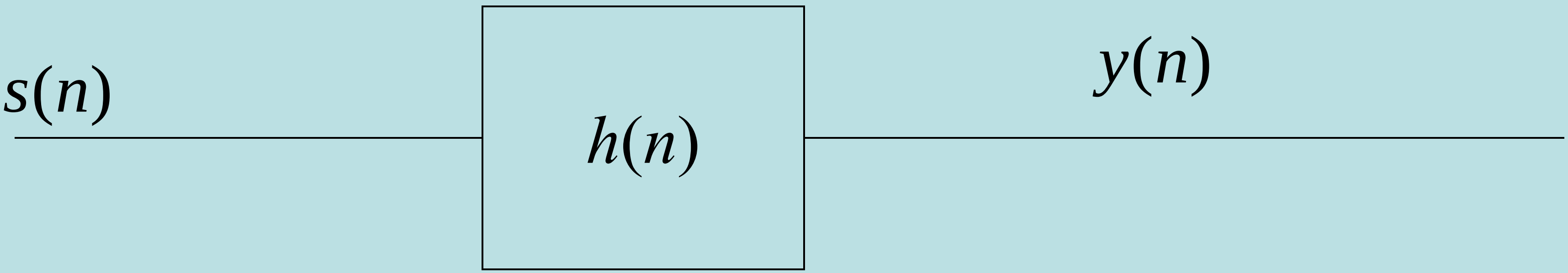
Motivation



$$y(n) = x(n) * h(n)$$

Fast convolution --- compute convolution using FFT

Outline

- Fast convolution of short sequences;
- Fast convolution of long sequences.

Convolution of short sequences

- *Let x*(*n*) be length *L*, (*n=0, 1, … L-1)*;
- *h*(*n*) be length *M, (n=0,1, …, M-1);*
- *y(n) should have L+M-1 samples, given by:*

$$y(n) = x(n) * h(n) = \sum_{m=0}^{N-1} h(m)x(n-m)$$

Where *n=0,1,…, N (N=L+M-1)*

This equation is referred to as linear convolution

Convolution of short sequences

- *Total computations (Assume M<L)*
  - *n=0*, 1 multiplication
  - *n*=1, *2* multiplications and 1 addition;
  - *n=2, 3* multiplications and 2 additions;
  - …
  - if *M-1<=n<=L*-1, *M* multiplications,…
  - ….
  - *n=L+M-2, 1* multiplication and no addition
  - **Hence, ML multiplications for convolving *x*(n) and *H*(n)**

Convolution of short sequences

- Let us see if DFT can be used for computing the convolution.
- As the length of *x(n)*,*h(n)* and *y(n)* are *L*,*M* and *(L+M-1)* respectively, we consider *N*-point DFTs of them, where *N>L+M-1*:

$$X(k) = \sum_{n=0}^{L-1} x(n) W_N^{nk}, \qquad H(k) = \sum_{n=0}^{M-1} h(n) W_N^{nk}$$

$$Y(k) = \sum_{n=0}^{L+M-1} y(n) W_N^{nk}$$

Convolution of short sequences

$$X(k)H(k) = \sum_{n=0}^{L-1} x(n)W_N^{nk} \sum_{m=0}^{M-1} h(m)W_N^{mk}$$

$$= \sum_{n=0}^{L-1}\sum_{m=0}^{M-1} x(n)h(m)W_N^{(n+m)k} \leftarrow let \qquad n+m=l$$

$$= \sum_{l=0}^{L+M-1}\sum_{m=0}^{M-1} x(l-m)h(m)W_N^{lk}$$

$$= \sum_{l=0}^{L+M-1} y(l)W_N^{lk} = Y(k)$$

y(*l*)

Convolution of short sequences

Hence convolution can be computed via DFT's:

- Step 1.

Compute N-point DFT of x(n) and h(n), where $N>L+M-1$

- Step 2.

Compute $Y(k)=X(k)H(k)$

- Step 3.

Compute N-point IDFT of Y(k) to get y(n)

Convolution of short sequences: Is it more efficient to use DFTs?

- Multiplications: (1/2)NlogN for each FFT and IFFT. Hence (3/2) NlogN +N complex multiplications are required; where N>=L+M-1
- The direct convolution involves ML real multiplications;
- Which one is more efficient? FFT is more efficient when L and M are large.
- For example: when L=M ……

Circular convolution

- Note that N must be bigger than L+M-1. Otherwise the result will not be correct. Why?

