EITF75 Systems and Signals

Lecture 11
More on DFTs

Fredrik Rusek

EITF75 Systems and Signals

Recap

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

$$X_{\mathrm{DFT}}(k) = \sum_{n=0}^{N-1} x(n) \mathrm{e}^{-\mathrm{j}2\pi \cdot \frac{k}{N} \cdot n} \qquad \text{for } k = 0, 1, \ldots, N-1$$

and the inverse transform (IDFT) as

$$x_{\mathrm{IDFT}}(n) = \frac{1}{N} \cdot \sum_{k=0}^{N-1} X(k) \mathrm{e}^{\mathrm{j}2\pi \cdot \frac{k}{N} \cdot n} \qquad \text{for } n = 0, 1, \ldots, N-1$$

DFTF

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

Formula for DTFT

EITF75 Systems and Signals

Recap

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

$$X_{\mathrm{DFT}}(k) = \sum_{n=0}^{N-1} x(n) \mathrm{e}^{-\mathrm{j}2\pi \cdot \frac{k}{N} \cdot n} \qquad \text{for } k = 0, 1, \ldots, N-1$$

and the inverse transform (IDFT) as

$$x_{\mathrm{IDFT}}(n) = \frac{1}{N} \cdot \sum_{k=0}^{N-1} X(k) \mathrm{e}^{\mathrm{j}2\pi \cdot \frac{k}{N} \cdot n} \qquad \text{for } n = 0, 1, \ldots, N-1$$

Suitable for computers

- Fast to compute
- Discrete)

DFTF

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

Suitable for analysis (by human)

EITF75 Systems and Signals

Convolutions

In real life, the input signal is very long, does not start/stop. Previous method (Lecture 10) fails

$$x(n) \longrightarrow \boxed{h(n)} \longrightarrow y(n) = x(n) \star h(n)$$

Three common methods

Overlap-add   to be described

Overlap-save

Overlap-discard

EITF75 Systems and Signals

Convolutions

$$x(n) \longrightarrow \boxed{h(n)} \longrightarrow y(n) = x(n) \star h(n)$$

Overlap-add

Visualization of input

.... x(57) x(58) x(59) x(60) x(1835466) ....

Visualization of impulse response

h(0) h(M-1)

EITF75 Systems and Signals

Convolutions

$$x(n) \rightarrow \boxed{h(n)} \rightarrow y(n) = x(n) \star h(n)$$

Overlap-add

Partion the input in blocks of length L

.... | $x_1(n)$ | $x_2(n)$ | etc | ....

Visualization of impulse response

| h(0) h(M-1) |

EITF75 Systems and Signals

Convolutions

$$x(n) \rightarrow \boxed{h(n)} \rightarrow y(n) = x(n) \star h(n)$$

Overlap-add

Partion the input in blocks of length L

.... | $x_1(n)$ | $x_2(n)$ | etc | ....

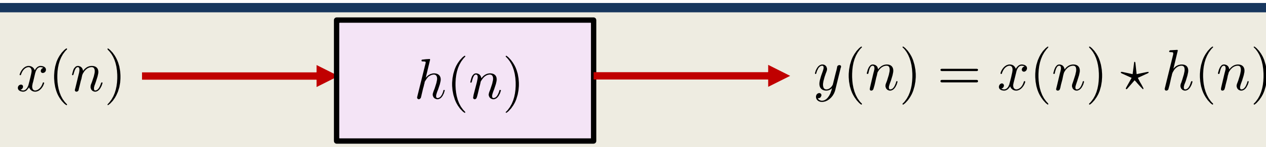
Visualization of impulse response

h(0) h(M-1)

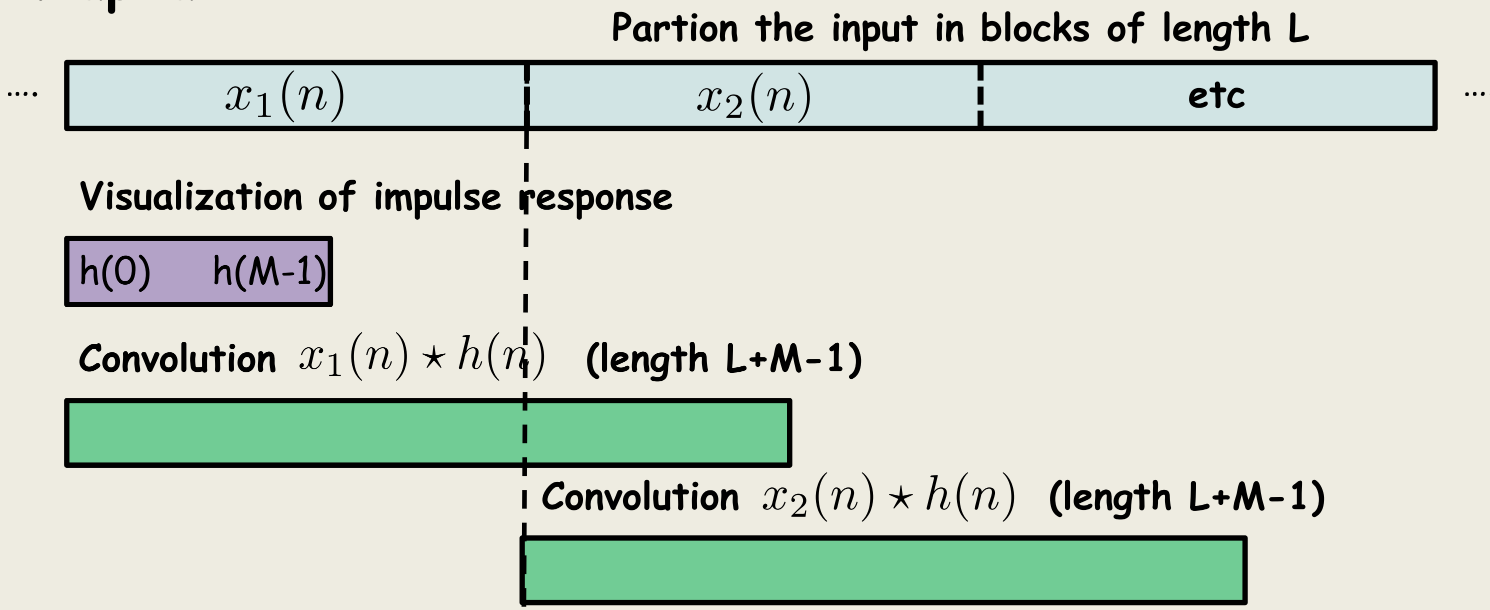
Convolution $x_1(n) \star h(n)$ (length L+M-1)

EITF75 Systems and Signals

Convolutions



Overlap-add

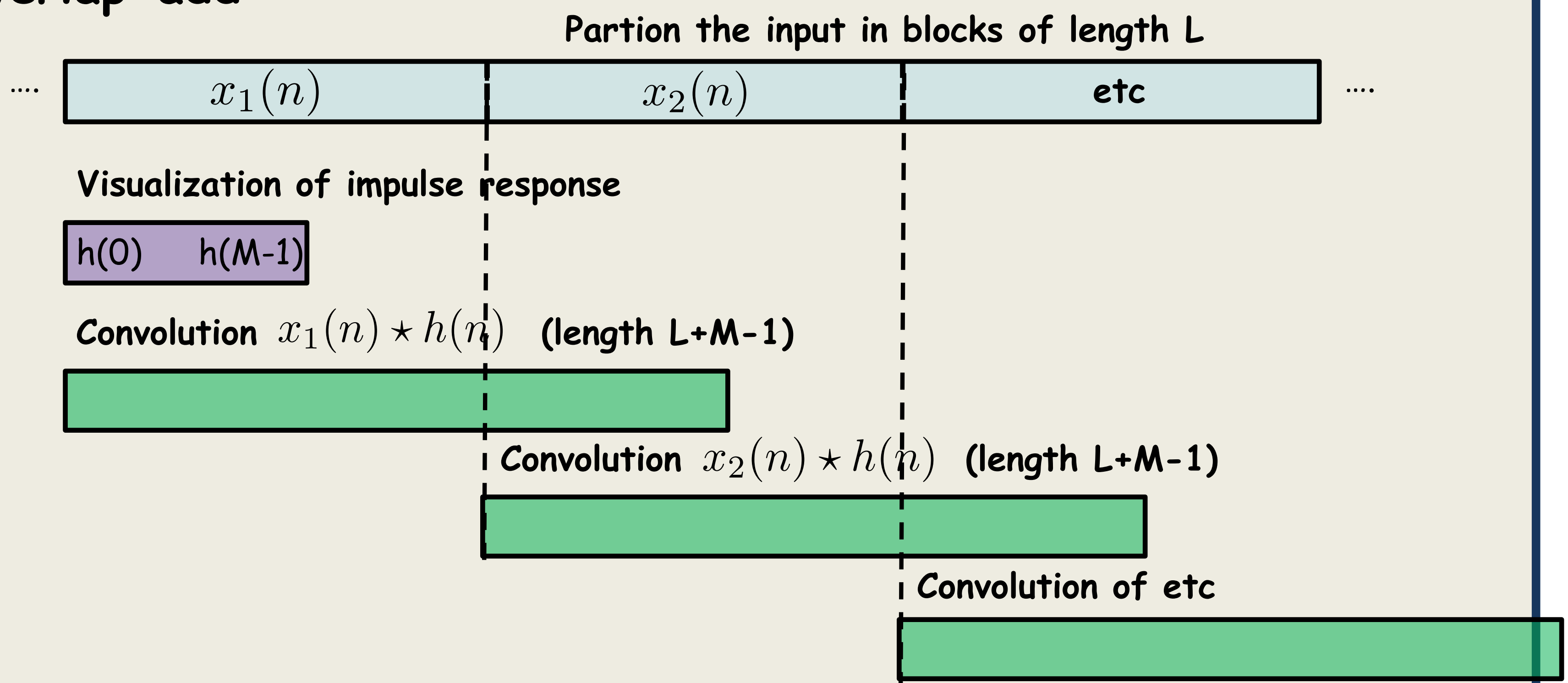


EITF75 Systems and Signals

Convolutions

$$x(n) \longrightarrow \boxed{h(n)} \longrightarrow y(n) = x(n) \star h(n)$$

Overlap-add

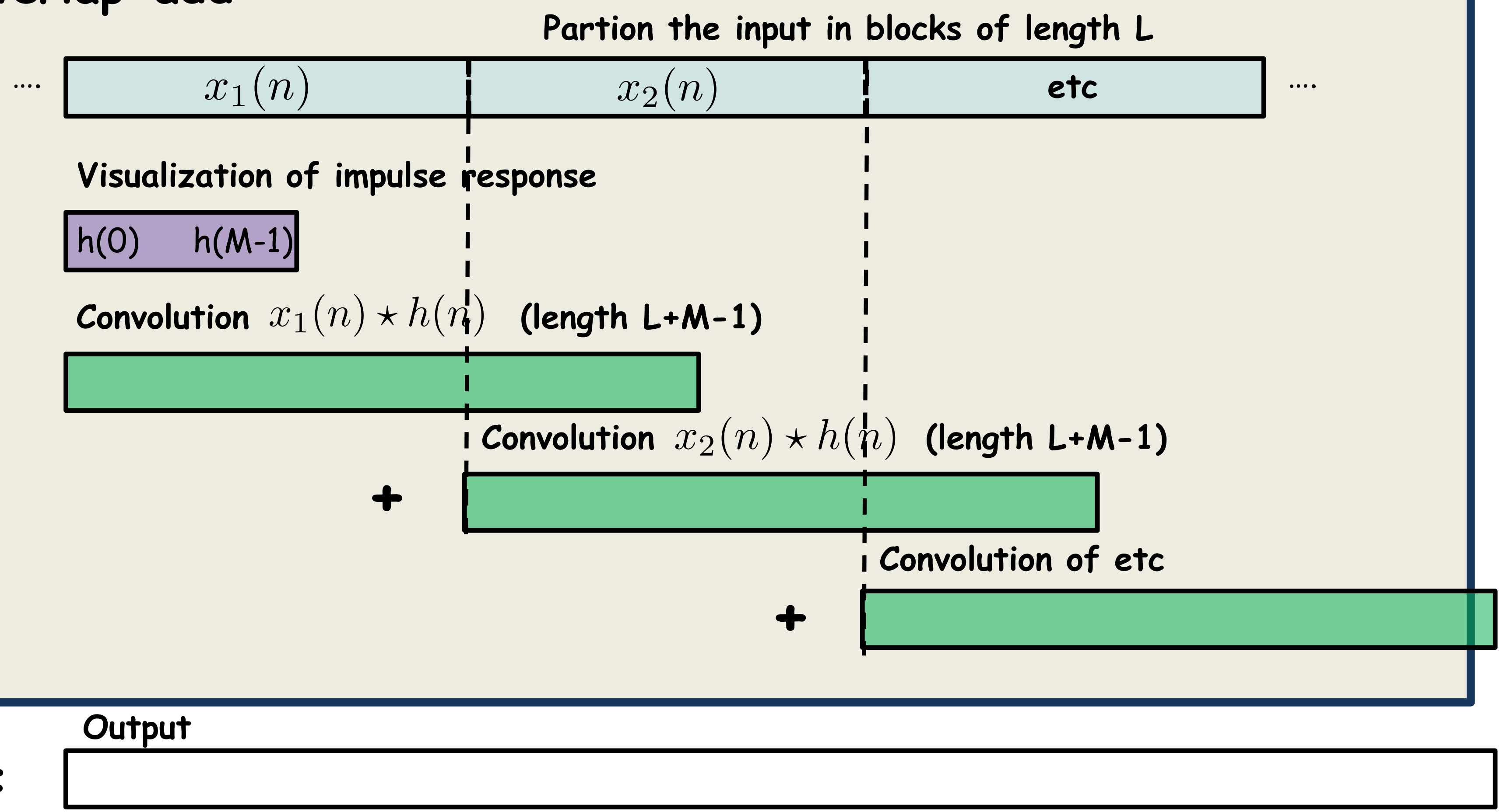


EITF75 Systems and Signals

Convolutions

$$x(n) \rightarrow \boxed{h(n)} \rightarrow y(n) = x(n) \star h(n)$$

Overlap-add

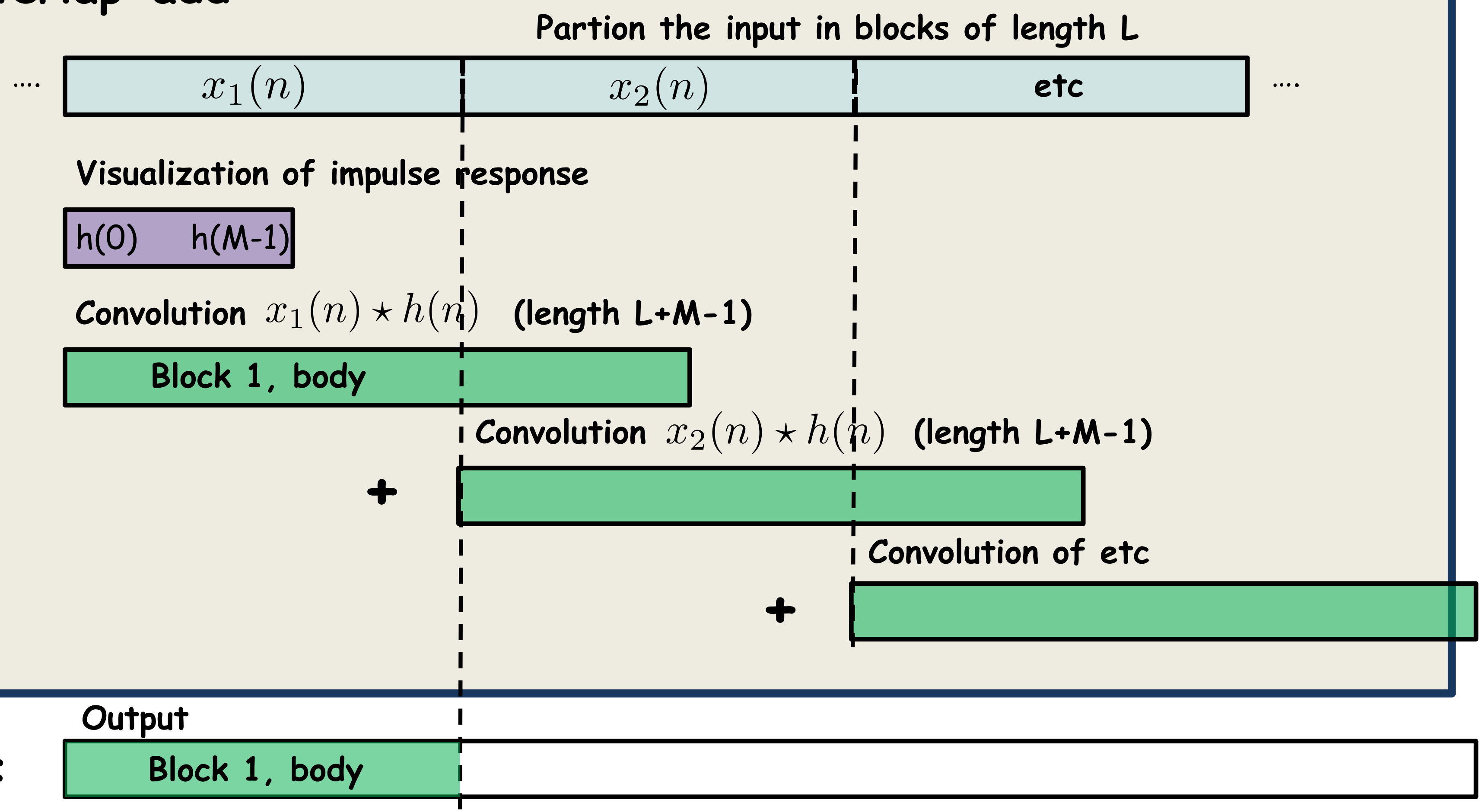


EITF75 Systems and Signals

Convolutions

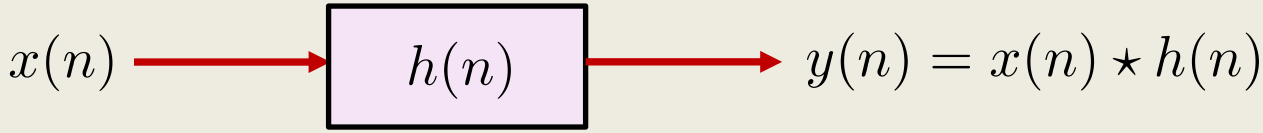
$$x(n) \rightarrow \boxed{h(n)} \rightarrow y(n) = x(n) \star h(n)$$

