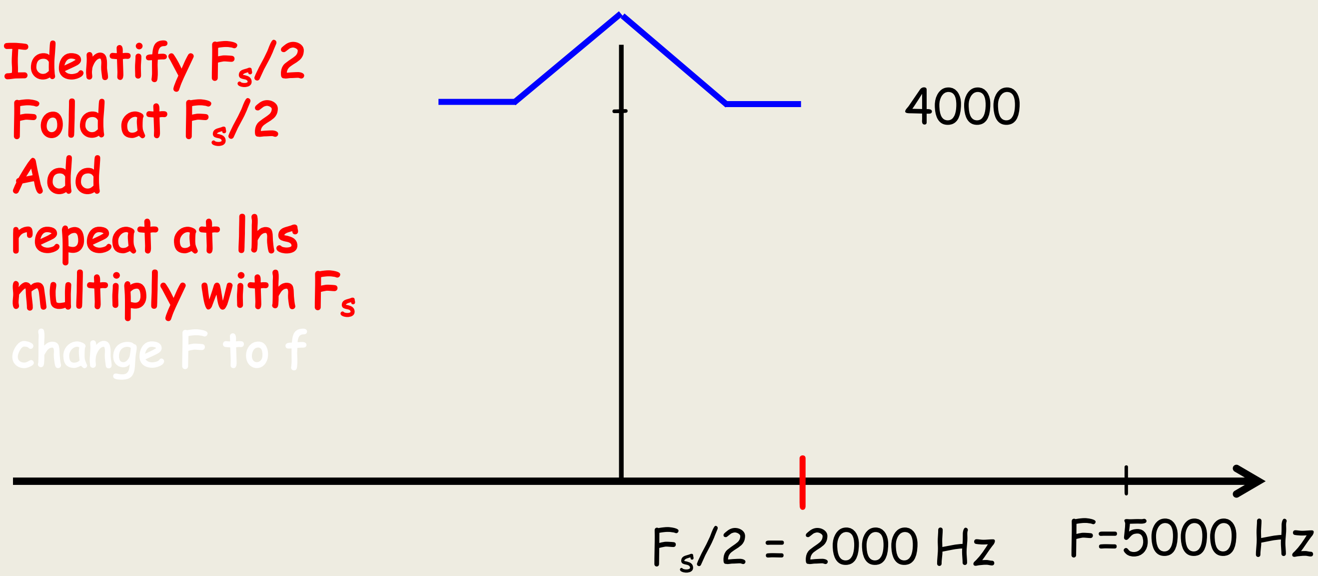
EITF75 Systems and Signals

Example: Folding



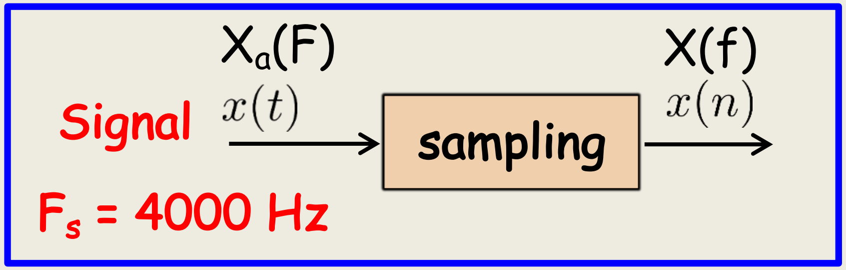
Folding

Step 1: Identify $F_s/2$
Step 2: Fold at $F_s/2$
Step 3: Add
Step 4: repeat at lhs
Step 5: multiply with F_s
Step 6: change F to f



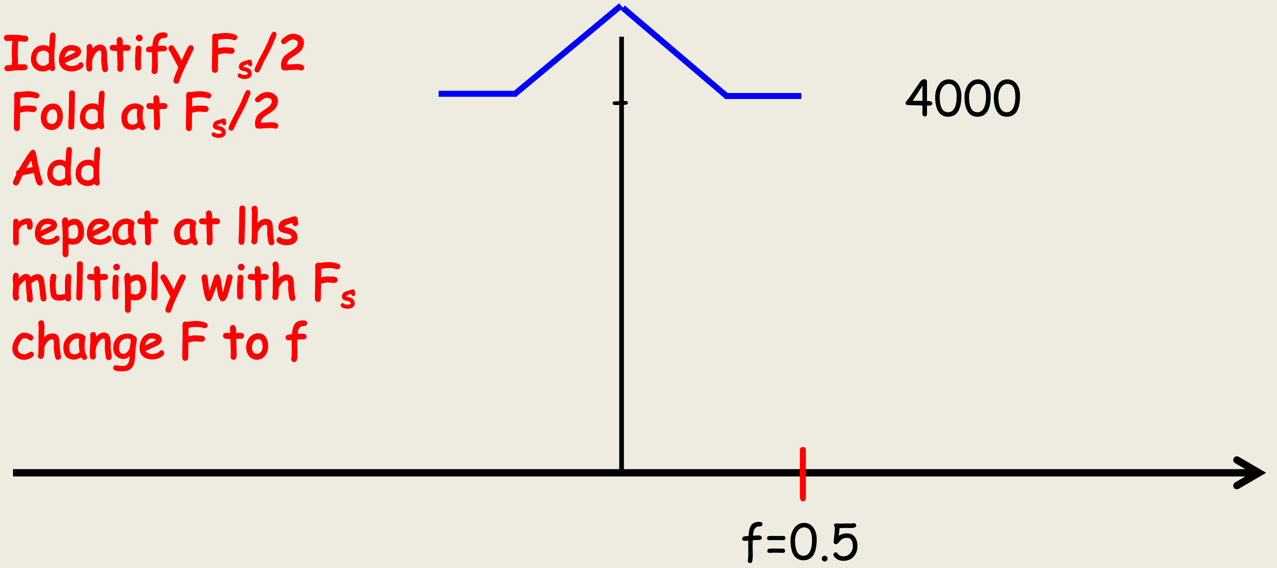
EITF75 Systems and Signals

Example: Folding



Folding

Step 1: Identify $F_s/2$
Step 2: Fold at $F_s/2$
Step 3: Add
Step 4: repeat at lhs
Step 5: multiply with F_s
Step 6: change F to f



EITF75 Systems and Signals

A/D and D/A

Signal $\quad \xrightarrow[x(t)]{X_a(F)} \boxed{\text{sampling}} \xrightarrow[x(n)]{X(f)}$

Key step is to understand what X(f) looks like in terms of $X_a(F)$

$$X(f) = F_s \sum_{k=-\infty}^{\infty} X_a((f-k)F_s)$$

$$X(f) = F_s X_a(f F_s)$$

k=0

If no aliasing
(e.g., sampling
Theorem fulfilled)

Sampling Theorem (Shannon 1948)

If $F_s > 2B$, where B is the highest frequency of the analog signal, then the analog signal can be recovered from its sampled version

If aliasing: In general not possible to revocer x(t) from x(n)

EITF75 Systems and Signals

A/D and D/A

$X_a(F)$ $\quad$ $X(f)$

Signal $x(t) \longrightarrow$ **sampling** (F_1 Hz) $\xrightarrow{x(n)}$ **Reconstruction** (F_2 Hz) $\longrightarrow \hat{x}(t)$

Note: sampling and reconstruction frequencies can differ. See lecture 9-10

$$X(f) = F_s \sum_{k=-\infty}^{\infty} X_a((f-k)F_s)$$

$$X(f) = F_s X_a(f F_s)$$

k=0

Reconstruction.

$$\hat{x}(t) = \sum_{n=-\infty}^{\infty} x(n)\operatorname{sinc}(F_s(t - n/F_s))$$

No aliasing

$$\hat{x}(t) = x(t)$$

Aliasing

$$\hat{x}(t) \neq x(t)$$

EITF75 Systems and Signals

LTI systems and z-transforms

EITF75 Systems and Signals

LTI systems

$x(n) \longrightarrow$ **system** $\longrightarrow y(n)$

A system is LTI if-and-only if:

- It is linear
- It is time-invariant

Linear system

$$x(n) = \alpha x_1(n) + \beta x_2(n)$$
$$\Longleftrightarrow$$
$$y(n) = \alpha y_1(n) + \beta y_2(n)$$

Time invariant system

$$x(n) \text{ replaced by } x(n-D)$$
$$\Longleftrightarrow$$
$$y(n) \text{ replaced by } y(n-D)$$

An LTI system is fully characterized by a difference equation ....or... an Impulse response $h(n)$

$$y(n) + \sum_{k=1}^{N} a_k y(n-k) = \sum_{k=0}^{M} b_k x(n-k)$$

Output if input is
$x(n) = \delta(n) = [1 \;\; 0 \;\; 0 \;\; \cdots]$
and system at rest

EITF75 Systems and Signals

LTI systems

$$x(n) \rightarrow \boxed{\text{system}} \rightarrow y(n)$$

A system is LTI if-and-only if:

- It is linear
- It is time-invariant

Assume that we turn on the circuit at n=0

System at rest if

$$y(-k) = 0, \; 1 \leq k \leq N$$

Not at rest if (has initial conditions)

$$\exists k, \; 1 \leq k \leq N, \; : \; y(-k) \neq 0$$

An LTI system is fully characterized by a difference equation ....or... an Impulse response $h(n)$

$$y(n) + \sum_{k=1}^{N} a_k y(n-k) = \sum_{k=0}^{M} b_k x(n-k)$$

Output if input is
$x(n) = \delta(n) = [1 \;\; 0 \;\; 0 \;\; \cdots]$
and system at rest

EITF75 Systems and Signals

$x(n) \rightarrow \boxed{\text{system}} \rightarrow y(n)$

What is output for a given input

Found by z-transform

The z-transform of $h(n)$ is defined as

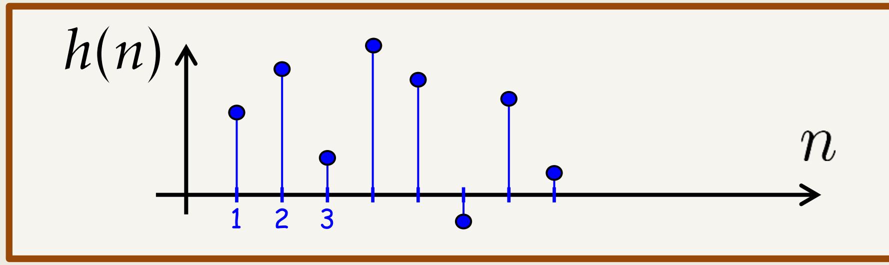
$$H(z) = \sum_{n=-\infty}^{\infty} h(n) z^{-n}$$

What is the z-transform?

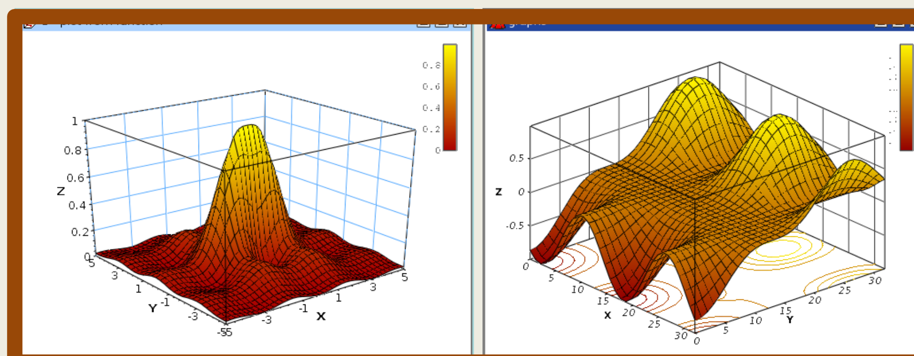
- A map from sequences to complex valued functions

What is H(z)?

- A complex function of a complex number



z-transform



If we want to plot H(z), we need 2 plots, one for the real part, one for the imaginary

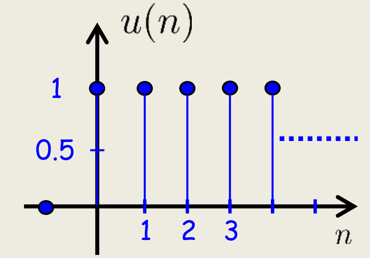
Z-transforms are not meant for "plotting and obtaining insights"

EITF75 Systems and Signals

An *important* example

$$h(n) = u(n) \quad \Leftrightarrow \quad H(z) = \frac{1}{1 - z^{-1}}$$

$$\text{ROC} \quad |z| > 1$$



$$H(z) = \sum_{n=-\infty}^{\infty} -u(-n-1) z^{-n}$$

Let's specify the ROC

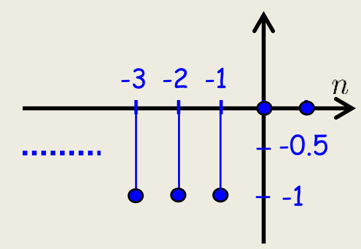
$$= -\sum_{n=-\infty}^{-1} z^{-n} = -\sum_{n=0}^{\infty} z^{n} + 1$$

$$= -\frac{1}{1-z} + 1 = \ldots = \frac{-z}{1-z}$$

$$= \frac{1}{1 - z^{-1}} \qquad \text{ROC} \quad |z| < 1$$

Anti-causal step

$$h(n) = -u(-n-1)$$



Different signals,
Same z-transform
Different ROC

EITF75, z-transform

Convention

If we are given an $X(z)$, and assume that the signal $x(n)$ is causal, then we can be sloppy with the ROC

There are many $x(n)$ for the same $X(z)$, and the ROC specifies the particular one. However, there is only one that is causal.