- Naturally multiplication in frequency domain is equivalent to circular convolution.

- If $N<L+M-1$, the circular convolution will involves overlaps .

Circular convolution

- Circular convolution of x(n) and h(n) is defined as the convolution of h(n) with a periodic signal $x_p(n)$:

$$y_p(n) = x_p(n) * h(n)$$

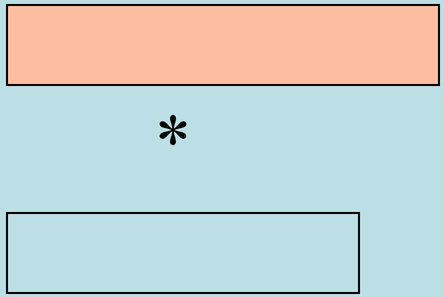
where

$$x_p(n) = x(n \bmod N), \qquad -\infty < n < \infty$$

Circular Convolution

x(n) length N

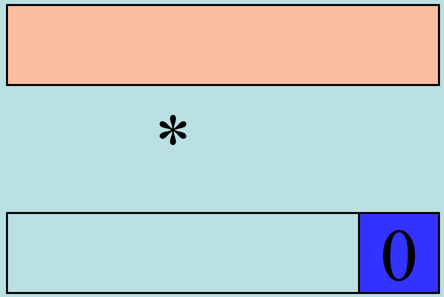