Overlap-add

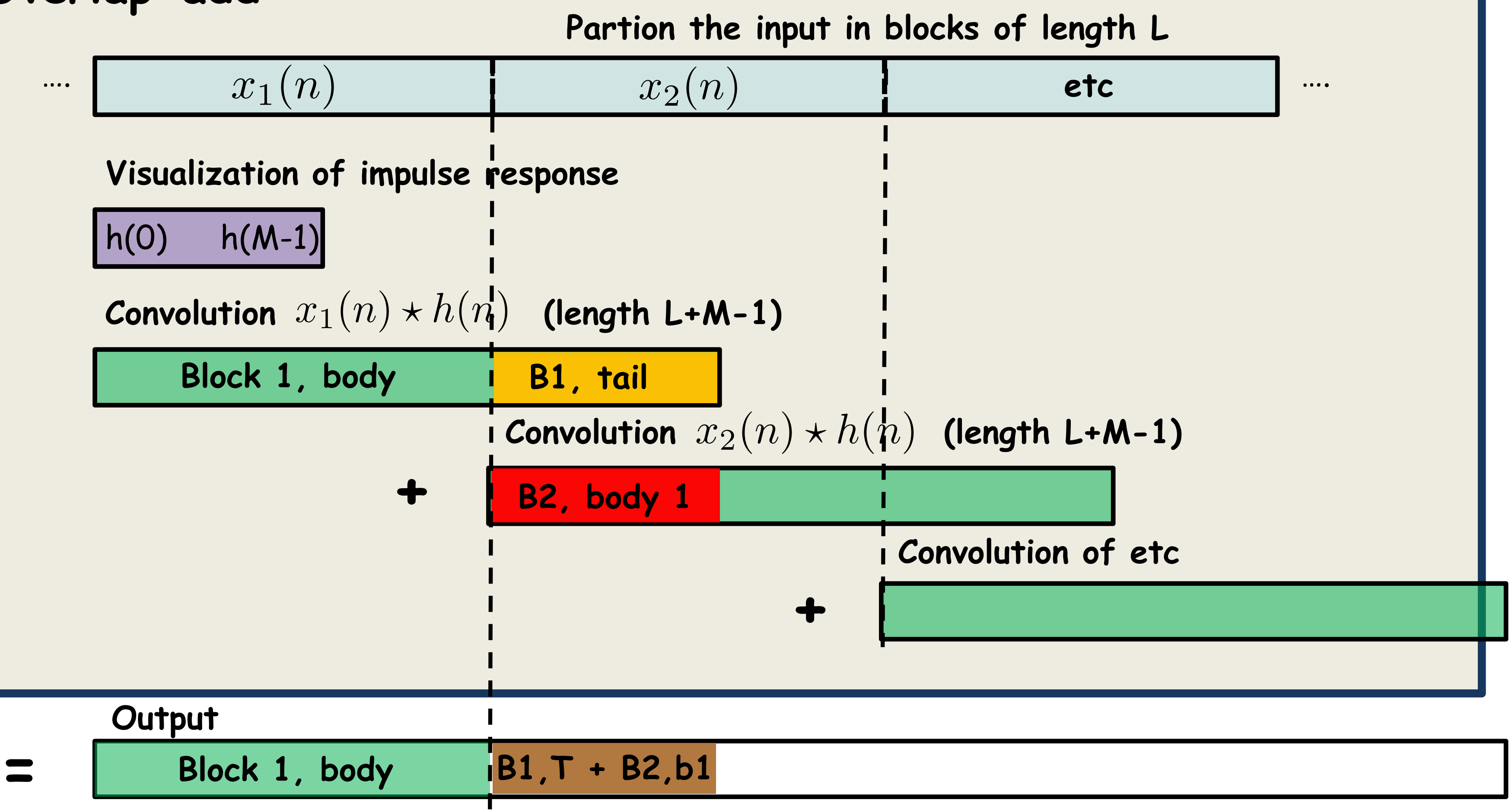


EITF75 Systems and Signals

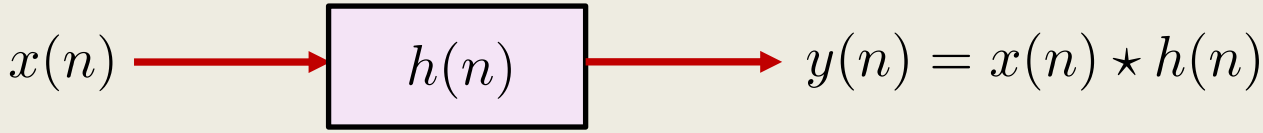
Convolutions



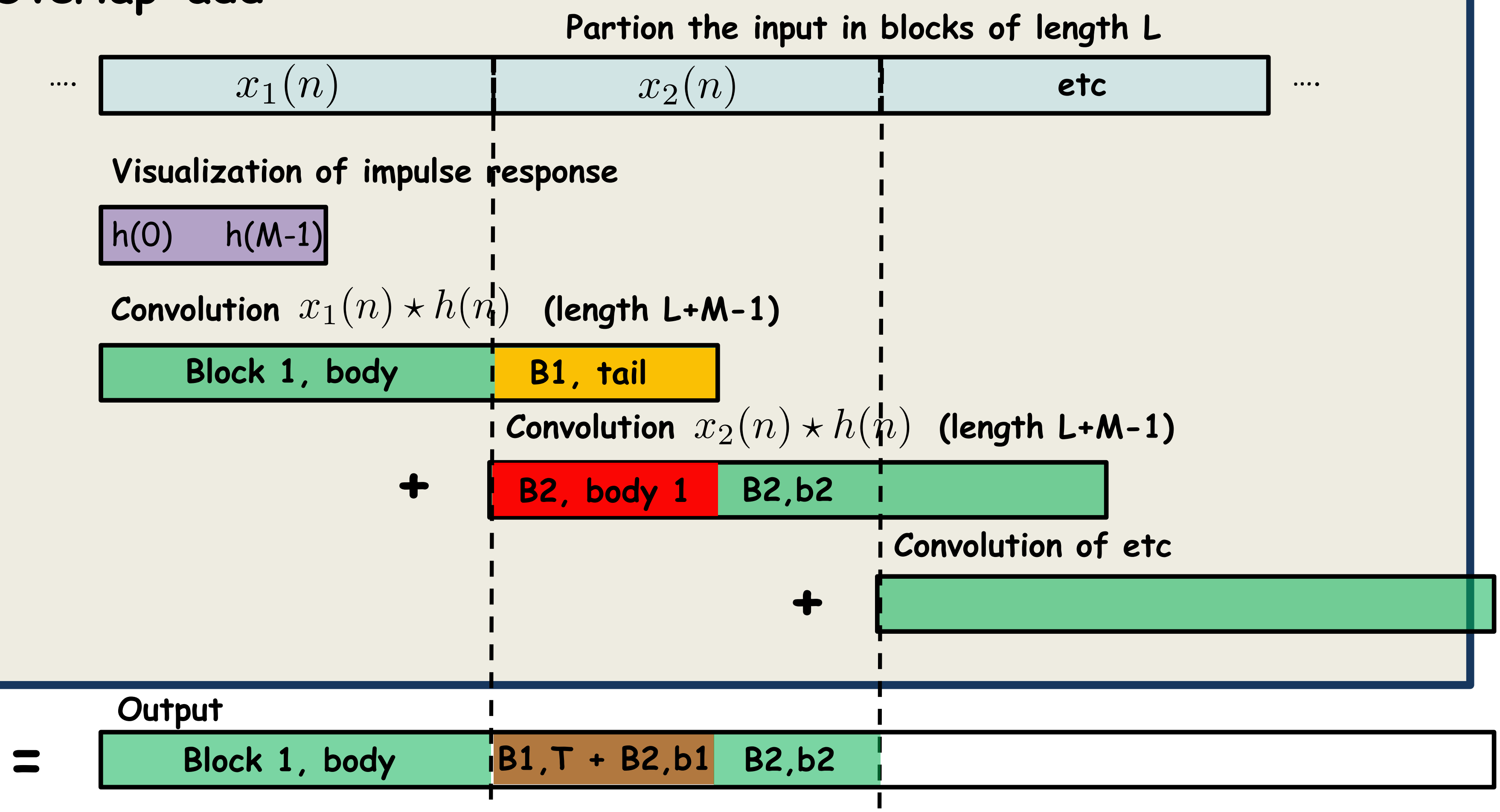
Overlap-add



Convolutions



Overlap-add

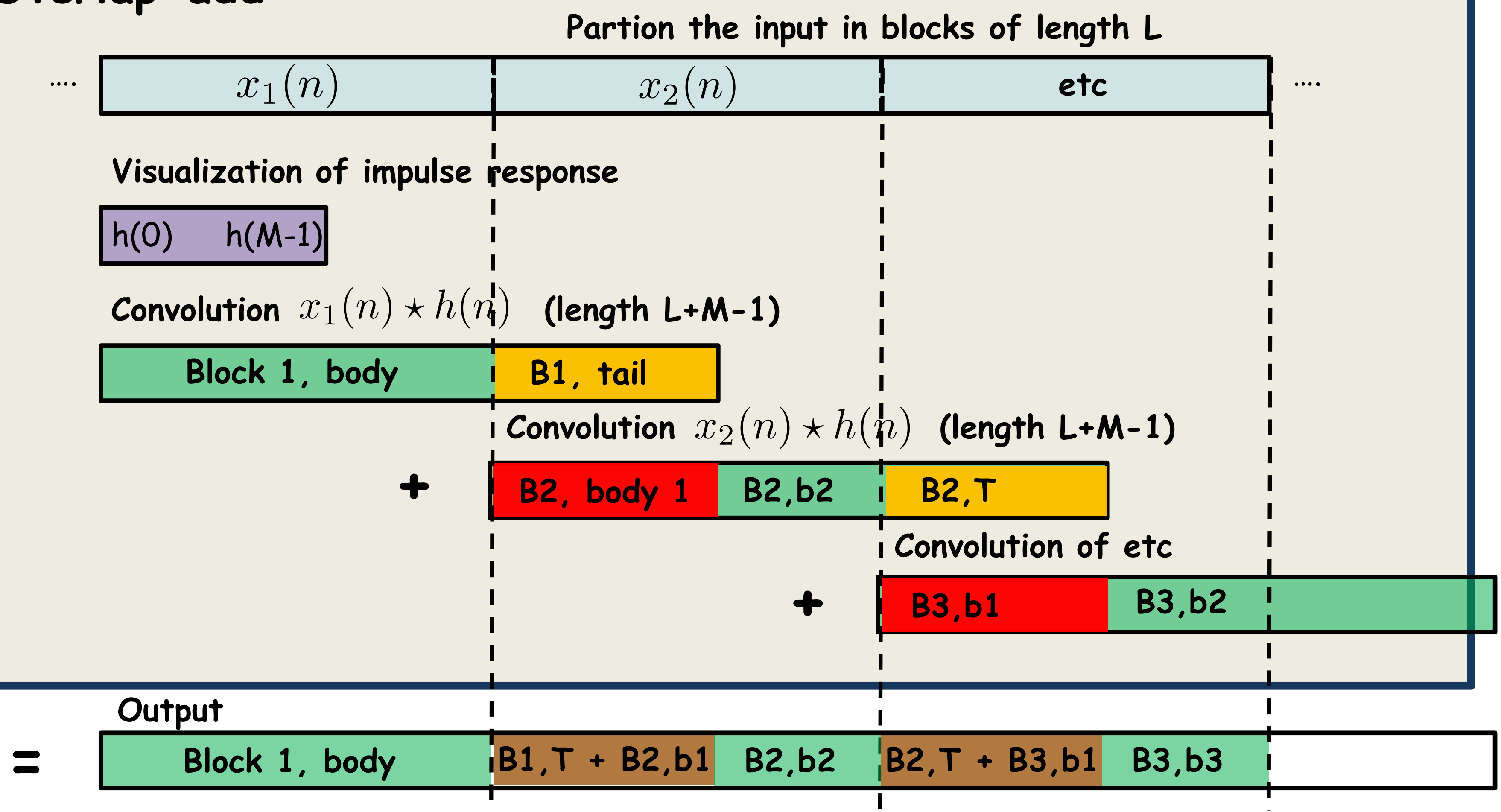


EITF75 Systems and Signals

Convolutions

$x(n) \rightarrow h(n) \rightarrow y(n) = x(n) \star h(n)$

Overlap-add

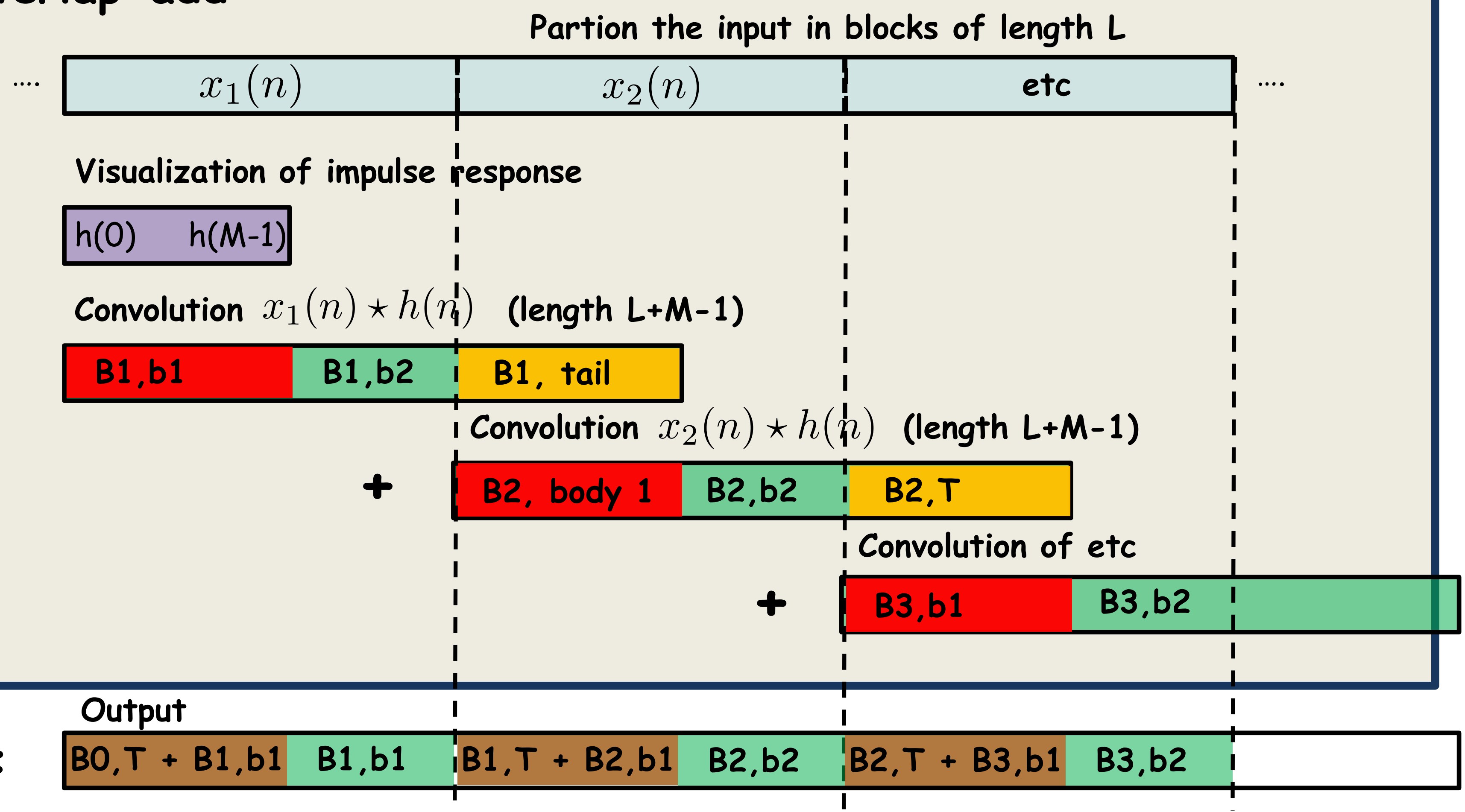


EITF75 Systems and Signals

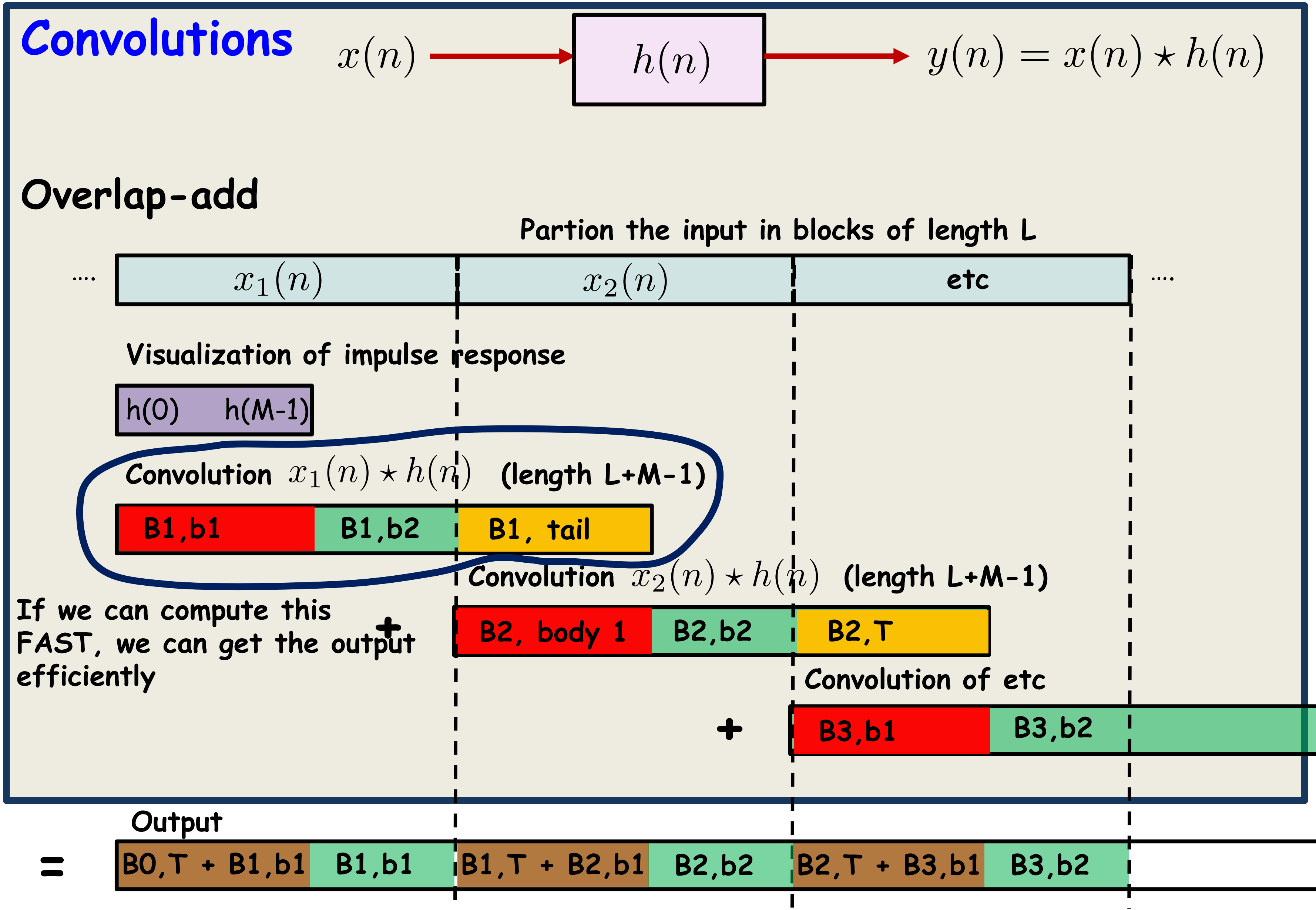
Convolutions

$$x(n) \longrightarrow \boxed{h(n)} \longrightarrow y(n) = x(n) \star h(n)$$

Overlap-add



EITF75 Systems and Signals



EITF75 Systems and Signals

Convolutions

$$x(n) \rightarrow \boxed{h(n)} \rightarrow y(n) = x(n) \star h(n)$$

Overlap-add

Partion the input in blocks of length L

.... | $x_1(n)$ | $x_2(n)$ | etc | ....

Visualization of impulse response

h(0) h(M-1)

Convolution $x_1(n) \star h(n)$ (length L+M-1)

Zero-pad Block 1 to length N=L+M-1

$x_1(n)$ 0 0 0 0 ... 0

EITF75 Systems and Signals

Convolutions

$$x(n) \longrightarrow \boxed{h(n)} \longrightarrow y(n) = x(n) \star h(n)$$

Overlap-add

Partion the input in blocks of length L

....	$x_1(n)$	$x_2(n)$	etc	....

Visualization of impulse response

h(0)	h(M-1)

Convolution $x_1(n) \star h(n)$ (length L+M-1)

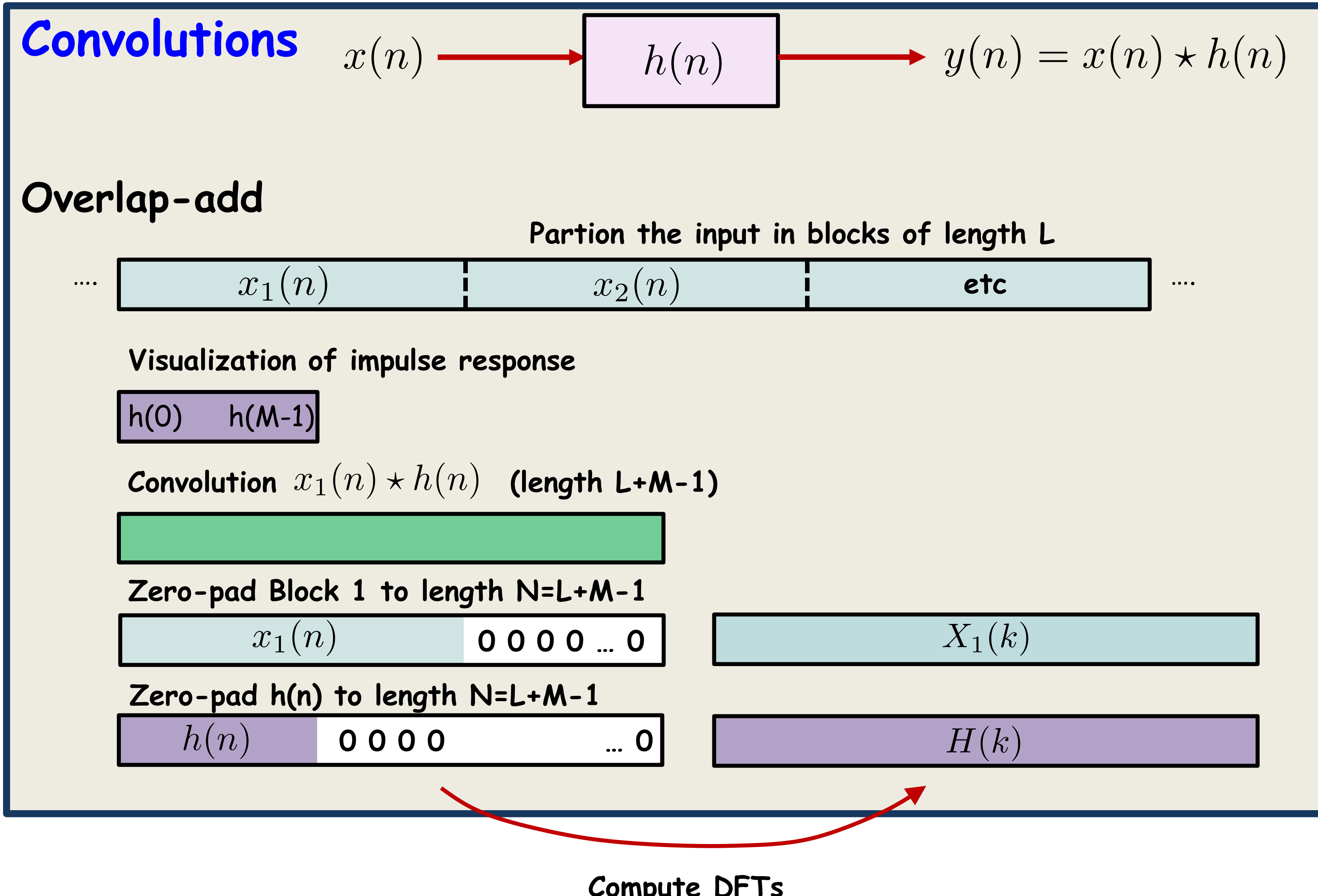
Zero-pad Block 1 to length N=L+M-1

$x_1(n)$	0 0 0 0 … 0

Zero-pad h(n) to length N=L+M-1

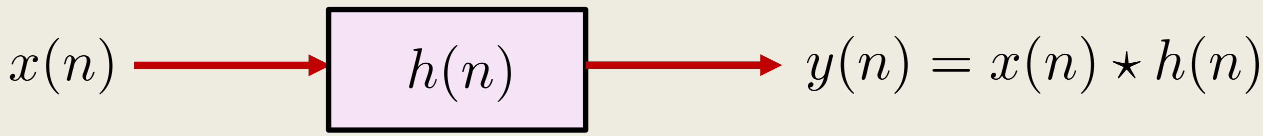
$h(n)$	0 0 0 0 … 0

EITF75 Systems and Signals

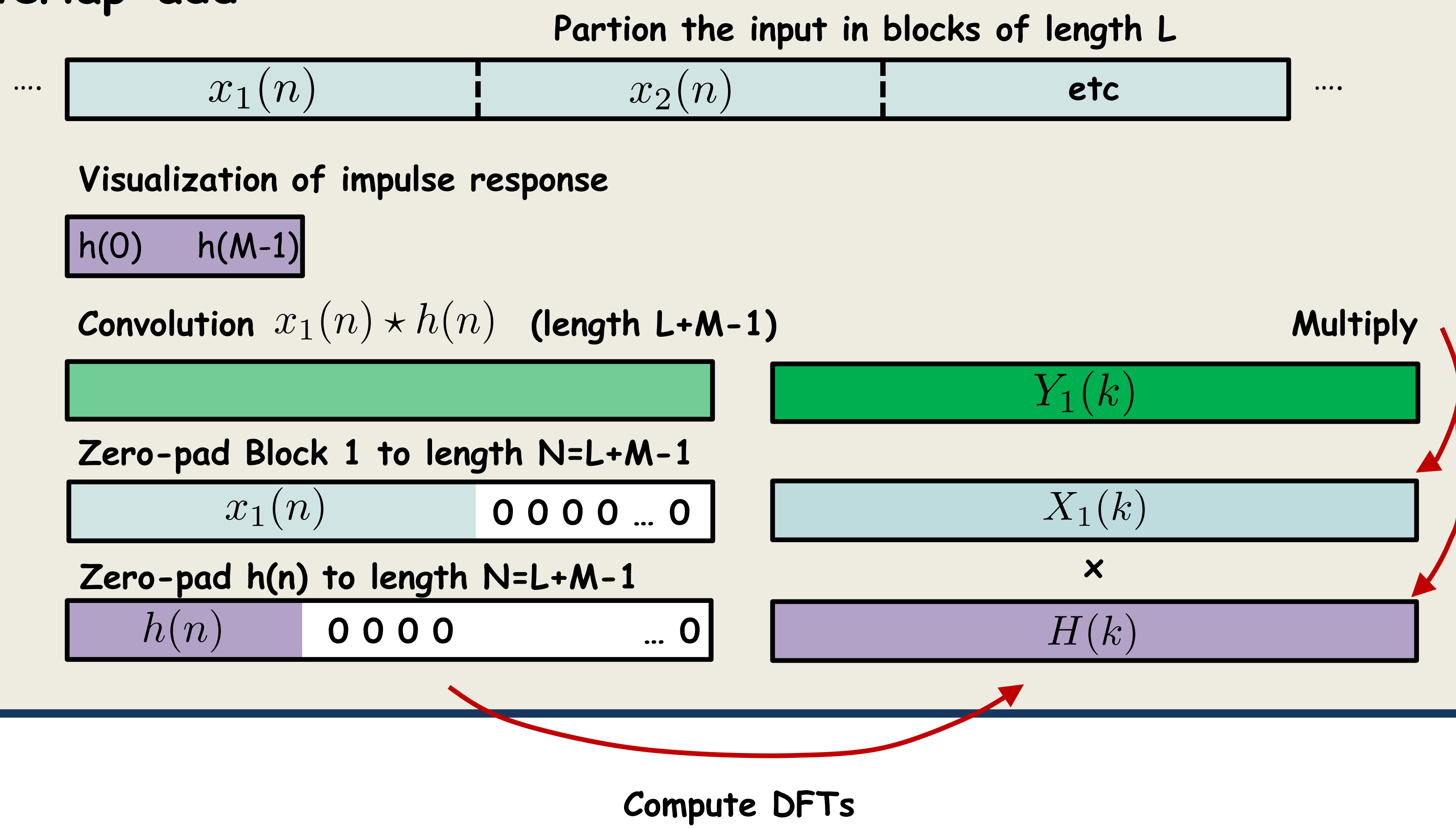


EITF75 Systems and Signals

Convolutions

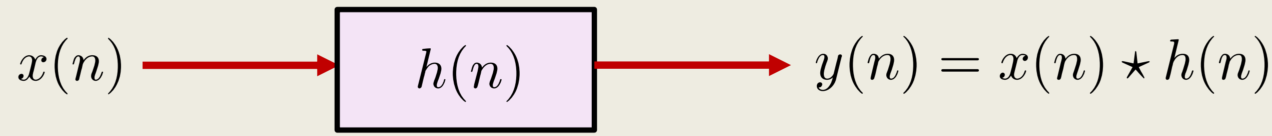


Overlap-add

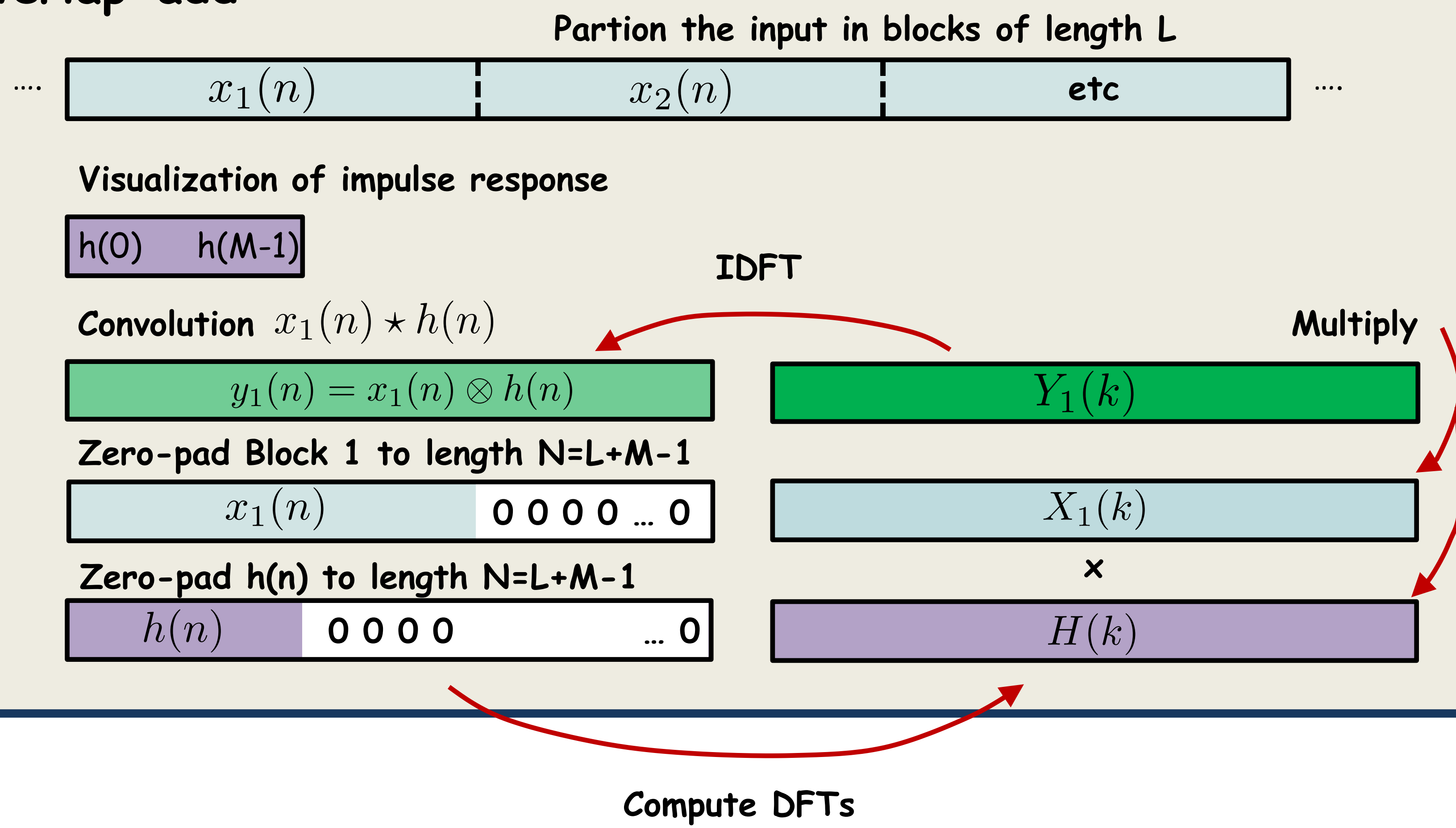


EITF75 Systems and Signals

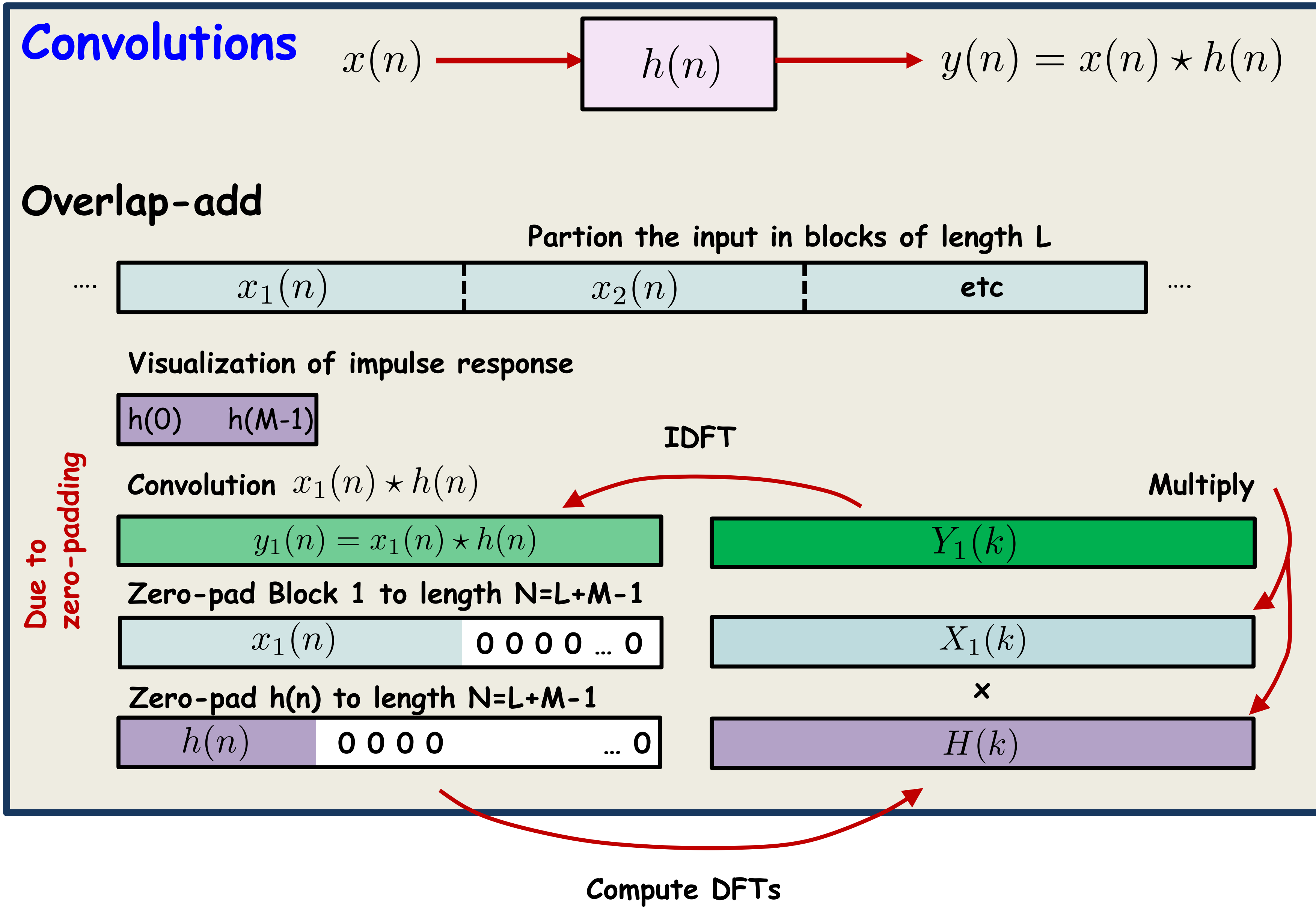
Convolutions



Overlap-add



EITF75 Systems and Signals

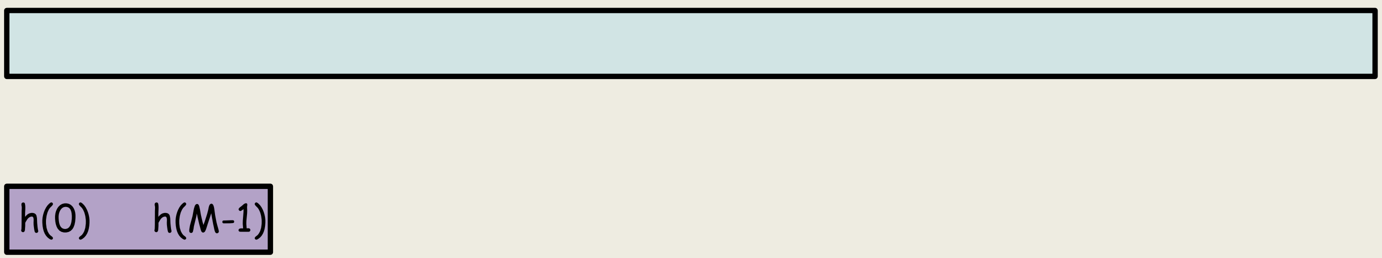


EITF75 Systems and Signals

Convolutions

$$x(n) \longrightarrow \boxed{h(n)} \longrightarrow y(n) = x(n) \star h(n)$$

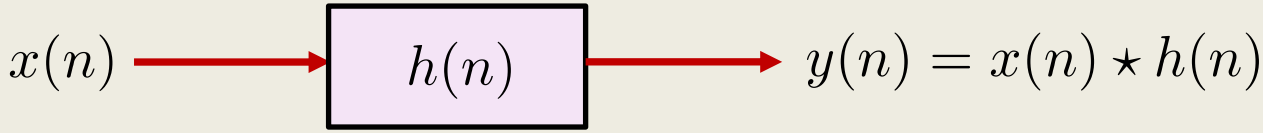
Overlap-add. **SUMMARY**



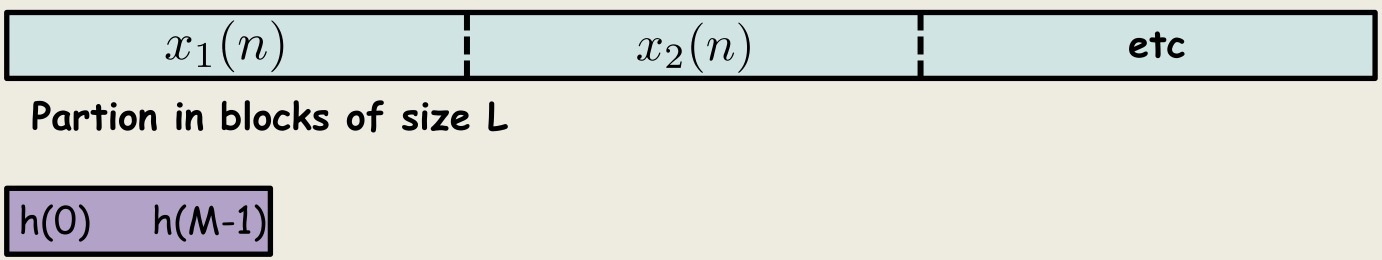
Compute DFTs

EITF75 Systems and Signals

Convolutions



Overlap-add. SUMMARY STEP 1



EITF75 Systems and Signals

Convolutions

$x(n) \rightarrow \boxed{h(n)} \rightarrow y(n) = x(n) \star h(n)$

Overlap-add. **SUMMARY** **STEP 2**

$x_1(n)$	$x_2(n)$	etc

Partion in blocks of size L

h(0) h(M-1)

Zero-pad to length N=L+M-1

$h(n)$	0 0 0 0 … 0

$H(k)$

Compute DFT (pre-processing)