EITF75 Systems and Signals

LTI systems

$$x(n) \longrightarrow \boxed{\text{system}} \longrightarrow y(n)$$

What is output for a given input

Found by z-transform

$$y(n) + \sum_{k=1}^{N} a_k y(n-k) = \sum_{k=0}^{M} b_k x(n-k)$$

EITF75 Systems and Signals

Analyzing a general difference equation (at rest)

Expression for general difference equation

$$y(n)+\sum_{k=1}^{N} a_k y(n-k)=\sum_{k=0}^{M} b_k x(n-k)$$

Solution for general difference equation (at rest)

$$Y(z)+\sum_{k=1}^{N} a_k z^{-k} Y(z)=\sum_{k=0}^{M} b_k z^{-k} X(z)$$

$$Y(z)=\underbrace{\boxed{\frac{b_0+b_1 z^{-1}+\cdots+b_M z^{-M}}{1+a_1 z^{-1}+\cdots+a_N z^{-N}}}}_{H(z)}\cdot X(z)$$

Step 1:

Change y(n-k) to z^{-k} Y(z)

Step 2:

Change x(n-k) to z^{-k} X(z)

Step 3:

Express Y(z) as H(z)X(z)

EITF75 Systems and Signals

Analyzing a general difference equation (at rest)

Expression for general difference equation

$$y(n)+\sum_{k=1}^{N} a_k y(n-k)=\sum_{k=0}^{M} b_k x(n-k)$$

Step 4:

Find the roots of the denominator and nominator of H(z). Roots should be in terms of z, not z^{-1}

Solution for general difference equation (at rest)

$$Y(z)+\sum_{k=1}^{N} a_k z^{-k} Y(z)=\sum_{k=0}^{M} b_k z^{-k} X(z)$$

$$Y(z)=\boxed{\frac{b_0+b_1 z^{-1}+\cdots+b_M z^{-M}}{1+a_1 z^{-1}+\cdots+a_N z^{-N}}}\cdot X(z)$$

$$Y(z)=\frac{z^{-M}}{z^{-N}}\,\frac{b_0 z^M+b_1 z^{M-1}+\cdots+b_M}{z^N+a_1 z^{N-1}+\cdots+a_N}X(z)$$

zeros

$$=b_0\cdot\frac{z^{-M}}{z^{-N}}\cdot\frac{(z-z_1)\cdots(z-z_M)}{(z-p_1)\cdots(z-p_N)}\cdot X(z)$$

poles

$H(z)$

EITF75 Systems and Signals

Analyzing a general difference equation (at rest)

Expression for general difference equation

$$y(n)+\sum_{k=1}^{N} a_k y(n-k)=\sum_{k=0}^{M} b_k x(n-k)$$

Step 4:

Find the roots of the denominator and nominator of H(z). Roots should be in terms of z, not z^{-1}

Solution for general difference equation (at rest)

$$Y(z)+\sum_{k=1}^{N} a_k z^{-k} Y(z)=\sum_{k=0}^{M} b_k z^{-k} X(z)$$

$$Y(z)=\boxed{\frac{b_0+b_1 z^{-1}+\cdots+b_M z^{-M}}{1+a_1 z^{-1}+\cdots+a_N z^{-N}}}\cdot X(z)$$

$H(z)$

$$Y(z)=\frac{z^{-M}}{z^{-N}}\frac{b_0 z^{M}+b_1 z^{M-1}+\cdots+b_M}{z^{N}+a_1 z^{N-1}+\cdots+a_N}X(z)$$

zeros

$$=b_0\cdot\frac{z^{-M}}{z^{-N}}\cdot\frac{(z-z_1)\cdots(z-z_M)}{(z-p_1)\cdots(z-p_N)}\cdot X(z)$$

poles

If degree of numerator >= degree of denominator. Perform polynomial division

EITF75 Systems and Signals

Analyzing a general difference equation (at rest)

Expression for general difference equation

$$y(n)+\sum_{k=1}^{N} a_k y(n-k)=\sum_{k=0}^{M} b_k x(n-k)$$

Step 4:

Find the roots of the denominator and nominator of H(z). Roots should be in terms of z, not z^{-1}

Solution for general difference equation (at rest)

$$Y(z)+\sum_{k=1}^{N} a_k z^{-k} Y(z)=\sum_{k=0}^{M} b_k z^{-k} X(z)$$

$$Y(z)=\boxed{\frac{b_0+b_1 z^{-1}+\cdots+b_M z^{-M}}{1+a_1 z^{-1}+\cdots+a_N z^{-N}}}\cdot X(z)$$

$$Y(z)=\frac{z^{-M}}{z^{-N}}\frac{b_0 z^M+b_1 z^{M-1}+\cdots+b_M}{z^N+a_1 z^{N-1}+\cdots+a_N}X(z)$$

zeros

$$=b_0\cdot\frac{z^{-M}}{z^{-N}}\cdot\frac{(z-z_1)\cdots(z-z_M)}{(z-p_1)\cdots(z-p_N)}\cdot X(z)$$

poles

$H(z)$

Will turn up in the time-domain as a delay
(can be negative delay)

EITF75 Systems and Signals

Analyzing a general difference equation (at rest)

Expression for general difference equation

$$\boxed{y(n)+\sum_{k=1}^{N}a_k y(n-k)=\sum_{k=0}^{M}b_k x(n-k)}$$

Assume $\boxed{X(z)=\frac{N(z)}{Q(z)}}$

$$Y(z)=\frac{\text{Some polynomial in } z}{(z-p_1)\cdots(z-p_N)\cdot(z-q_1)\cdots(z-q_L)}$$

Assuming all poles are real and distinct

Assuming deg(num) < deg(denom)

$$Y(z)=\sum_{k=1}^{N}\frac{A_k}{1-z^{-1}p_k}+\sum_{k=1}^{L}\frac{Q_k}{1-z^{-1}q_k}$$

$$Y(z)=\frac{z^{-M}}{z^{-N}}\frac{b_0z^M+b_1z^{M-1}+\cdots+b_M}{z^N+a_1z^{N-1}+\cdots+a_N}X(z)$$

$$=b_0\cdot\frac{z^{-M}}{z^{-N}}\cdot\frac{(z-z_1)\cdots(z-z_M)}{(z-p_1)\cdots(z-p_N)}\cdot X(z)$$

Perform partial fraction expansion

EITF75 Systems and Signals

Analyzing a general difference equation (at rest)

Expression for general difference equation

$$y(n) + \sum_{k=1}^{N} a_k y(n-k) = \sum_{k=0}^{M} b_k x(n-k)$$

$$Y(z) = \frac{z^{-M}}{z^{-N}} \frac{b_0 z^M + b_1 z^{M-1} + \cdots + b_M}{z^N + a_1 z^{N-1} + \cdots + a_N} X(z)$$

$$= b_0 \cdot \frac{z^{-M}}{z^{-N}} \cdot \frac{(z-z_1)\cdots(z-z_M)}{(z-p_1)\cdots(z-p_N)} \cdot X(z)$$

Assume $\boxed{X(z) = \frac{N(z)}{Q(z)}}$

$$Y(z) = \frac{\textit{Some polynomial in z}}{(z-p_1)\cdots(z-p_N)\cdot(z-q_1)\cdots(z-q_L)}$$

Assuming all poles are real and distinct

Assuming deg(num) < deg(denom)

$$Y(z) = \sum_{k=1}^{N} \frac{A_k}{1 - z^{-1} p_k} + \sum_{k=1}^{L} \frac{Q_k}{1 - z^{-1} q_k}$$

Invert

$$y(n) = \sum_{k=1}^{N} A_k (p_k)^n u(n) + \sum_{k=1}^{L} Q_k (q_k)^n u(n)$$

EITF75 Systems and Signals

Analyzing a general difference equation (at rest)

Expression for general difference equation

$$Y(z) = \sum_{k=1}^{N} \frac{A_k}{1 - z^{-1} p_k} + \sum_{k=1}^{L} \frac{Q_k}{1 - z^{-1} q_k}$$

This...

...generates that

$$y(n) = \sum_{k=1}^{N} A_k (p_k)^n u(n) + \sum_{k=1}^{L} Q_k (q_k)^n u(n)$$

EITF75 Systems and Signals

Analyzing a general difference equation (at rest)

Important: To get stable output, all poles must be inside the unit circle

Assume $X(z) = \frac{N(z)}{Q(z)}$

$$Y(z) = \frac{1}{(z-p_1)\cdots(z-p_N)} \cdot \frac{\text{Some polynomial in } z}{(z-q_1)\cdots(z-q_L)}$$

$$Y(z) = \sum_{k=1}^{N} \frac{A_k}{1-z^{-1}p_k} + \sum_{k=1}^{L} \frac{Q_k}{1-z^{-1}q_k}$$

Important: poles in H(z) and in X(z) determines the output structure:

"You can never get a term in y(n) that doesn't exist in either X(z) or H(z)"

$$Y(z) = \frac{z^{-M}}{z^{-N}} \frac{b_0 z^M + b_1 z^{M-1} + \cdots + b_M}{z^N + a_1 z^{N-1} + \cdots + a_N} X(z)$$

$$= b_0 \cdot \frac{z^{-M}}{z^{-N}} \cdot \frac{(z-z_1)\cdots(z-z_M)}{(z-p_1)\cdots(z-p_N)} \cdot X(z)$$

$$y(n) = \sum_{k=1}^{N} A_k (p_k)^n u(n) + \sum_{k=1}^{L} Q_k (q_k)^n u(n)$$

EITF75 Systems and Signals

A complex conjugated pair of poles

$$h(n) = r^n \cdot \sin(\omega n)u(n)$$

$$h(n) = r^n \cdot \cos(\omega n)u(n)$$

$$H(z) = \frac{r\sin(\omega)z^{-1}}{1 - 2r\cos(\omega)z^{-1} + r^2 z^{-2}}$$

$$H(z) = \frac{1 - r\cos(\omega)z^{-1}}{1 - 2r\cos(\omega)z^{-1} + r^2 z^{-2}}$$

Polar coordinates: r is "length" and w is angle of the pole.
To get stable output: r<1 (poles inside the unit circle)

Example

Quite messy to invert a mixture of the two above: Make sure you know how to do that.

Invert $$H(z) = z^{-1} \cdot \frac{1 - z^{-1}}{1 - 1.27z^{-1} + 0.81z^{-2}}$$

EITF75 Systems and Signals

Systems not at rest

Use the one-sided z-transform

$$X^+(z) = \sum_{n=0}^{\infty} x(n) z^{-n}$$

EITF75 Systems and Signals

Systems not at rest

Use the one-sided z-transform

$$X^+(z) = \sum_{n=0}^{\infty} x(n) z^{-n}$$

End result: The solution at rest + contribution from initial conditions

$$Y(z) = H(z)X(z) + \frac{N_0(z)}{A(z)} = \frac{B(z)}{A(z)} X(z) + \frac{N_0(z)}{A(z)}$$

$$N_0(z) = -\sum_{k=1}^{N} a_k z^{-k} \sum_{n=1}^{k} y(-n) z^n$$

N: highest power of z^{-1} in A(z)

EITF75 Systems and Signals

Fourier analysis. 4 cases

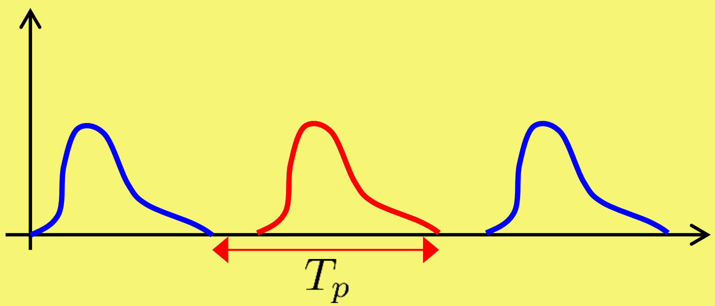
Periodic/aperiodic signal
Continuous/discrete signal