h(n) length M



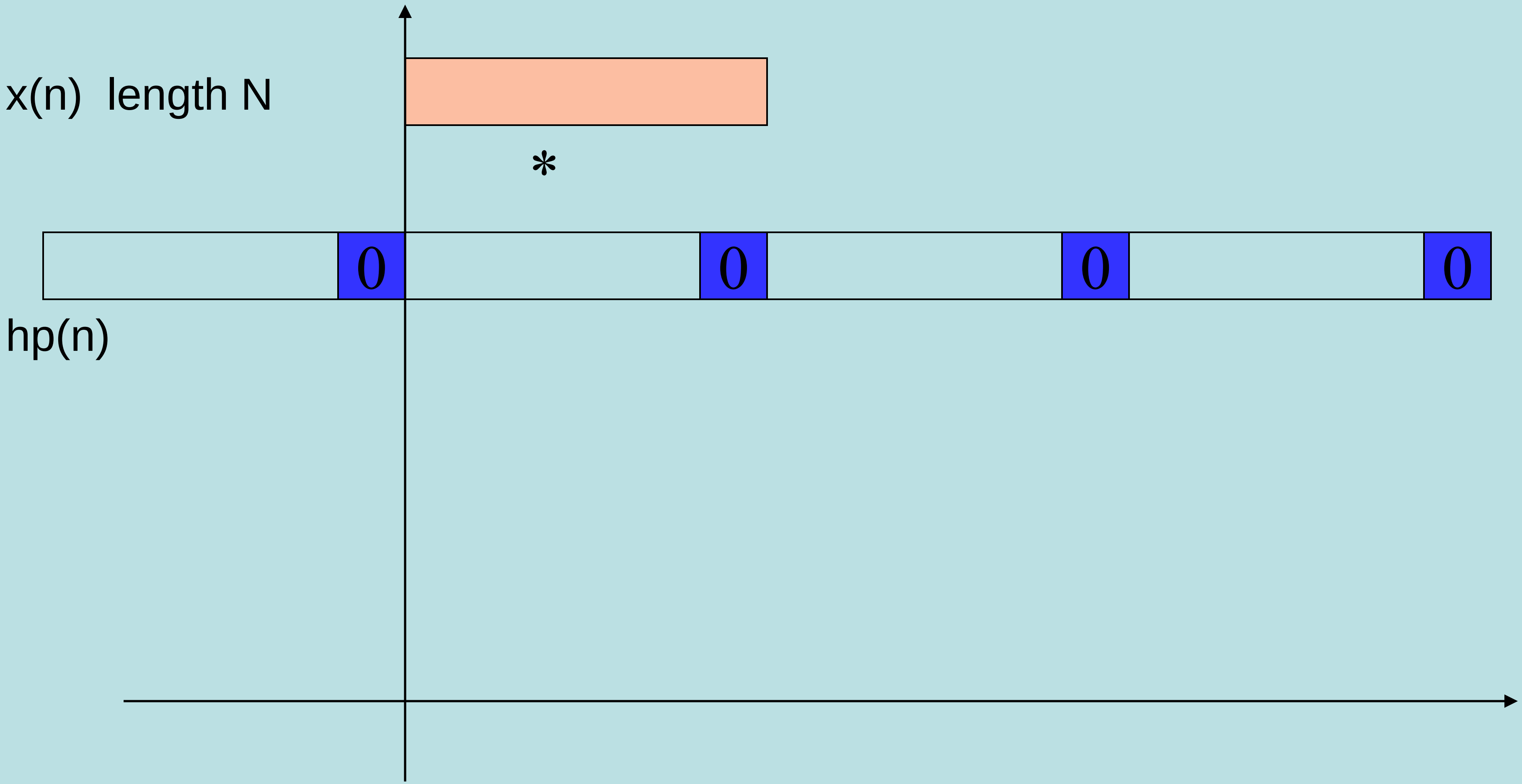
Circular Convolution

x(n) length N

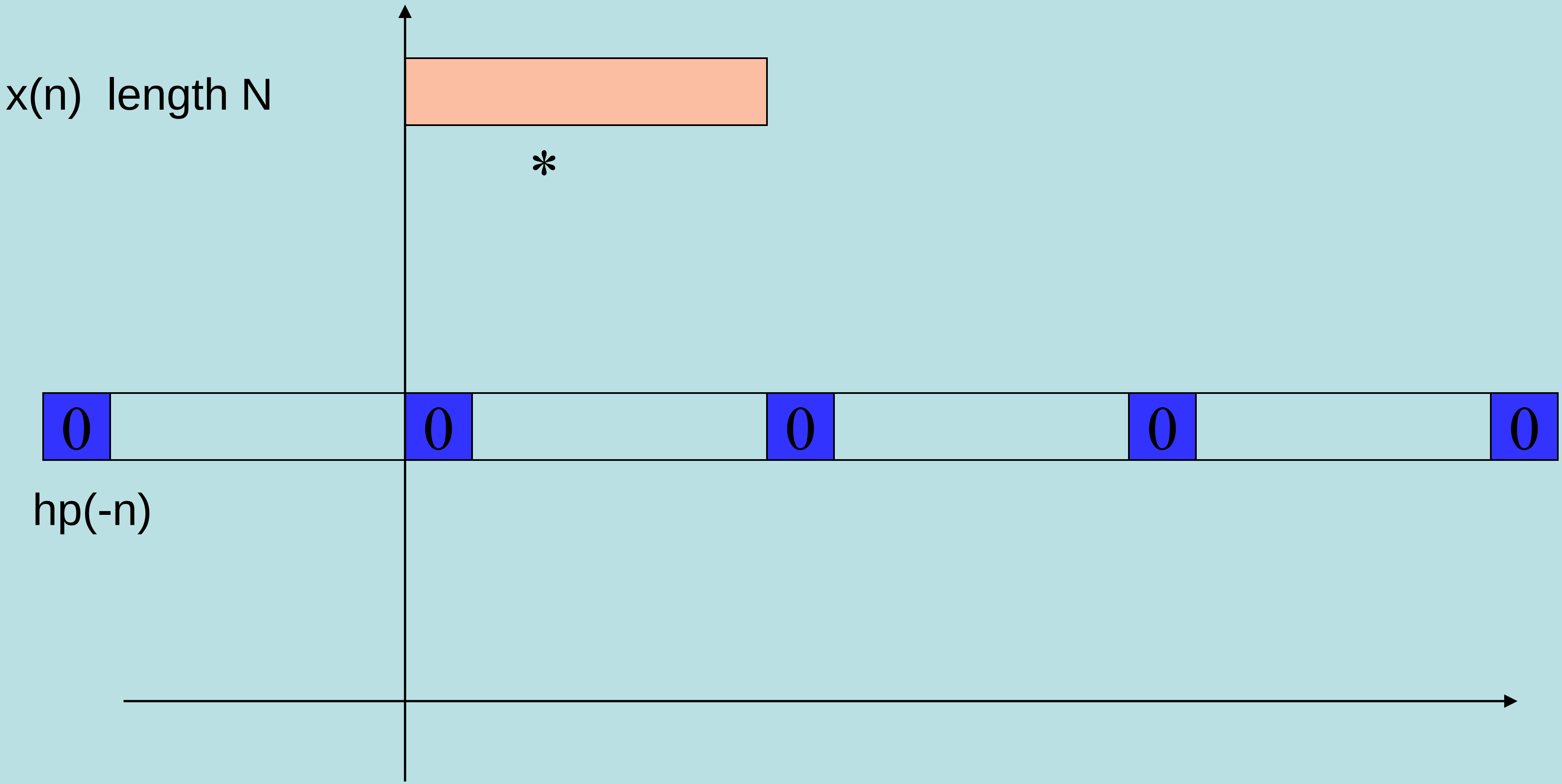
h(n) length M



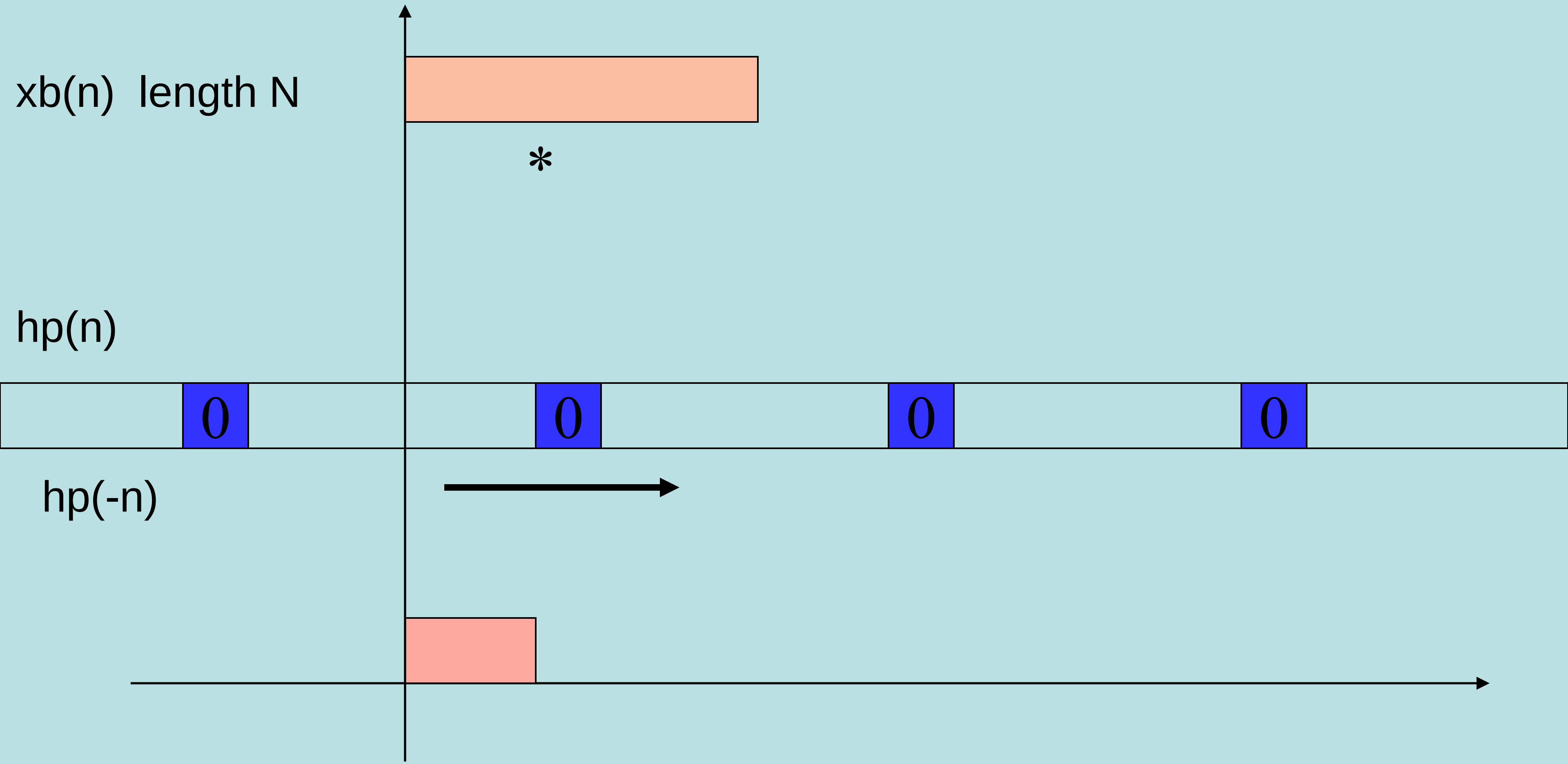
Circular Convolution



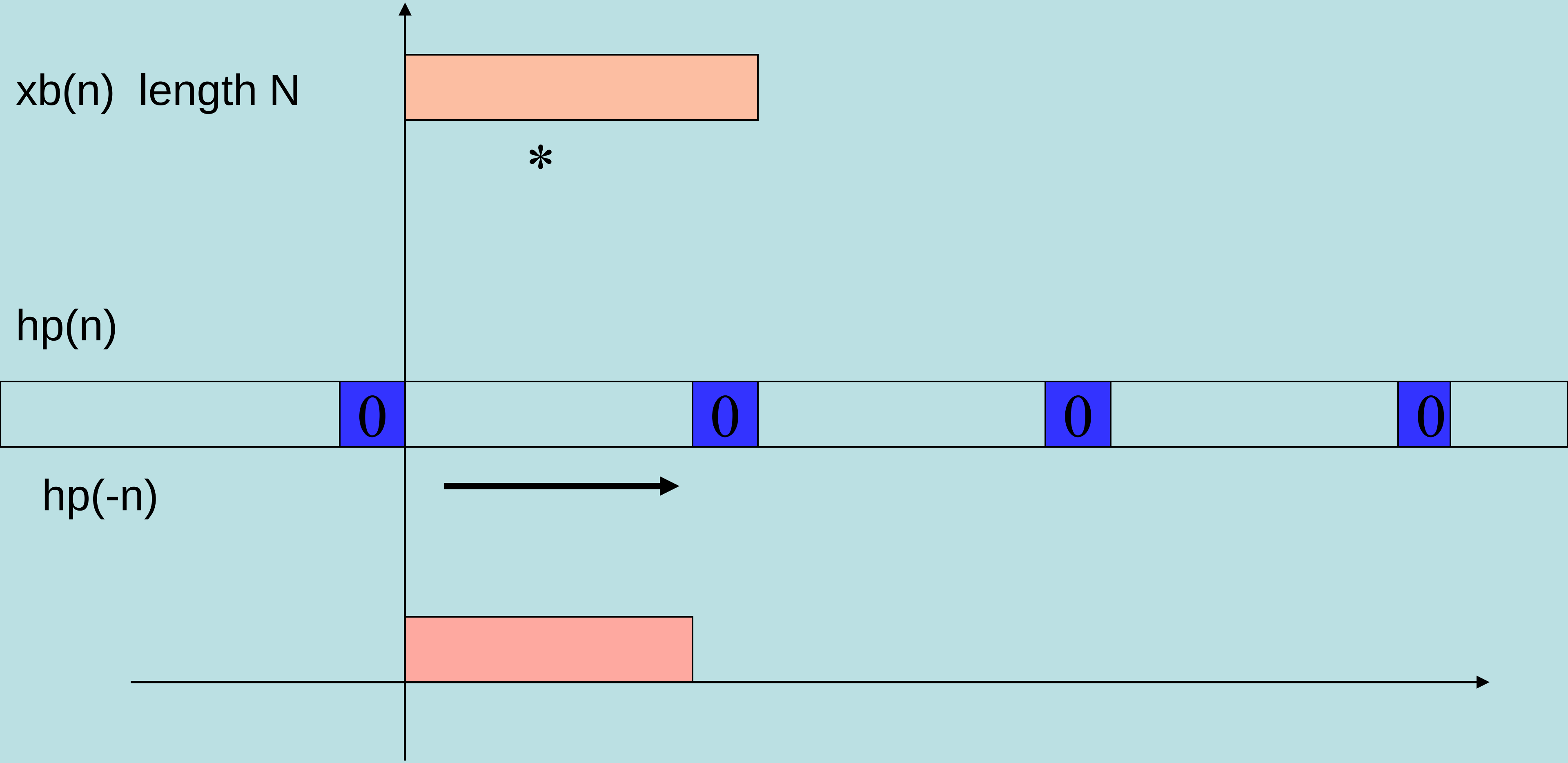