EITF75 Systems and Signals

Convolutions

$$x(n) \longrightarrow \boxed{h(n)} \longrightarrow y(n) = x(n) \star h(n)$$

Overlap-add. SUMMARY STEP 2

$x_1(n)$	$x_2(n)$	etc

Partion in blocks of size L

h(n) no longer needed

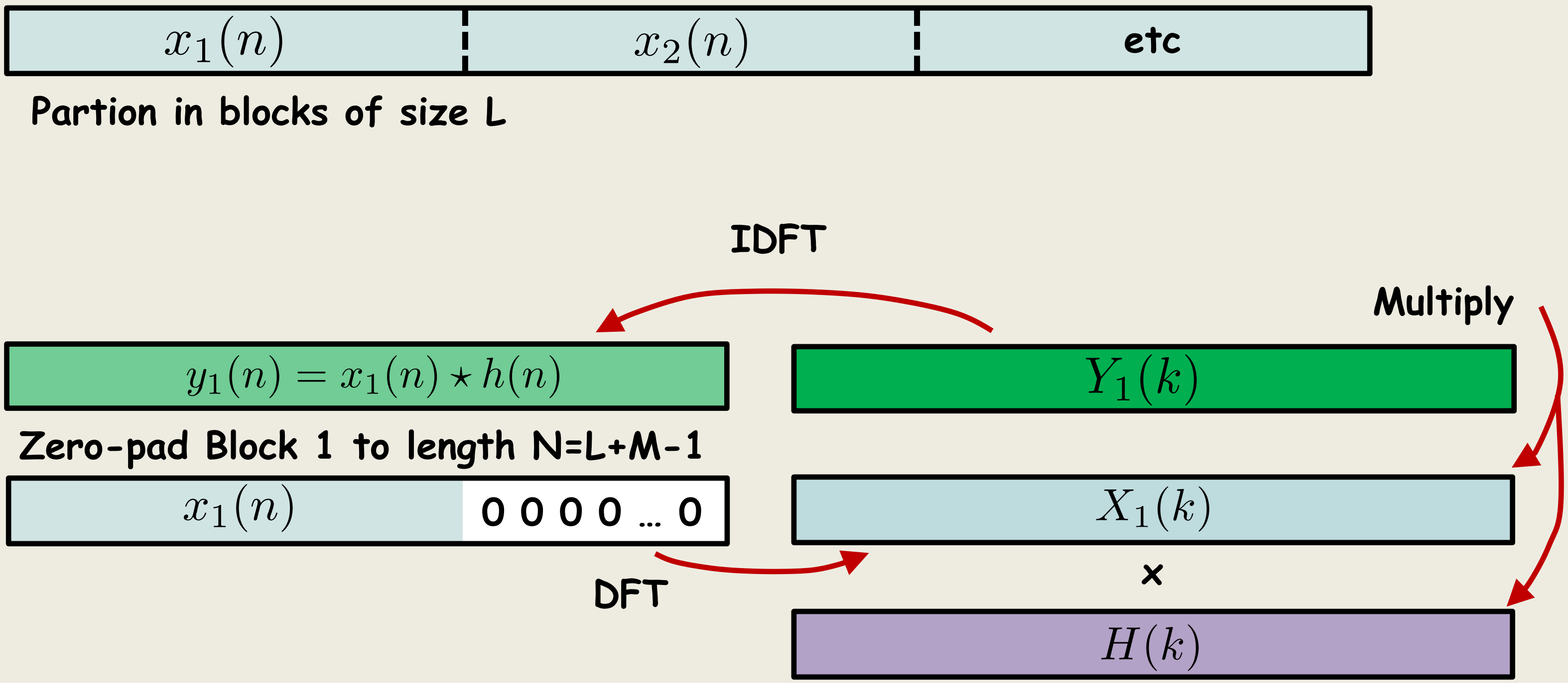
$H(k)$

EITF75 Systems and Signals

Convolutions

$$x(n) \rightarrow \boxed{h(n)} \rightarrow y(n) = x(n) \star h(n)$$

Overlap-add. SUMMARY STEP 3 For block 1



EITF75 Systems and Signals

Convolutions

$x(n)$ → $h(n)$ → $y(n) = x(n) \star h(n)$

Overlap-add. **SUMMARY** **STEP 4** For block 1

$x_2(n)$	etc

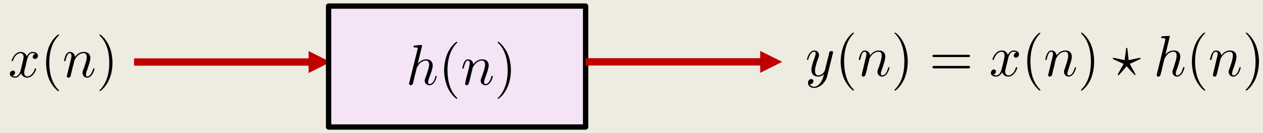
Partion in blocks of size L

$y_1(n) = x_1(n) \star h(n)$

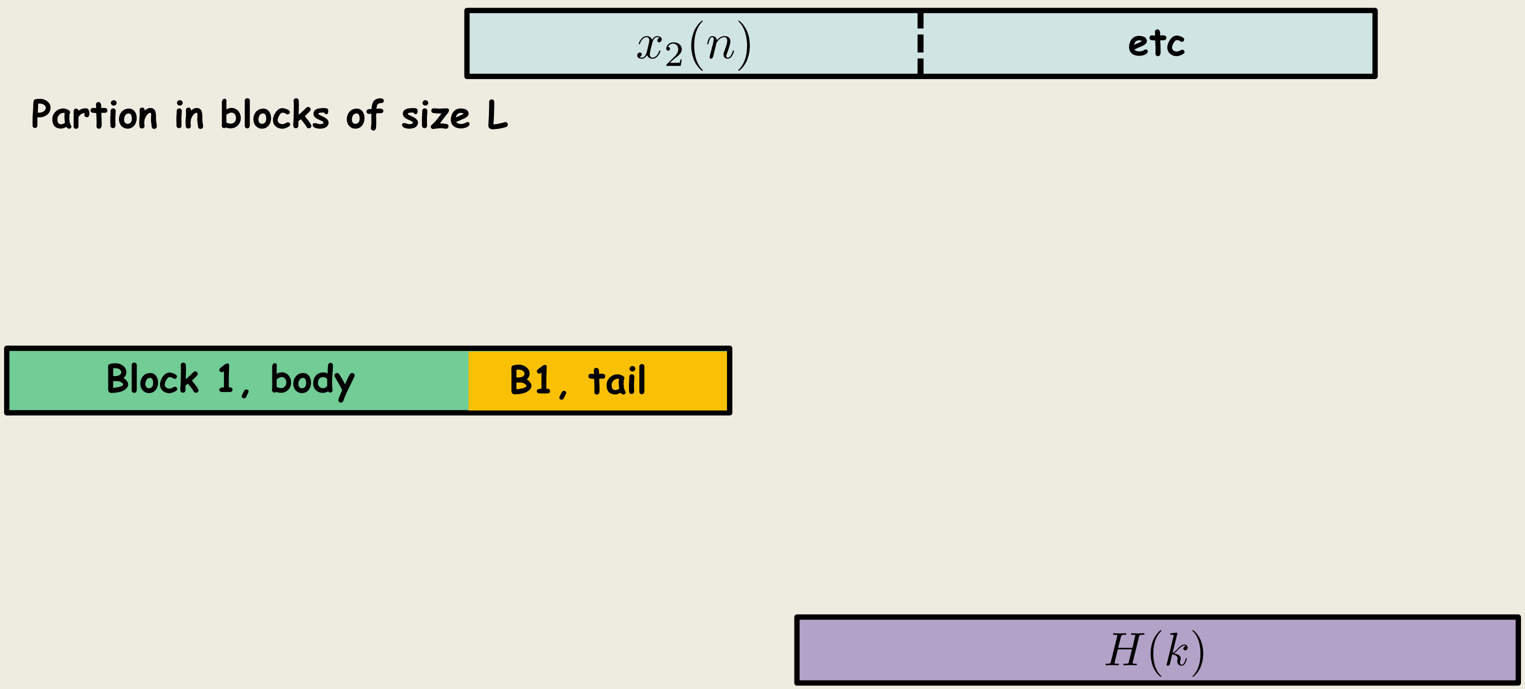
$H(k)$

EITF75 Systems and Signals

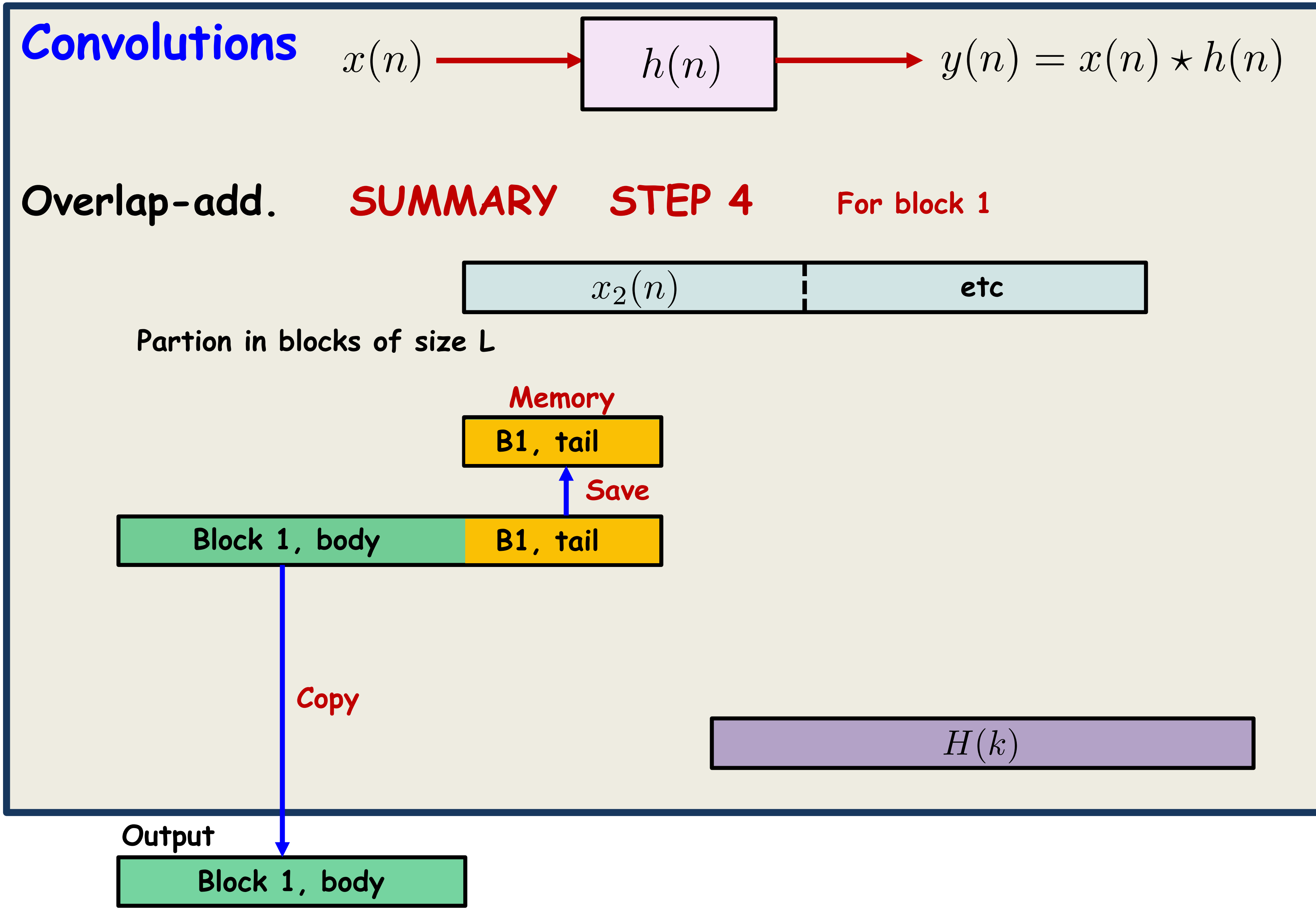
Convolutions



Overlap-add. SUMMARY STEP 4 For block 1



EITF75 Systems and Signals



EITF75 Systems and Signals

Convolutions

$$x(n) \longrightarrow \boxed{h(n)} \longrightarrow y(n) = x(n) \star h(n)$$

Overlap-add. SUMMARY STEP 4 For block 1

$x_2(n)$	etc

Partion in blocks of size L

Memory

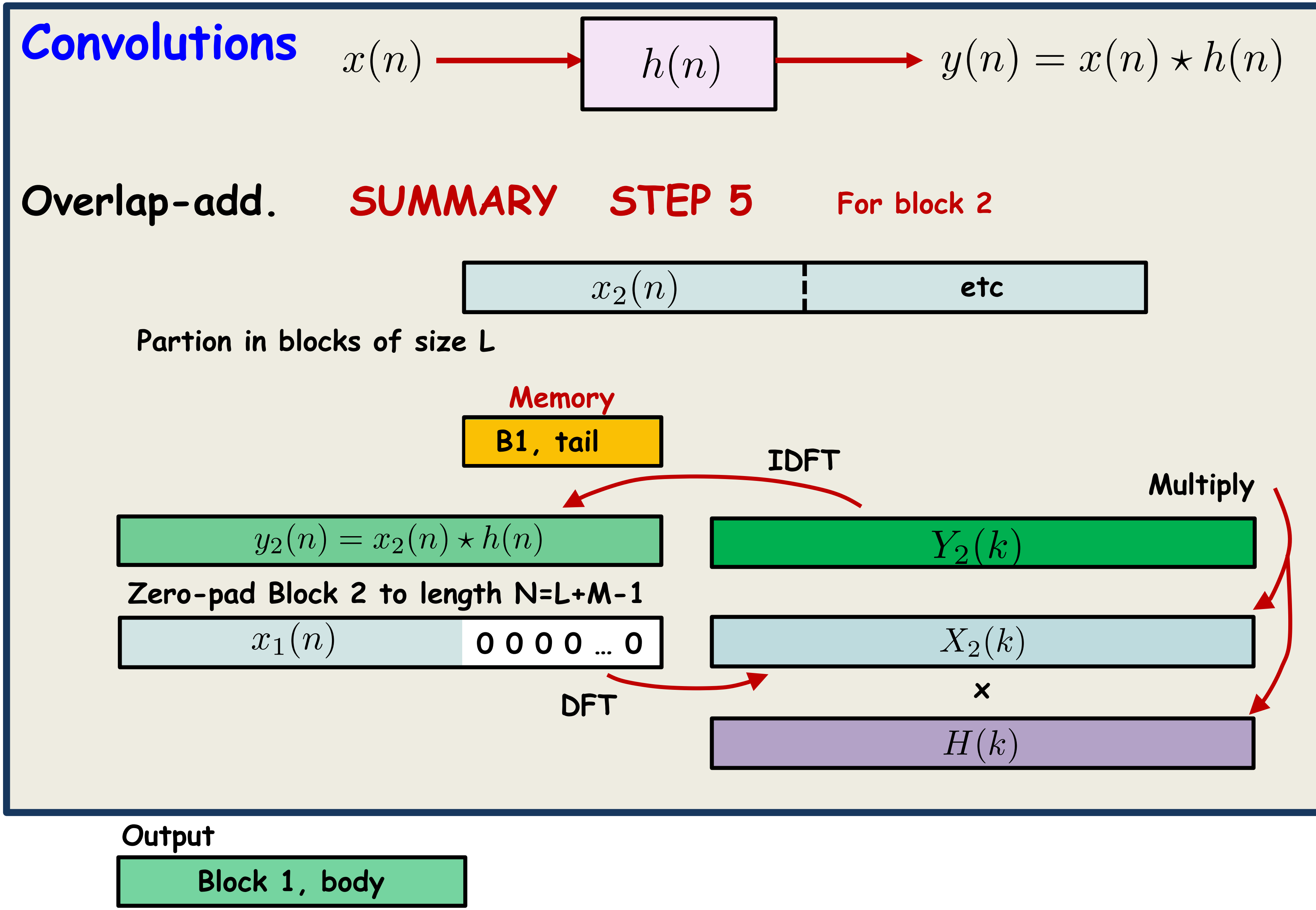
B1, tail

$H(k)$

Output

Block 1, body

EITF75 Systems and Signals



EITF75 Systems and Signals

Convolutions

$x(n) \rightarrow \boxed{h(n)} \rightarrow y(n) = x(n) \star h(n)$

Overlap-add. SUMMARY STEP 5 For block 2

etc

Partion in blocks of size L

Memory

B1, tail

$y_2(n) = x_2(n) \star h(n)$

$H(k)$

Output

Block 1, body

EITF75 Systems and Signals

Convolutions

$$x(n) \longrightarrow \boxed{h(n)} \longrightarrow y(n) = x(n) \star h(n)$$

Overlap-add. SUMMARY STEP 5 For block 2

etc

Partion in blocks of size L

Memory

B1, tail

$y_2(n) = x_2(n) \star h(n)$

$H(k)$

Output

Block 1, body

EITF75 Systems and Signals

Convolutions

$$x(n) \longrightarrow \boxed{h(n)} \longrightarrow y(n) = x(n) \star h(n)$$

Overlap-add. **SUMMARY** **STEP 6** For block 2

etc

Partion in blocks of size L

Memory

B1, tail

B2, body 1	B2,b2	B2,T

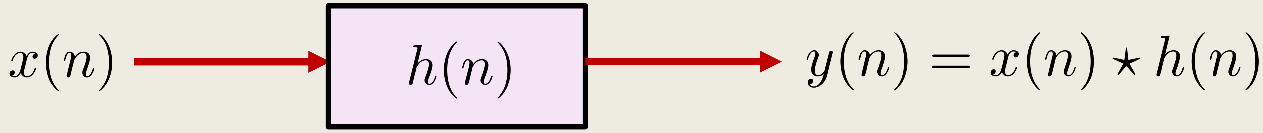
$H(k)$

Output

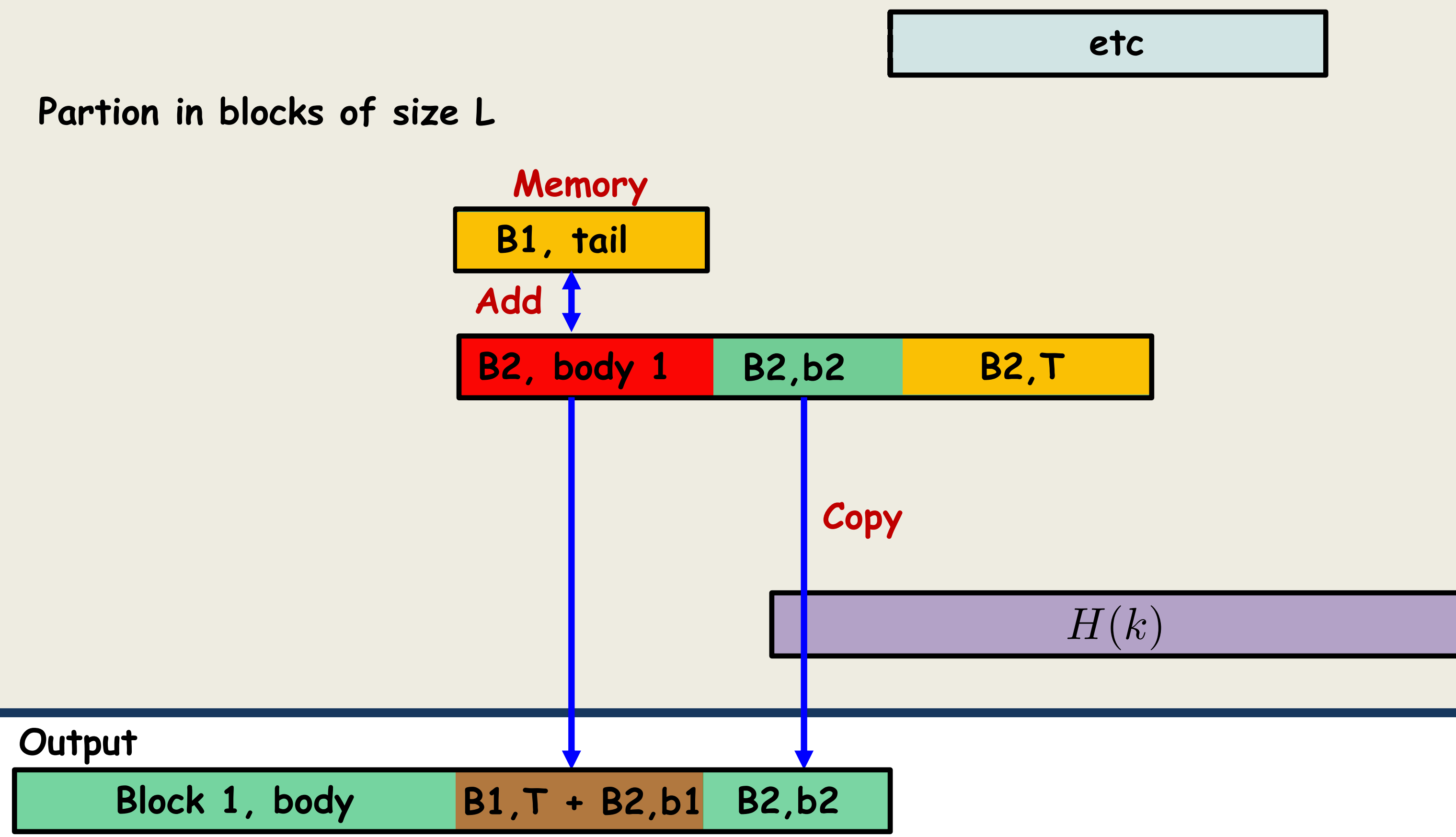
Block 1, body

EITF75 Systems and Signals

Convolutions



Overlap-add. SUMMARY STEP 6 For block 2



EITF75 Systems and Signals

Convolutions

$x(n) \rightarrow \boxed{h(n)} \rightarrow y(n) = x(n) \star h(n)$

Overlap-add. SUMMARY STEP 6 For block 2

etc

Partion in blocks of size L

Memory

B2,T

Save

B2,T

$H(k)$

Output

Block 1, body	B1,T + B2,b1	B2,b2

EITF75 Systems and Signals

Convolutions

$x(n) \rightarrow \boxed{h(n)} \rightarrow y(n) = x(n) \star h(n)$

Overlap-add. SUMMARY STEP 6 For block 2

etc

Partion in blocks of size L

Memory

B2,T

$H(k)$

Output

Block 1, body	B1,T + B2,b1	B2,b2

EITF75 Systems and Signals

Convolutions

$$x(n) \longrightarrow \boxed{h(n)} \longrightarrow y(n) = x(n) \star h(n)$$

Overlap-add. **SUMMARY** **Repeat 5&6** For blocks >2

etc

Partion in blocks of size L

Memory

B2,T

$H(k)$

Output

Block 1, body	B1,T + B2,b1	B2,b2

EITF75 Systems and Signals

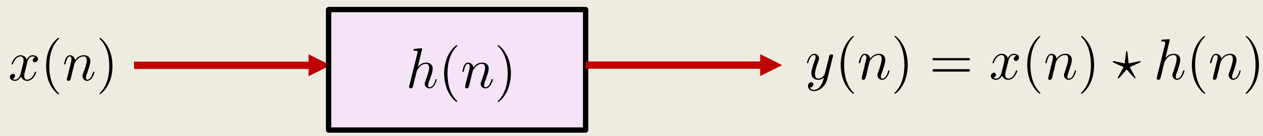
Convolutions

$$x(n) \rightarrow \boxed{h(n)} \rightarrow y(n) = x(n) \star h(n)$$

Overlap-add. **Computational Complexity**

EITF75 Systems and Signals

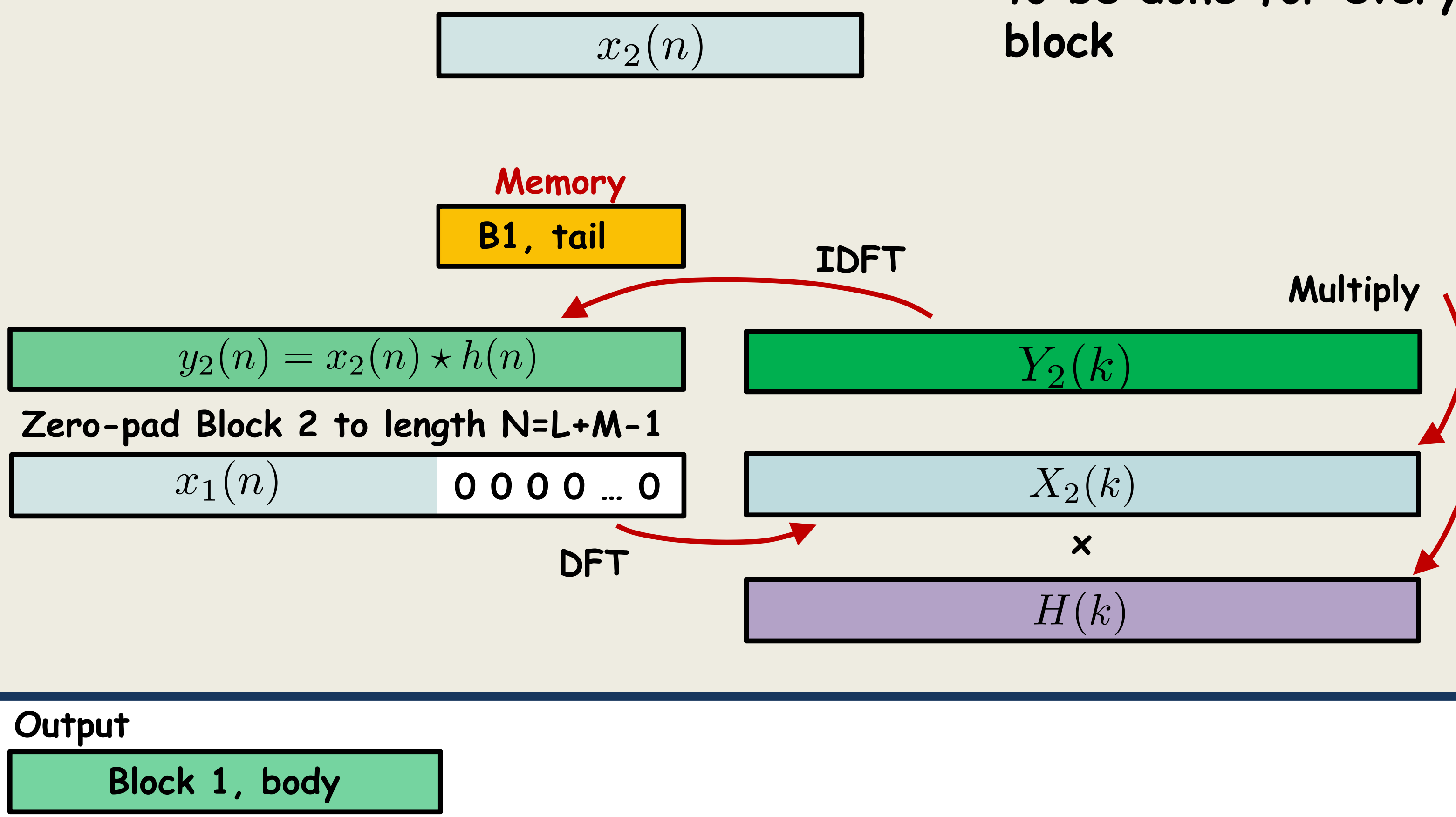
Convolutions



Overlap-add.

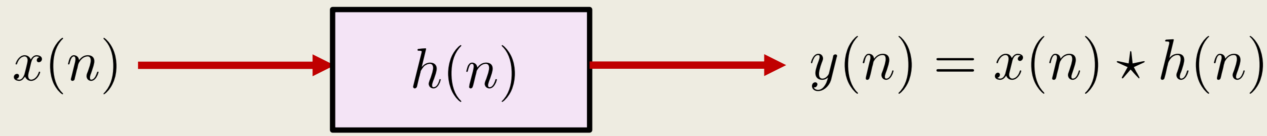
Computational Complexity

What is below needs to be done for every block



EITF75 Systems and Signals

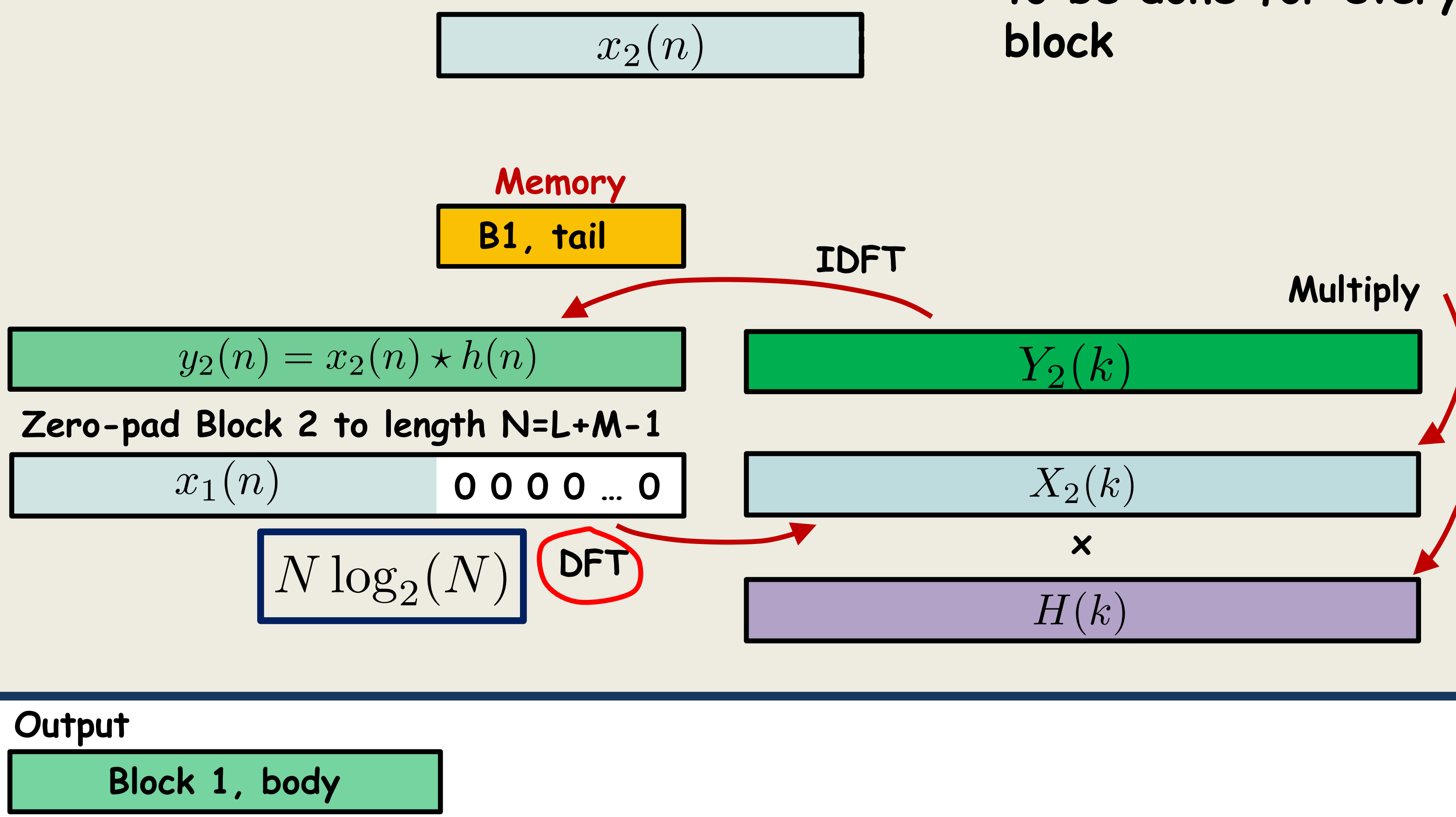
Convolutions



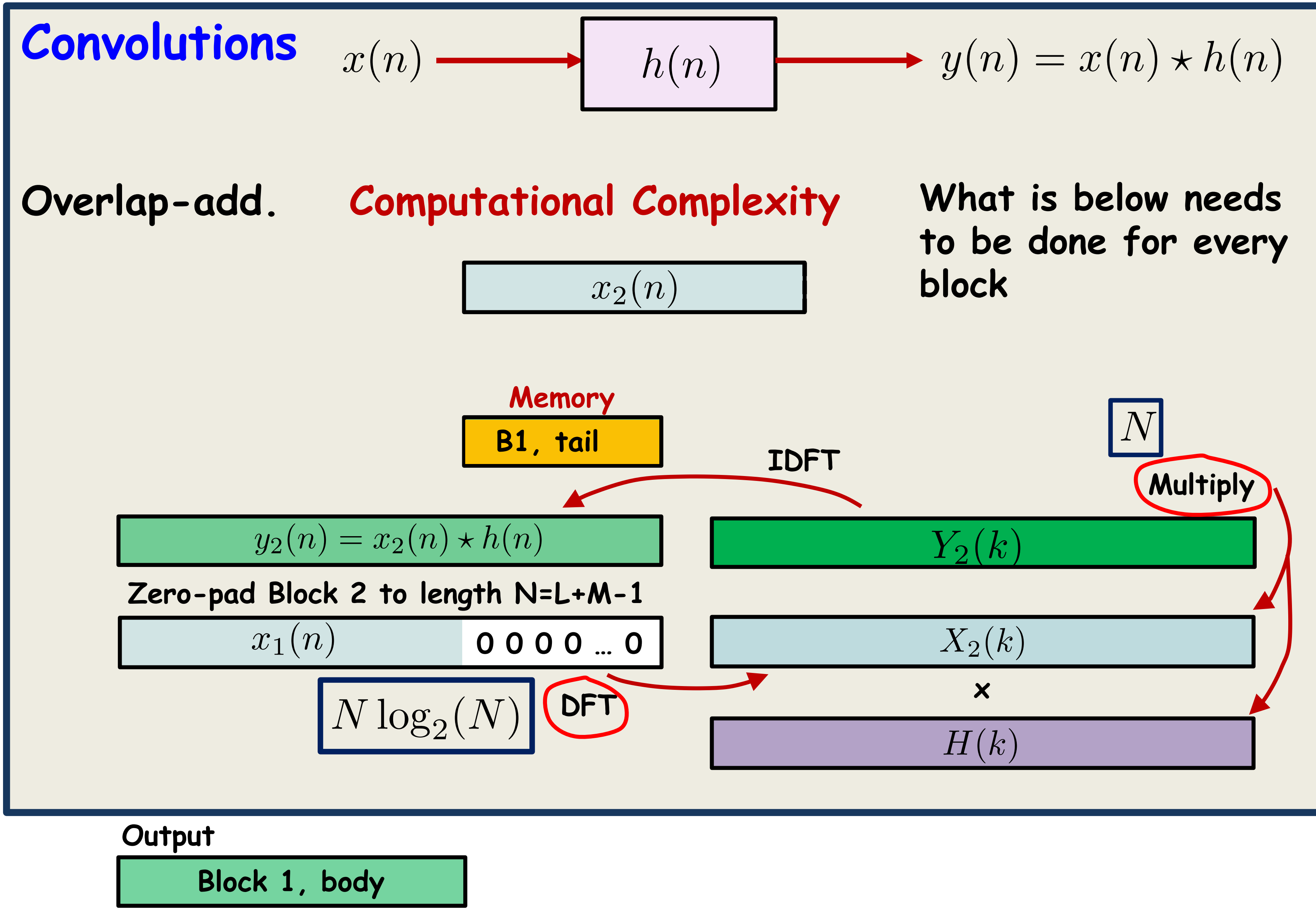
Overlap-add.