EITF75, Fourier transforms

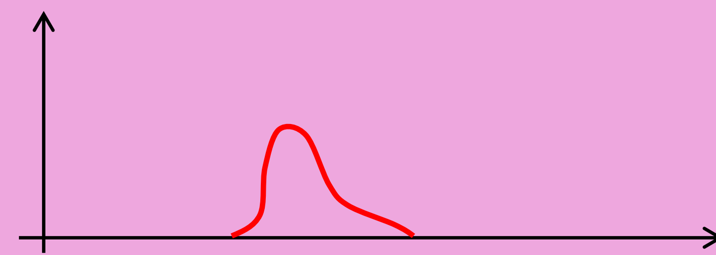
4 different type of signals

Time signals shown, not Fourier transforms

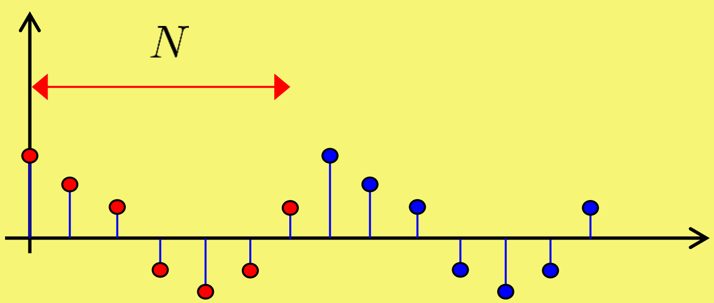
Continuous and periodic



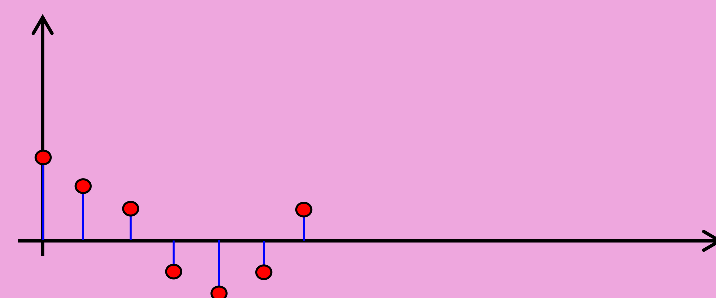
Continuous and aperiodic



Discrete and periodic



Discrete and aperiodic



Discrete spectra

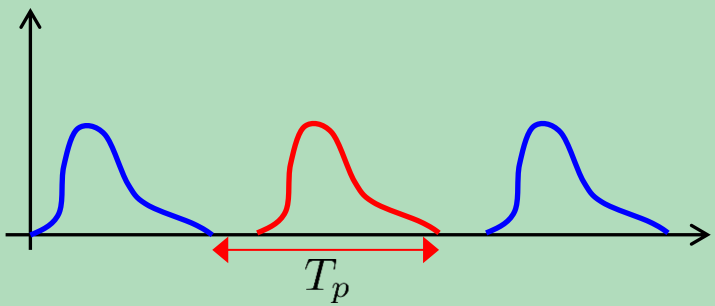
Continuous spectra

EITF75, Fourier transforms

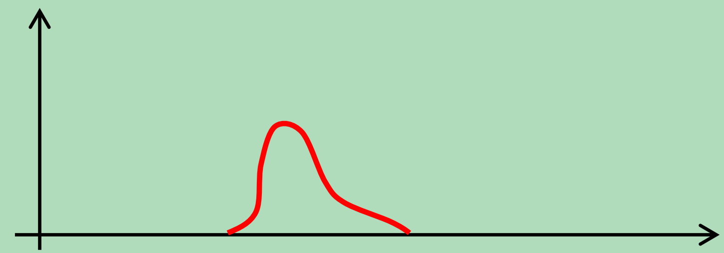
4 different type of signals

Aperiodic spectra

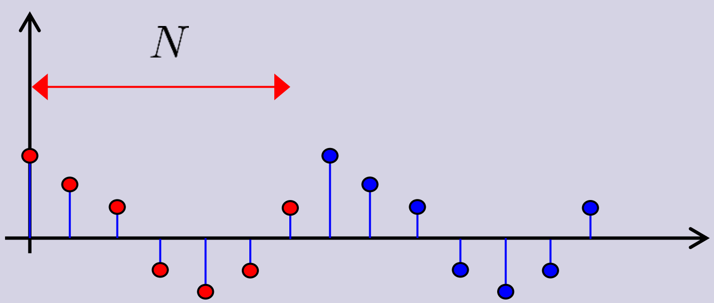
Continuous and periodic



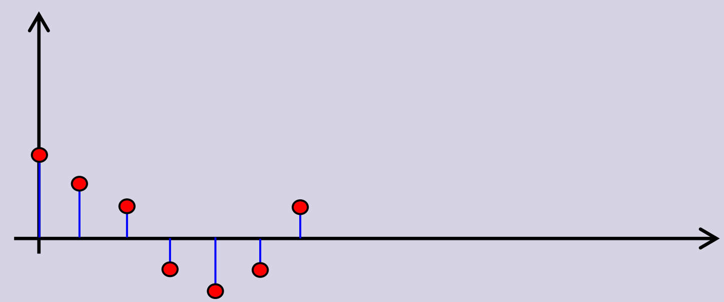
Continuous and aperiodic



Discrete and periodic

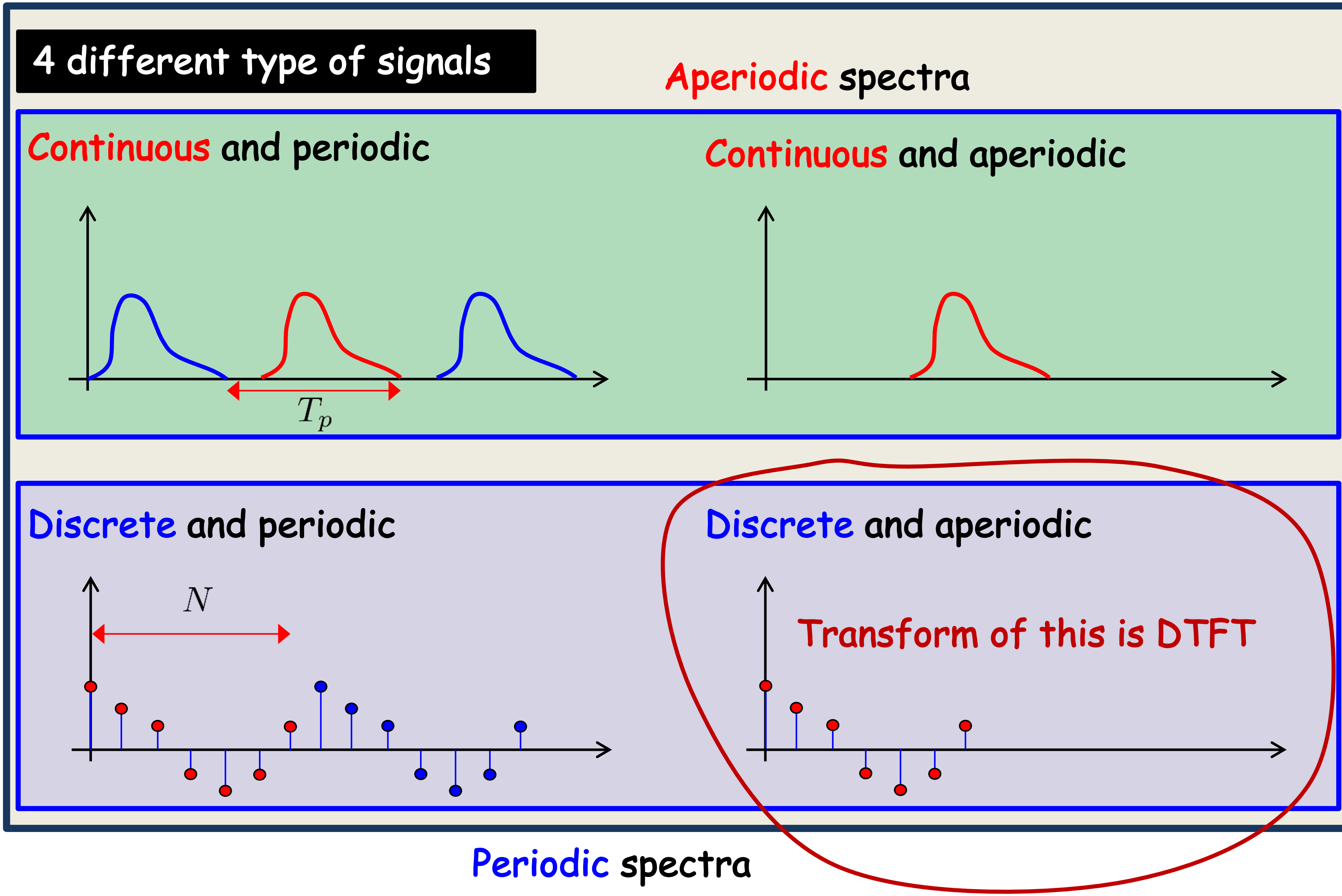


Discrete and aperiodic



Periodic spectra

EITF75, Fourier transforms



EITF75 Systems and Signals

Z-transform

$$X(z) = \sum_{n=-\infty}^{\infty} x(n) z^{-n}$$

DTFT

(discrete time Fourier transform)

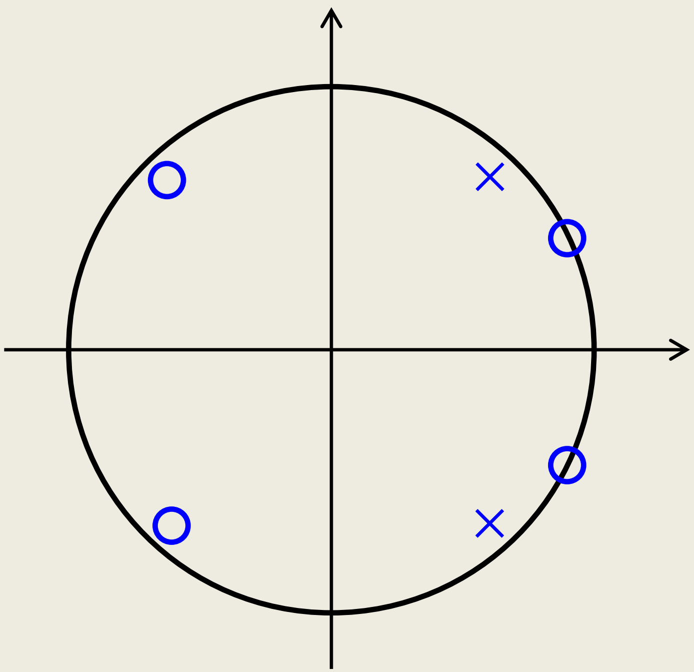
$$X(f) = \sum_{n=-\infty}^{\infty} x(n) \exp(-i2\pi n f)$$

$$= X(z|z = \exp(i2\pi f))$$

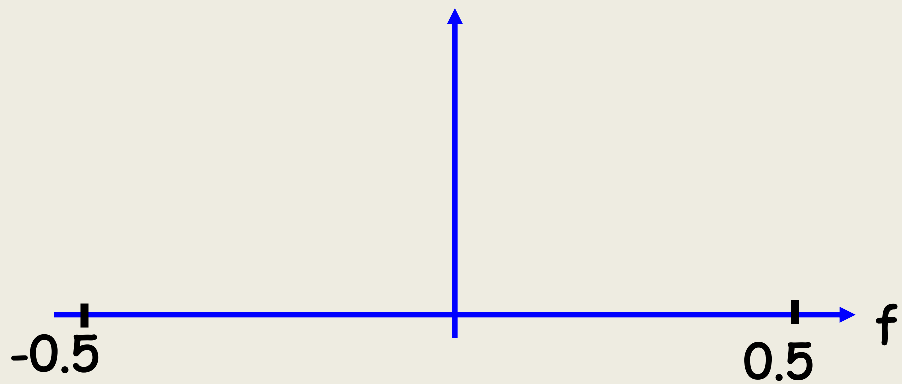
Important: DTFT is z-transform evaluated at unit circle

EITF75, DTFT

Pole-zero plot



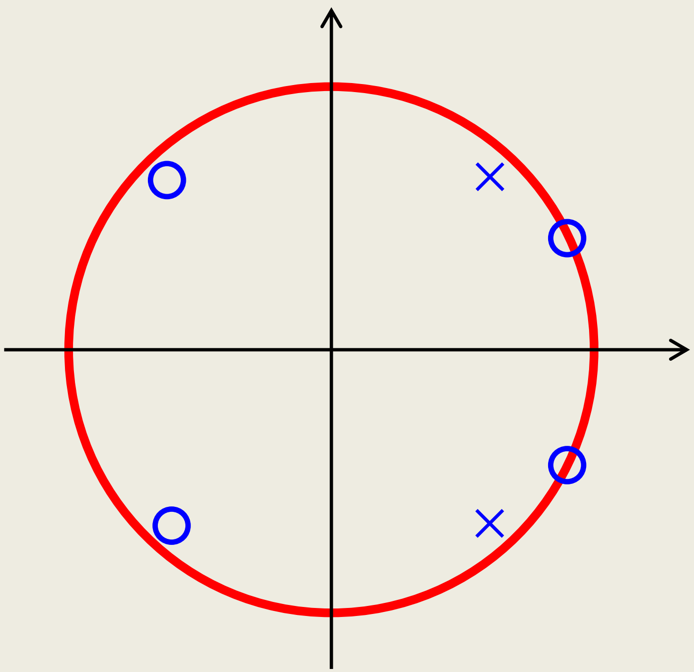
DTFT



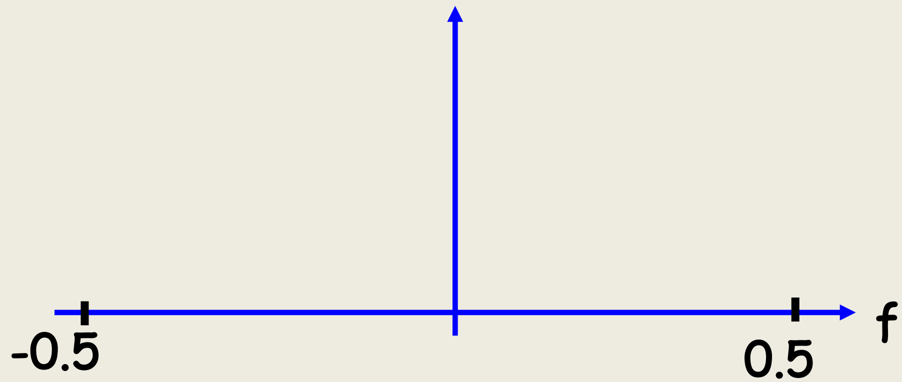
Book makes a big deal out of this. But quite easy....

EITF75, DTFT

Pole-zero plot



DTFT is H(z) at unit circle

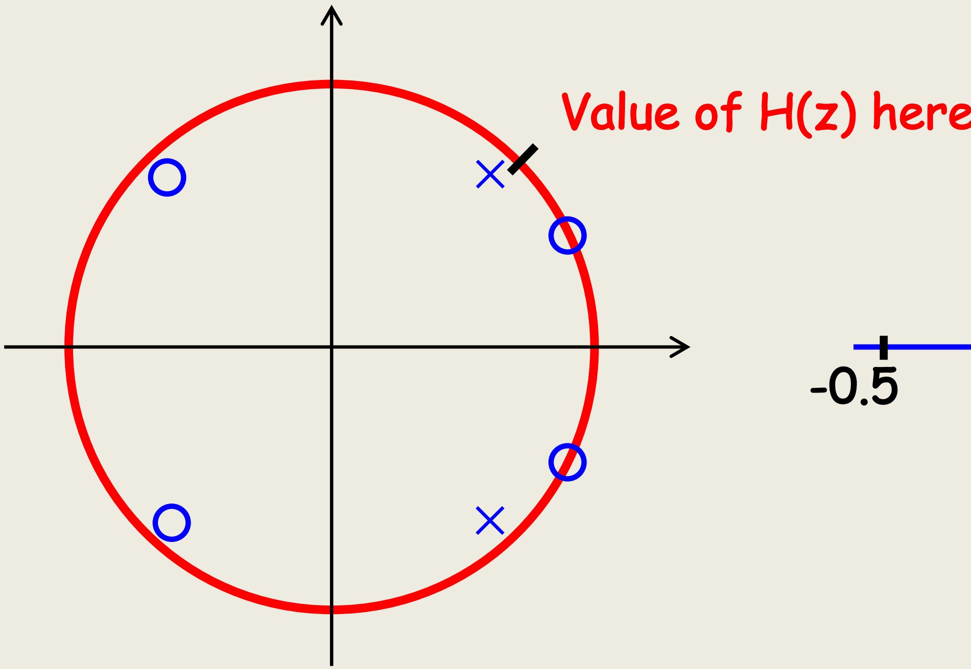


Recall

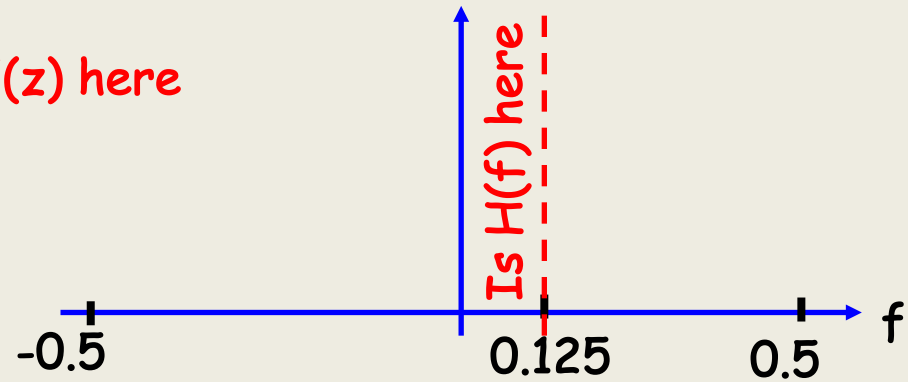
$$H(z) = \frac{(z-z_1)\cdots(z-z_M)}{(z-p_1)\cdots(z-p_N)}$$

EITF75, DTFT

Pole-zero plot



DTFT is H(z) at unit circle

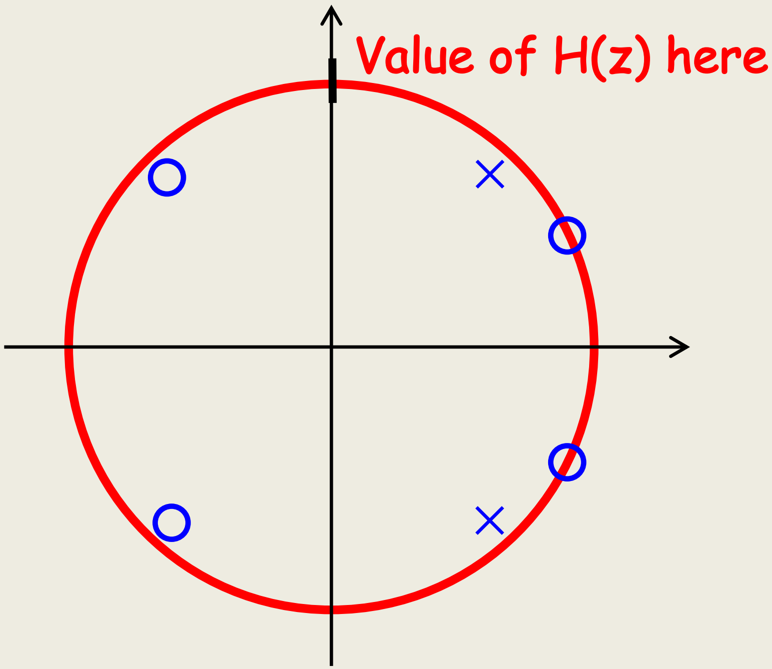


Recall

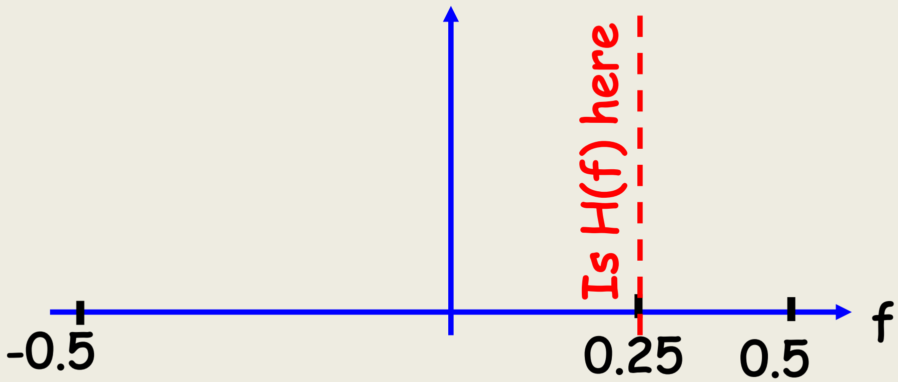
$$H(z) = \frac{(z - z_1) \cdots (z - z_M)}{(z - p_1) \cdots (z - p_N)}$$