Circular Convolution



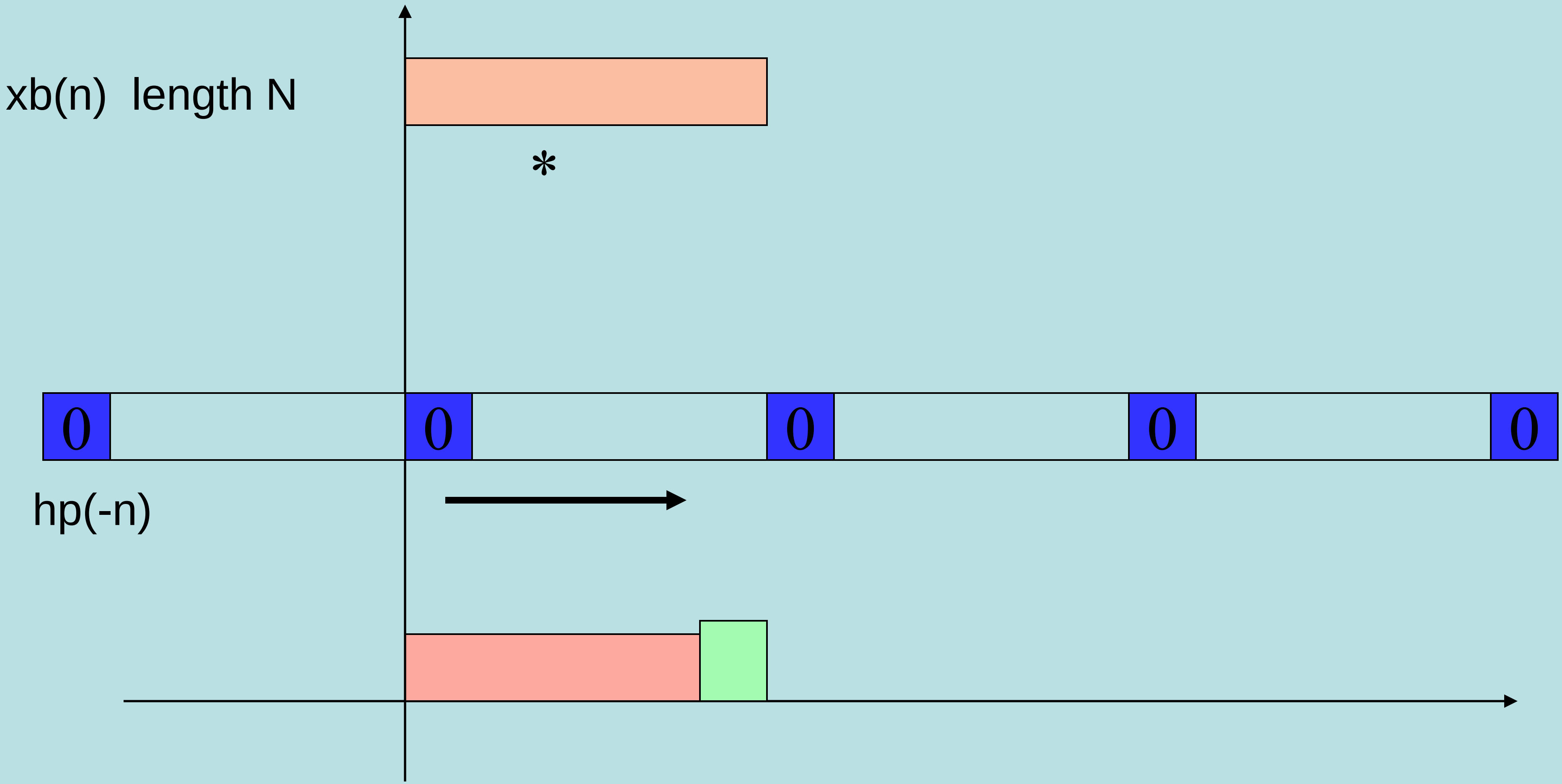
Circular Convolution



Circular Convolution



Circular Convolution



Examples

- Let {x(n)}={1,2,3} and {h(n)}={1,1,1}, then the result should be {y(n)}={1,3,6,5,3}
- With L=M=3, we should choose N=5
- however if we take N=4, the extended signals are
  - {x(n)}={1,2,3,0} and {h(n)}={1,1,1,0}
- The DFT yields
  - X(k)={6,-2-2j,2,-2+2j}
  - H(k)={3,-j,1,j}
  - Y(k)={18,-2+2j,2,-2-2j}
  - Hence y(n)={4,3,6,5}