Computational Complexity

What is below needs to be done for every block



EITF75 Systems and Signals



EITF75 Systems and Signals

Convolutions

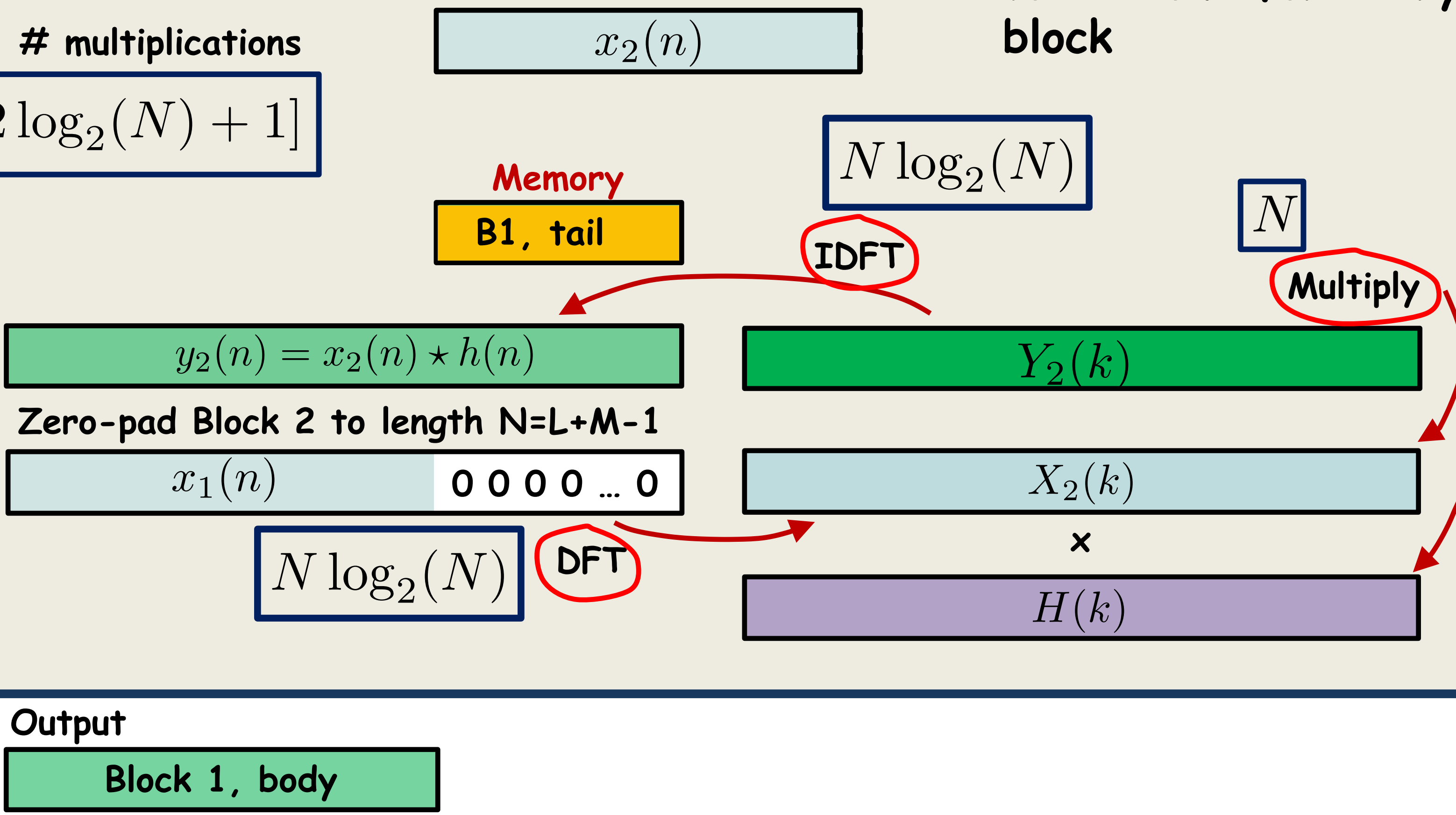
$$x(n) \rightarrow \boxed{h(n)} \rightarrow y(n) = x(n) \star h(n)$$

Overlap-add. **Computational Complexity**

What is below needs to be done for every block

Total # multiplications

$$\boxed{N\left[2\log_2(N) + 1\right]}$$



EITF75 Systems and Signals

Convolutions

$$x(n) \longrightarrow \boxed{h(n)} \longrightarrow y(n) = x(n) \star h(n)$$

Overlap-add. **Computational Complexity**

Total # multiplications

$$\boxed{N\left[2\log_2(N)+1\right]}$$

For these multiplications, we get L = N-M outputs

#multiplications/output

$$\boxed{\frac{N}{N-M}\left[2\log_2(N)+1\right]}$$

$$\boxed{\frac{N}{N-M} \approx 1,\ L \gg M}$$

EITF75 Systems and Signals

Convolutions

$$x(n) \longrightarrow \boxed{h(n)} \longrightarrow y(n) = x(n) \star h(n)$$

Overlap-add. **Computational Complexity**

Total # multiplications

$$\boxed{N\left[2\log_2(N)+1\right]}$$

For these multiplications, we get L = N-M outputs

#multiplications/output

$$\boxed{\frac{N}{N-M}\left[2\log_2(N)+1\right]}$$

Easy to solve numerically (no closed form)

$$\boxed{\frac{\partial}{\partial N}\frac{N}{N-M}\left[2\log_2(N)+1\right]=0}$$

$$\boxed{\frac{N}{N-M}\approx 1,\ L \gg M}$$

EITF75 Systems and Signals

Convolutions

$$x(n) \longrightarrow \boxed{h(n)} \longrightarrow y(n) = x(n) \star h(n)$$

Overlap-add. **Computational Complexity**

Total # multiplications

$$\boxed{N\,[2\log_2(N) + 1]}$$

For these multiplications, we get L = N-M outputs

#multiplications/output

$$\boxed{\frac{N}{N-M}\,[2\log_2(N) + 1]}$$

Easy to solve numerically (no closed form)

$$\boxed{\frac{\partial}{\partial N}\frac{N}{N-M}\,[2\log_2(N) + 1] = 0}$$

$$\boxed{\frac{N}{N-M} \approx 1,\ L \gg M}$$

Convolutions

In real life, the input signal is very long, does not start/stop. Previous method fails

$$x(n) \longrightarrow \boxed{h(n)} \longrightarrow y(n) = x(n) \star h(n)$$

Three common methods

Overlap-add to be described

Overlap-save to be briefly described

Overlap-discard not to be described

EITF75 Systems and Signals

Convolutions

$$x(n) \longrightarrow \boxed{h(n)} \longrightarrow y(n) = x(n) \star h(n)$$

Overlap-save

Partion the input in blocks of length L

....	$x_1(n)$	$x_2(n)$	etc	....

Visualization of impulse response

h(0)	h(M-1)

Zero-pad Block 1 to length N=L+M-1

0 0 0 0 … 0	$x_1(n)$

EITF75 Systems and Signals

Convolutions

$x(n) \rightarrow \boxed{h(n)} \rightarrow y(n) = x(n) \star h(n)$

Overlap-save

Partion the input in blocks of length L

.... $x_1(n)$ | $x_2(n)$ | etc ....

Visualization of impulse response

h(0) h(M-1)

Zero-pad Block 1 to length N=L+M-1

0 0 0 0 … 0 $x_1(n)$

Circular Convolution of zero-padded signal and h(n)

EITF75 Systems and Signals

Convolutions

$x(n) \rightarrow \boxed{h(n)} \rightarrow y(n) = x(n) \star h(n)$

Overlap-save

Partion the input in blocks of length L

....	$x_1(n)$	$x_2(n)$	etc	....

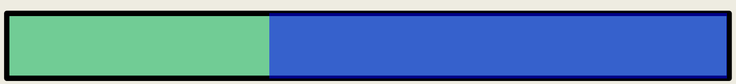
Visualization of impulse response

h(0) h(M-1)

Zero-pad Block 1 to length N=L+M-1

0 0 0 0 … 0	$x_1(n)$

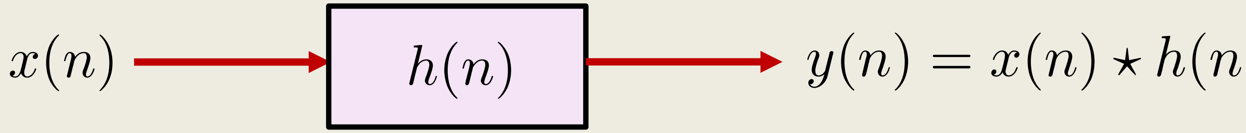
Circular Convolution of zero-padded signal and h(n)



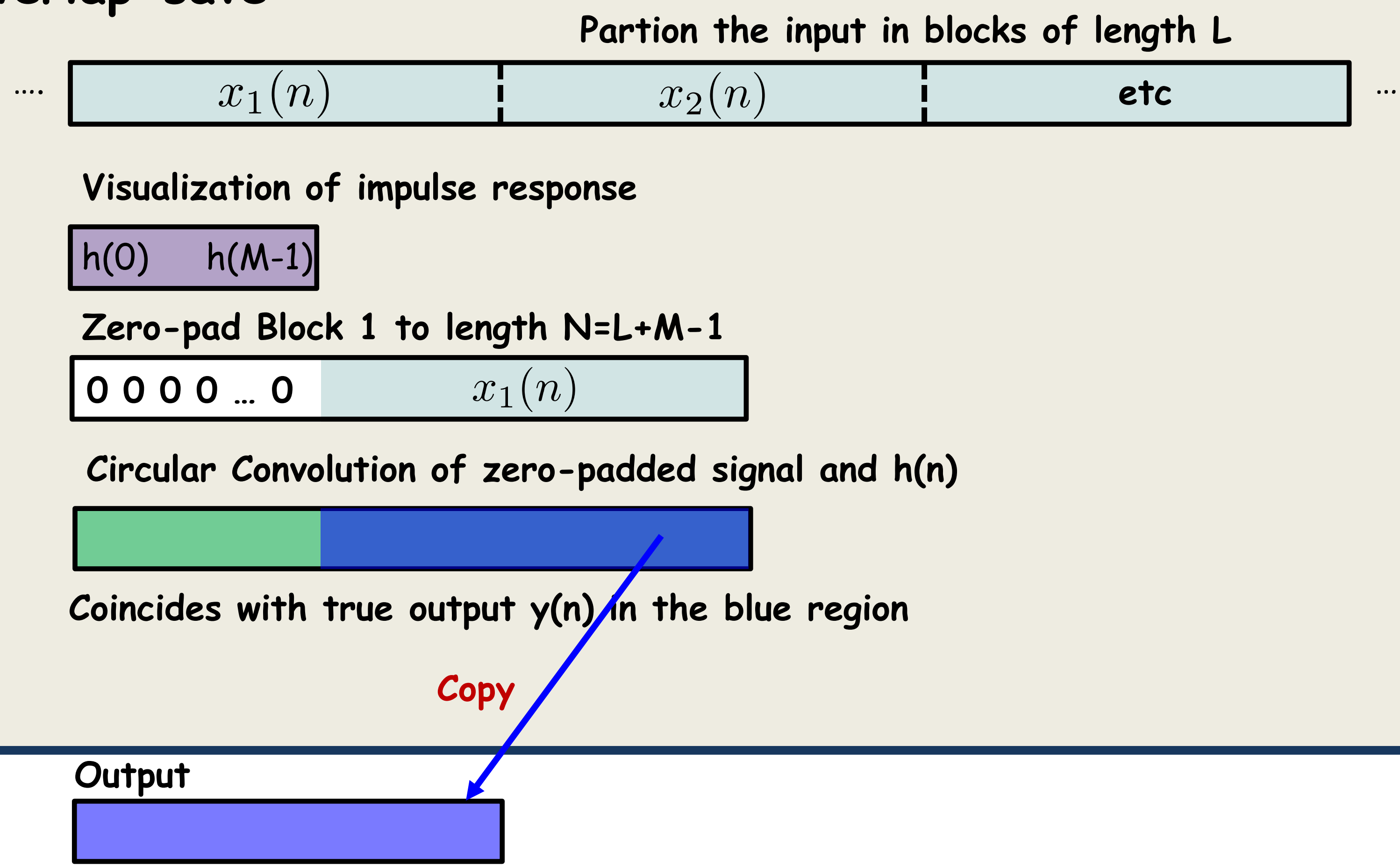
Coincides with true output y(n) in the blue region

EITF75 Systems and Signals

Convolutions



Overlap-save



EITF75 Systems and Signals

Convolutions

$$x(n) \longrightarrow \boxed{h(n)} \longrightarrow y(n) = x(n) \star h(n)$$

Overlap-save

Partion the input in blocks of length L

.... | $x_1(n)$ | $x_2(n)$ | etc | ....

Visualization of impulse response

h(0) h(M-1)

Block 2: Borrow a bit from block 1

$x_2(n)$

Output

EITF75 Systems and Signals

Convolutions

$$x(n) \longrightarrow \boxed{h(n)} \longrightarrow y(n) = x(n) \star h(n)$$

Overlap-save

Partion the input in blocks of length L

.... $x_1(n)$ | $x_2(n)$ | etc ....

Visualization of impulse response

h(0) h(M-1)

Block 2: Borrow a bit from block 1

$x_2(n)$

Output

EITF75 Systems and Signals

Convolutions

$$x(n) \longrightarrow \boxed{h(n)} \longrightarrow y(n) = x(n) \star h(n)$$

Overlap-save

Partion the input in blocks of length L

.... | $x_1(n)$ | $x_2(n)$ | etc | ....

Visualization of impulse response

h(0) h(M-1)

Block 2: Borrow a bit from block 1

$x_2(n)$

Circular Convolution of above signal and h(n)

Output

EITF75 Systems and Signals

Convolutions

$$x(n) \rightarrow \boxed{h(n)} \rightarrow y(n) = x(n) \star h(n)$$

Overlap-save

Partion the input in blocks of length L

.... | $x_1(n)$ | $x_2(n)$ | etc | ....

Visualization of impulse response

h(0) h(M-1)

Block 2: Borrow a bit from block 1

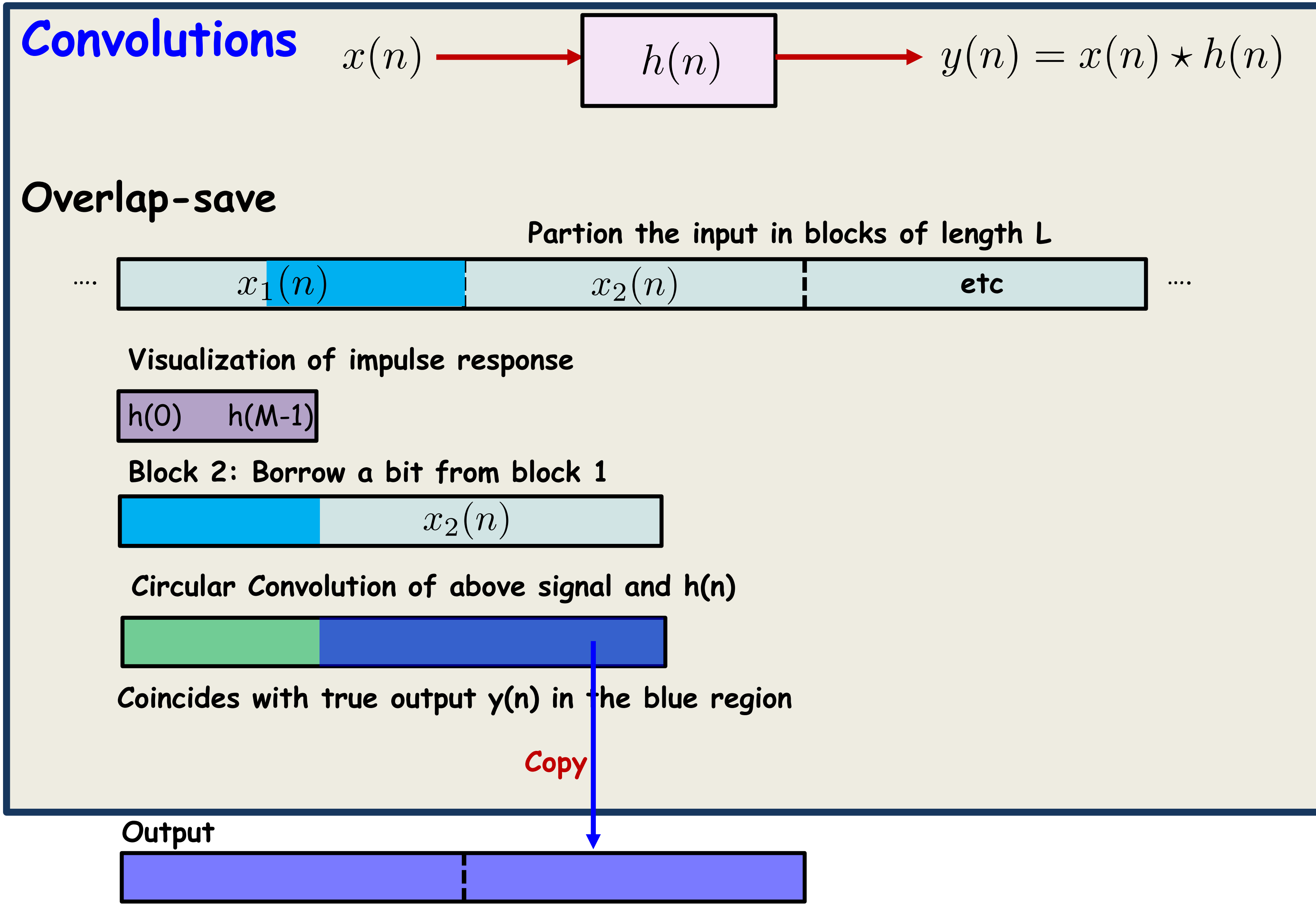
$x_2(n)$

Circular Convolution of above signal and h(n)

Coincides with true output y(n) in the blue region

Output

EITF75 Systems and Signals



EITF75 Systems and Signals

Convolutions

$$x(n) \longrightarrow \boxed{h(n)} \longrightarrow y(n) = x(n) \star h(n)$$

Assume an IIR filter

Implications for overlap-add/save ?

Convolutions

$$x(n) \longrightarrow \boxed{h(n)} \longrightarrow y(n) = x(n) \star h(n)$$

Assume an IIR filter

Implications for overlap-add/save ?   Does not work, since L>M for overlap to work

EITF75 Systems and Signals

Convolutions

$$x(n) \longrightarrow \boxed{h(n)} \longrightarrow y(n) = x(n) \star h(n)$$

Assume an IIR filter

Implications for overlap-add/save ? Does not work, since L>M for overlap to work

However, an IIR filter is implemented via a difference equation

$$y(n) = \sum_{\ell=1}^{L} a_\ell y(n-\ell) + \sum_{k=0}^{K} b_k x(n-k)$$

L+K+1 multiplications to get 1 output

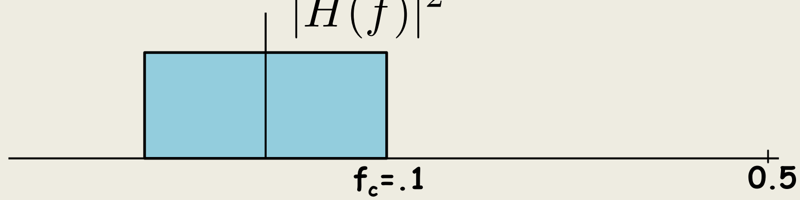
EITF75 Systems and Signals

Convolutions

$$x(n) \longrightarrow \boxed{h(n)} \longrightarrow y(n) = x(n) \star h(n)$$

Case study: Low pass filter implemented as $\begin{cases} \text{FIR} \\ \text{IIR} \end{cases}$

Target



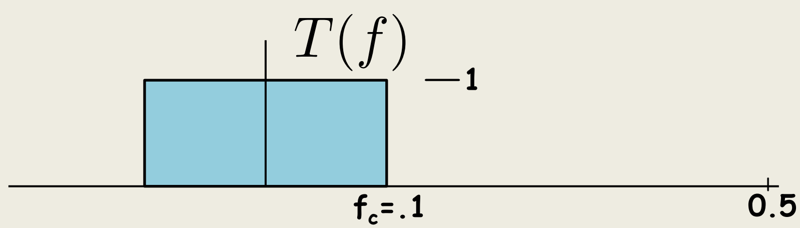
EITF75 Systems and Signals

Convolutions

$$x(n) \rightarrow \boxed{h(n)} \rightarrow y(n) = x(n) \star h(n)$$

Case study: Low pass filter implemented as $\left\{\begin{array}{l}\textbf{FIR}\\ \textbf{IIR}\end{array}\right.$

Target



<u>FIR:</u> $$H(f) = \sum_{n=0}^{N-1} b_n \mathrm{e}^{-i2\pi f n}$$

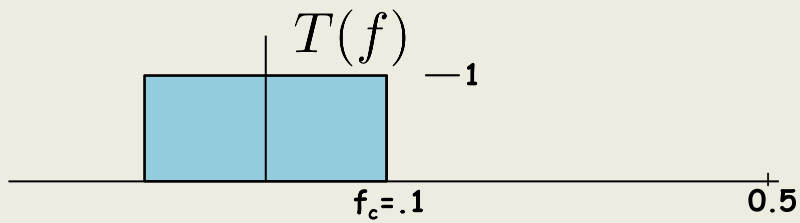
EITF75 Systems and Signals

Convolutions

$$x(n) \longrightarrow \boxed{h(n)} \longrightarrow y(n) = x(n) \star h(n)$$

Case study: Low pass filter implemented as $\begin{cases} \text{FIR} \\ \text{IIR} \end{cases}$

Target



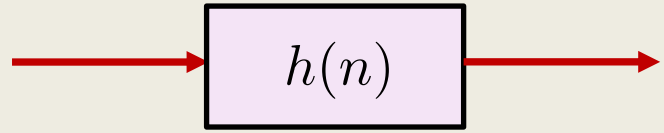
FIR: $$H(f) = \sum_{n=0}^{N-1} b_n \mathrm{e}^{-i2\pi f n}$$

Error $$E(f) = \sqrt{T(f)} - \sum_{n=0}^{N-1} b_n \mathrm{e}^{-i2\pi f n}$$

EITF75 Systems and Signals

Convolutions

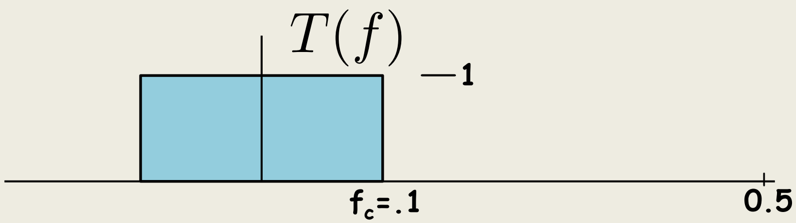
$x(n)$



$y(n) = x(n) \star h(n)$

Case study: Low pass filter implemented as $\begin{cases} \text{FIR} \\ \text{IIR} \end{cases}$

Target



FIR: $H(f) = \sum_{n=0}^{N-1} b_n \mathrm{e}^{-i2\pi f n}$ **Error** $E(f) = \sqrt{T(f)} - \sum_{n=0}^{N-1} b_n \mathrm{e}^{-i2\pi f n}$

To be minimized: $\delta = \int_{-0.5}^{0.5} |E(f)|^2 \mathrm{d}f$

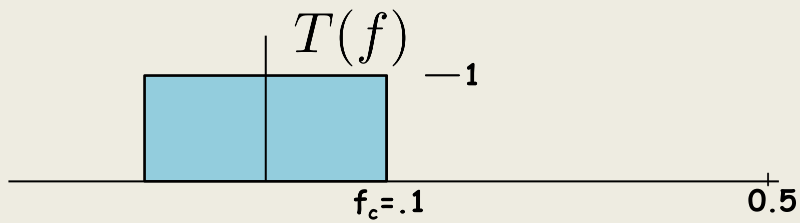
EITF75 Systems and Signals

Convolutions

$$x(n) \longrightarrow \boxed{h(n)} \longrightarrow y(n) = x(n) \star h(n)$$

Case study: Low pass filter implemented as $\begin{cases} \textbf{FIR} \\ \textbf{IIR} \end{cases}$

Target



<u>FIR:</u> $$H(f) = \sum_{n=0}^{N-1} b_n \mathrm{e}^{-i2\pi f n}$$

Error $$E(f) = \sqrt{T(f)} - \sum_{n=0}^{N-1} b_n \mathrm{e}^{-i2\pi f n}$$

To be minimized: $$\delta = \int_{-0.5}^{0.5} |E(f)|^2 \mathrm{d}f = \int_{-0.5}^{0.5} |H(f)|^2 \mathrm{d}f + \int_{-0.5}^{0.5} T(f)\mathrm{d}f - 2\mathrm{R}\left\{\int_{-0.5}^{0.5} \sum_{n=0}^{N-1} b_n T(f) \mathrm{e}^{-i2\pi f n} \mathrm{d}f\right\}$$

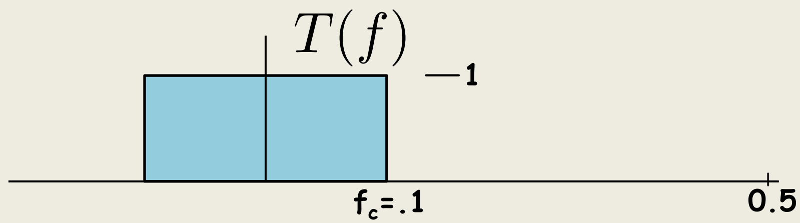
EITF75 Systems and Signals

Convolutions

$$x(n) \longrightarrow \boxed{h(n)} \longrightarrow y(n) = x(n) \star h(n)$$

Case study: Low pass filter implemented as $\begin{cases} \textbf{FIR} \\ \textbf{IIR} \end{cases}$

Target



FIR: $$H(f) = \sum_{n=0}^{N-1} b_n \mathrm{e}^{-i2\pi f n}$$

Error $$E(f) = \sqrt{T(f)} - \sum_{n=0}^{N-1} b_n \mathrm{e}^{-i2\pi f n}$$

Parseval

To be minimized: $$\delta = \int_{-0.5}^{0.5} |E(f)|^2 \mathrm{d}f = \sum_{n=0}^{N-1} |b_n|^2 + \int_{-0.5}^{0.5} T(f)\mathrm{d}f - 2\mathrm{R}\left\{ \int_{-0.5}^{0.5} \sum_{n=0}^{N-1} b_n T(f) \mathrm{e}^{-i2\pi f n} \mathrm{d}f \right\}$$

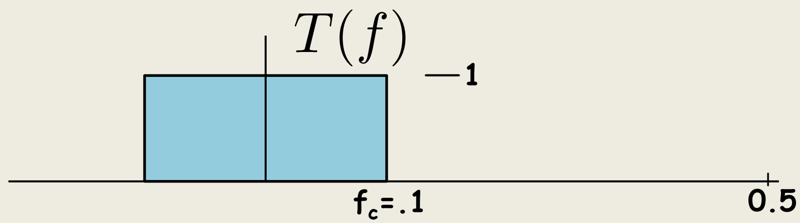
EIT75 Systems and Signals

Convolutions

$$x(n) \longrightarrow \boxed{h(n)} \longrightarrow y(n) = x(n) \star h(n)$$

Case study: Low pass filter implemented as $\begin{cases} \text{FIR} \\ \text{IIR} \end{cases}$

Target



FIR: $$H(f) = \sum_{n=0}^{N-1} b_n \mathrm{e}^{-i2\pi f n}$$

Error $$E(f) = \sqrt{T(f)} - \sum_{n=0}^{N-1} b_n \mathrm{e}^{-i2\pi f n}$$

Parseval

To be minimized: $$\delta = \int_{-0.5}^{0.5} |E(f)|^2 \mathrm{d}f = \sum_{n=0}^{N-1} |b_n|^2 + 0.2 - 2\mathrm{R}\left\{ \int_{-0.5}^{0.5} \sum_{n=0}^{N-1} b_n T(f) \mathrm{e}^{-i2\pi f n} \mathrm{d}f \right\}$$

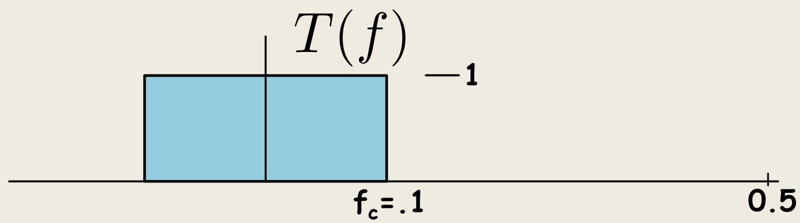
EITF75 Systems and Signals

Convolutions

$$x(n) \rightarrow \boxed{h(n)} \rightarrow y(n) = x(n) \star h(n)$$

Case study: Low pass filter implemented as $\left\{ \begin{array}{l} \textbf{FIR} \\ \textbf{IIR} \end{array} \right.$

Target



FIR: $H(f) = \sum_{n=0}^{N-1} b_n \mathrm{e}^{-i2\pi f n}$ **Error** $E(f) = T(f) - \sum_{n=0}^{N-1} b_n \mathrm{e}^{-i2\pi f n}$

To be minimized: $\delta = \int_{-0.5}^{0.5} |E(f)|^2 \mathrm{d}f$

$$= \sum_{n=0}^{N-1} |b_n|^2 + 0.2$$

$$-2 \int_{-0.5}^{0.5} \sum_{n=0}^{N-1} b_n T(f) \mathrm{e}^{-i2\pi f n} \mathrm{d}f$$

From symmetry: real

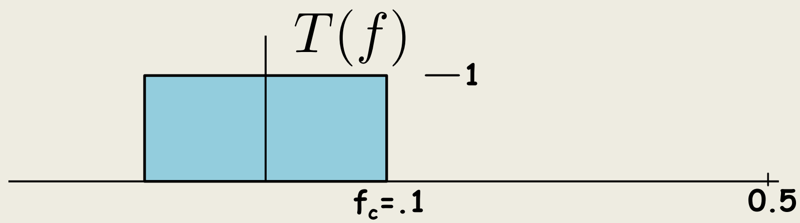
EITF75 Systems and Signals

Convolutions

$$x(n) \longrightarrow \boxed{h(n)} \longrightarrow y(n) = x(n) \star h(n)$$

Case study: Low pass filter implemented as $\left\{ \begin{array}{l} \textbf{FIR} \\ \textbf{IIR} \end{array} \right.$

Target



FIR: $$H(f) = \sum_{n=0}^{N-1} b_n \mathrm{e}^{-i2\pi f n}$$

Error $$E(f) = \sqrt{T(f)} - \sum_{n=0}^{N-1} b_n \mathrm{e}^{-i2\pi f n}$$

To be minimized: $$\delta = \int_{-0.5}^{0.5} |E(f)|^2 \mathrm{d}f = \sum_{n=0}^{N-1} |b_n|^2 + 0.2 - 2\sum_{n=0}^{N-1} b_n \int_{-0.1}^{0.1} T(f)\mathrm{e}^{-i2\pi f n}\mathrm{d}f$$

Change order

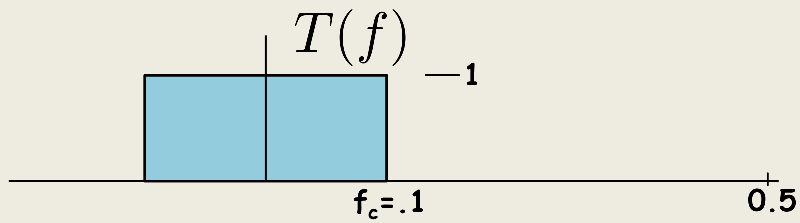
EITF75 Systems and Signals

Convolutions

$$x(n) \longrightarrow \boxed{h(n)} \longrightarrow y(n) = x(n) \star h(n)$$

Case study: Low pass filter implemented as $\begin{cases} \text{FIR} \\ \text{IIR} \end{cases}$

Target



FIR: $$H(f) = \sum_{n=0}^{N-1} b_n \mathrm{e}^{-i2\pi f n}$$

Error $$E(f) = \sqrt{T(f)} - \sum_{n=0}^{N-1} b_n \mathrm{e}^{-i2\pi f n}$$

To be minimized: $$\delta = \int_{-0.5}^{0.5} |E(f)|^2 \mathrm{d}f = \sum_{n=0}^{N-1} |b_n|^2 + 0.2 - 2\sum_{n=0}^{N-1} b_n T_n$$

$$T_n = \int_{-0.1}^{0.1} T(f)\mathrm{e}^{-i2\pi f n}\mathrm{d}f = \cdots = \frac{\sin(\pi n/5)}{\pi n}$$