EITF75, DTFT

Pole-zero plot



DTFT is H(z) at unit circle

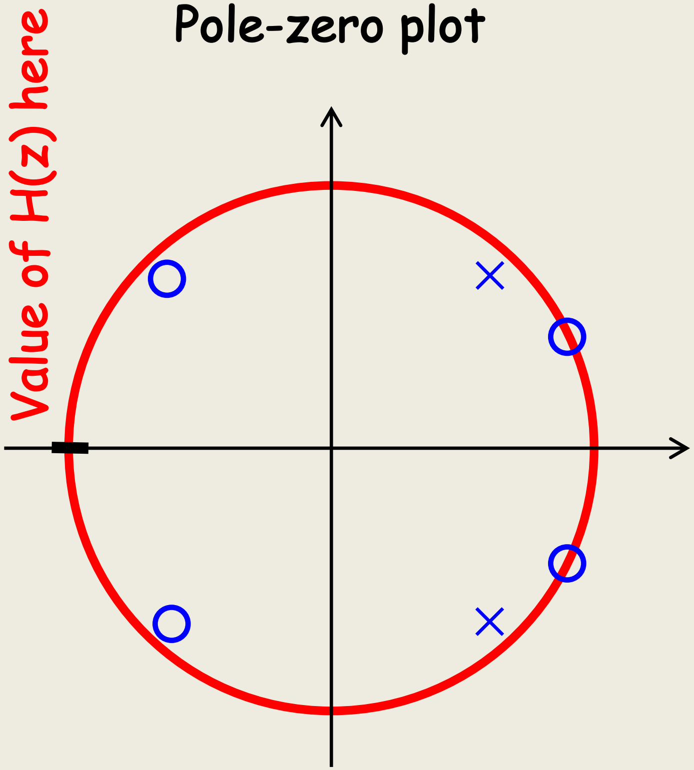


Recall

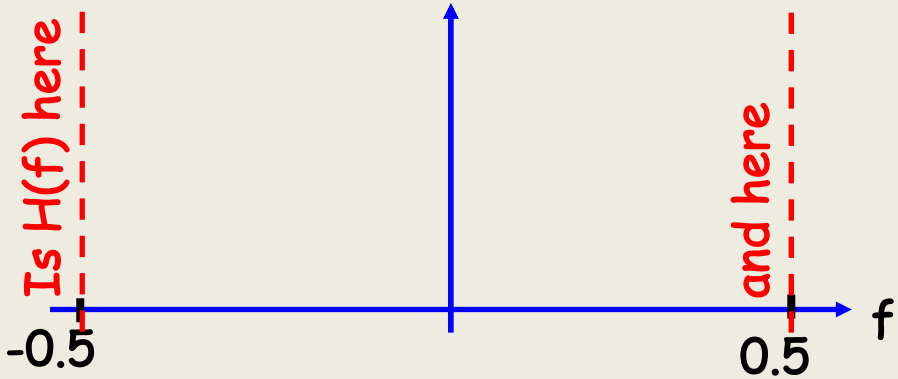
$$H(z) = \frac{(z - z_1)\cdots(z - z_M)}{(z - p_1)\cdots(z - p_N)}$$

EITF75, DTFT

Pole-zero plot



DTFT is H(z) at unit circle

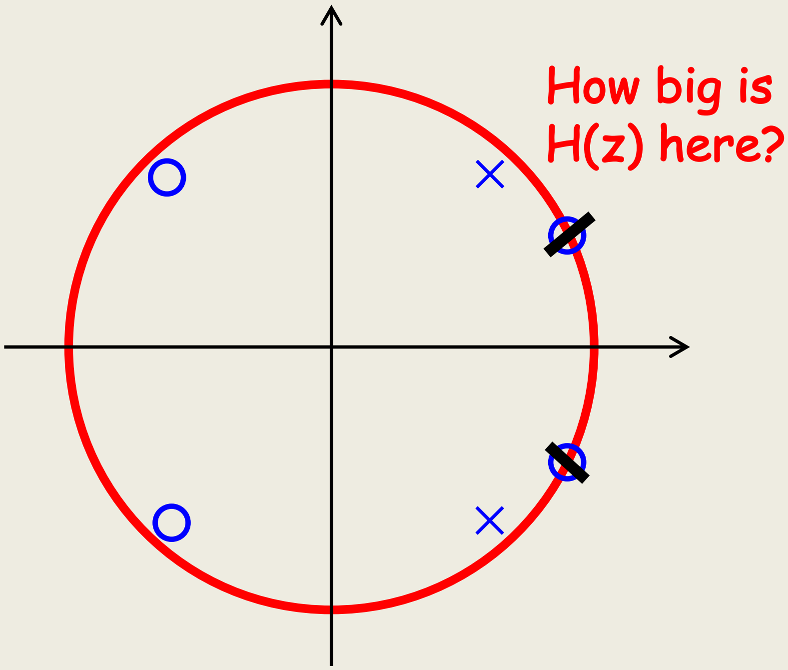


Recall

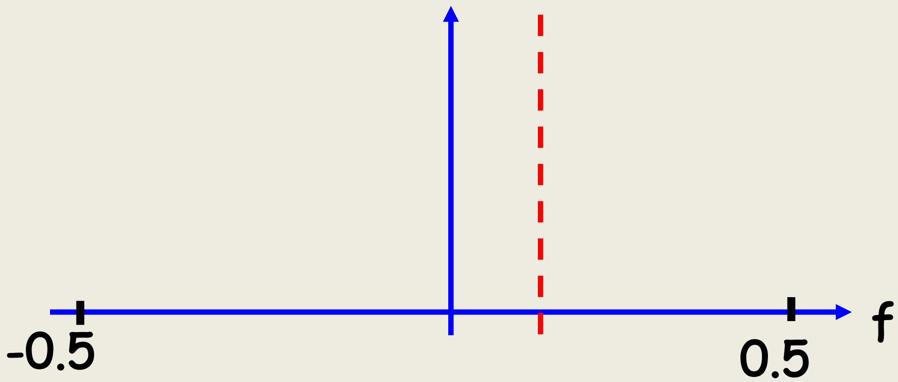
$$H(z) = \frac{(z - z_1)\cdots(z - z_M)}{(z - p_1)\cdots(z - p_N)}$$

EITF75, DTFT

Pole-zero plot



DTFT is H(z) at unit circle

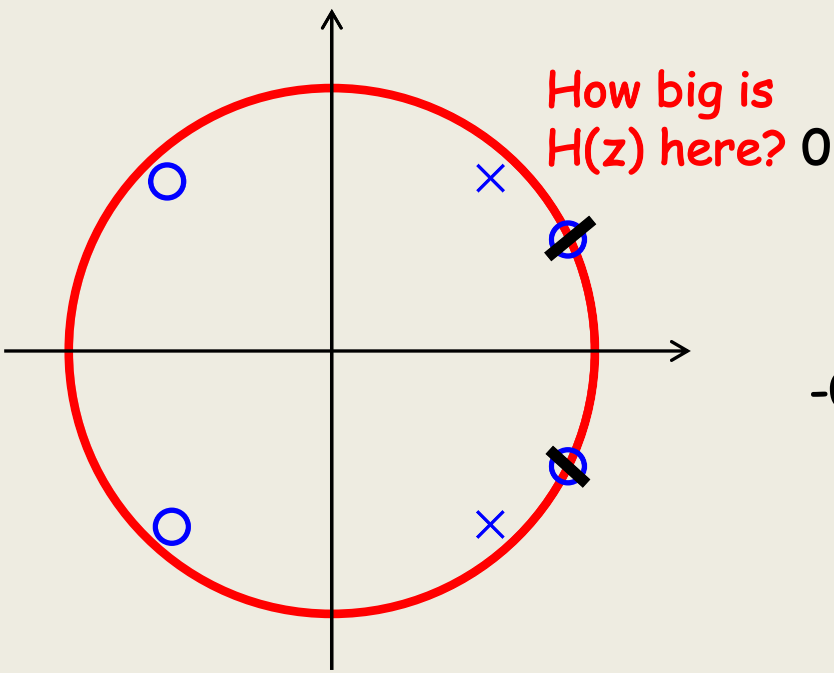


Recall

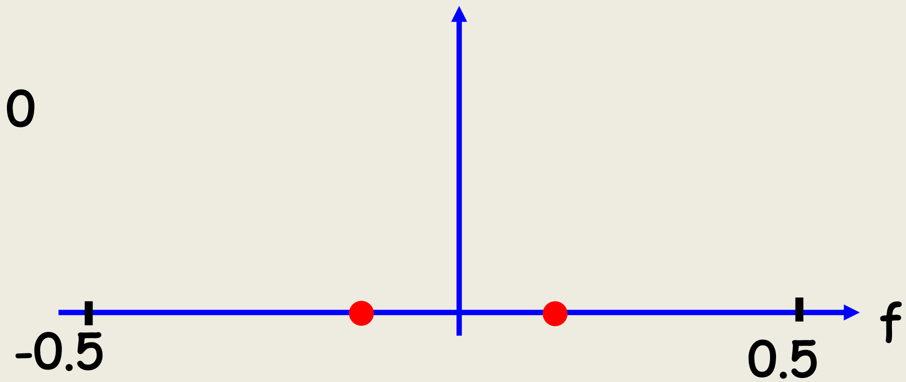
$$H(z) = \frac{(z-z_1)\cdots(z-z_M)}{(z-p_1)\cdots(z-p_N)}$$

EITF75, DTFT

Pole-zero plot



DTFT is H(z) at unit circle

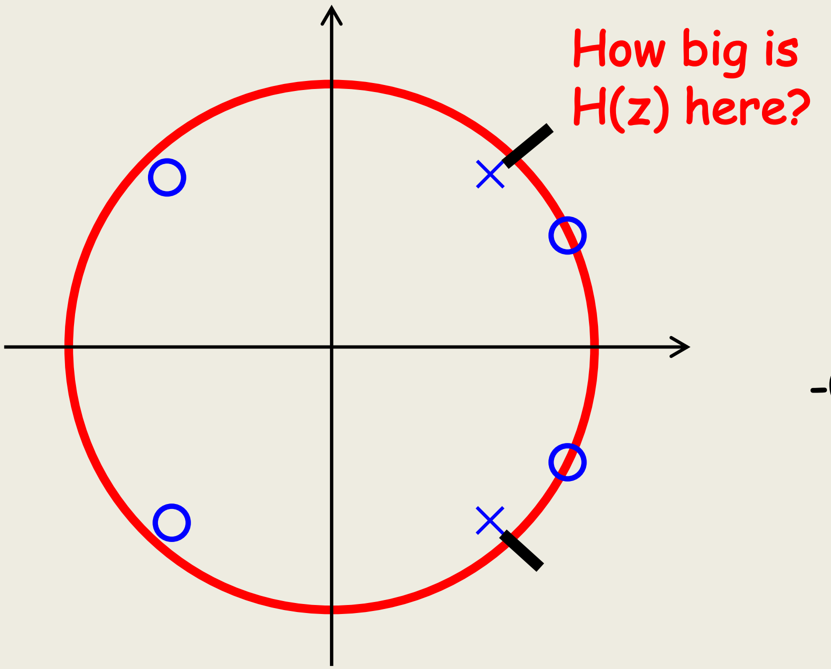


We are at a zero

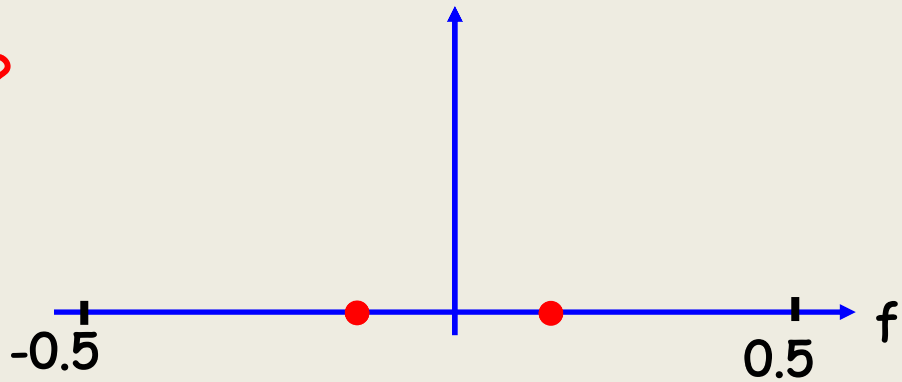
$$H(z) = \frac{\boxed{(z-z_1)}\cdots(z-z_M)}{(z-p_1)\cdots(z-p_N)}$$