Integral independent of b_n

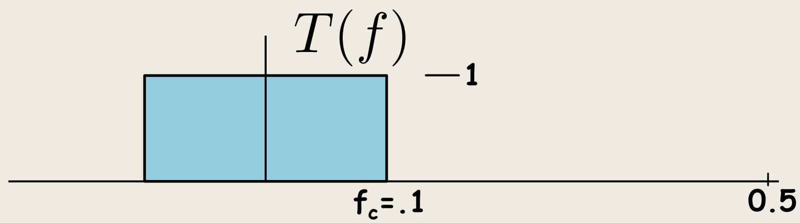
EITF75 Systems and Signals

Convolutions

$$x(n) \longrightarrow \boxed{h(n)} \longrightarrow y(n) = x(n) \star h(n)$$

Case study: Low pass filter implemented as $\begin{cases} \text{FIR} \\ \text{IIR} \end{cases}$

Target



FIR: $$H(f) = \sum_{n=0}^{N-1} b_n \mathrm{e}^{-i2\pi f n}$$

Error $$E(f) = \sqrt{T(f)} - \sum_{n=0}^{N-1} b_n \mathrm{e}^{-i2\pi f n}$$

$$\min_{\{b_n\}_{n=0}^{N-1}} \left[\sum_{n=0}^{N-1} |b_n|^2 - 2 \sum_{n=0}^{N-1} b_n T_n + 0.2 \right]$$

$$\sum_{n=0}^{N-1} |b_n|^2 \quad + \quad 0.2$$

$$-2 \sum_{n=0}^{N-1} b_n T_n$$

Optimization problem

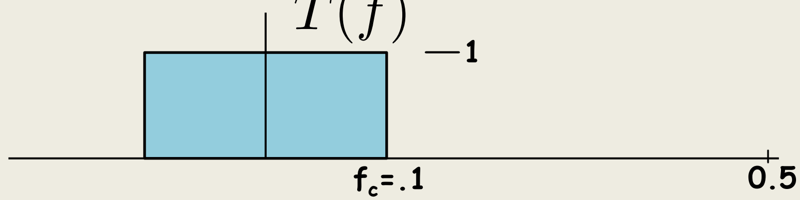
EITF75 Systems and Signals

Convolutions

$$x(n) \longrightarrow \boxed{h(n)} \longrightarrow y(n) = x(n) \star h(n)$$

Case study: Low pass filter implemented as $\left\{\begin{array}{l}\textbf{FIR}\\ \textbf{IIR}\end{array}\right.$

Target



FIR: $H(f) = \sum_{n=0}^{N-1} b_n \mathrm{e}^{-i2\pi f n}$ **Error** $E(f) = \sqrt{T(f)} - \sum_{n=0}^{N-1} b_n \mathrm{e}^{-i2\pi f n}$

$$\min_{\{b_n\}_{n=0}^{N-1}} \left[\sum_{n=0}^{N-1} |b_n|^2 - 2 \sum_{n=0}^{N-1} b_n T_n + 0.2 \right]$$

$$\boxed{\frac{\partial}{\partial b_n} \cdots = 0 \rightarrow b_n = T_n}$$

Optimization problem

$$T_n = \int_{-0.1}^{0.1} T(f) \mathrm{e}^{-i2\pi f n} \mathrm{d}f = \cdots = \frac{\sin(\pi n/5)}{\pi n}$$

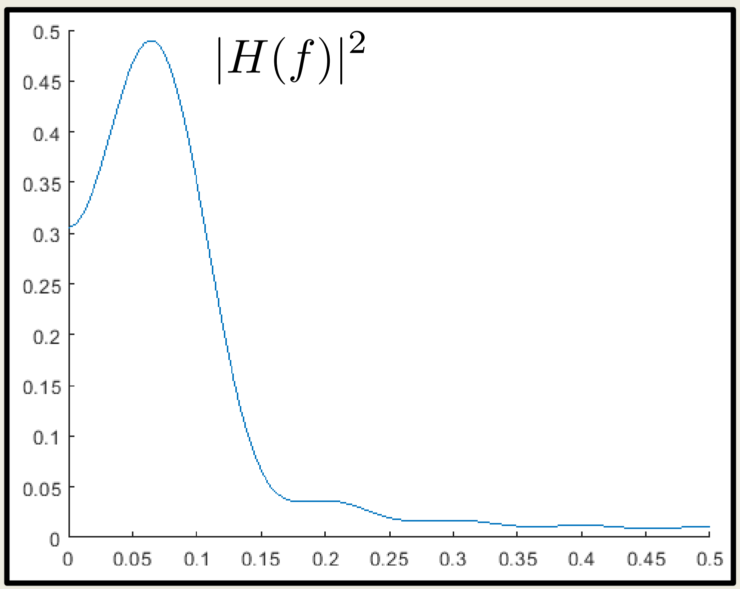
EITF75 Systems and Signals

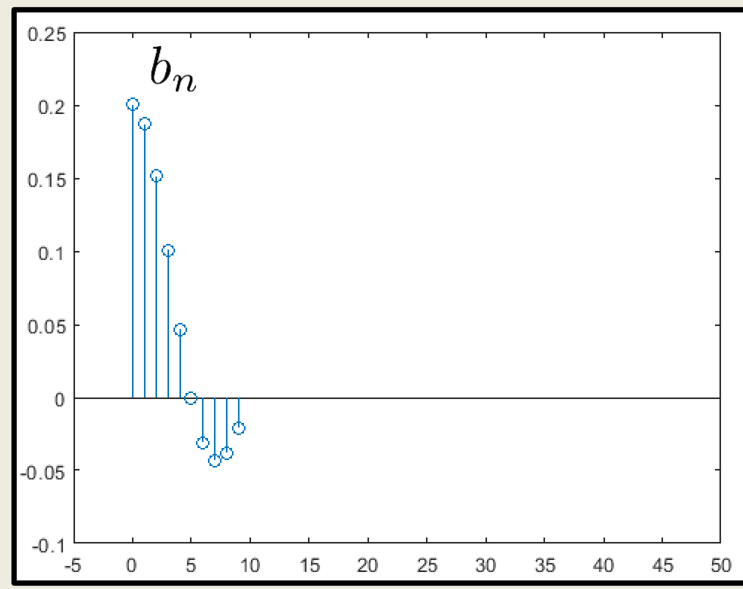
Convolutions

$$x(n) \longrightarrow \boxed{h(n)} \longrightarrow y(n) = x(n) \star h(n)$$

Case study: Low pass filter implemented as $\begin{cases} \textbf{FIR} \\ \textbf{IIR} \end{cases}$

FIR, N=10





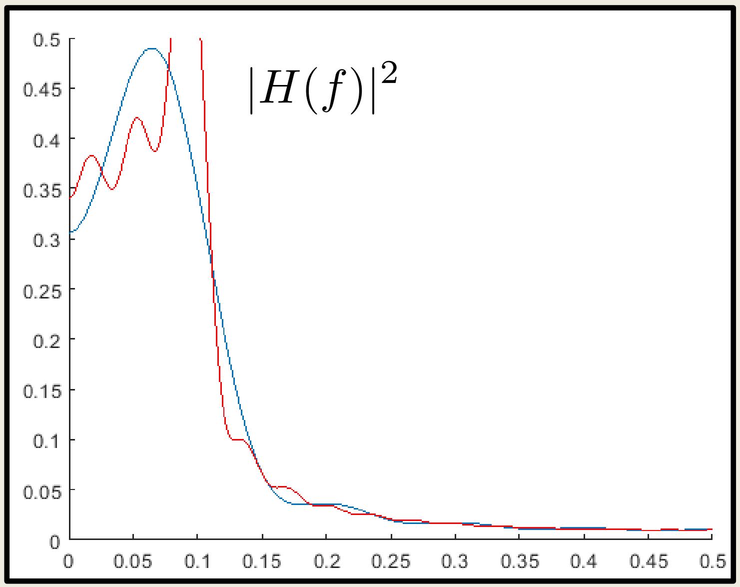
EITF75 Systems and Signals

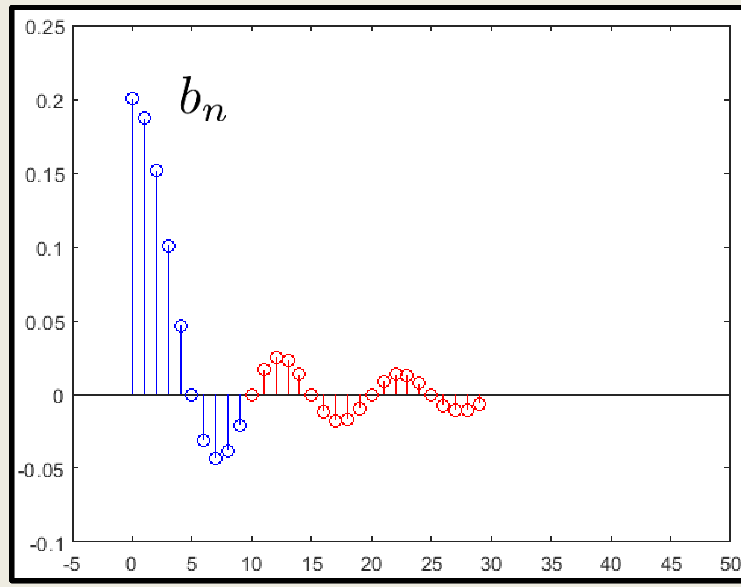
Convolutions

$$x(n) \longrightarrow \boxed{h(n)} \longrightarrow y(n) = x(n) \star h(n)$$

Case study: Low pass filter implemented as $\begin{cases} \text{FIR} \\ \text{IIR} \end{cases}$

FIR, N=10, N=30





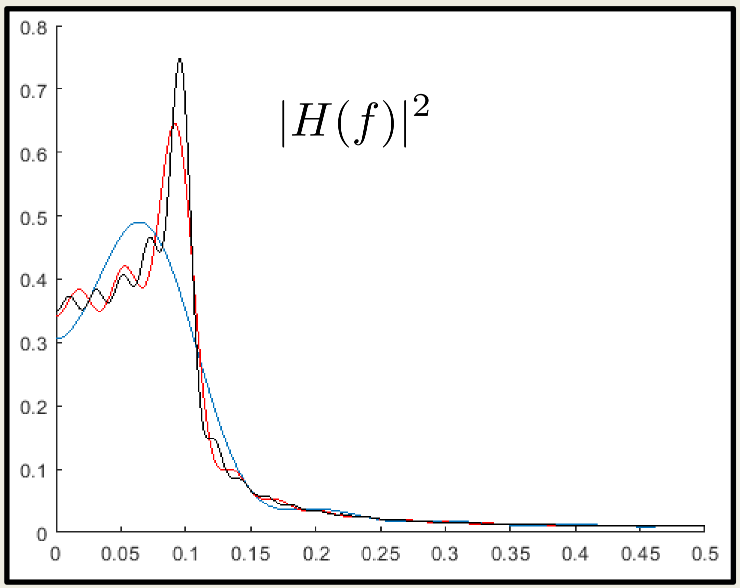
EITF75 Systems and Signals

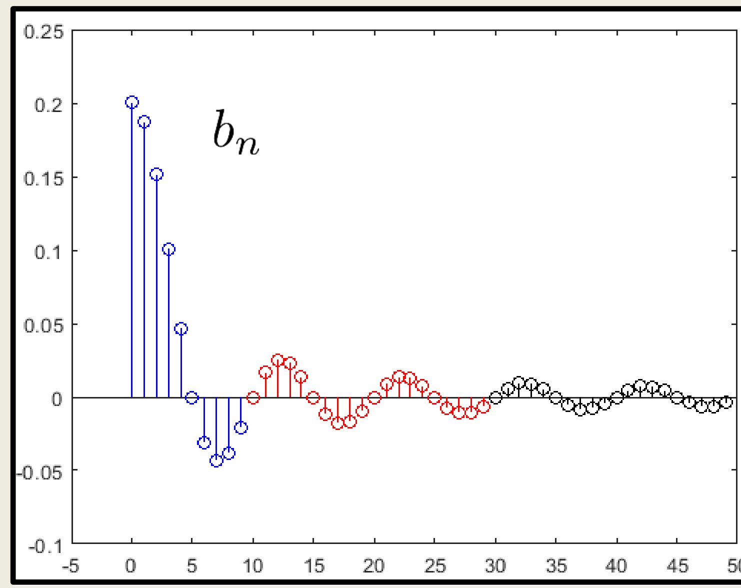
Convolutions

$$x(n) \longrightarrow \boxed{h(n)} \longrightarrow y(n) = x(n) \star h(n)$$

Case study: Low pass filter implemented as $\left\{ \begin{array}{l} \textbf{FIR} \\ \textbf{IIR} \end{array} \right.$

FIR, N=10, N=30, N=50





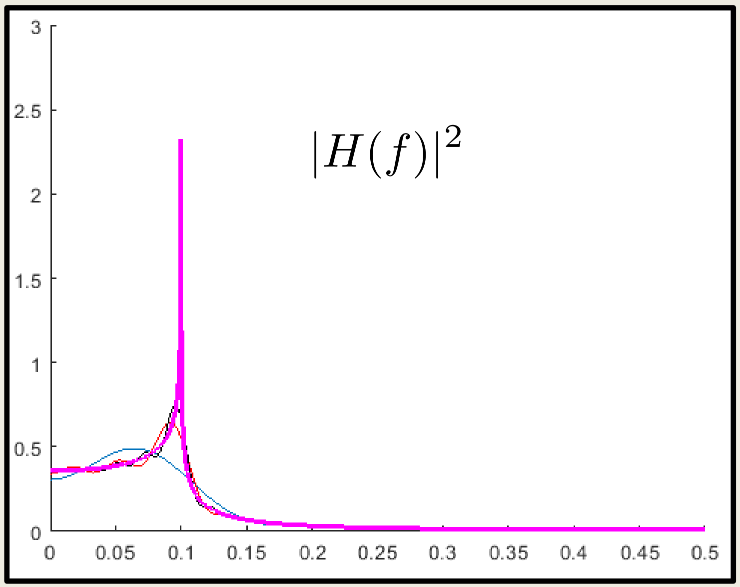
EITF75 Systems and Signals

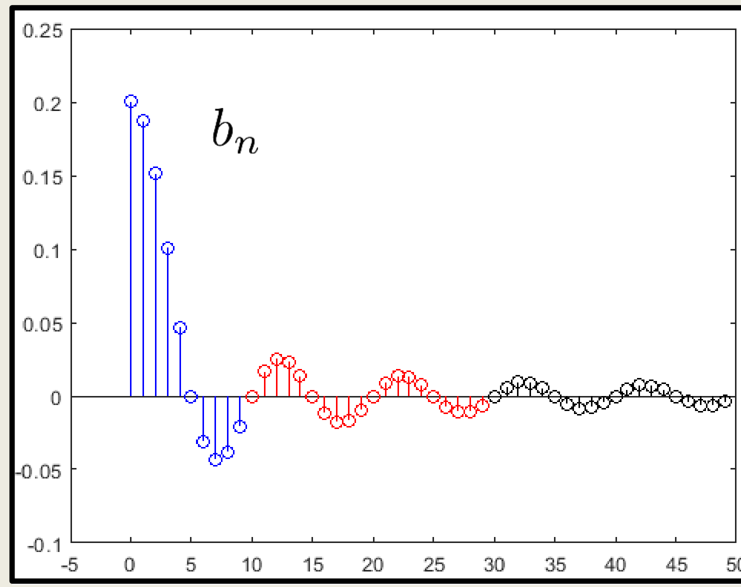
Convolutions

$$x(n) \longrightarrow \boxed{h(n)} \longrightarrow y(n) = x(n) \star h(n)$$

Case study: Low pass filter implemented as $\begin{cases} \text{FIR} \\ \text{IIR} \end{cases}$

FIR, N=10, N=30, N=50, N=1000





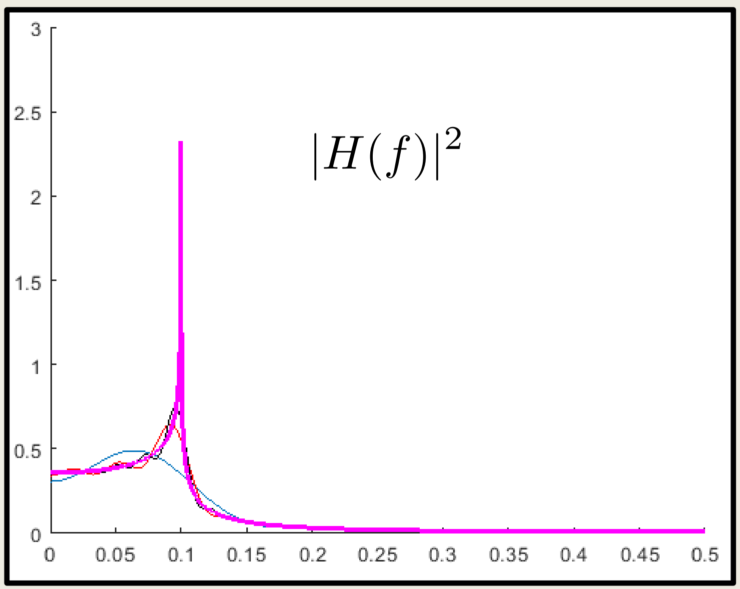
EITF75 Systems and Signals

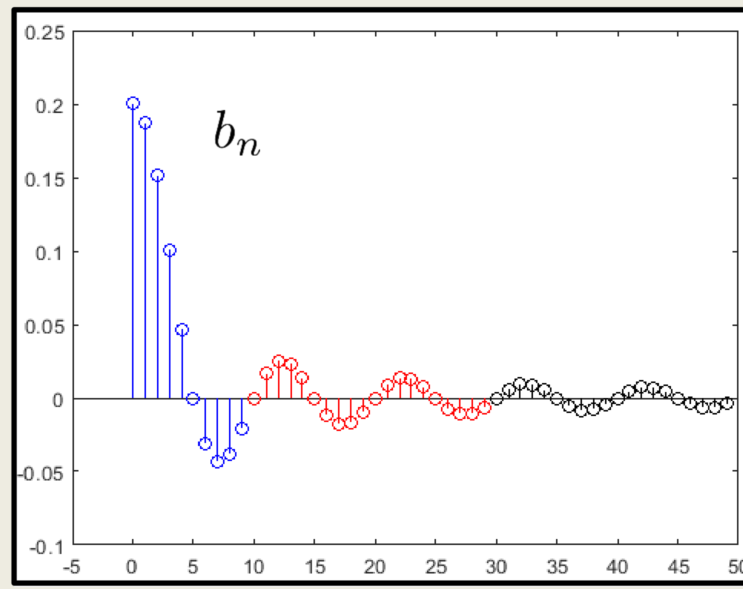
Convolutions

$$x(n) \longrightarrow \boxed{h(n)} \longrightarrow y(n) = x(n) \star h(n)$$

Case study: Low pass filter implemented as $\left\{ \begin{array}{l} \textbf{FIR} \\ \textbf{IIR} \end{array} \right.$

FIR, N=10, N=30, N=50, N=1000





EITF75 Systems and Signals

Convolutions

$x(n) \rightarrow \boxed{h(n)} \rightarrow y(n) = x(n) \star h(n)$

Case study: Low pass filter implemented as $\begin{cases} \text{FIR} \\ \text{IIR} \end{cases}$

IIR, Butterworth filter, P poles, cutoff f_c=0.1

```
>> [B,A]=butter(4,2*.1)
B =
    0.0048    0.0193    0.0289    0.0193    0.0048
A =
    1.0000   -2.3695    2.3140   -1.0547    0.1874
```

Numerator polynomial

Denominator polynomial

EITF75 Systems and Signals

Convolutions

$$x(n) \longrightarrow \boxed{h(n)} \longrightarrow y(n) = x(n) \star h(n)$$

Case study: Low pass filter implemented as $\begin{cases} \text{FIR} \\ \text{IIR} \end{cases}$

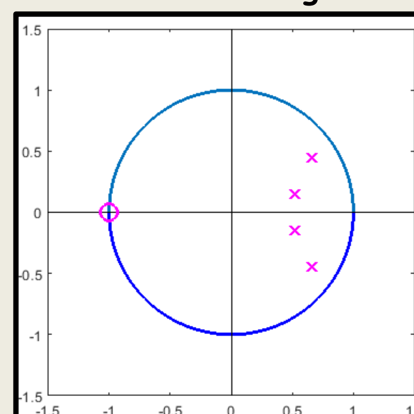
IIR, Butterworth filter, P poles, cutoff f_c=0.1

```
>> [B,A]=butter(4,2*.1)
B =
    0.0048     0.0193     0.0289     0.0193     0.0048
A =
    1.0000    -2.3695     2.3140    -1.0547     0.1874
```

Numerator polynomial

Denominator polynomial

Pole-zero diagram



EITF75 Systems and Signals

Convolutions

$$x(n) \longrightarrow \boxed{h(n)} \longrightarrow y(n) = x(n) \star h(n)$$

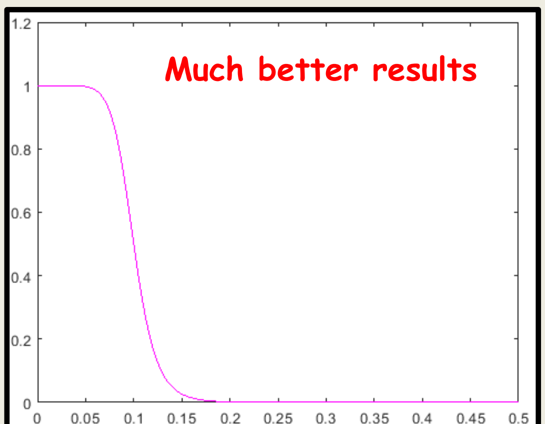
Case study: Low pass filter implemented as $\begin{cases} \text{FIR} \\ \text{IIR} \end{cases}$

IIR, Butterworth filter, P poles, cutoff f_c=0.1

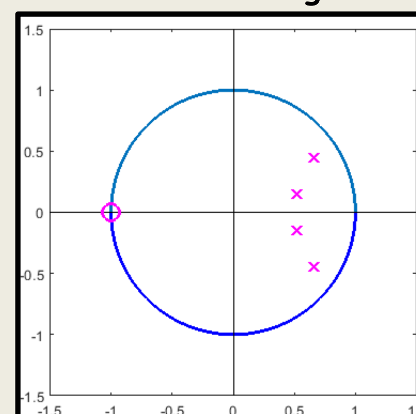
```
>> [B,A]=butter(4,2*.1)
B =
    0.0048     0.0193     0.0289     0.0193     0.0048
A =
    1.0000    -2.3695     2.3140    -1.0547     0.1874
```

Numerator polynomial

Denominator polynomial



Pole-zero diagram



EITF75 Systems and Signals

Convolutions

$$x(n) \rightarrow \boxed{h(n)} \rightarrow y(n) = x(n) \star h(n)$$

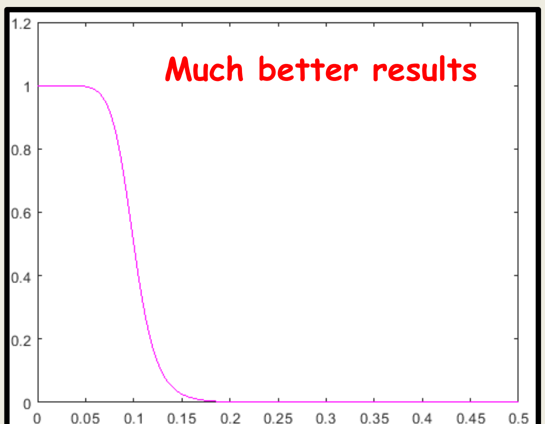
Case study: Low pass filter implemented as $\begin{cases} \text{FIR} \\ \text{IIR} \end{cases}$

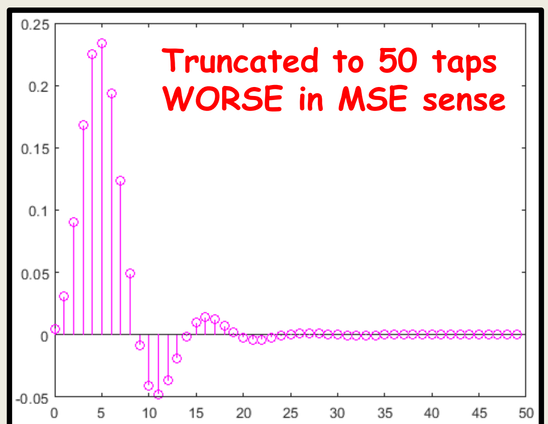
IIR, Butterworth filter, P poles, cutoff f_c=0.1

```
>> [B,A]=butter(4,2*.1)
B =
     0.0048     0.0193     0.0289     0.0193     0.0048
A =
     1.0000    -2.3695     2.3140    -1.0547     0.1874
```

Numerator polynomial

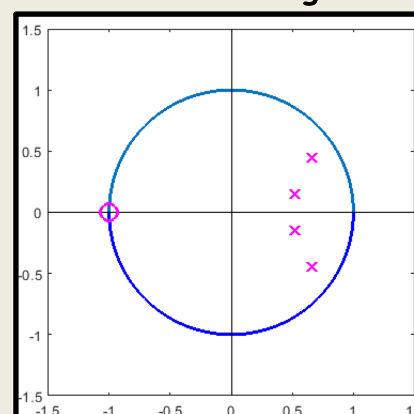
Denominator polynomial






```
>> h=filter(B,A,[1 zeros(1,1000)]);
```

Pole-zero diagram



EITF75 Systems and Signals

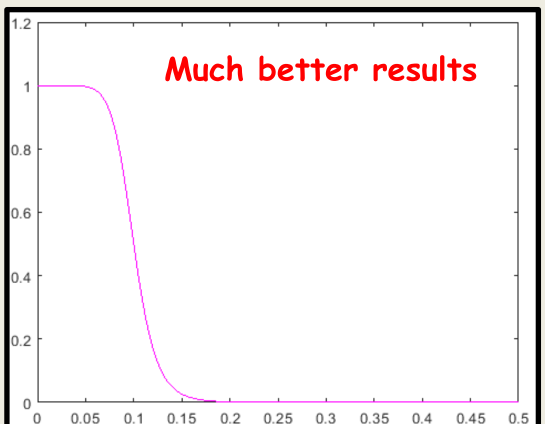
Convolutions

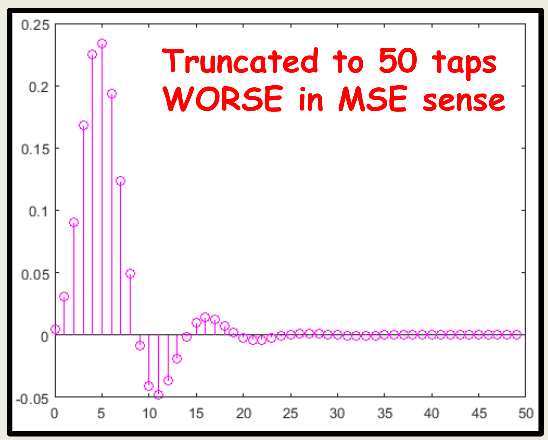
$$x(n) \longrightarrow \boxed{h(n)} \longrightarrow y(n) = x(n) \star h(n)$$

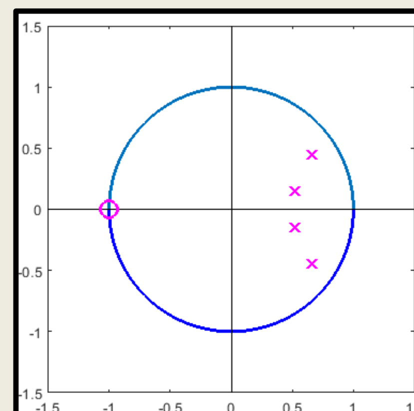
Case study: Low pass filter implemented as $\begin{cases} \text{FIR} \\ \text{IIR} \end{cases}$

IIR, Butterworth filter, P poles, cutoff f_c=0.1

```
>> [B,A]=butter(8,2*.1)
B =
    0.0000    0.0002    0.0007    0.0013    0.0017    0.0013    0.0007    0.0002    0.0000
A =
    1.0000   -4.7845   10.4450  -13.4577   11.1293   -6.0253    2.0793   -0.4172    0.0372
```







Pole-zero diagram

EITF75 Systems and Signals

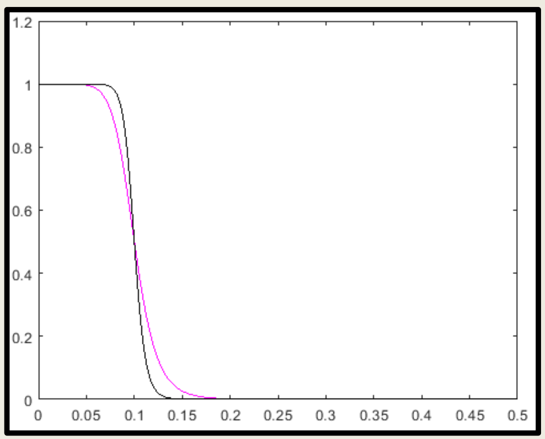
Convolutions

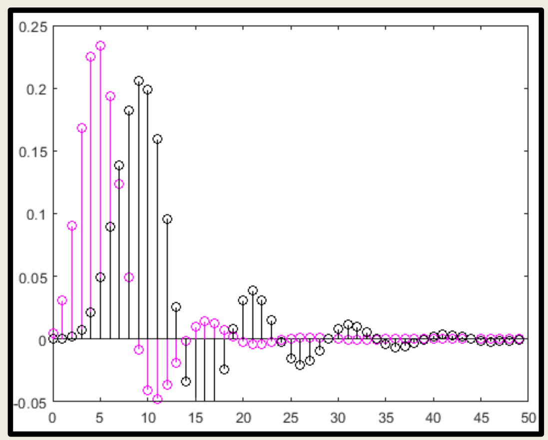
$$x(n) \longrightarrow \boxed{h(n)} \longrightarrow y(n) = x(n) \star h(n)$$

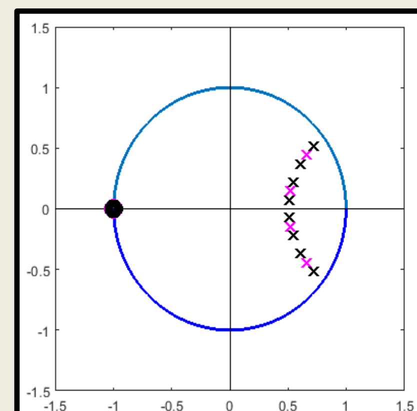
Case study: Low pass filter implemented as $\left\{ \begin{array}{l} \text{FIR} \\ \text{IIR} \end{array} \right.$

IIR, Butterworth filter, P poles, cutoff f_c=0.1

```
>> [B,A]=butter(8,2*.1)
B =
    0.0000    0.0002    0.0007    0.0013    0.0017    0.0013    0.0007    0.0002    0.0000
A =
    1.0000   -4.7845   10.4450  -13.4577   11.1293   -6.0253    2.0793   -0.4172    0.0372
```







Pole-zero diagram

EITF75 Systems and Signals

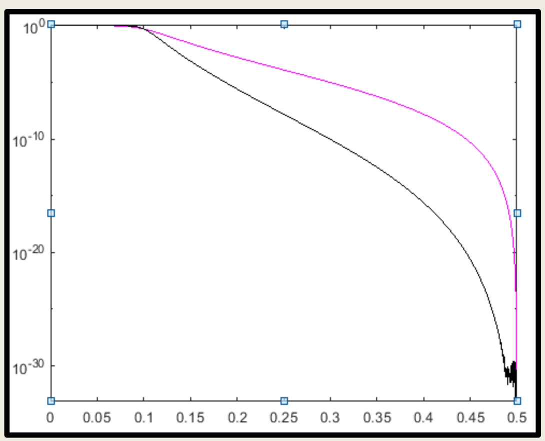
Convolutions

$$x(n) \longrightarrow \boxed{h(n)} \longrightarrow y(n) = x(n) \star h(n)$$

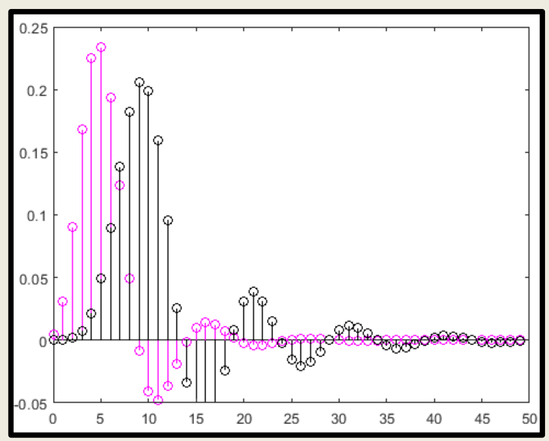
Case study: Low pass filter implemented as $\begin{cases} \text{FIR} \\ \text{IIR} \end{cases}$

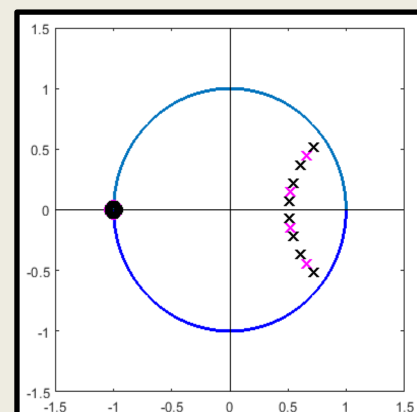
IIR, Butterworth filter, P poles, cutoff f_c=0.1

```
>> [B,A]=butter(8,2*.1)
B =
    0.0000    0.0002    0.0007    0.0013    0.0017    0.0013    0.0007    0.0002    0.0000
A =
    1.0000   -4.7845   10.4450  -13.4577   11.1293   -6.0253    2.0793   -0.4172    0.0372
```



Log scale





Pole-zero diagram

EITF75 Systems and Signals

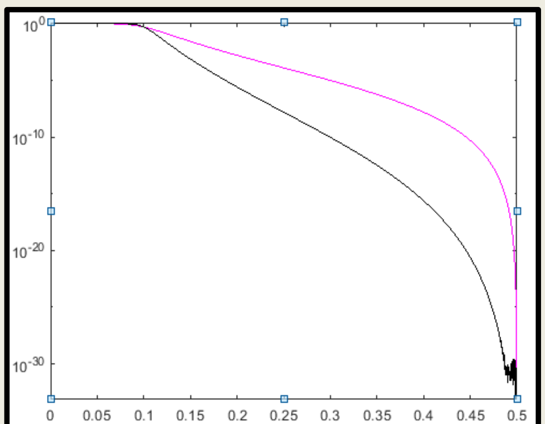
Convolutions

$$x(n) \longrightarrow \boxed{h(n)} \longrightarrow y(n) = x(n) \star h(n)$$

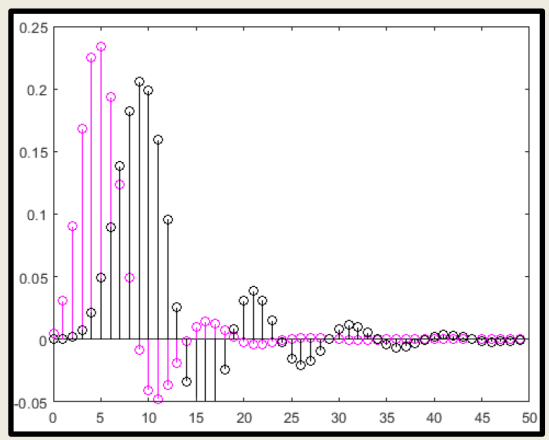
Case study: Low pass filter implemented as $\begin{cases} \text{FIR} \\ \text{IIR} \end{cases}$

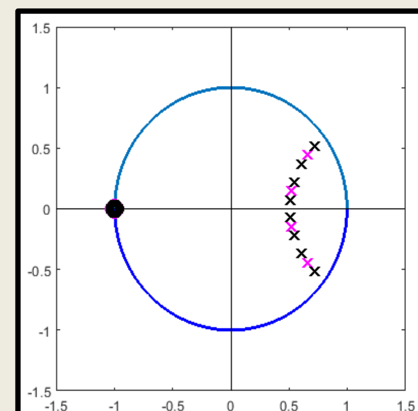
IIR, Butterworth filter, P poles, cutoff f_c=0.1

```
>> [B,A]=butter(8,2*.1)
B =
    0.0000    0.0002    0.0007    0.0013    0.0017    0.0013    0.0007    0.0002    0.0000
A =
    1.0000   -4.7845   10.4450  -13.4577   11.1293   -6.0253    2.0793   -0.4172    0.0372
```



Log scale





Pole-zero diagram

Lesson learded: IIR filters superior. Simple implementation, good results

EITF75 Systems and Signals

DFT of sinusoids

$$x(n) = \cos\left(2\pi \frac{k_0}{N} n\right)$$

Find DFT

Note, for k_0 an integer, an integer number of periods

EITF75 Systems and Signals

DFT of sinusoids

$$x(n) = \cos\left(2\pi \frac{k_0}{N} n\right)$$

Find DFT

Note, for k_0 an integer, an integer number of periods

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-j2\pi \cdot \frac{k}{N} \cdot n}$$

By definition

EITF75 Systems and Signals

DFT of sinusoids

$$x(n) = \cos\left(2\pi\frac{k_0}{N}n\right)$$

Find DFT

Note, for k_0 an integer, an integer number of periods

$$X(k) = \sum_{n=0}^{N-1} x(n)e^{-j2\pi\cdot\frac{k}{N}\cdot n}$$

$$= \sum_{n=0}^{N-1} \frac{1}{2}\cdot\left[e^{j2\pi\cdot\frac{k_0}{N}\cdot n} + e^{-j2\pi\cdot\frac{k_0}{N}\cdot n}\right]\cdot e^{-j2\pi\cdot\frac{k}{N}\cdot n}$$

Euler

EITF75 Systems and Signals

DFT of sinusoids

$$x(n) = \cos\left(2\pi \frac{k_0}{N} n\right)$$

Find DFT

Note, for k_0 an integer, an integer number of periods

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-j2\pi \cdot \frac{k}{N} \cdot n}$$

$$= \sum_{n=0}^{N-1} \frac{1}{2} \cdot \left[e^{j2\pi \cdot \frac{k_0}{N} \cdot n} + e^{-j2\pi \cdot \frac{k_0}{N} \cdot n} \right] \cdot e^{-j2\pi \cdot \frac{k}{N} \cdot n}$$

$$= \sum_{n=0}^{N-1} \frac{1}{2} \cdot e^{-j2\pi \cdot \frac{k-k_0}{N} \cdot n} + \sum_{n=0}^{N-1} \frac{1}{2} \cdot e^{-j2\pi \cdot \frac{k+k_0}{N} \cdot n}$$

Preparation for geometric series

EITF75 Systems and Signals

DFT of sinusoids

$$x(n) = \cos\left(2\pi \frac{k_0}{N} n\right)$$

Find DFT

Note, for k_0 an integer, an integer number of periods

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-j2\pi \cdot \frac{k}{N} \cdot n}$$

$$= \sum_{n=0}^{N-1} \frac{1}{2} \cdot \left[e^{j2\pi \cdot \frac{k_0}{N} \cdot n} + e^{-j2\pi \cdot \frac{k_0}{N} \cdot n} \right] \cdot e^{-j2\pi \cdot \frac{k}{N} \cdot n}$$

$$= \sum_{n=0}^{N-1} \frac{1}{2} \cdot e^{-j2\pi \cdot \frac{k-k_0}{N} \cdot n} + \sum_{n=0}^{N-1} \frac{1}{2} \cdot e^{-j2\pi \cdot \frac{k+k_0}{N} \cdot n}$$

$$= \frac{N}{2} \cdot \left[\delta(k - k_0 \mod N) + \delta(k + k_0 \mod N) \right]$$

We have seen this before (lecture 10)

EITF75 Systems and Signals

DFT of sinusoids

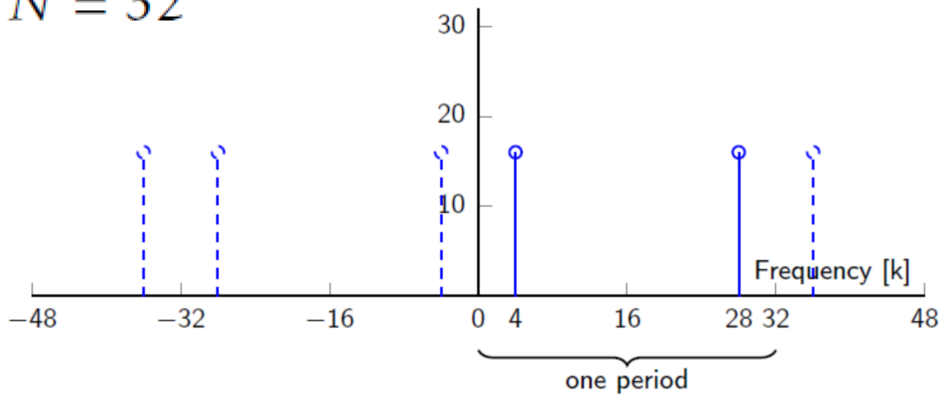
$$x(n) = \cos\left(2\pi \frac{k_0}{N} n\right)$$

Find DFT

Note, for k_0 an integer, an integer number of periods

$k_0 = 4$

$N = 32$



$$X(k) = \frac{N}{2} \cdot [\delta(k - k_0 \mod N) + \delta(k + k_0 \mod N)]$$

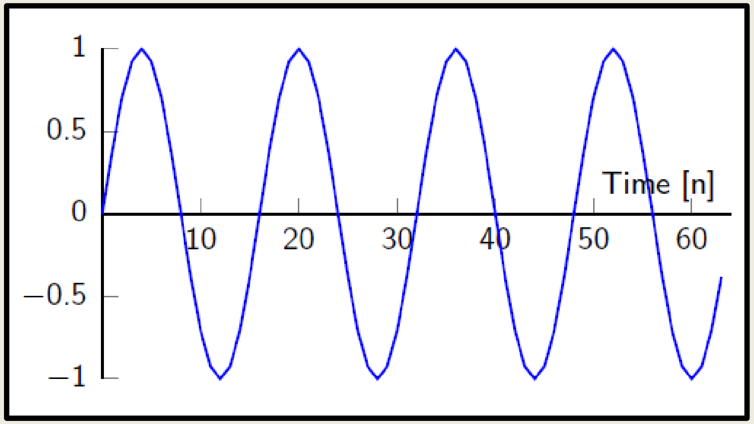
EITF75 Systems and Signals

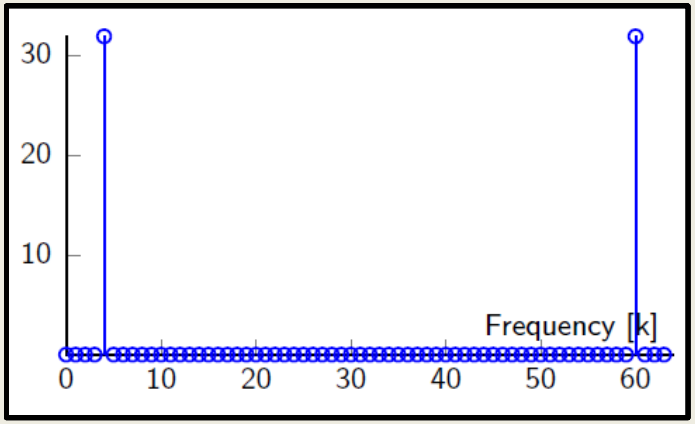
DFT of sinusoids

$$x(n) = \cos\left(2\pi \frac{k_0}{N} n\right)$$

Find DFT

Note, for k_0 an integer, an integer number of periods





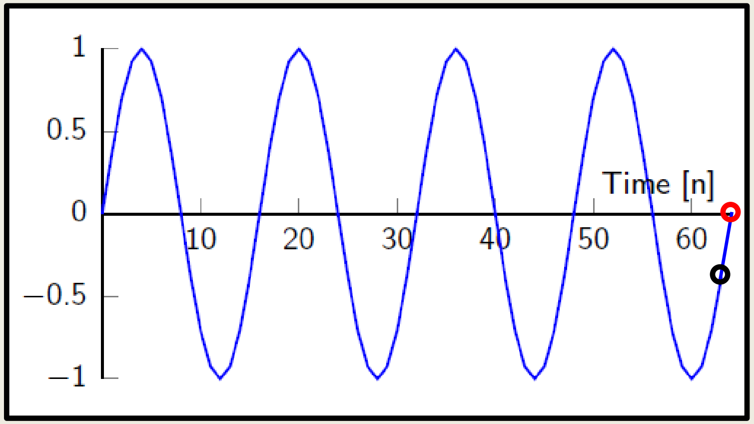
EITF75 Systems and Signals

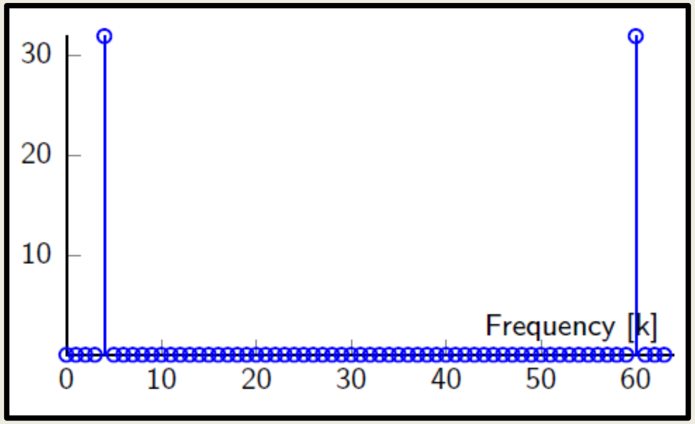
DFT of sinusoids

$$x(n) = \cos\left(2\pi \frac{k_0}{N} n\right)$$

Find DFT

Note, for k_0 an integer, an integer number of periods





This IS an even number of periods

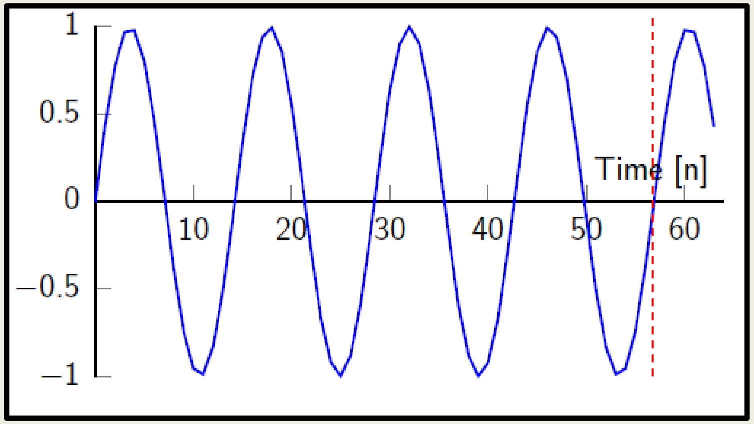
EITF75 Systems and Signals

DFT of sinusoids

$$x(n) = \cos\left(2\pi \frac{k_0}{N} n\right)$$

Find DFT

Note, for k_0 an integer, an integer number of periods



??

What would happen if k_0 not an integer

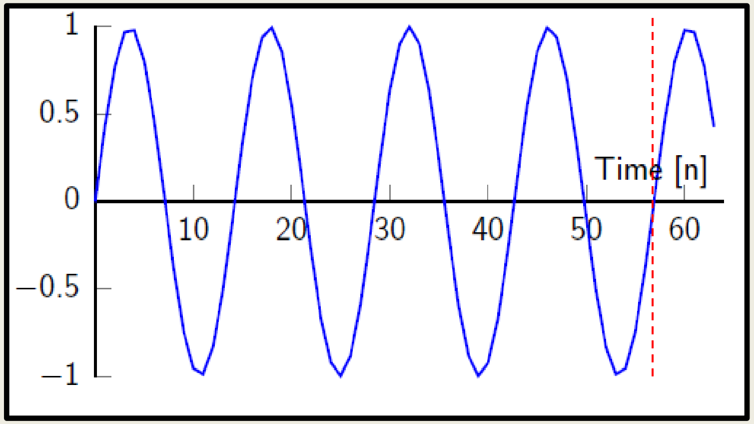
EITF75 Systems and Signals

DFT of sinusoids

$$x(n) = \cos\left(2\pi \frac{k_0}{N} n\right)$$

Find DFT

Note, for k_0 an integer, an integer number of periods



Essentially, we get a delta at k_0, but this is not an integer, so not computed by the DFT

What would happen if k_0 not an integer

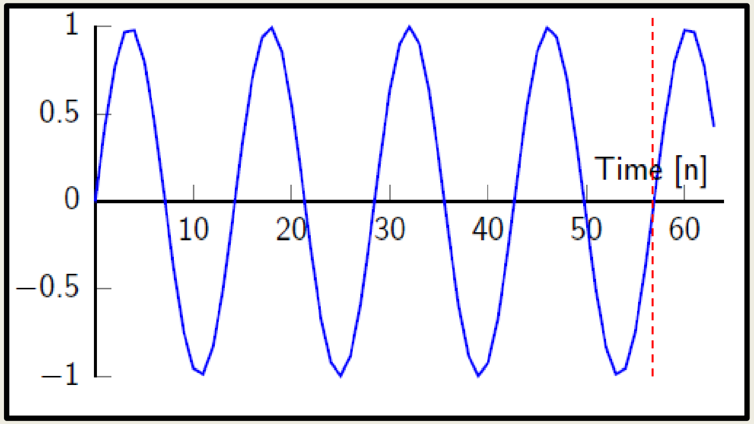
EITF75 Systems and Signals

DFT of sinusoids

$$x(n) = \cos\left(2\pi \frac{k_0}{N} n\right)$$

Find DFT

Note, for k_0 an integer, an integer number of periods



For the above case, $4<k_0<5$

Essentially, we get a delta at k_0, but this is not an integer, so not computed by the DFT

Spectral leakage: We get something around k_0

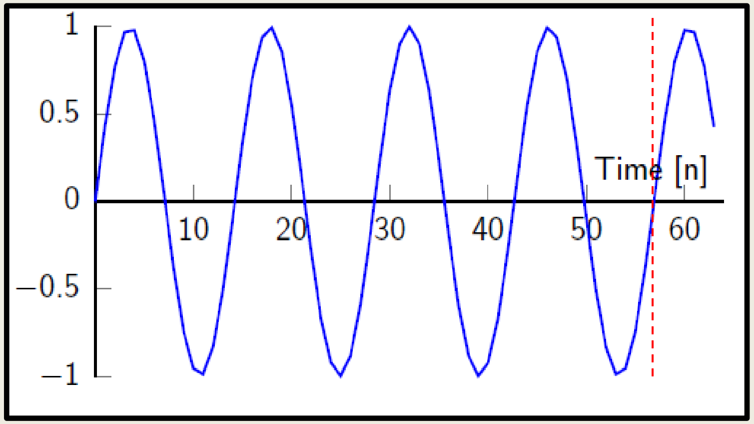
EITF75 Systems and Signals

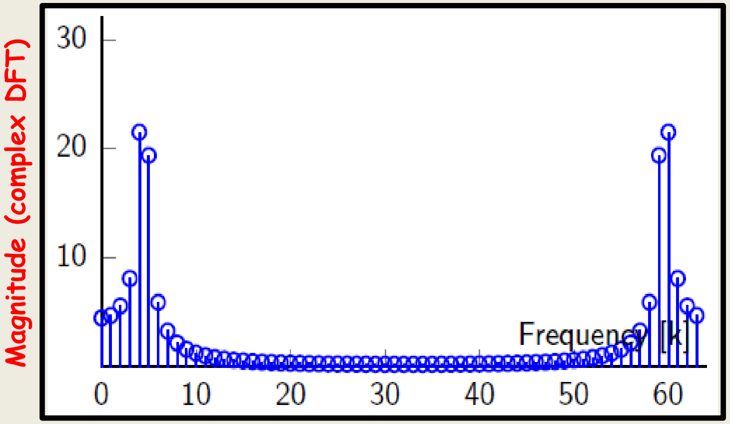
DFT of sinusoids

$$x(n) = \cos\left(2\pi \frac{k_0}{N} n\right)$$

Find DFT

Note, for k_0 an integer, an integer number of periods





For the above case, $4<k_0<5$

Observe something around 4-5

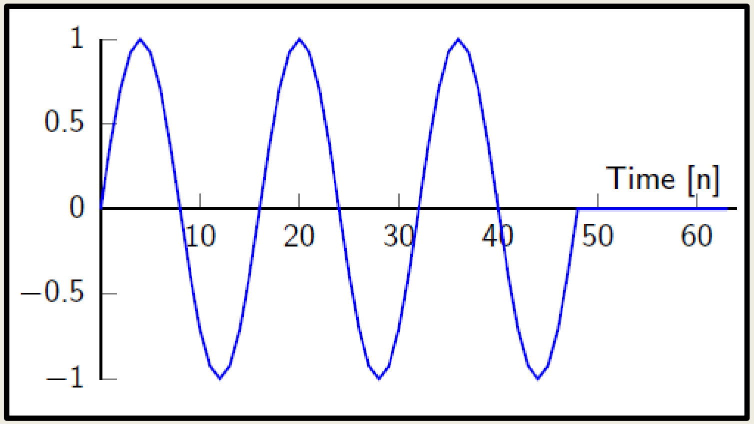
EITF75 Systems and Signals

DFT of sinusoids

$$x(n) = \cos\left(2\pi \frac{k_0}{N} n\right)$$

Find DFT

Note, for k_0 an integer, an integer number of periods



??

DFT of integer k_0, but with zero-padding

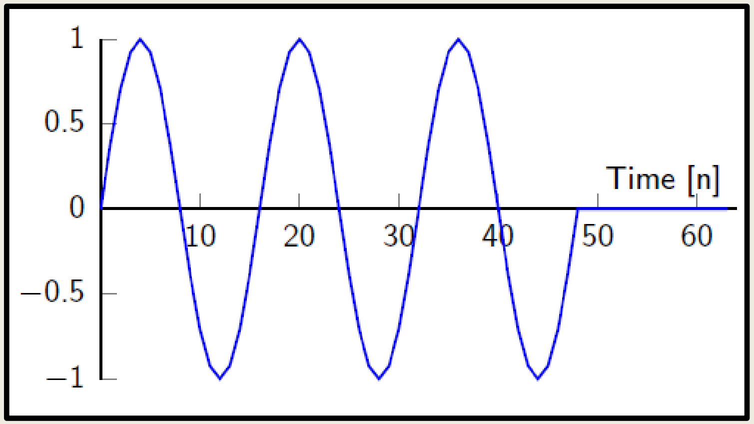
EITF75 Systems and Signals

DFT of sinusoids

$$x(n) = \cos\left(2\pi \frac{k_0}{N} n\right)$$

Find DFT

Note, for k_0 an integer, an integer number of periods



DFT of integer k_0, but with zero-padding

We cannot get a delta at k=4, since that is what we obtain without zero-padding

Convolution theorem duality [edit]

It can also be shown that:

$$\mathcal{F}\{\mathbf{x} \cdot \mathbf{y}\}_k \triangleq \sum_{n=0}^{N-1} x_n \cdot y_n \cdot e^{-i\frac{2\pi}{N}kn}$$

$$= \frac{1}{N}(\mathbf{X} * \mathbf{Y_N})_k,$$ which is the circular convolution of $\mathbf{X}$ and $\mathbf{Y}$.

WIKIPEDIA
The Free Encyclopedia

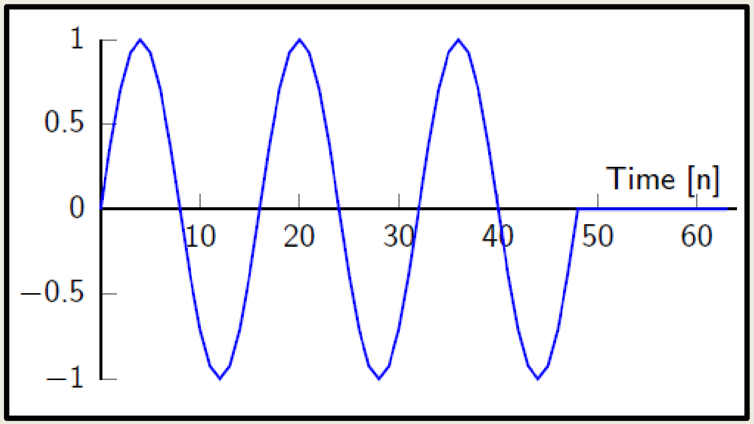
EITF75 Systems and Signals

DFT of sinusoids

$$x(n) = \cos\left(2\pi \frac{k_0}{N} n\right)$$

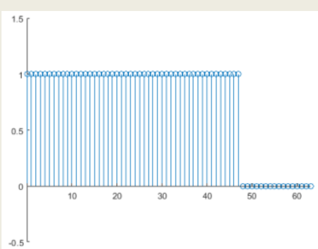
Find DFT

Note, for k_0 an integer, an integer number of periods

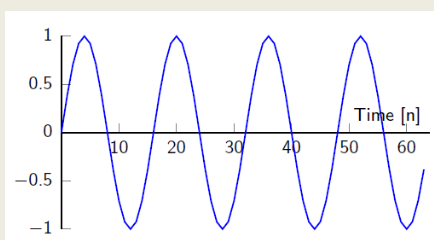


We cannot get a delta at k=4, since that is what we obtain without zero-padding

=



*



Convolution theorem duality [edit]

It can also be shown that:

$$\mathcal{F}\{\mathbf{x}\cdot\mathbf{y}\}_k \triangleq \sum_{n=0}^{N-1} x_n \cdot y_n \cdot e^{-i\frac{2\pi}{N}kn}$$

$$= \frac{1}{N}(\mathbf{X} * \mathbf{Y_N})_k,$$ which is the circular convolution of $\mathbf{X}$ and $\mathbf{Y}$.

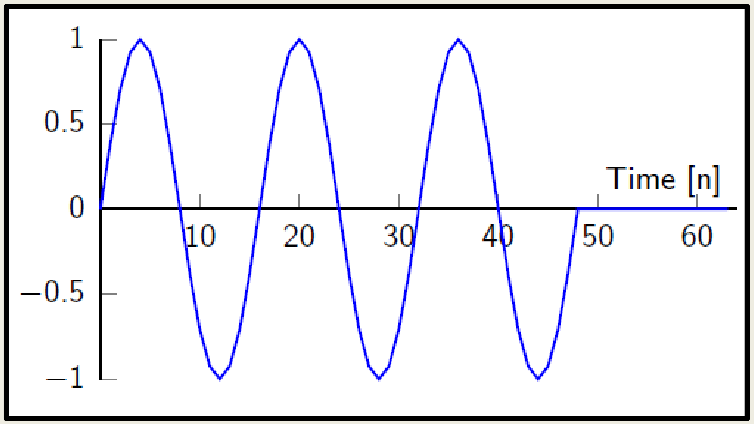
WIKIPEDIA
The Free Encyclopedia

DFT of sinusoids

$$x(n) = \cos\left(2\pi \frac{k_0}{N} n\right)$$

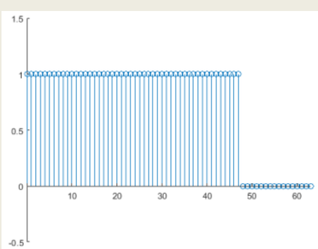
Find DFT

Note, for k_0 an integer, an integer number of periods

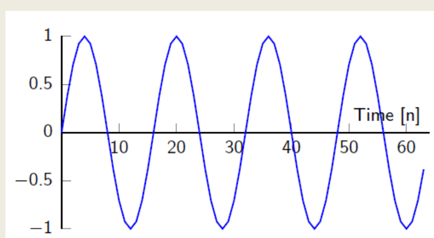


DFT of zero-padded signal will be circular convolution of DFTs of the two bottom pictures

=



*



Convolution theorem duality [edit]

It can also be shown that:

$$\mathcal{F}\{\mathbf{x}\cdot\mathbf{y}\}_k \triangleq \sum_{n=0}^{N-1} x_n \cdot y_n \cdot e^{-i\frac{2\pi}{N}kn}$$

$$= \frac{1}{N}(\mathbf{X} * \mathbf{Y_N})_k,$$ which is the circular convolution of $\mathbf{X}$ and $\mathbf{Y}$.

WIKIPEDIA
The Free Encyclopedia

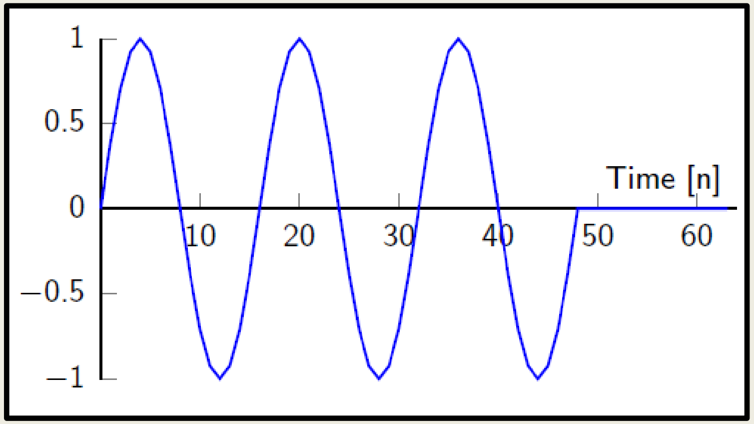
EITF75 Systems and Signals

DFT of sinusoids

$$x(n) = \cos\left(2\pi \frac{k_0}{N} n\right)$$

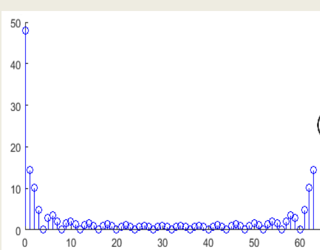
Find DFT

Note, for k_0 an integer, an integer number of periods

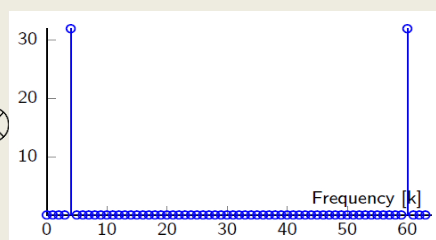


DFT of zero-padded signal will be circular convolution of DFTs of the two bottom pictures

Magnitude (complex DFT)



$\otimes$



Convolution theorem duality [edit]

It can also be shown that:

$$\mathcal{F}\{\mathbf{x}\cdot\mathbf{y}\}_k \triangleq \sum_{n=0}^{N-1} x_n \cdot y_n \cdot e^{-i\frac{2\pi}{N}kn}$$

$$= \frac{1}{N}(\mathbf{X} * \mathbf{Y_N})_k,$$ which is the circular convolution of $\mathbf{X}$ and $\mathbf{Y}$.