EITF75, DTFT

Pole-zero plot



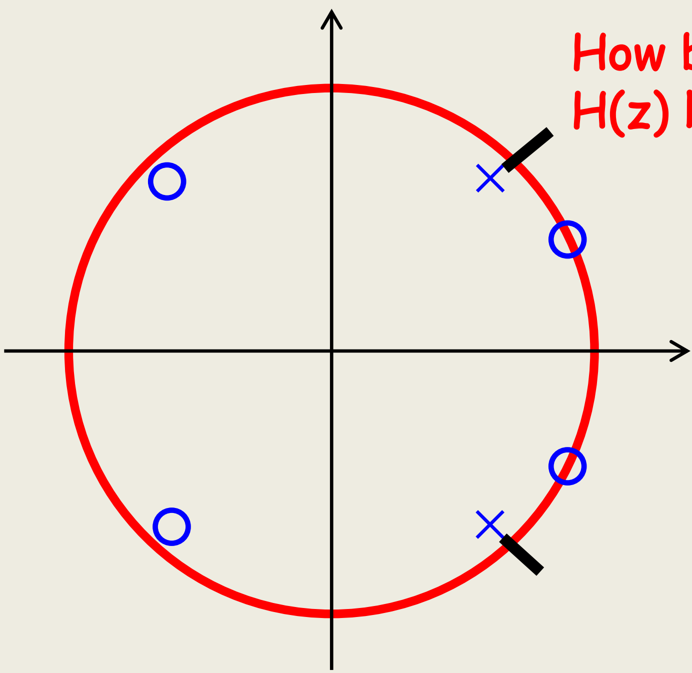
DTFT is H(z) at unit circle



$$H(z) = \frac{(z - z_1) \cdots (z - z_M)}{(z - p_1) \cdots (z - p_N)}$$

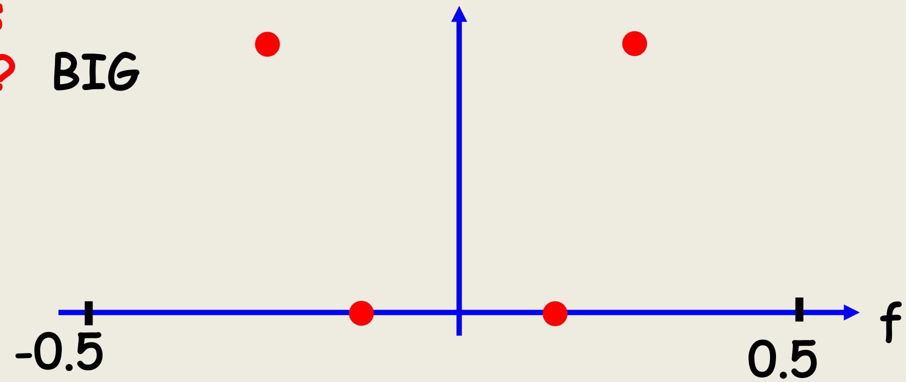
EITF75, DTFT

Pole-zero plot



How big is H(z) here? BIG

DTFT is H(z) at unit circle



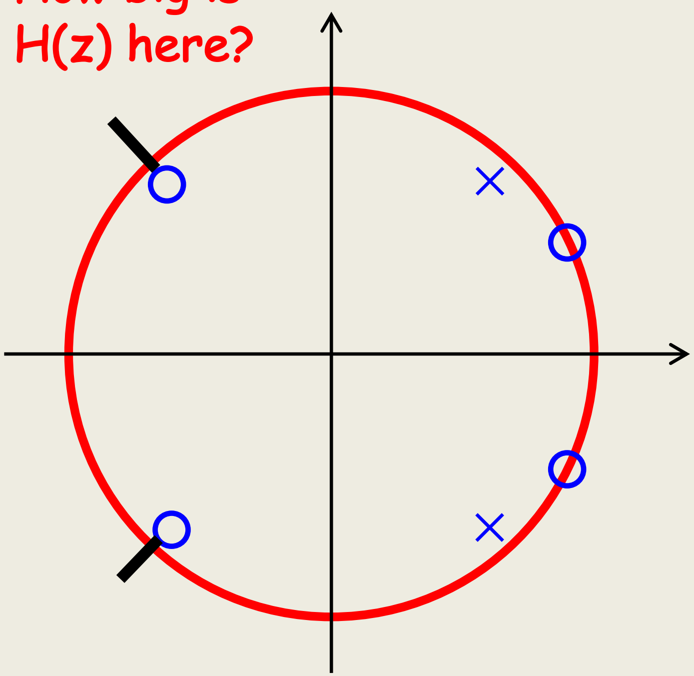
We are close to a pole

$$\boxed{H(z) = \frac{(z-z_1)\cdots(z-z_M)}{\boxed{(z-p_1)}\cdots(z-p_N)}}$$

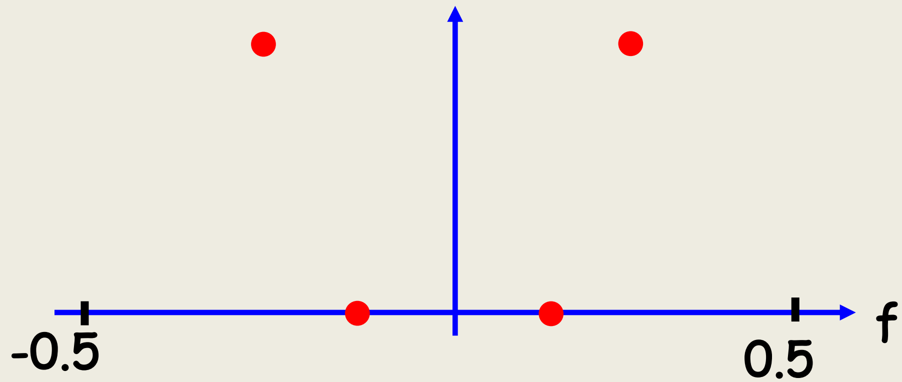
EITF75, DTFT

Pole-zero plot

How big is H(z) here?

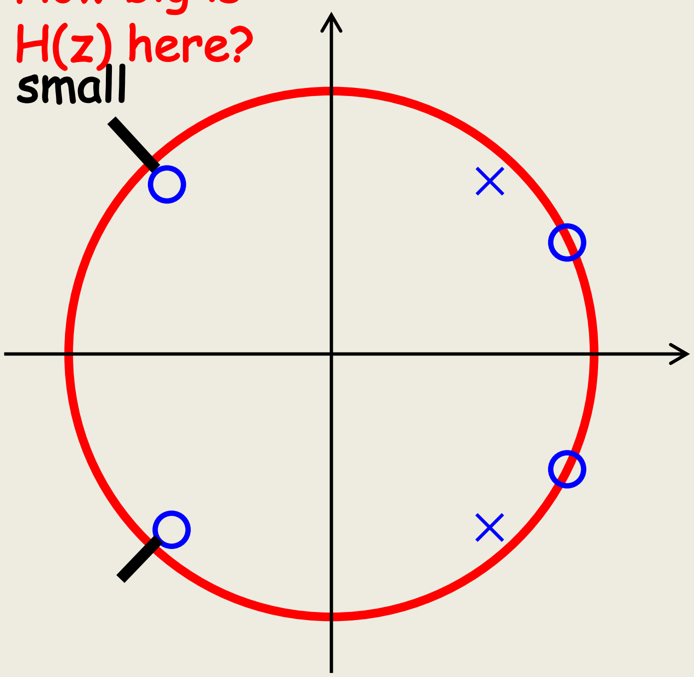


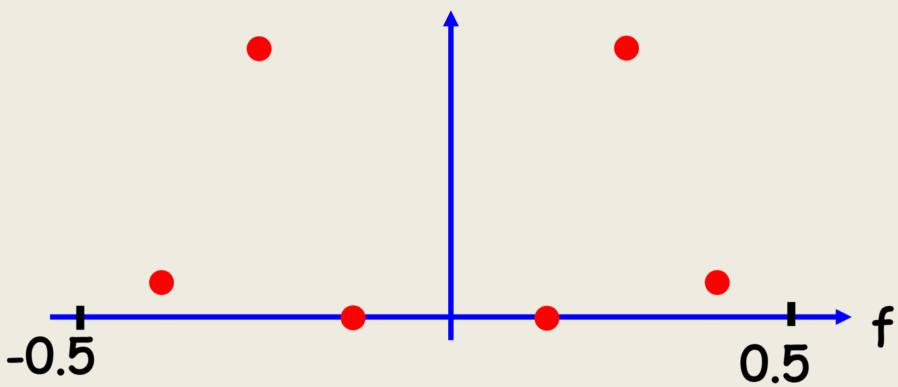
DTFT is H(z) at unit circle



$$H(z) = \frac{(z - z_1)\cdots(z - z_M)}{(z - p_1)\cdots(z - p_N)}$$

EITF75, DTFT

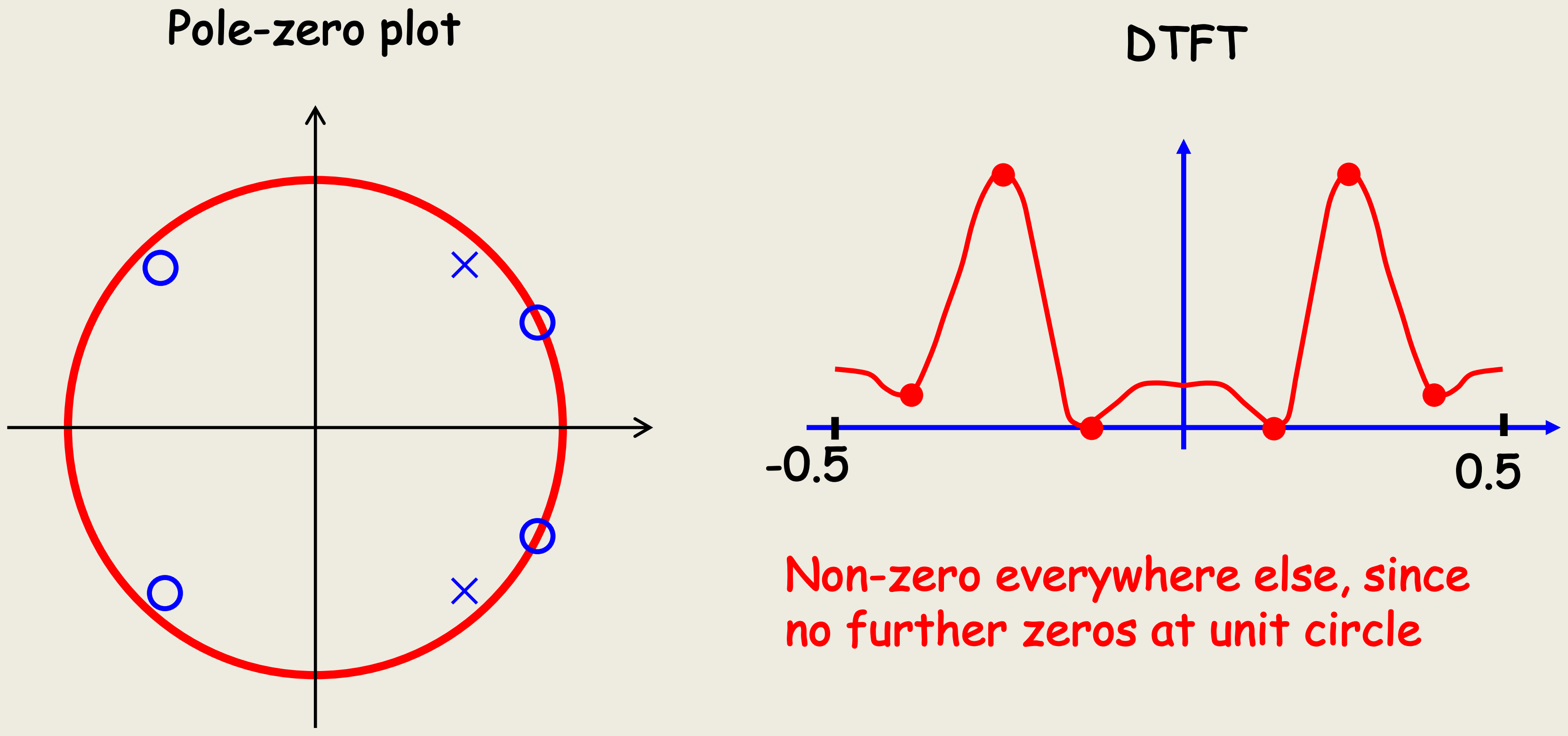




We are close to a zero

$$H(z) = \frac{(z - z_1)\cdots\boxed{(z - z_M)}}{(z - p_1)\cdots(z - p_N)}$$

EITF75, DTFT

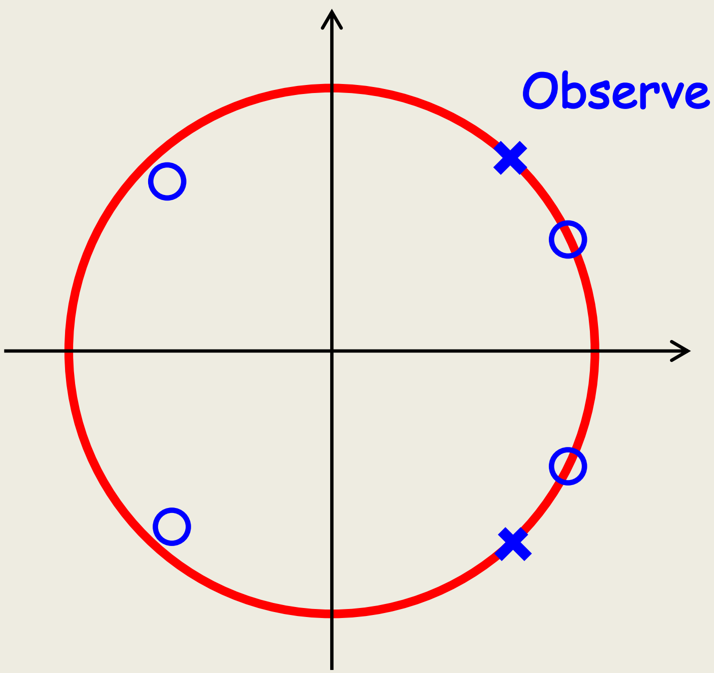


Non-zero everywhere else, since
no further zeros at unit circle

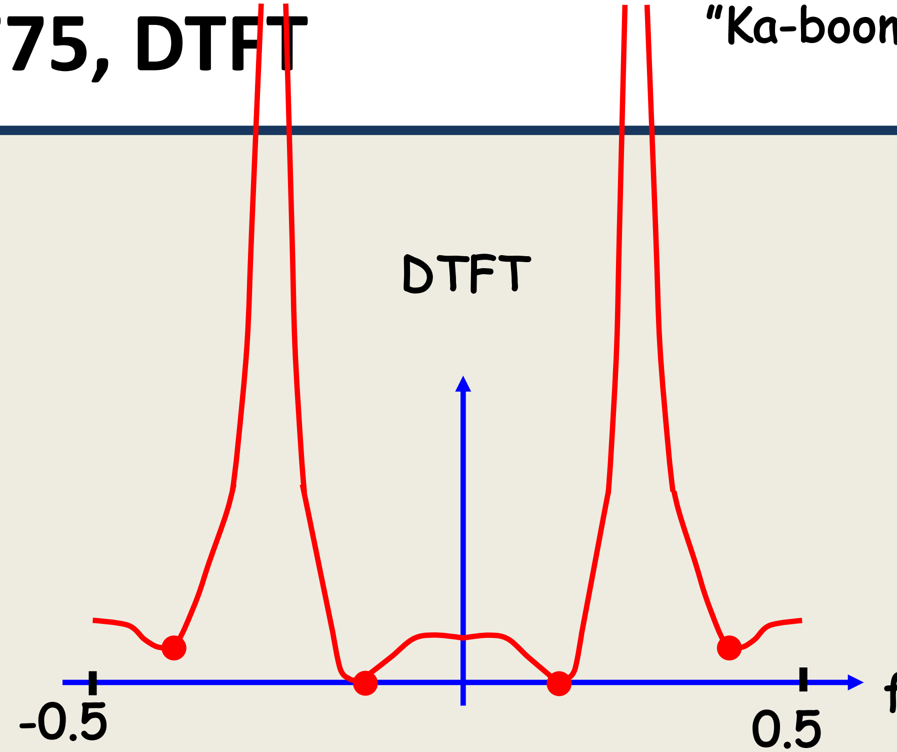
EITF75, DTFT

"Ka-boom"

Pole-zero plot

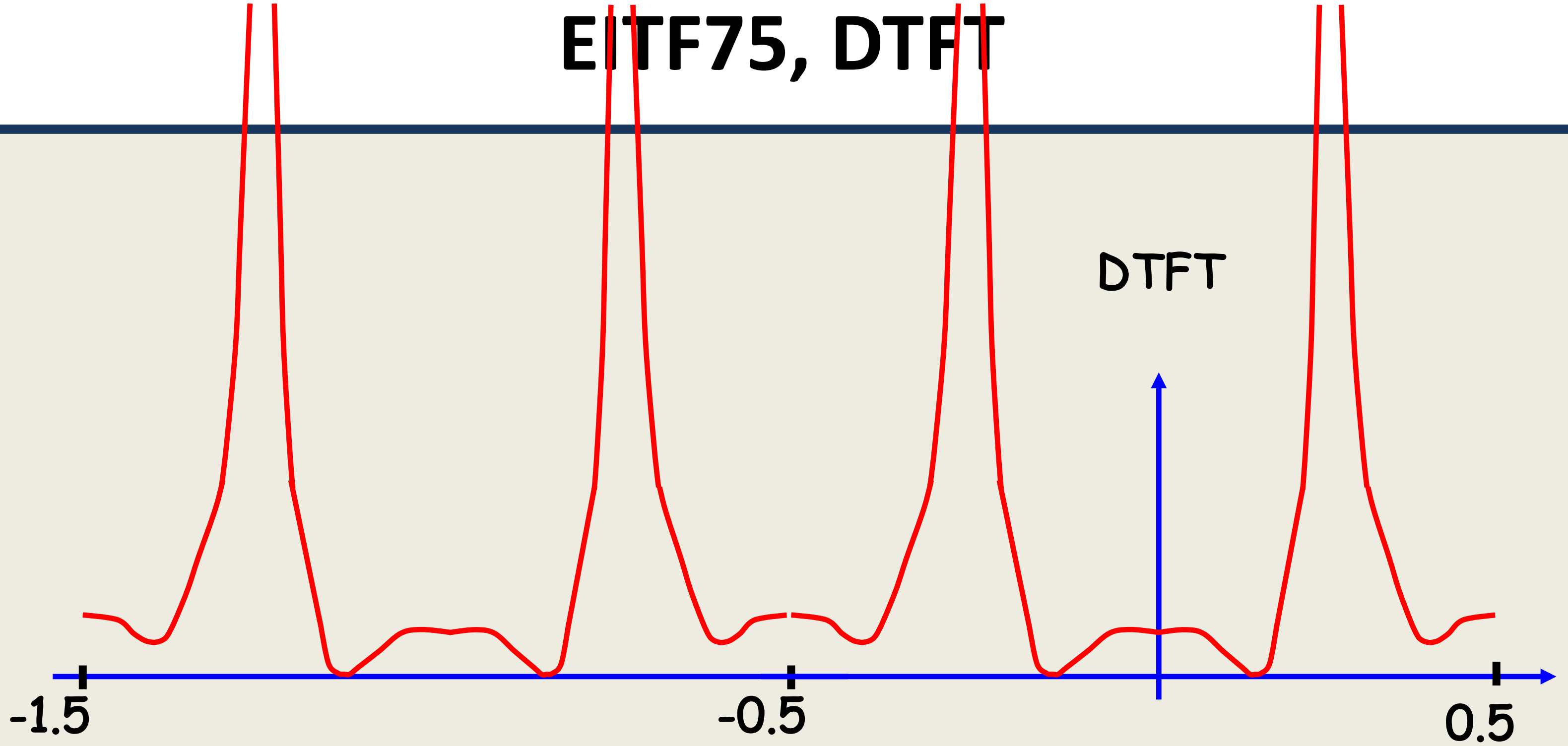


DTFT



Unstable

EITF75, DTFT



Final remark: X(f) is periodic

EITF75 Systems and Signals

DTFT Z-transform

For stable h(n) $H(f) = H\left(e^{i2\pi f}\right)$

For input $x(n) = \exp(i2\pi f_0 n)$

We get the output $y(n) = H(f_0)\exp(i2\pi f_0 n)$

Important: An LTI system cannot create frequencies not present in the input signal