WIKIPEDIA
The Free Encyclopedia

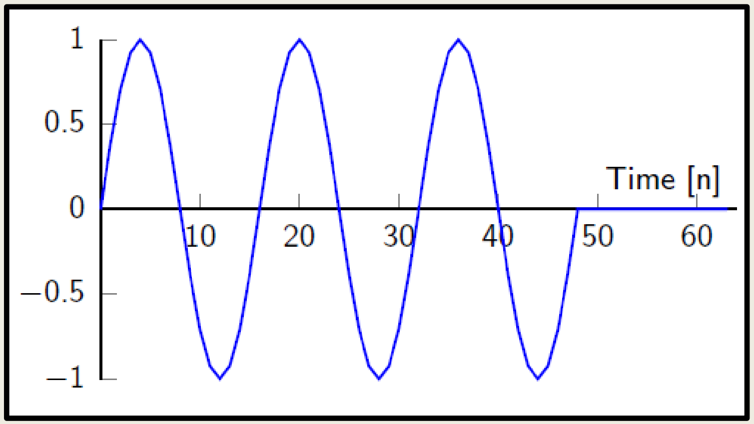
EITF75 Systems and Signals

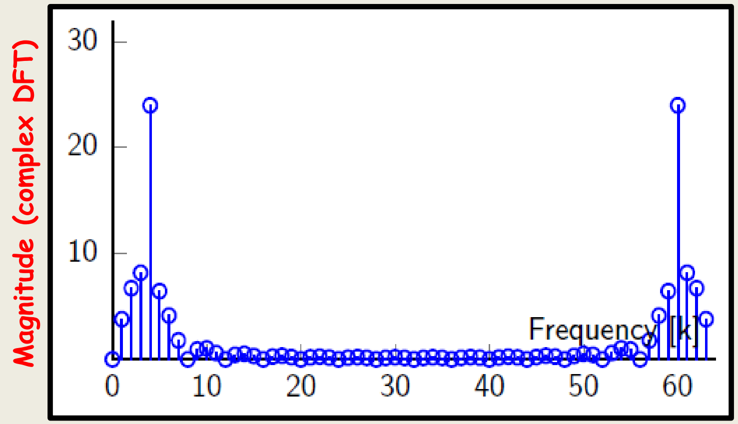
DFT of sinusoids

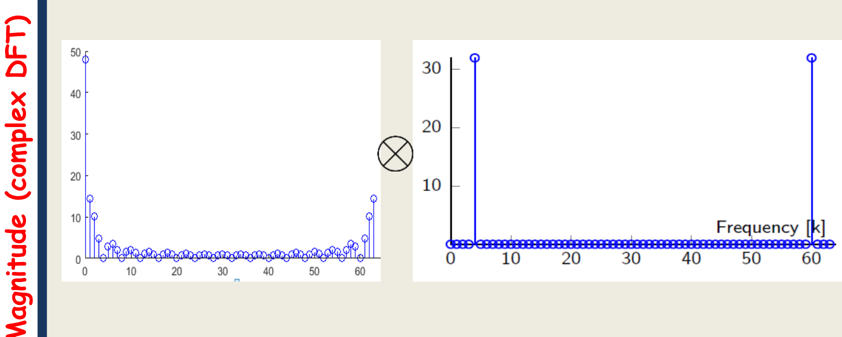
$$x(n) = \cos\left(2\pi \frac{k_0}{N} n\right)$$

Find DFT

Note, for k_0 an integer, an integer number of periods







Convolution theorem duality [edit]

It can also be shown that:

$$\mathcal{F}\{\mathbf{x}\cdot\mathbf{y}\}_k \triangleq \sum_{n=0}^{N-1} x_n \cdot y_n \cdot e^{-i\frac{2\pi}{N}kn}$$

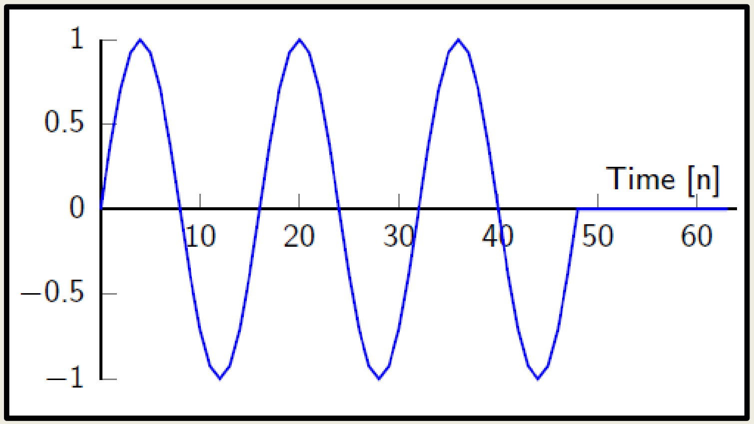
$$= \frac{1}{N}(\mathbf{X} * \mathbf{Y_N})_k,$$ which is the circular convolution of $\mathbf{X}$ and $\mathbf{Y}$.

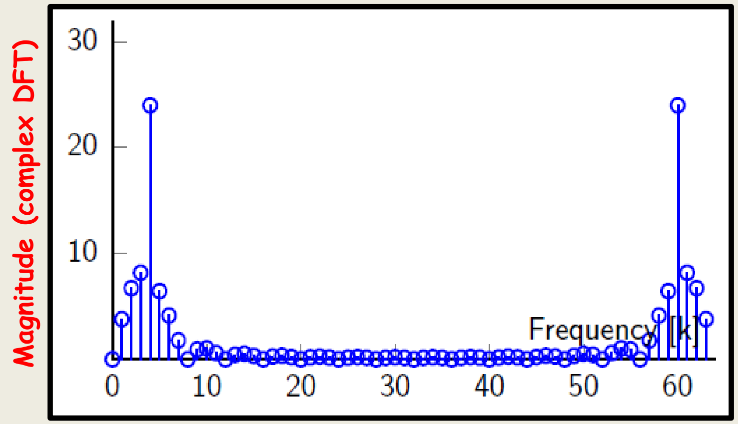
Wikipedia
The Free Encyclopedia

EITF75 Systems and Signals

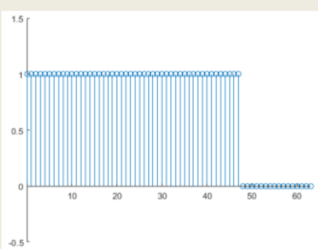
DFT of sinusoids

$$x(n) = \cos\left(2\pi \frac{k_0}{N} n\right)$$ **Find DFT**

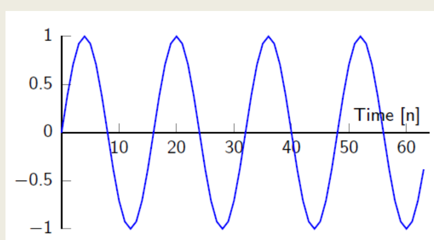




=



*



Convolution theorem duality [edit]

It can also be shown that:

$$\mathcal{F}\{\mathbf{x}\cdot\mathbf{y}\}_k \triangleq \sum_{n=0}^{N-1} x_n \cdot y_n \cdot e^{-i\frac{2\pi}{N}kn}$$

$$= \frac{1}{N}(\mathbf{X} * \mathbf{Y_N})_k,$$ which is the circular convolution of $\mathbf{X}$ and $\mathbf{Y}$.

WIKIPEDIA
The Free Encyclopedia

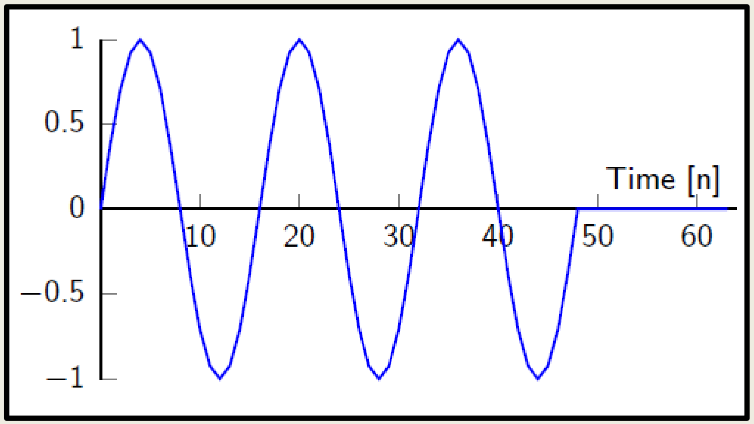
Can we see the peak at n=4 sharper if we window with something else ?

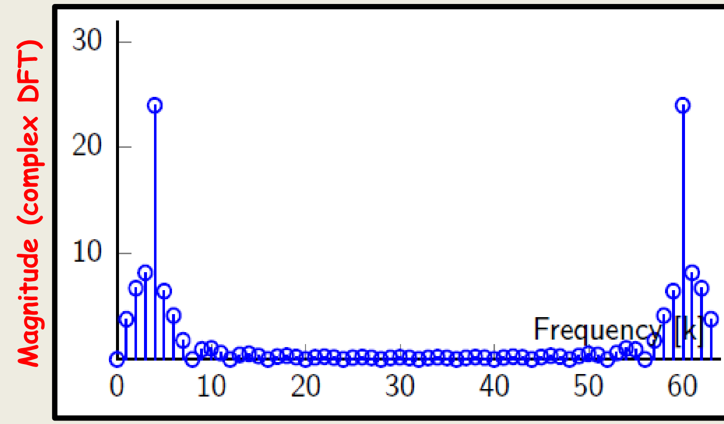
EITF75 Systems and Signals

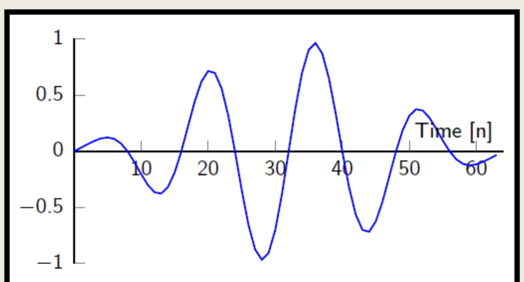
DFT of sinusoids

$$x(n) = \cos\left(2\pi \frac{k_0}{N} n\right)$$

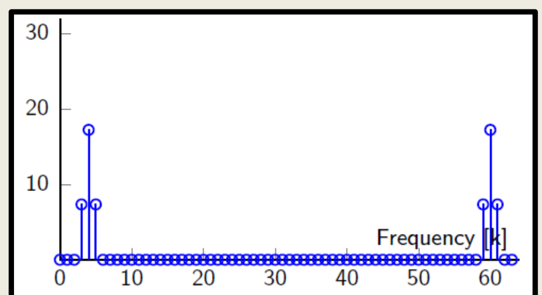
Find DFT







DFT $\longleftrightarrow$

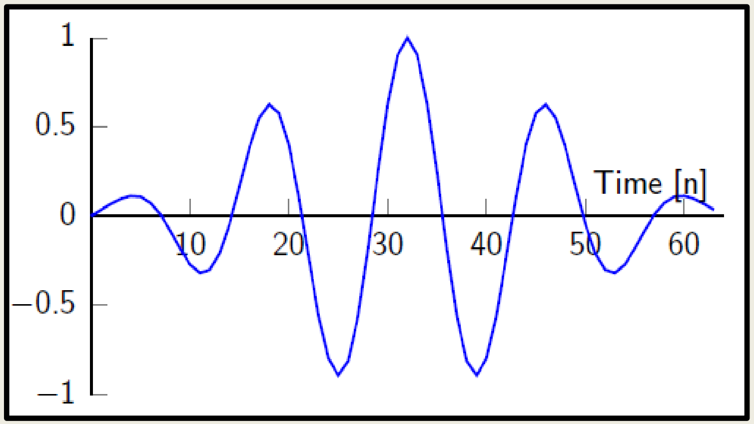


Hamming window $w_{\text{hamming}}(n) = 0.54 + 0.46\cos\left(2\pi \cdot \frac{1}{N-1} \cdot \left(n - \frac{N-1}{2}\right)\right)$

EITF75 Systems and Signals

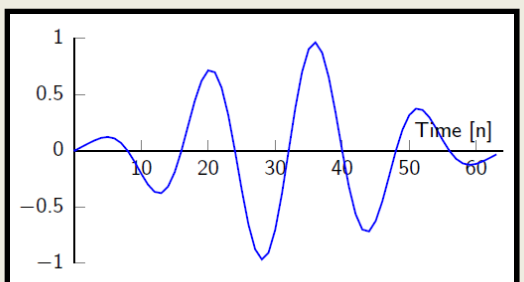
DFT of sinusoids

$$x(n) = \cos\left(2\pi \frac{k_0}{N} n\right)$$ **Find DFT**

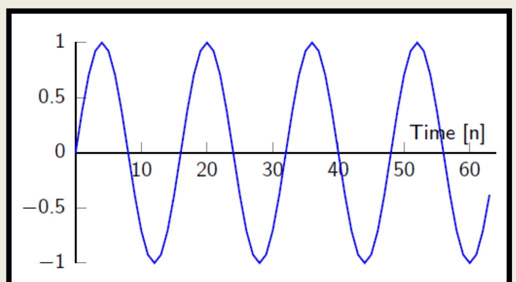


??

=



*

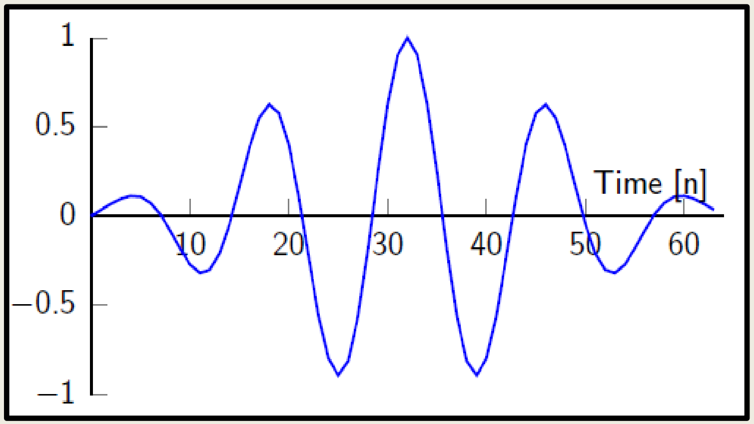


Hamming window $$w_{\text{hamming}}(n) = 0.54 + 0.46\cos\left(2\pi \cdot \frac{1}{N-1} \cdot \left(n - \frac{N-1}{2}\right)\right)$$

DFT of sinusoids

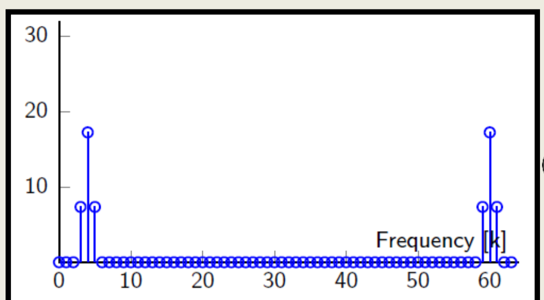
$$x(n) = \cos\left(2\pi \frac{k_0}{N} n\right)$$

Find DFT

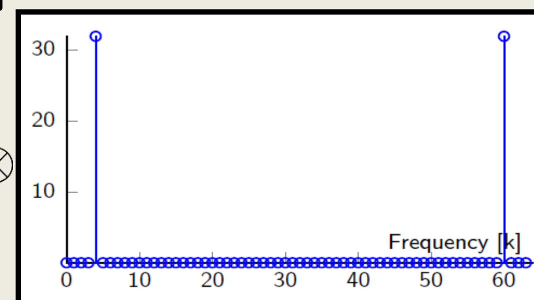


??

=



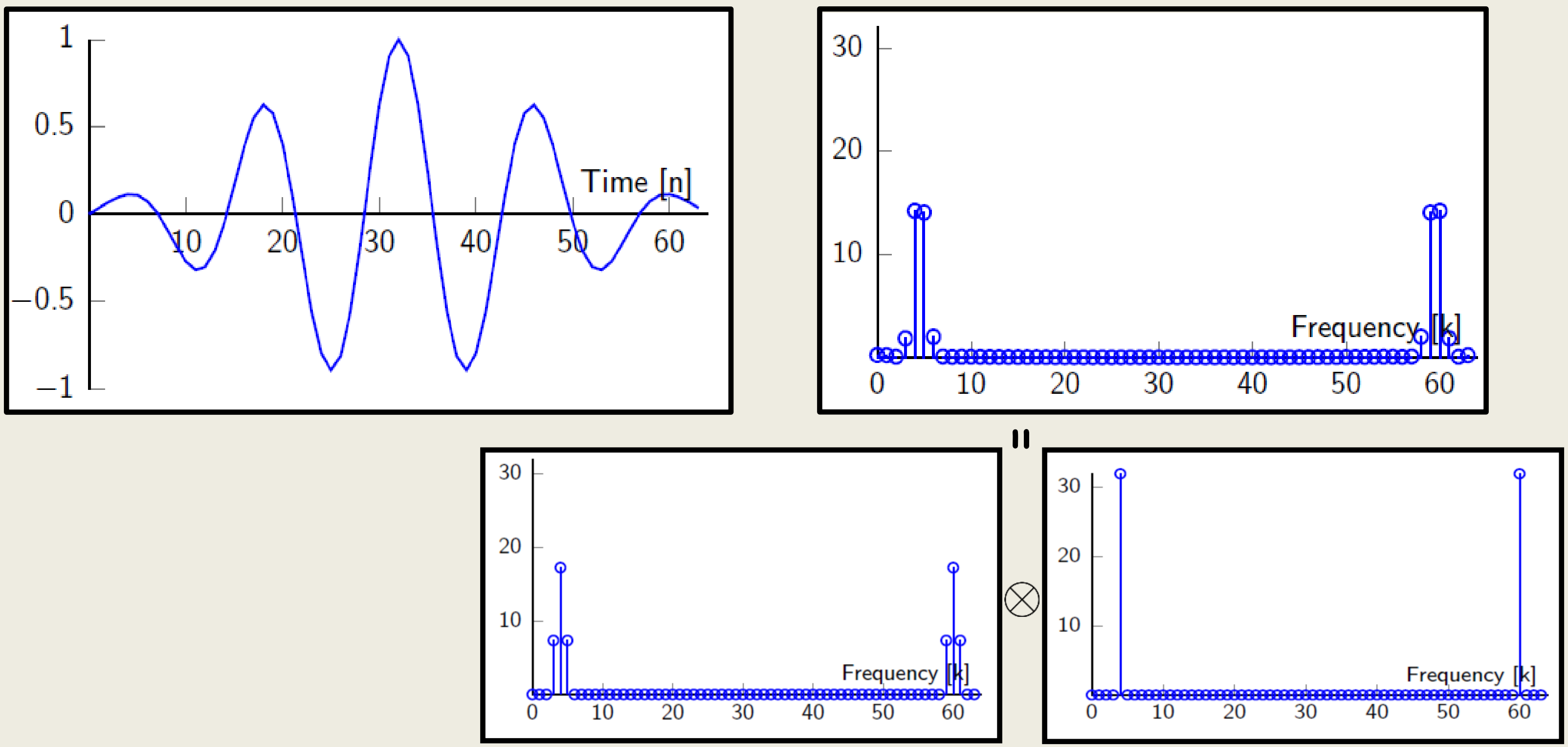
$\otimes$



EITF75 Systems and Signals

DFT of sinusoids

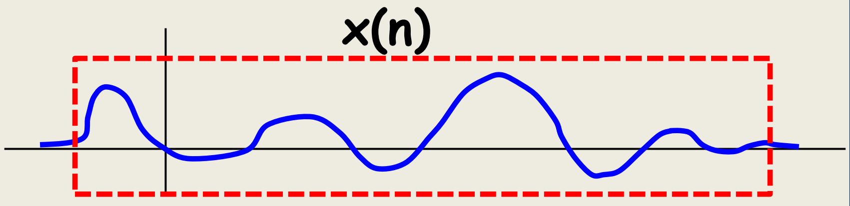
$$x(n) = \cos\left(2\pi \frac{k_0}{N} n\right)$$ **Find DFT**



EITF75 Systems and Signals

Summary of windowing:

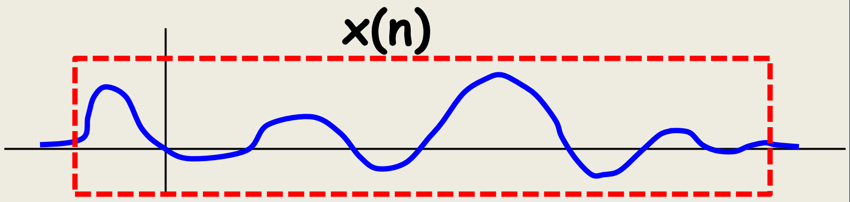
1. If we encounter an unknown signal x(n)



EITF75 Systems and Signals

Summary of windowing:

1. If we encounter an unknown signal x(n)
2. There may be edge effects (since we likely removed the tails)



EITF75 Systems and Signals

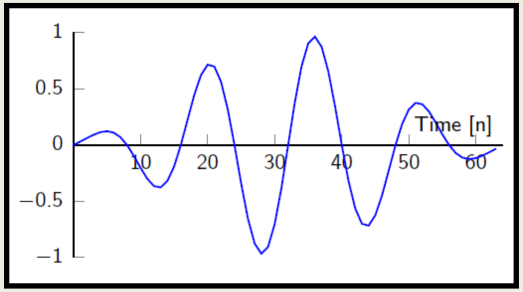
Summary of windowing:

1. If we encounter an unknown signal x(n)
2. There may be edge effects (since we likely removed the tails)
3. We can reduce the effects of edge effects on DFT

EITF75 Systems and Signals

Summary of windowing:

1. If we encounter an unknown signal x(n)
2. There may be edge effects (since we likely removed the tails)
3. We can reduce the effects of edge effects on DFT
4. By windowing with a Hamming window



EITF75 Systems and Signals

Application of DFT in 5G: Orthogonal Frequency division multiplexing

Data signal $x(n) = \ldots +1\ -1\ -1\ +1\ +1\ -1\ \ldots$ (very long)

Channel $h(n) = h(0)\ h(1)\ h(2) \qquad h(M-1)$ (quite long, say 50 taps)

Observation $y(n) = x(n) * h(n) + w(n)$ (w(n) noise, Gaussian distributed)

EITF75 Systems and Signals

Application of DFT in 5G: Orthogonal Frequency division multiplexing

Data signal $x(n) = \ldots\ +1\ \ -1\ \ -1\ \ +1\ \ +1\ \ -1\ \ldots$ (very long)

Channel $h(n) = h(0)\ h(1)\ h(2) \qquad h(M-1)$ (quite long, say 50 taps)

Observation $y(n) = x(n) * h(n) + w(n)$ (w(n) noise, Gaussian distributed)

Problem: get back x(n) from y(n)

EITF75 Systems and Signals

Application of DFT in 5G: Orthogonal Frequency division multiplexing

Data signal $x(n)$ = … +1 -1 -1 +1 +1 -1 … (very long)

Channel $h(n) = h(0)\ h(1)\ h(2) \quad h(M-1)$ (quite long, say 50 taps)

Observation $y(n) = x(n) * h(n) + w(n)$ (w(n) noise, Gaussian distributed)

Problem: get back $x(n)$ from $y(n)$

Naive solution: $Y(f) = X(f)H(f) + W(f)$ DTFT

EITF75 Systems and Signals

Application of DFT in 5G: Orthogonal Frequency division multiplexing

Data signal $x(n) = \ldots +1 \; -1 \; -1 \; +1 \; +1 \; -1 \; \ldots$ (very long)

Channel $h(n) = h(0)\; h(1)\; h(2) \qquad h(M-1)$ (quite long, say 50 taps)

Observation $y(n) = x(n) * h(n) + w(n)$ (w(n) noise, Gaussian distributed)

Problem: get back $x(n)$ from $y(n)$

Naive solution: $Y(f) = X(f)H(f) + W(f)$ DTFT

$$Z(f) = Y(f)/H(f) = X(f) + W(f)/H(f)$$

EITF75 Systems and Signals

Application of DFT in 5G: Orthogonal Frequency division multiplexing

Data signal $x(n) = \ldots +1 \ -1 \ -1 \ +1 \ +1 \ -1 \ \ldots$ (very long)

Channel $h(n) = h(0)\ h(1)\ h(2) \qquad h(M-1)$ (quite long, say 50 taps)

Observation $y(n) = x(n) * h(n) + w(n)$ ($w(n)$ noise, Gaussian distributed)

Problem: get back $x(n)$ from $y(n)$

Naive solution: $Y(f) = X(f)H(f) + W(f)$ DTFT

$Z(f) = Y(f)/H(f) = X(f) + W(f)/H(f) = X(f) + E(f)$

$z(n) = x(n) + e(n)$ IDTFT

EITF75 Systems and Signals

Application of DFT in 5G: Orthogonal Frequency division multiplexing

Data signal $x(n)$ = … +1 -1 -1 +1 +1 -1 … (very long)

Channel $h(n) = h(0)\ h(1)\ h(2) \qquad h(M-1)$ (quite long, say 50 taps)

Observation $y(n) = x(n) * h(n) + w(n)$ (w(n) noise, Gaussian distributed)

Problem: get back $x(n)$ from $y(n)$

Naive solution: $Y(f) = X(f)H(f) + W(f)$ DTFT

$$Z(f) = Y(f)/H(f) = X(f) + W(f)/H(f) = X(f) + E(f)$$

$z(n) = x(n) + e(n)$ IDTFT

$$\hat{x}(n) = \text{Threshold}[\ z(n)\ ]$$

EITF75 Systems and Signals

Application of DFT in 5G: Orthogonal Frequency division multiplexing

Data signal $x(n) = \ldots\ +1\ \ -1\ \ -1\ \ +1\ \ +1\ \ -1\ \ \ldots$

Channel $h(n) = h(0)\ h(1)\ h(2)\ \qquad h(M-1)$

Observation $y(n) = x(n) * h(n) + w(n)$

Problem: get back $x(n)$ from $y(n)$

Naive solution: $Y(f) = X(f)H(f) + W(f)$

$$Z(f) = Y(f)/H(f) = X(f) + E(f)$$

$$z(n) = x(n) + e(n)$$

$$\hat{x}(n) = \text{Threshold}[\ z(n)\ ]$$

Problems

DTFT hard to use

Remedy: use overlap-add

EITF75 Systems and Signals

Application of DFT in 5G: Orthogonal Frequency division multiplexing

Data signal $x(n) = \dots +1 \ -1 \ -1 \ +1 \ +1 \ -1 \ \dots$

Channel $h(n) = h(0)\ h(1)\ h(2) \qquad h(M-1)$

Observation $y(n) = x(n) * h(n) + w(n)$

Problem: get back $x(n)$ from $y(n)$

Naive solution: $Y(f) = X(f)H(f) + W(f)$

$Z(f) = Y(f)/H(f) = X(f) + E(f)$

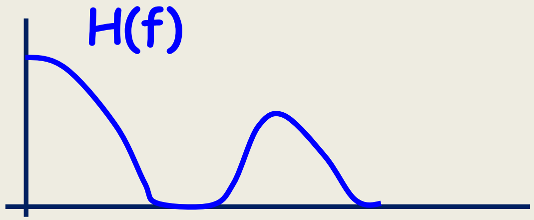
$z(n) = x(n) + e(n)$

$\hat{x}(n) = \text{Threshold}[\ z(n)\ ]$

Problems

DTFT hard to use

Remedy: use overlap-add



EITF75 Systems and Signals

Application of DFT in 5G: Orthogonal Frequency division multiplexing

Data signal $x(n)$ = … +1 -1 -1 +1 +1 -1 …

Channel $h(n) = h(0)\ h(1)\ h(2) \qquad h(M-1)$

Observation $y(n) = x(n) * h(n) + w(n)$

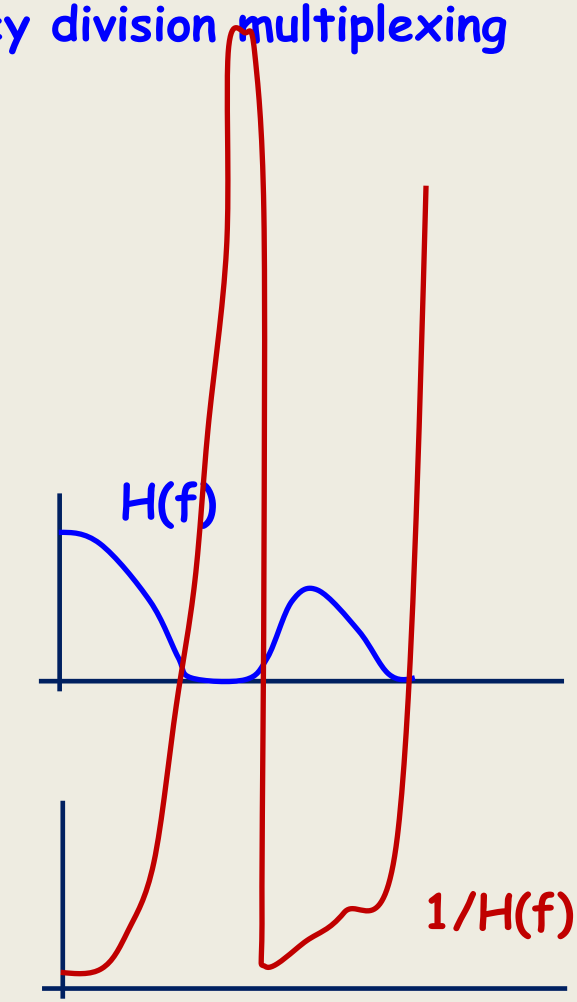
Problem: get back $x(n)$ from $y(n)$

Naive solution: $Y(f) = X(f)H(f) + W(f)$

$Z(f) = Y(f)/H(f) = X(f) + E(f)$

$z(n) = x(n) + e(n)$

$\hat{x}(n) = \text{Threshold}[\ z(n)\ ]$



EITF75 Systems and Signals

Application of DFT in 5G: Orthogonal Frequency division multiplexing

Data signal $x(n) = \dots +1 \ -1 \ -1 \ +1 \ +1 \ -1 \ \dots$

Channel $h(n) = h(0)\ h(1)\ h(2) \quad\quad h(M-1)$

Observation $y(n) = x(n) * h(n) + w(n)$

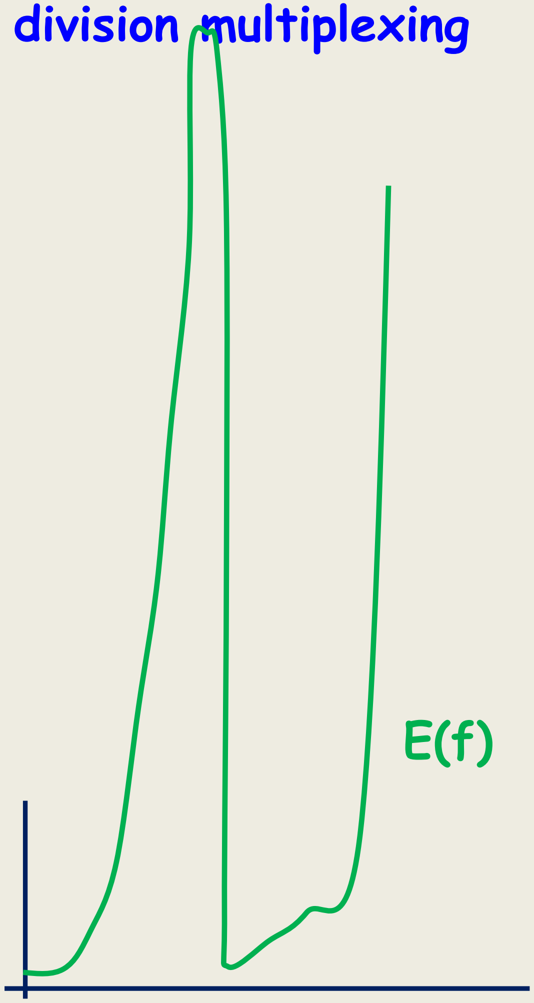
Problem: get back $x(n)$ from $y(n)$

Naive solution: $Y(f) = X(f)H(f) + W(f)$

$Z(f) = Y(f)/H(f) = X(f) + E(f)$

$z(n) = x(n) + e(n)$

$\hat{x}(n) = \text{Threshold}[\, z(n)\,]$



EITF75 Systems and Signals

Application of DFT in 5G: Orthogonal Frequency division multiplexing

Data signal $x(n) = \ldots +1 \;\; -1 \;\; -1 \;\; +1 \;\; +1 \;\; -1 \;\; \ldots$