EITF75 Systems and Signals

For cos/sin

Inputs

$$x(n) = \cos(2\pi f_0 n)$$

$$x(n) = \sin(2\pi f_0 n)$$

Outputs (LTI system)

$$y(n) = |H(f_0)| \cos(2\pi f_0 + \Theta(f_0))$$

$$y(n) = |H(f_0)| \sin(2\pi f_0 + \Theta(f_0))$$

ot

EITF75 Systems and Signals

Assume oscillating input, but turned on at n=0

$$x(n) = \cos(\omega_0 n) \cdot u(n) \quad \omega_0 = \frac{2\pi}{16}$$

Steady state solution (i.e., y(n) at big n) is the same as before. At small n, there is a transient behavior.

$$Y(z) = H(z)X(z) = \frac{B(z)}{A(z)}X(z)$$

$$= \frac{B(z)}{A(z)} \frac{1-\cos(\omega_0)z^{-1}}{1-2\cos(\omega_0)z^{-1}+z^{-2}}$$

$$= \sum_{n=1}^{N} \frac{A_n}{1-p_n z^{-1}} + \frac{X_1 + X_2 z^{-1}}{1-2\cos(\omega_0)z^{-1}+z^{-2}}$$

Transient (if all poles inside unit circle)

Steady state (same as for infinite cos)

EITF75 Systems and Signals

Parseval's formula

$$\sum_{n=-\infty}^{\infty} x(n)y^*(n) = \int_{-0.5}^{0.5} X(f)Y^*(f)\mathrm{d}f$$

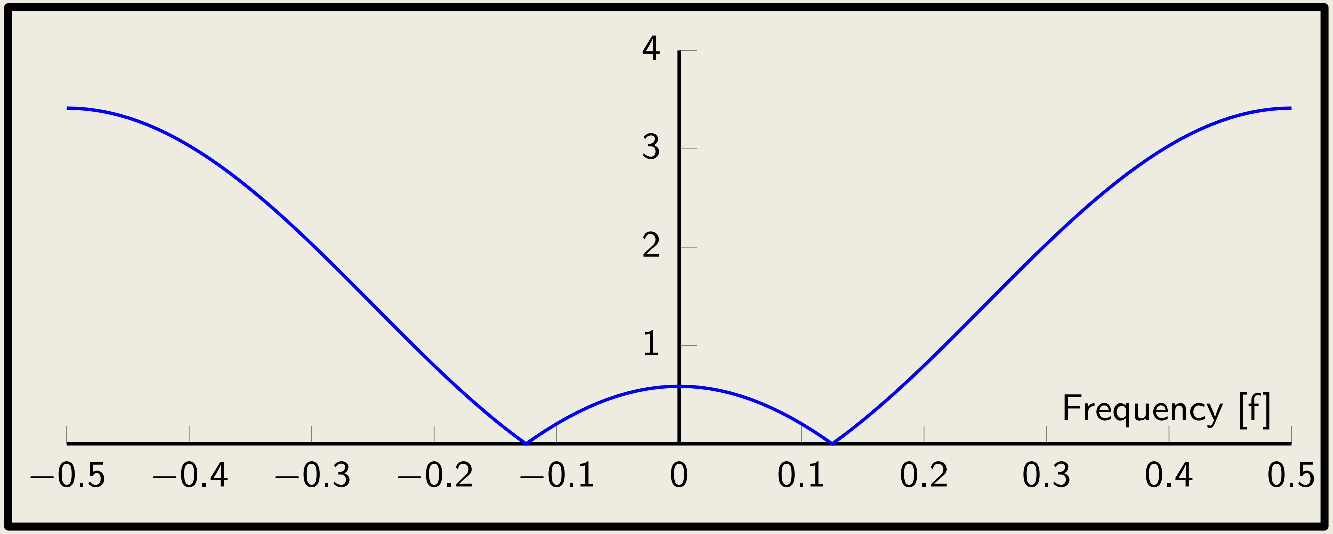
Special case: $y(n) = x(n)$

$$\sum_{n=-\infty}^{\infty} |x(n)|^2 = \int_{-0.5}^{0.5} |X(f)|^2\mathrm{d}f$$

EITF75 Systems and Signals

Some filter design

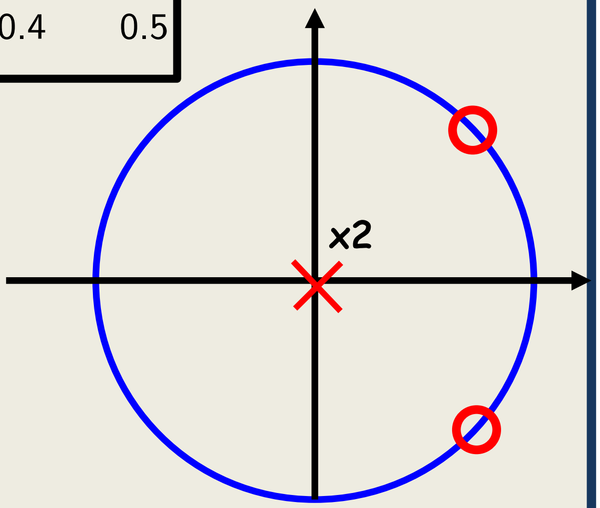
EITF75 Systems and Signals



Magnitude response

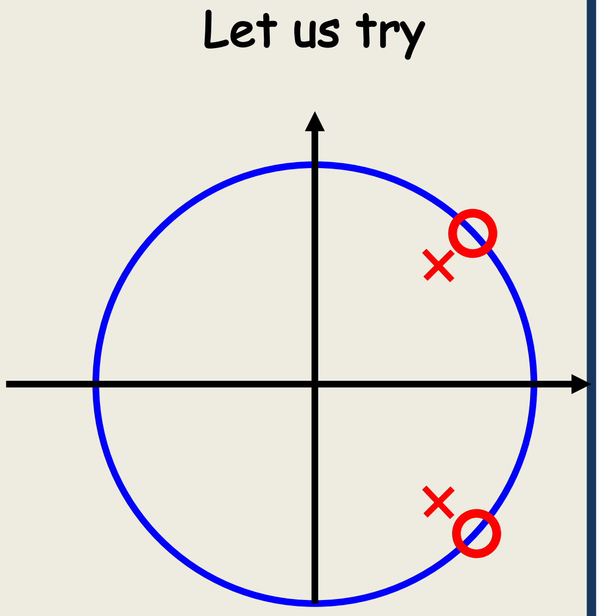
FIASCO

An attempt to cancel f=0.125 by using two zeros

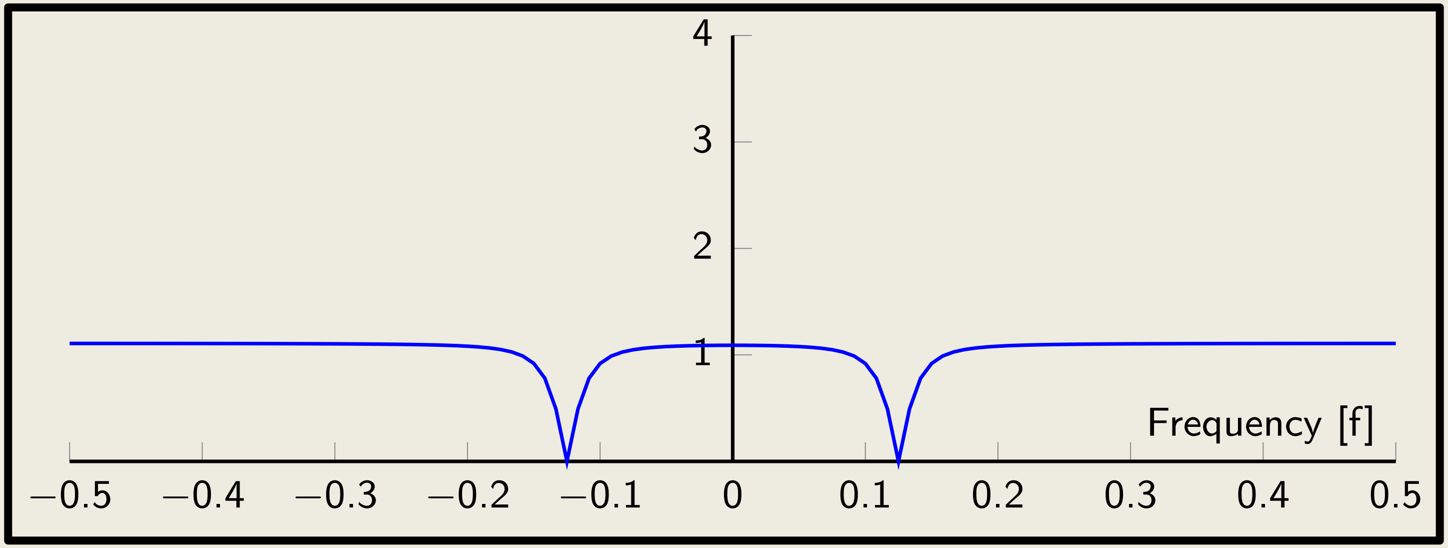


$h(n) = \{ \underline{1} \quad -2\cos(w_0) \quad 1 \}$

EITF75 Systems and Signals



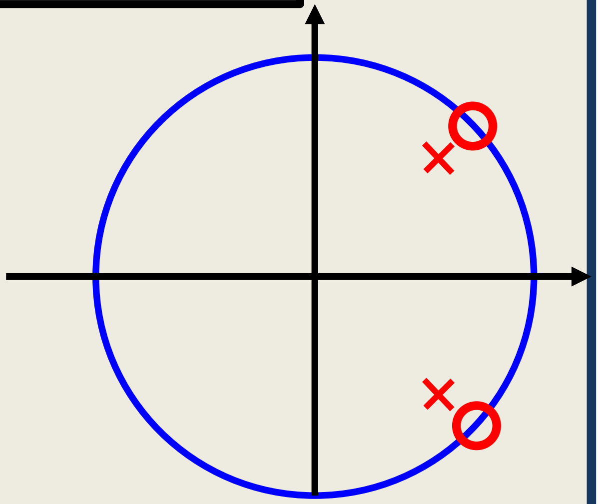
EITF75 Systems and Signals



Magnitude response

Much better

NOTCH filter



EITF75 Systems and Signals

FIR filters with linear phase

Linear phase is desirable since it delays all frequencies equally much

Linear phase is defined as $$\Theta(\omega) = \kappa\omega + \pi\ell$$

Whenever there is a phase jump with π, this should be seen as a magnitude response that is negative

$h(n) = h(-n)$	Symmetry around n=0. Not causal
$h(n) = h(N-n)$	Symmetry around n=(N-1)/2.
$h(n) = -h(N-n)$	Anti-symmetry around n=(N-1)/2.

Three types of linear phase filters

EITF75 Systems and Signals

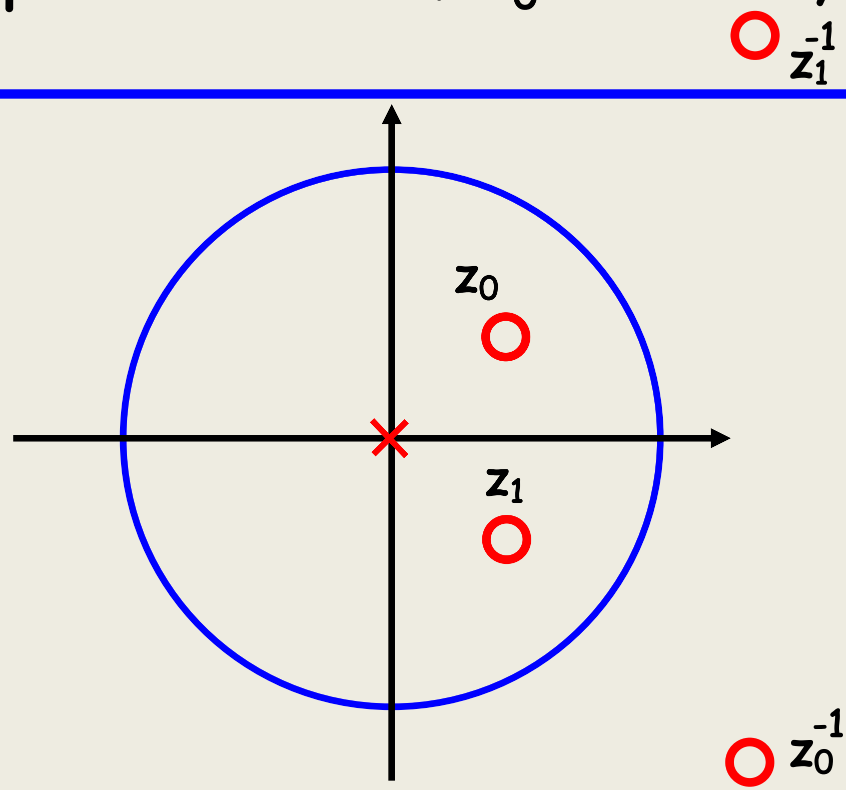
Example TYPE 1

$$h(n) = \left\{ \begin{matrix} 1 & 2 & \underline{3} & 2 & 1 \end{matrix} \right\} \qquad H(z) = 1 + 2z^{-1} + 3z^{-2} + 2z^{-3} + z^{-4}$$