Channel $h(n) = h(0)\; h(1)\; h(2) \qquad h(M-1)$

Observation $y(n) = x(n) * h(n) + w(n)$

Problem: get back $x(n)$ from $y(n)$

Naive solution: $Y(f) = X(f)H(f) + W(f)$

$$Z(f) = Y(f)/H(f) = X(f) + E(f)$$

$$z(n) = x(n) + e(n)$$

$$\hat{x}(n) = \text{Threshold}[\, z(n) \,]$$

EITF75 Systems and Signals

Application of DFT in 5G: Orthogonal Frequency division multiplexing

Data signal $x(n) = \ldots +1 \ -1 \ -1 \ +1 \ +1 \ -1 \ \ldots$

Channel $h(n) = h(0)\ h(1)\ h(2) \qquad h(M-1)$

Observation $y(n) = x(n) * h(n) + w(n)$

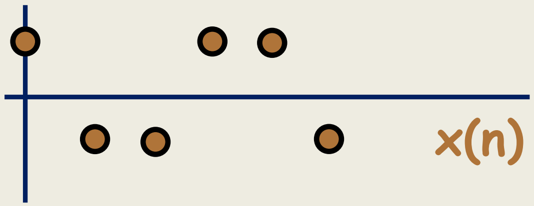
Problem: get back $x(n)$ from $y(n)$

Naive solution: $Y(f) = X(f)H(f) + W(f)$

$Z(f) = Y(f)/H(f) = X(f) + E(f)$

$z(n) = x(n) + e(n)$

$\hat{x}(n) = \text{Threshold}[\ z(n)\ ]$



EITF75 Systems and Signals

Application of DFT in 5G: Orthogonal Frequency division multiplexing

Data signal $x(n)$ = … +1 -1 -1 +1 +1 -1 …

Channel $h(n) = h(0)\ h(1)\ h(2) \quad h(M-1)$

Observation $y(n) = x(n) * h(n) + w(n)$

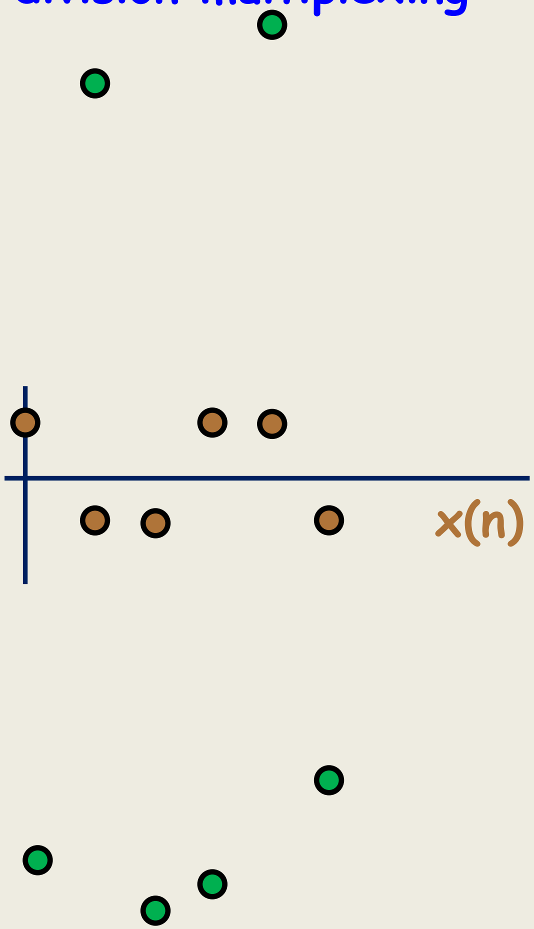
Problem: get back $x(n)$ from $y(n)$

Naive solution: $Y(f) = X(f)H(f) + W(f)$

$Z(f) = Y(f)/H(f) = X(f) + E(f)$

$z(n) = x(n) + e(n)$

$\hat{x}(n) = \text{Threshold}[\ z(n)\ ]$



EITF75 Systems and Signals

Application of DFT in 5G: Orthogonal Frequency division multiplexing

Data signal $x(n) = \ldots +1 \ -1 \ -1 \ +1 \ +1 \ -1 \ \ldots$

Channel $h(n) = h(0)\ h(1)\ h(2) \qquad h(M-1)$

Observation $y(n) = x(n) * h(n) + w(n)$

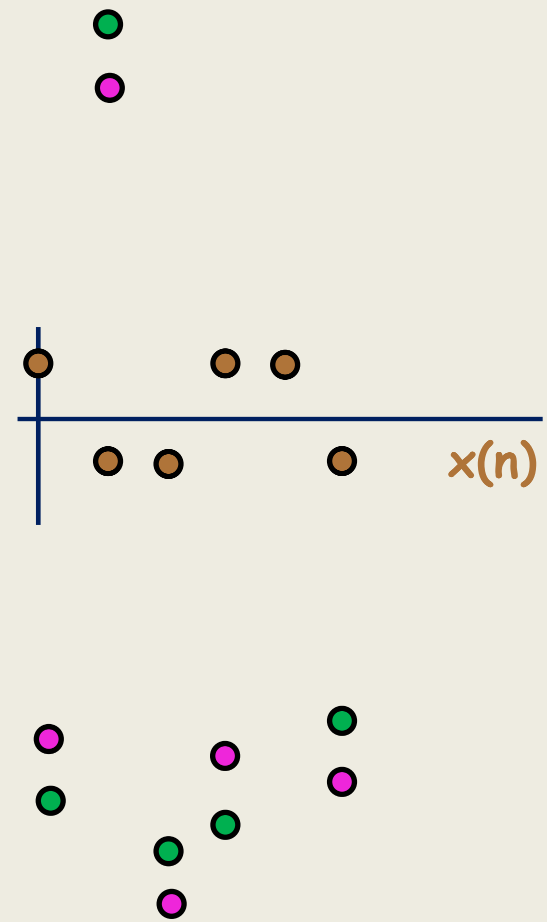
Problem: get back $x(n)$ from $y(n)$

Naive solution: $Y(f) = X(f)H(f) + W(f)$

$Z(f) = Y(f)/H(f) = X(f) + E(f)$

random $\longrightarrow$ $z(n) = x(n) + e(n)$

$\hat{x}(n) = \text{Threshold}[\, z(n) \,]$



EITF75 Systems and Signals

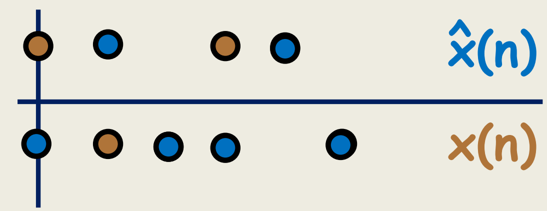
Application of DFT in 5G: Orthogonal Frequency division multiplexing

Data signal $x(n)$ = … +1 -1 -1 +1 +1 -1 …

Channel $h(n) = h(0)\ h(1)\ h(2) \qquad h(M-1)$

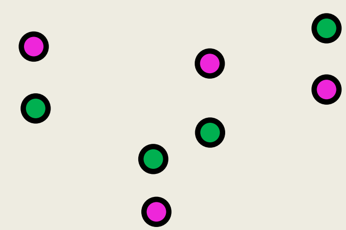
Observation $y(n) = x(n) * h(n) + w(n)$

Problem: get back $x(n)$ from $y(n)$



Naive solution: $Y(f) = X(f)H(f) + W(f)$

$$Z(f) = Y(f)/H(f) = X(f) + E(f)$$



random ⟶ $z(n) = x(n) + e(n)$

50% error ⟶ $\hat{x}(n) = \text{Threshold}[\ z(n)\ ]$

EITF75 Systems and Signals

Application of DFT in 5G: Orthogonal Frequency division multiplexing

Orthogonal Frequency division multiplexing solution

Data signal $x(n)$ = … +1 -1 -1 +1 +1 -1 …

$x_1(n)$	$x_2(n)$	$x_3(n)$

Block the data signal. Block size L

EITF75 Systems and Signals

Application of DFT in 5G: Orthogonal Frequency division multiplexing

Orthogonal Frequency division multiplexing solution

Data signal $x(n)$ = … +1 -1 -1 +1 +1 -1 …

$x_1(n)$	$x_2(n)$	$x_3(n)$

Block the data signal. Block size L

Apply IDFT to each block

$a_1(n)$	$a_2(n)$	$a_3(n)$

$a_1(n) = \text{IDFT}(x_1(n))$

$a_2(n) = \text{IDFT}(x_2(n))$

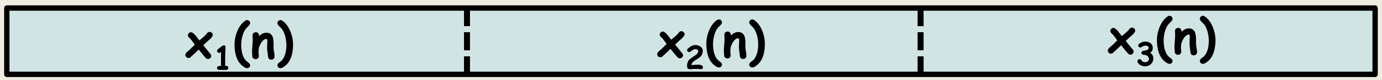
$a_3(n) = \text{IDFT}(x_3(n))$

EITF75 Systems and Signals

Application of DFT in 5G: Orthogonal Frequency division multiplexing

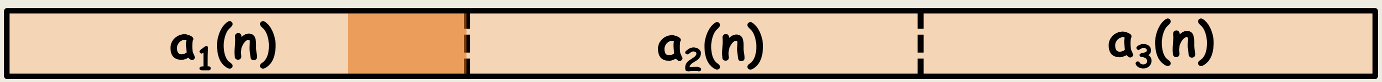
Orthogonal Frequency division multiplexing solution

Data signal $x(n)$ = … +1 -1 -1 +1 +1 -1 …



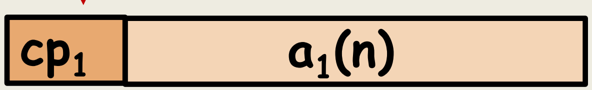
Block the data signal. Block size L

Apply IDFT to each block



$a_k(n) = \text{IDFT}(x_k(n))$

Copy tail and append at front (cyclic prefix)

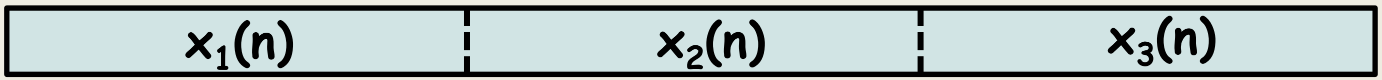


EITF75 Systems and Signals

Application of DFT in 5G: Orthogonal Frequency division multiplexing

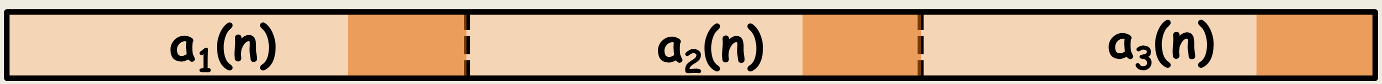
Orthogonal Frequency division multiplexing solution

Data signal $x(n)$ = … +1 -1 -1 +1 +1 -1 …



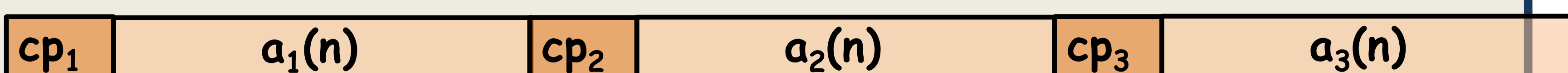
Block the data signal. Block size L

Apply IDFT to each block



$a_k(n)$ = IDFT($x_k(n)$)

Repeat for all blocks



EITF75 Systems and Signals

Application of DFT in 5G: Orthogonal Frequency division multiplexing

Orthogonal Frequency division multiplexing solution

Data signal $x(n)$ = … +1 -1 -1 +1 +1 -1 …

$x_1(n)$	$x_2(n)$	$x_3(n)$

Block the data signal. Block size L

Apply IDFT to each block

$a_1(n)$	$a_2(n)$	$a_3(n)$

$a_k(n)$ = IDFT($x_k(n)$)

Transmit over channel $h(n)$

cp_1	$a_1(n)$	cp_2	$a_2(n)$	cp_3	$a_3(n)$

EITF75 Systems and Signals

Application of DFT in 5G: Orthogonal Frequency division multiplexing

Orthogonal Frequency division multiplexing solution

Data signal $x(n)$ = … +1 -1 -1 +1 +1 -1 …

$x_1(n)$	$x_2(n)$	$x_3(n)$

Block the data signal. Block size L

Apply IDFT to each block

$a_1(n)$	$a_2(n)$	$a_3(n)$

$a_k(n) = \mathrm{IDFT}(x_k(n))$

Transmit over channel $h(n)$

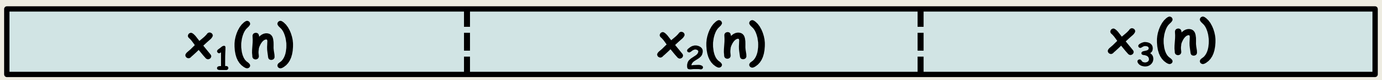
cp_1	$a_1(n)$	cp_2	$a_2(n)$	cp_3	$a_3(n)$
ycp_1	$y_1(n)$	ycp_2	$y_2(n)$	ycp_3	$y_3(n)$

EITF75 Systems and Signals

Application of DFT in 5G: Orthogonal Frequency division multiplexing

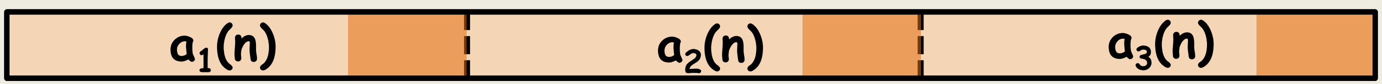
Orthogonal Frequency division multiplexing solution

Data signal $x(n)$ = … +1 -1 -1 +1 +1 -1 …



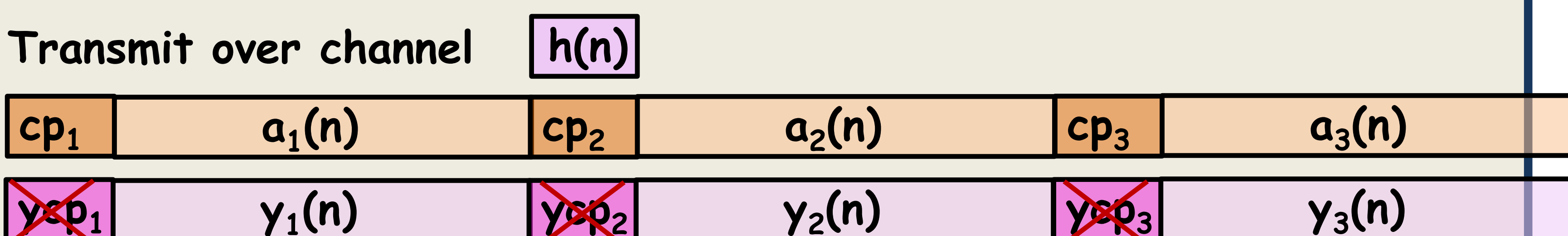
Block the data signal. Block size L

Apply IDFT to each block



$a_k(n) = \mathrm{IDFT}(x_k(n))$

Transmit over channel



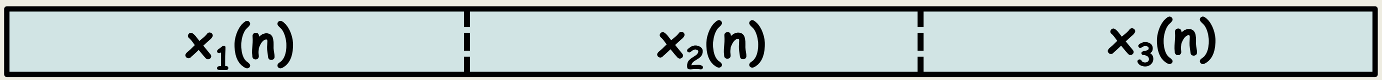
Throw away

EITF75 Systems and Signals

Application of DFT in 5G: Orthogonal Frequency division multiplexing

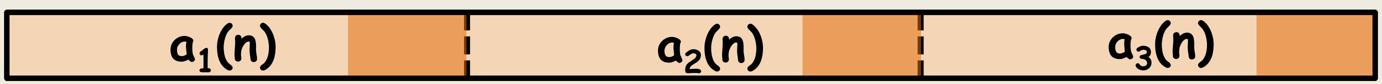
Orthogonal Frequency division multiplexing solution

Data signal $x(n)$ = … +1 -1 -1 +1 +1 -1 …



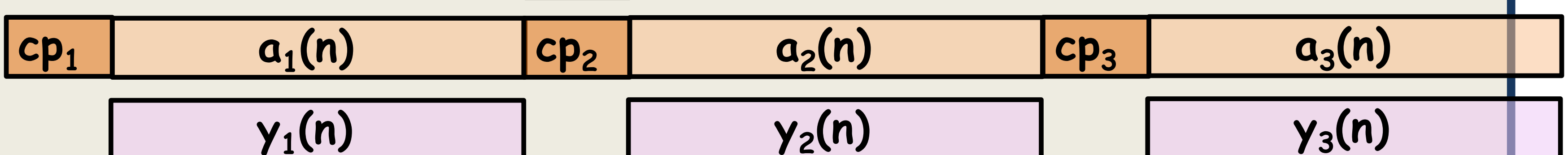
Block the data signal. Block size L

Apply IDFT to each block



$a_k(n)$ = IDFT($x_k(n)$)

Transmit over channel $h(n)$



EITF75 Systems and Signals

Application of DFT in 5G: Orthogonal Frequency division multiplexing

Orthogonal Frequency division multiplexing solution

Data signal $x(n) = \ldots +1 \ -1 \ -1 \ +1 \ +1 \ -1 \ \ldots$

$x_1(n)$ | $x_2(n)$ | $x_3(n)$

Block the data signal. Block size L

Apply IDFT to each block

$a_1(n)$ | $a_2(n)$ | $a_3(n)$

$a_k(n) = \mathrm{IDFT}(x_k(n))$

Transmit over channel $h(n)$

cp_1 | $a_1(n)$ | cp_2 | $a_2(n)$ | cp_3 | $a_3(n)$

$y_1(n)$ | $y_2(n)$ | $y_3(n)$

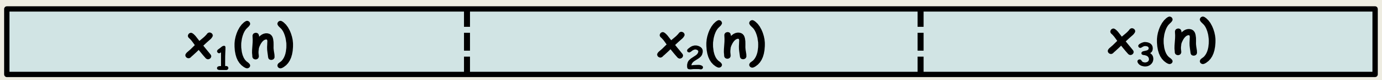
Math formula for $y_1(n)$?

EITF75 Systems and Signals

Application of DFT in 5G: Orthogonal Frequency division multiplexing

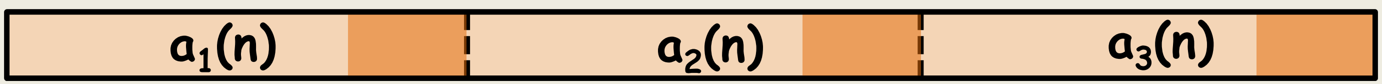
Orthogonal Frequency division multiplexing solution

Data signal $x(n)$ = … +1 -1 -1 +1 +1 -1 …



Block the data signal. Block size L

Apply IDFT to each block



$a_k(n) = \mathrm{IDFT}(x_k(n))$

Transmit over channel $h(n)$

Math formula for $y_1(n)$? $y_1(n) = a_1(n) \otimes h(n) + w_1(n)$

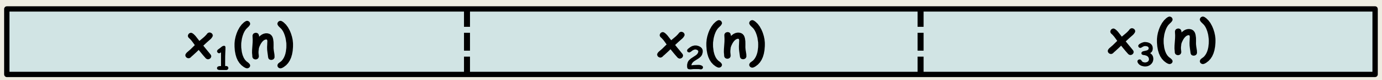
Due to CP

EITF75 Systems and Signals

Application of DFT in 5G: Orthogonal Frequency division multiplexing

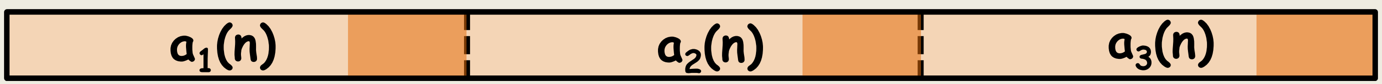
Orthogonal Frequency division multiplexing solution

Data signal $x(n)$ = … +1 -1 -1 +1 +1 -1 …



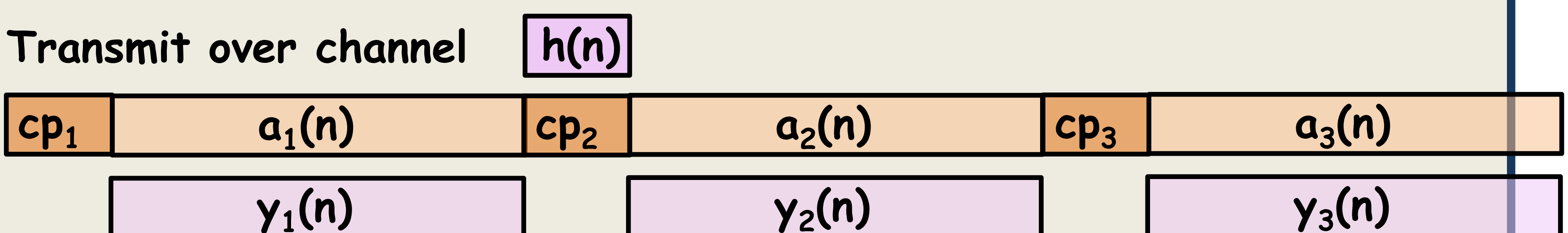
Block the data signal. Block size L

Apply IDFT to each block



$a_k(n)$ = IDFT($x_k(n)$)

Transmit over channel



Math formula for $y_1(n)$? $\quad y_1(n) = a_1(n) \otimes h(n) + w_1(n)$

Apply DFT $\quad Y_1(k) = x_1(k)\, H(k) + W_1(k)$

EITF75 Systems and Signals

Application of DFT in 5G: Orthogonal Frequency division multiplexing

Orthogonal Frequency division multiplexing solution

Data signal $x(n)$ = … +1 -1 -1 +1 +1 -1 …

$x_1(n)$ | $x_2(n)$ | $x_3(n)$

Block the data signal. Block size L

Apply IDFT to each block

$a_1(n)$ | $a_2(n)$ | $a_3(n)$

$a_k(n) = \mathrm{IDFT}(x_k(n))$

Transmit over channel $h(n)$

cp_1 | $a_1(n)$ | cp_2 | $a_2(n)$ | cp_3 | $a_3(n)$

$y_1(n)$ | $y_2(n)$ | $y_3(n)$

Math formula for $y_1(n)$? $y_1(n) = a_1(n) \otimes h(n) + w_1(n)$

Apply DFT $Z_1(k) = x_1(k) + W_1(k)/H(k)$

EITF75 Systems and Signals

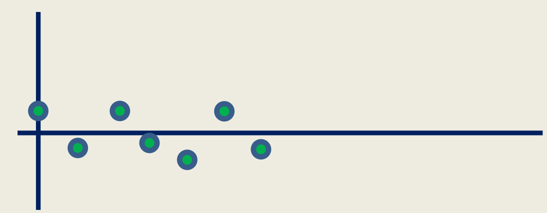
Application of DFT in 5G: Orthogonal Frequency division multiplexing

Orthogonal Frequency division multiplexing solution

Data signal $x(n)$ = … +1 -1 -1 +1 +1 -1 …

$x_1(n)$

$W_1(k)$



Transmit over channel $h(n)$

cp_1	$a_1(n)$	cp_2	$a_2(n)$	cp_3	$a_3(n)$
	$y_1(n)$		$y_2(n)$		$y_3(n)$

Math formula for $y_1(n)$? $\quad y_1(n) = a_1(n) \otimes h(n) + w_1(n)$

Apply DFT $\quad Z_1(k) = x_1(k) + W_1(k)/H(k)$

EITF75 Systems and Signals

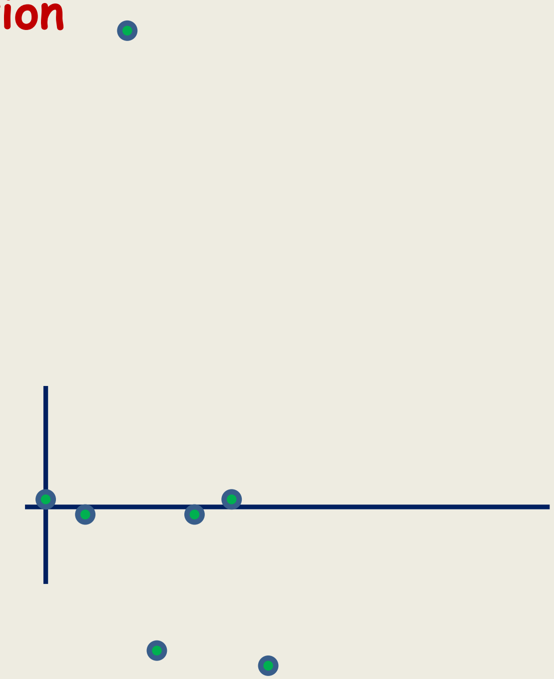
Application of DFT in 5G: Orthogonal Frequency division multiplexing

Orthogonal Frequency division multiplexing solution

Data signal $x(n)$ = … +1 -1 -1 +1 +1 -1 …

$x_1(n)$

$W_1(k)/H(k)$



Transmit over channel $h(n)$

cp_1	$a_1(n)$	cp_2	$a_2(n)$	cp_3	$a_3(n)$
	$y_1(n)$		$y_2(n)$		$y_3(n)$

Math formula for $y_1(n)$? $y_1(n) = a_1(n) \otimes h(n) + w_1(n)$

Apply DFT $Z_1(k) = x_1(k) + W_1(k)/H(k)$

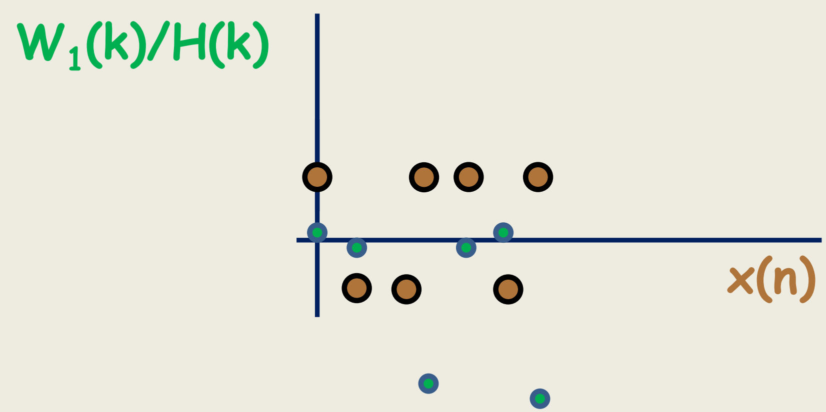
EITF75 Systems and Signals

Application of DFT in 5G: Orthogonal Frequency division multiplexing

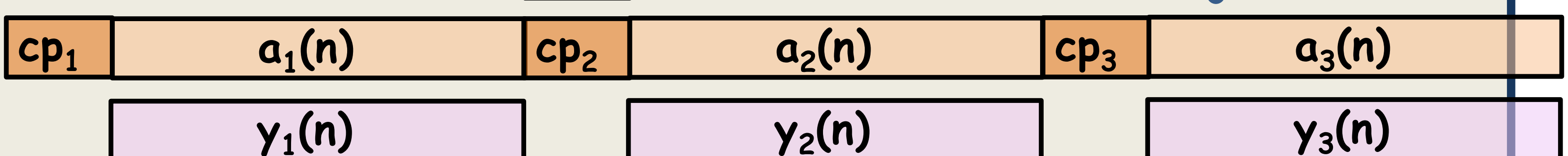
Orthogonal Frequency division multiplexing solution

Data signal $x(n)$ = … +1 -1 -1 +1 +1 -1 …

$x_1(n)$



Transmit over channel $h(n)$



Math formula for $y_1(n)$? $y_1(n) = a_1(n) \otimes h(n) + w_1(n)$

Apply DFT $Z_1(k) = x_1(k) + W_1(k)/H(k)$

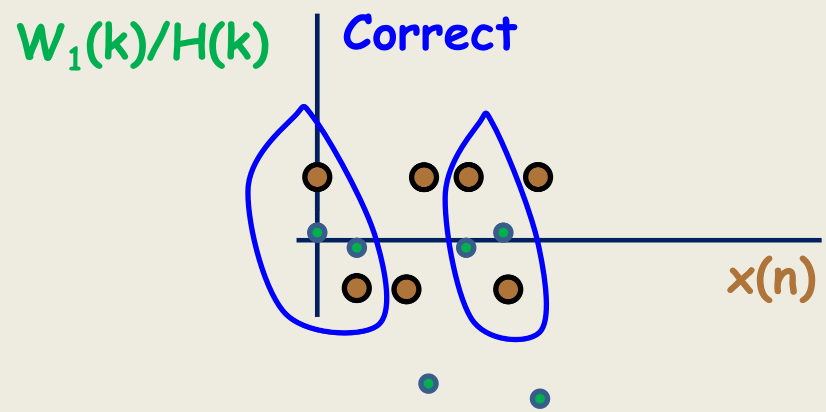
EITF75 Systems and Signals

Application of DFT in 5G: Orthogonal Frequency division multiplexing

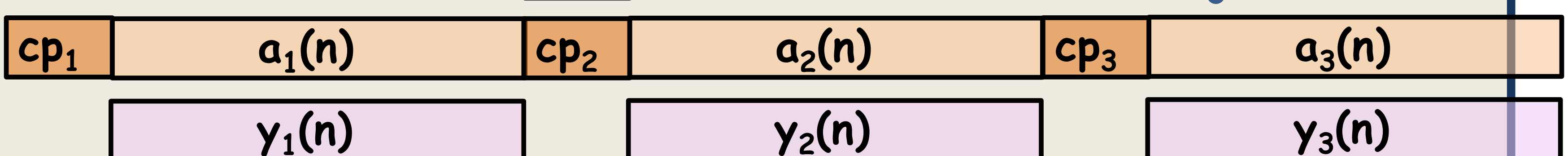
Orthogonal Frequency division multiplexing solution

Data signal $x(n)$ = … +1 -1 -1 +1 +1 -1 …

$x_1(n)$



Transmit over channel $h(n)$



Math formula for $y_1(n)$? $y_1(n) = a_1(n) \otimes h(n) + w_1(n)$

Apply DFT $Z_1(k) = x_1(k) + W_1(k)/H(k)$

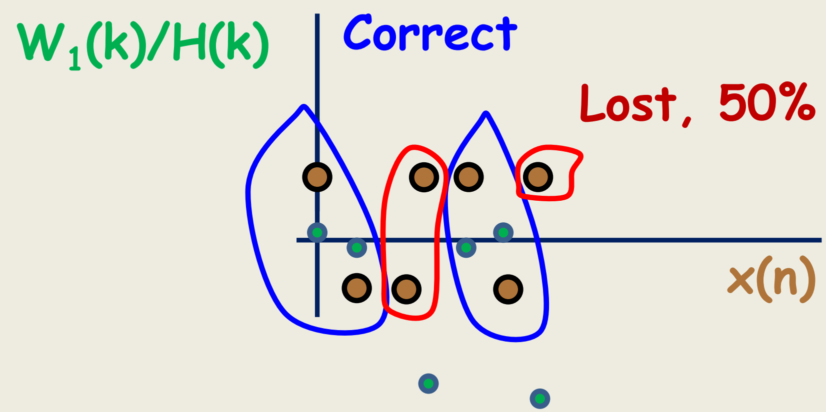
EITF75 Systems and Signals

Application of DFT in 5G: Orthogonal Frequency division multiplexing

Orthogonal Frequency division multiplexing solution

Data signal $x(n)$ = … +1 -1 -1 +1 +1 -1 …

$x_1(n)$



Transmit over channel $h(n)$

cp_1	$a_1(n)$	cp_2	$a_2(n)$	cp_3	$a_3(n)$
	$y_1(n)$		$y_2(n)$		$y_3(n)$

Math formula for $y_1(n)$? $y_1(n) = a_1(n) \otimes h(n) + w_1(n)$

Apply DFT $Z_1(k) = x_1(k) + W_1(k)/H(k)$

EITF75 Systems and Signals

Application of DFT in 5G: Orthogonal Frequency division multiplexing

Orthogonal Frequency division multiplexing solution

Summary:

By inserting the CP (of length M = channel duration) and using DFT/IDFT, we can recover the data where the channel is good

The transmitter can avoid sending data at bad channel frequencies

FFT can be used for efficient implementation. 1024 - 4096 in 5G

CP is pure rate and power loss, but needed