EITF75 Systems and Signals

Pole-Zero diagram for linear phase FIR filters

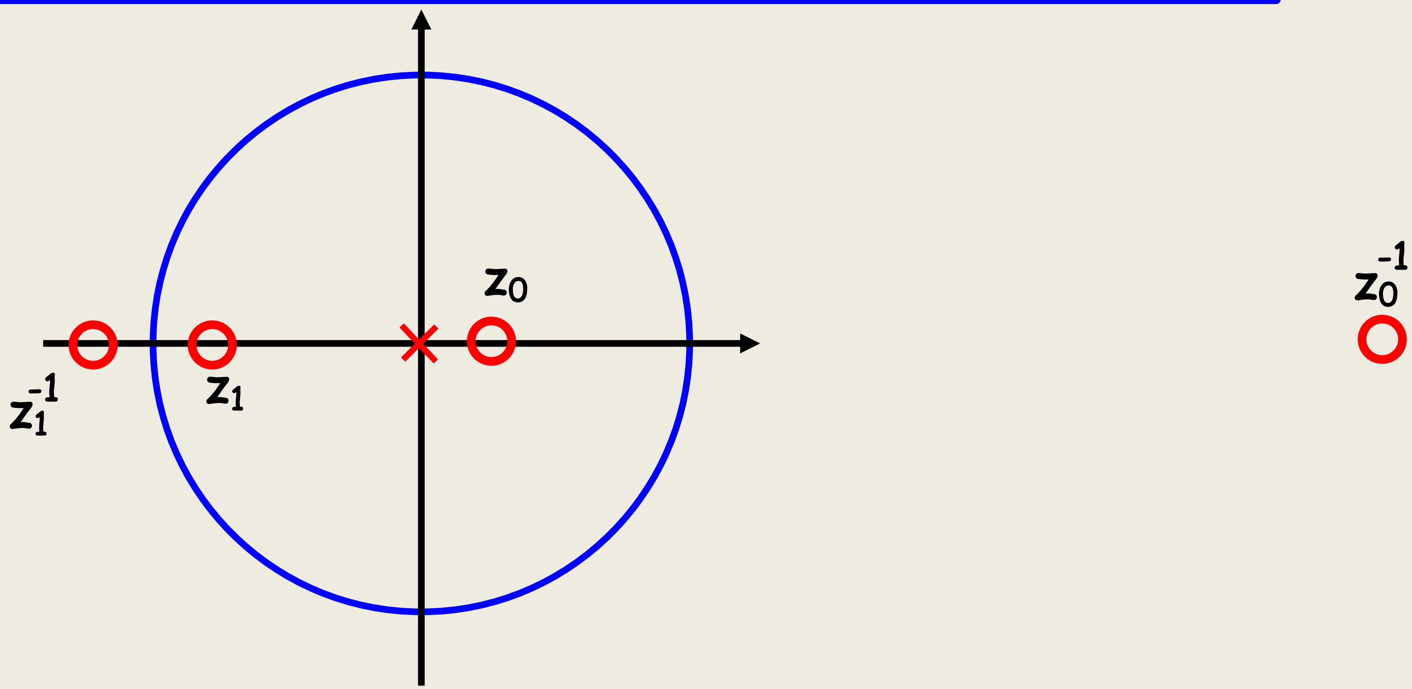
Linear phase <-> If z_0 is a zero, so is z_0^{-1}



EITF75 Systems and Signals

Pole-Zero diagram for linear phase FIR filters

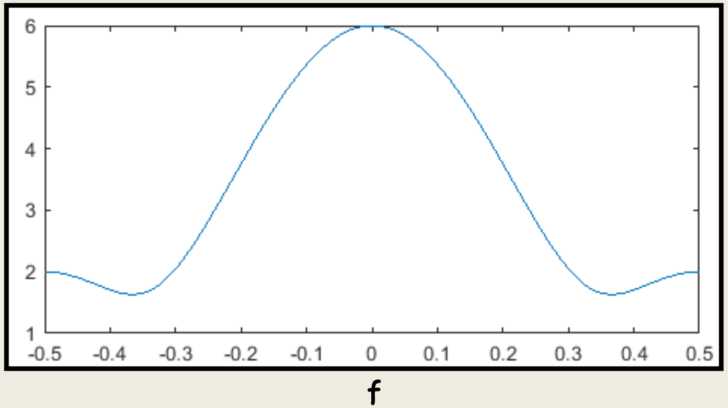
Linear phase <-> If z_0 is a zero, so is z_0^{-1}



EITF75 Systems and Signals

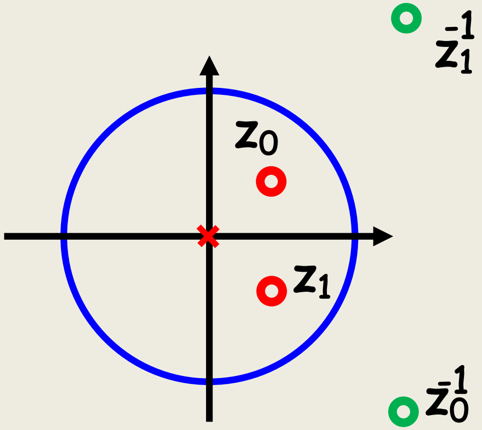
Minimum phase filters

$$|H(f)| = |H^*(f)|$$

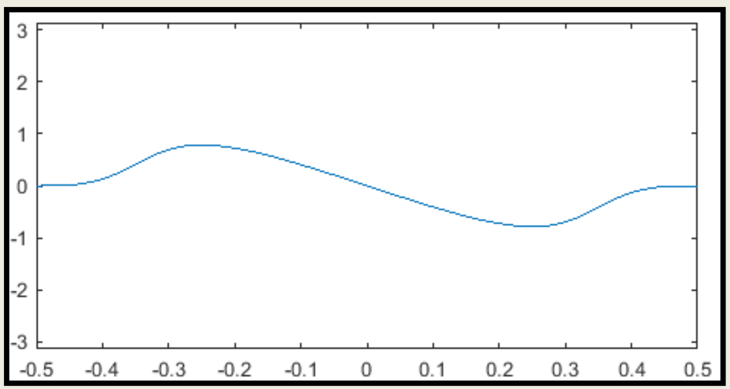


This is a general rule:
A filter with all zeros inside the unit circle has smaller phase.

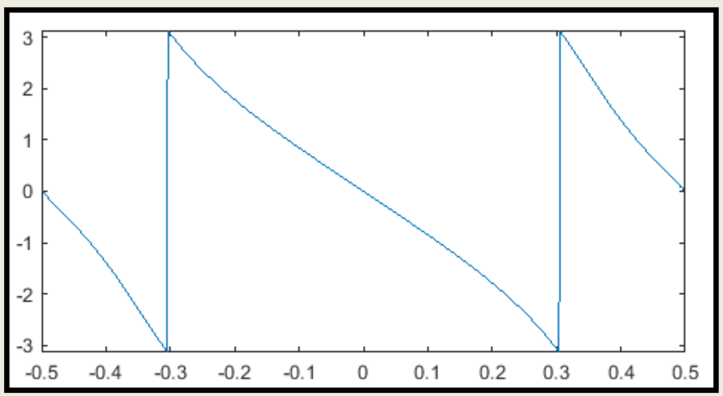
Minimum phase filter
Maximum phase filter



$\Theta(f)$



$\Theta(f)$



EITF75 Systems and Signals

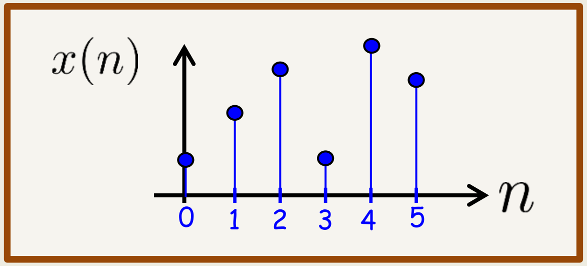
The DFT

EITF75 Systems and Signals

Background and motivation for yet another transform

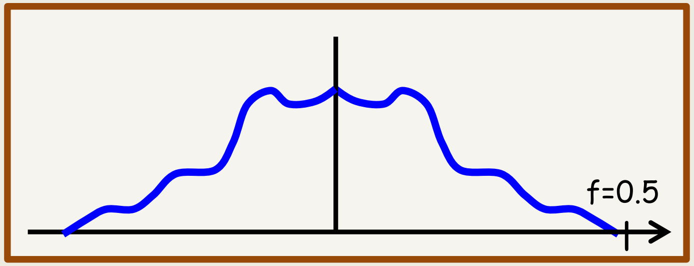
The discrete Fourier Transform (DFT) in one sentence:
A Fourier version of x(n) with 6 numbers

Besides, the DTFT is terribly inefficient



These 6 numbers, are in the frequency domain represened by a continuous curve !

It should be possible to Fourier represent x(n) by 6 numbers as well



EITF75 Systems and Signals

Formal definition

For a sequence $x(n)$ of arbitrary length, the N-point DFT is defined as

$$X_{\mathrm{DFT}}(k) = \sum_{n=0}^{N-1} x(n) e^{-j2\pi \cdot \frac{k}{N} \cdot n} \qquad \text{for } k = 0, 1, \ldots, N-1$$

and the inverse transform (IDFT) as

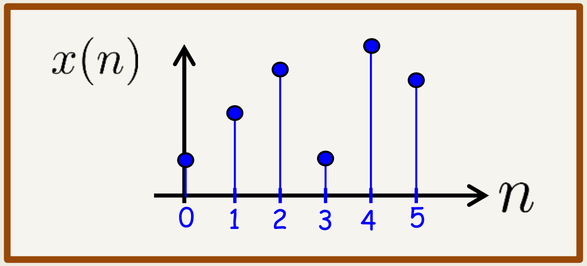
$$x_{\mathrm{IDFT}}(n) = \frac{1}{N} \cdot \sum_{k=0}^{N-1} X(k) e^{j2\pi \cdot \frac{k}{N} \cdot n} \qquad \text{for } n = 0, 1, \ldots, N-1$$

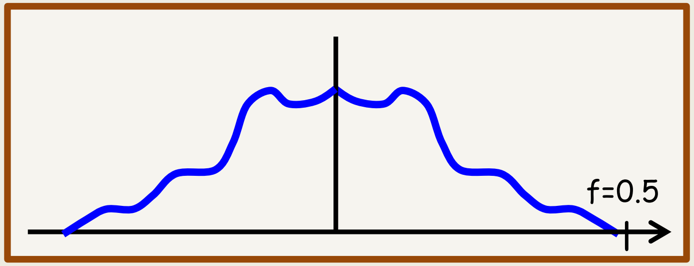
Result

if the length of $x(n)$ is N, then

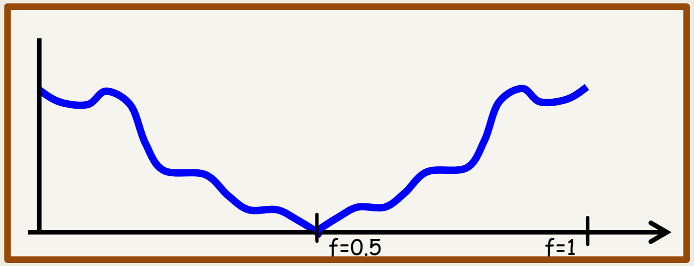
$x_{\mathrm{IDFT}}(n) = x(n)$ and $X_{\mathrm{DFT}}(k) = X(f \mid f = k/N)$

EITF75 Systems and Signals



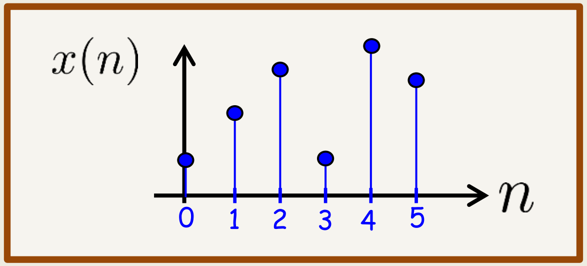


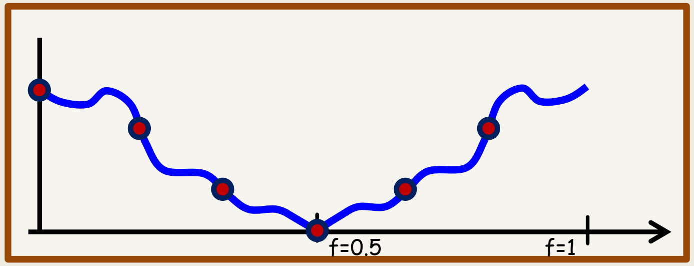
The DTFT is periodic



We can represent it like this

EITF75 Systems and Signals



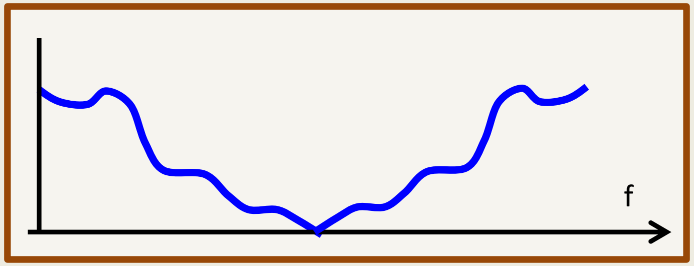


A 6-point DFT would compute the samples of the DTFT

This is sufficient to represent x(n)

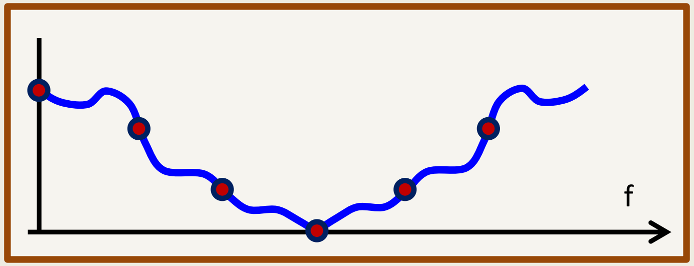
Important: The DFT size must be at least as long as the signal, otherwise there is a loss (aliasing in time)

EITF75 Systems and Signals



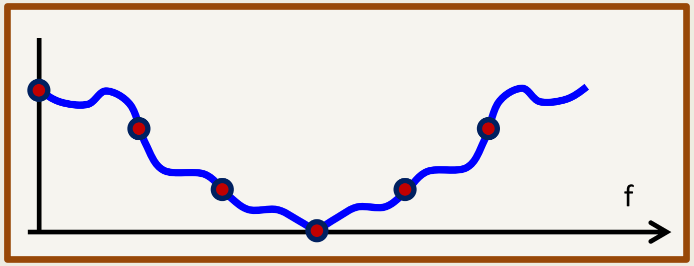
Assume a DTFT of a 6-tap signal

EITF75 Systems and Signals

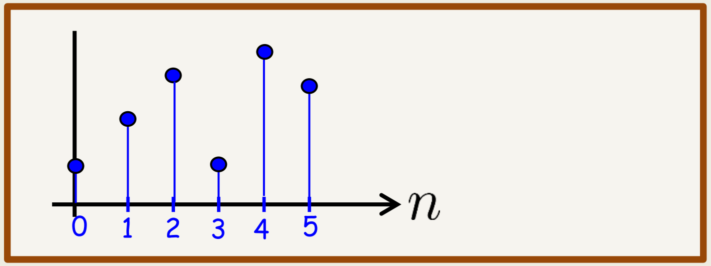


Sample it

EITF75 Systems and Signals



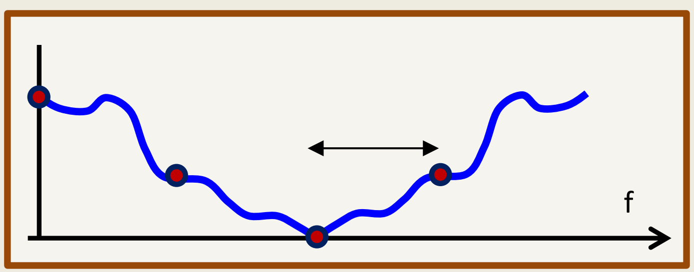
Sample it



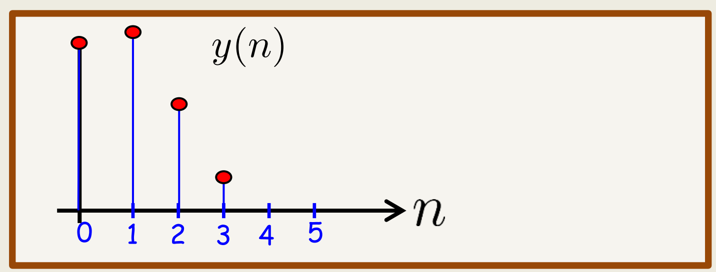
Compute the "other domain" representation from samples. In this case, the time domain

EITF75 Systems and Signals

The time-aliasing only occurs if we are not careful with the DFT size. If it is equal or larger than the length of the signal, there is no time-aliasing



But if sample spacing is too small…



There is aliasing

Periodically extended

Aliasing $y(n) = \sum_{m=-\infty}^{\infty} x(n - mN)$

No aliasing $y(n) = x(n)$

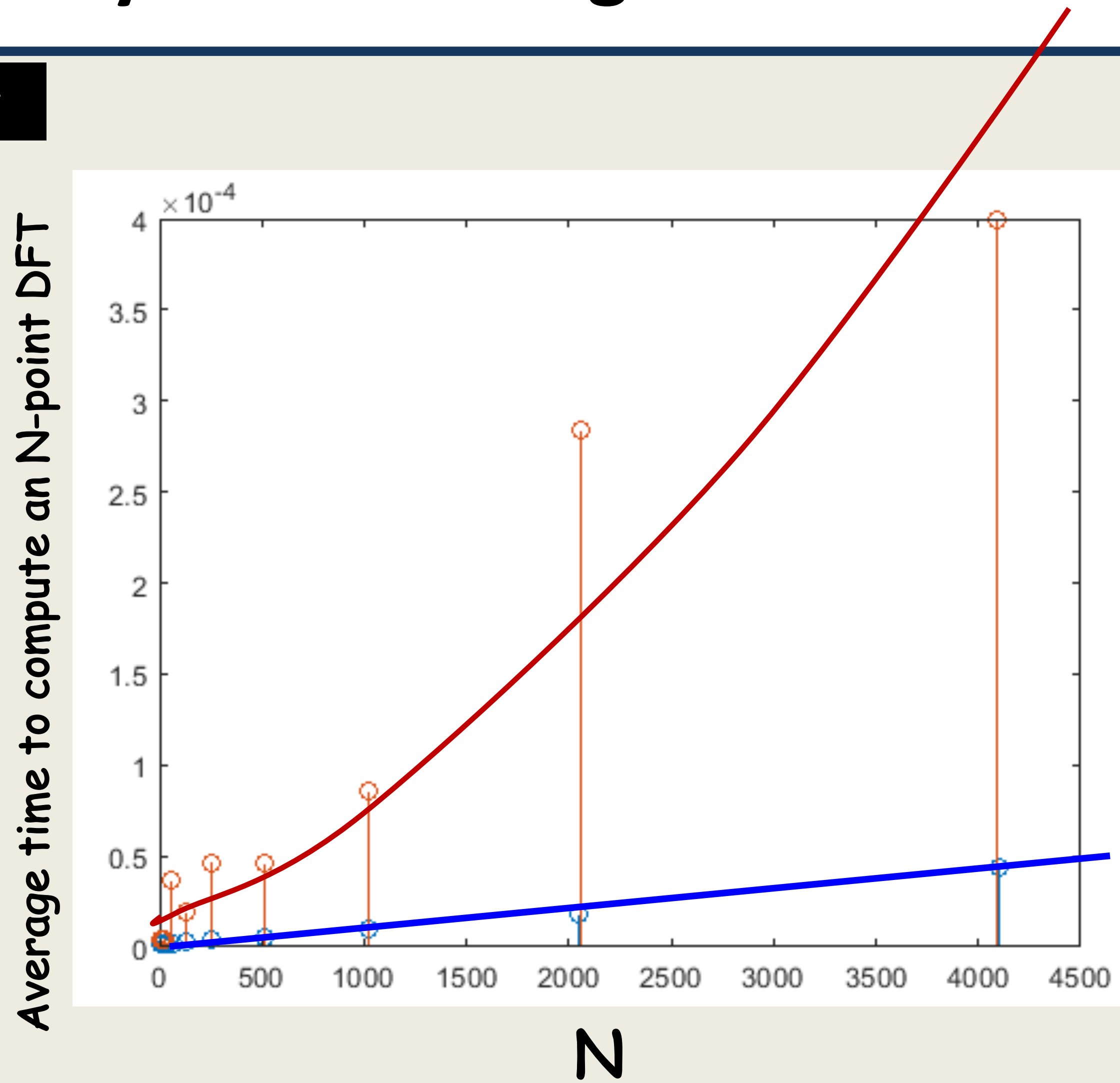
EITF75 Systems and Signals

Computational complexity

Test in Matlab

2048 is 2^{11}

Significant speed-up possible for $N=2^k$



EITF75 Systems and Signals

Computational complexity

FFT not included in course, but good to know about

Cooley and Tukey 1965

Method known to, and used by, Gauss in 1805

Fast Fourier transform (FFT)

If $N=2^k$, then $N \log_2(N)$ complexity to compute

$$X_{\mathrm{DFT}}(k) = \sum_{n=0}^{N-1} x(n) e^{-j2\pi \cdot \frac{k}{N} \cdot n} \qquad \text{for } k = 0, 1, \ldots, N-1$$

Made possible by some algebraic manipulations and tricks.

The importance of the FFT cannot be underestimated. WIFI and 4G, etc could not been implemented without the FFT

For a computer,

1. It can avoid the continuous DTFT
2. It can compute the DFT extremely fast

EITF75 Systems and Signals

Properties

For DTFTs, we have

$$x(n) \star y(n) \leftrightarrow X(f)Y(f)$$

$$x(n) \leftrightarrow X(f) \qquad x(n-n_0) \leftrightarrow X(f)\mathrm{e}^{-i2\pi f n_0}$$

For DFTs, we have

$$x_1(n) \otimes x_2(n) \leftrightarrow X(k)Y(k)$$

$$x(n-n_0 \bmod N) \leftrightarrow X(k)\mathrm{e}^{-i2\pi k n_0/N}$$

where $x_1(n) \otimes x_2(n) = \sum_{k=0}^{N-1} x_1(k)x_2(n-k \bmod N)$

Circular convolution

EITF75 Systems and Signals

Example

Linear convolution computed via DFTs

Given: Two length N sequences, x(n), y(n)

Task: Compute their linear convolution by using DFT and its inverse IDFT

```
>> x=[1 2 3 4];
>> y=[2 2 1 1];
>> yL=conv(x,y)
yL =
     2     6    11    17    13     7     4
```

This is the result, But not computed via DFT

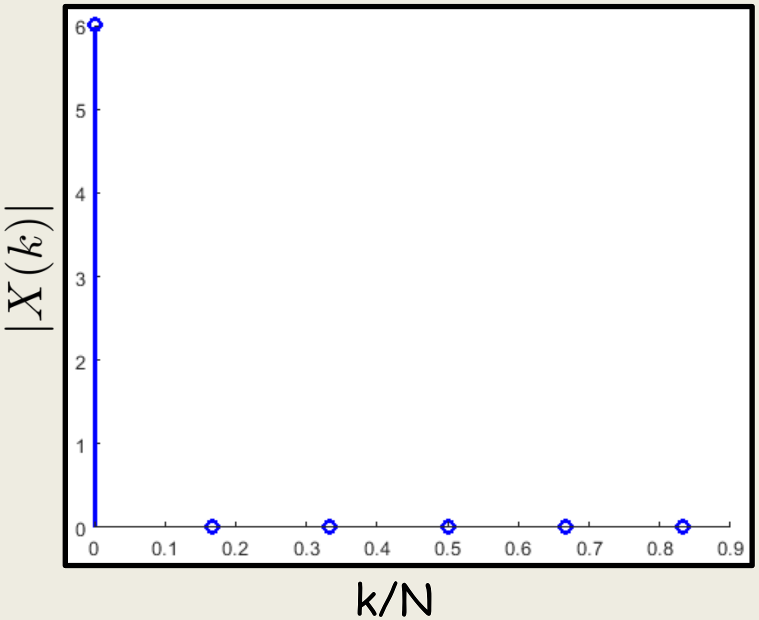
```
>> xp=[1 2 3 4 0 0 0 0];
>> yp=[2 2 1 1 0 0 0 0];
>> yL=ifft(fft(xp).*fft(yp))
yL =
    2.0000    6.0000   11.0000   17.0000   13.0000    7.0000    4.0000   -0.0000
```

Still a circular convolution carried out, but due to zero-padding, it behaves linear.

More examples: Resolution increase

$$x(n) = \{1\ 1\ 1\ 1\ 1\ 1\}$$

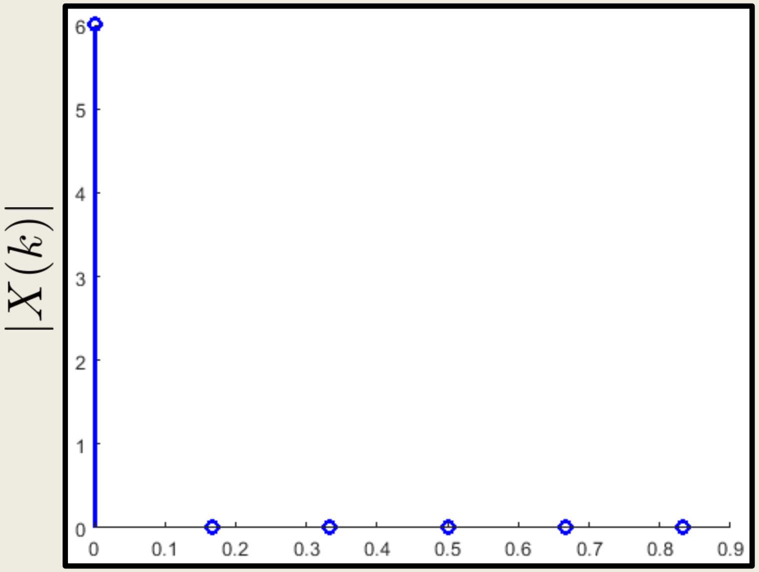
Compute DFT (N=6)



More examples: Resolution increase

$$x(n) = \{1\ \ 1\ \ 1\ \ 1\ \ 1\ \ 1\ \ 0\ \ 0\}$$

Compute DFT (N=8)



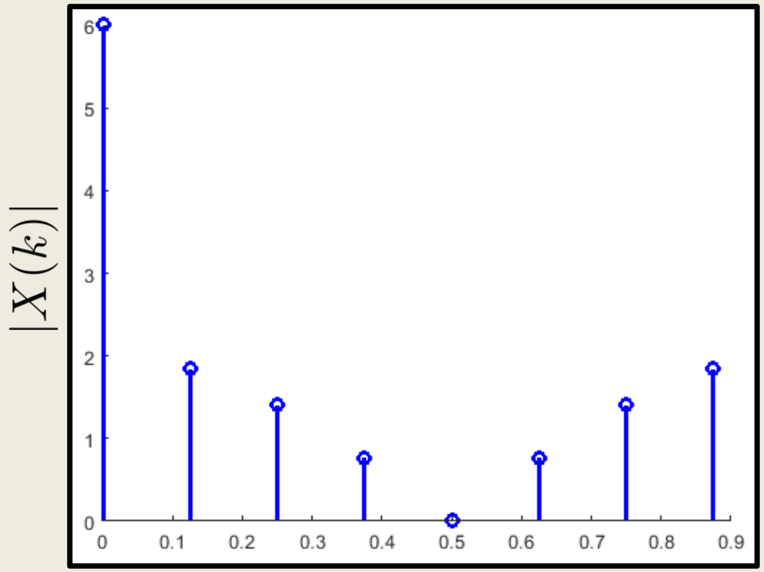
k/N

EITF75 Systems and Signals

More examples: Resolution increase

$$x(n) = \{1 \;\; 1 \;\; 1 \;\; 1 \;\; 1 \;\; 1 \;\; 0 \;\; 0\}$$

Compute DFT (N=8)



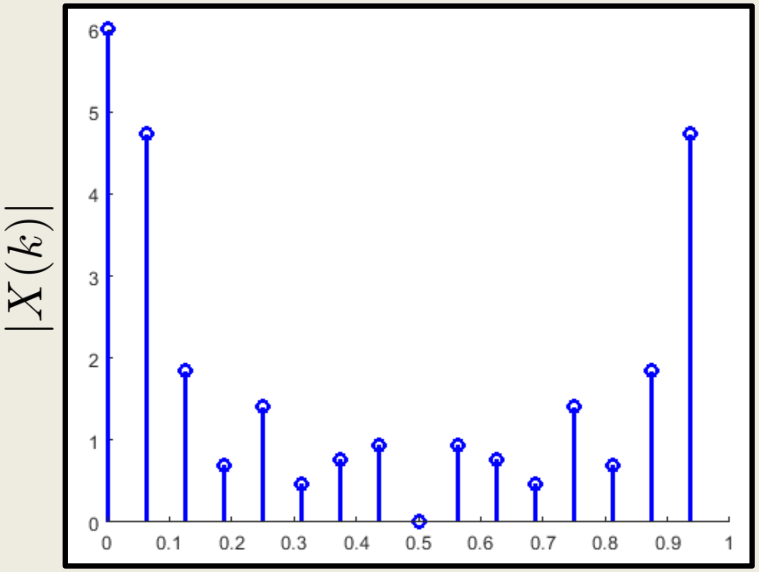
k/N

EITF75 Systems and Signals

More examples: Resolution increase

$$x(n) = \{1\ 1\ 1\ 1\ 1\ 1\ 0\ 0 \cdots\cdots\}$$

Compute DFT (N=16)



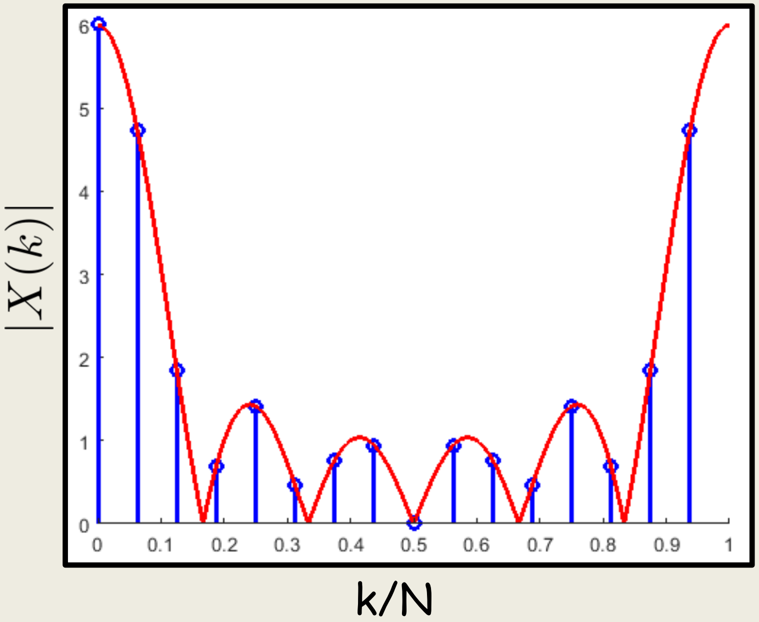
k/N

EITF75 Systems and Signals

More examples: Resolution increase

$$x(n) = \{1\ 1\ 1\ 1\ 1\ 1\ 0\ 0 \cdots\cdots\}$$

Compute DFT (N=16)



What is this line?

DFT size larger-or-equal to the length of x(n)

Therefore, DFT samples of DTFT