Examples

$x_p(n)$= {…..3,0 1,2,3,0,1,2,3,0,1,2,3,0,1,2,3,0,1,2,3,0….}

---------------------------------------+-----------------------------→

$x_p(-n)$={….3,2,1,0,3,2,1,0, 3,2,1,0,3,2,1,0,3,2,1,0, ….}

{h(n)}={1,1,1},

y(n)= 4,3,6,5

Examples

If x(n)={1,2,3,0,0} ➔ 5 point DFT

h(n)={1,1,1,0,0} ➔ 5 point DFT

we can get y(n)={1,3,6,5,3}

{1,2,3,0,0,1,2,3,0,0,1,2,3,0,0,1,2,3,0,0,1,2,3,0,0}

{1,1,1}

x_p(n)={…..,2,3,0,0,1,2,3,0,0,1,2,3,0,0,1,2,3,0,0,1,2,3,0…}

---*----------------------------------→

x_p(-n)={…..0,0,3,2,1,0,0,3,2,1,0,0,3,2,1,0,0,3,2,1,0,0,…}

{h(n)}= {1,1,1},

y(n)=1,3,6,5,3

Convolution of long sequences

Convolution of long sequences

Convolution of long sequences

Convolution of long sequences

Convolution of long sequences

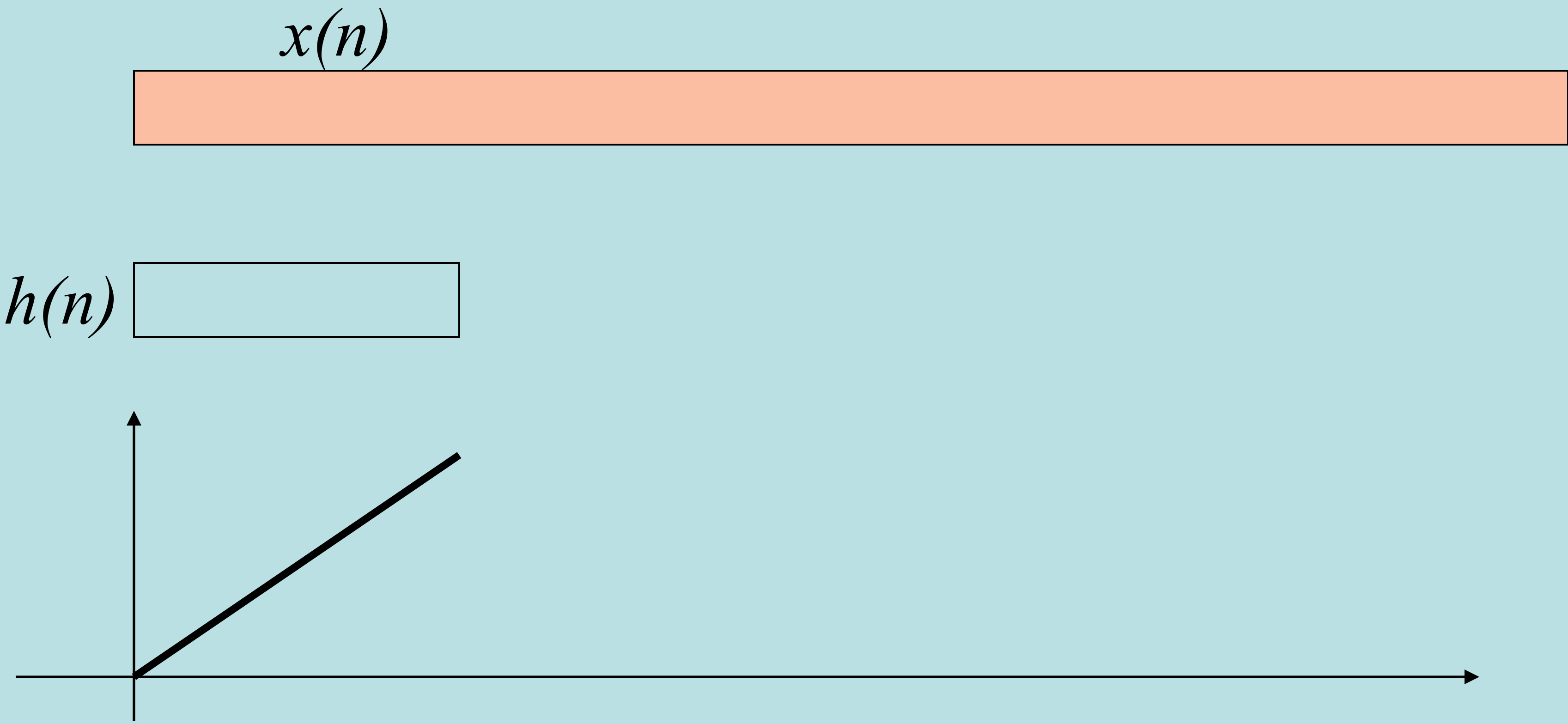
Convolution of long sequences

Convolution of long sequences

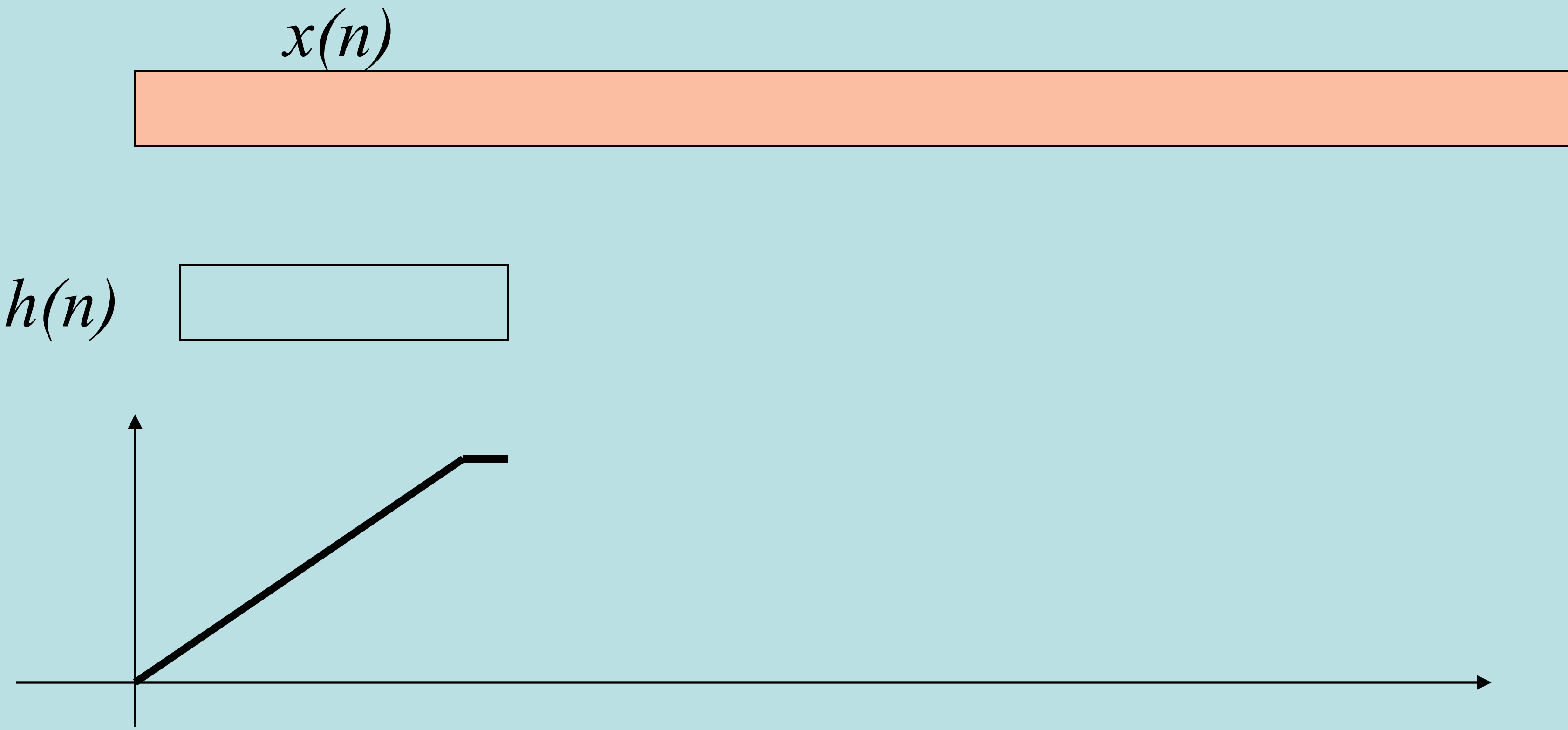
Convolution of long sequences

Convolution of long sequences

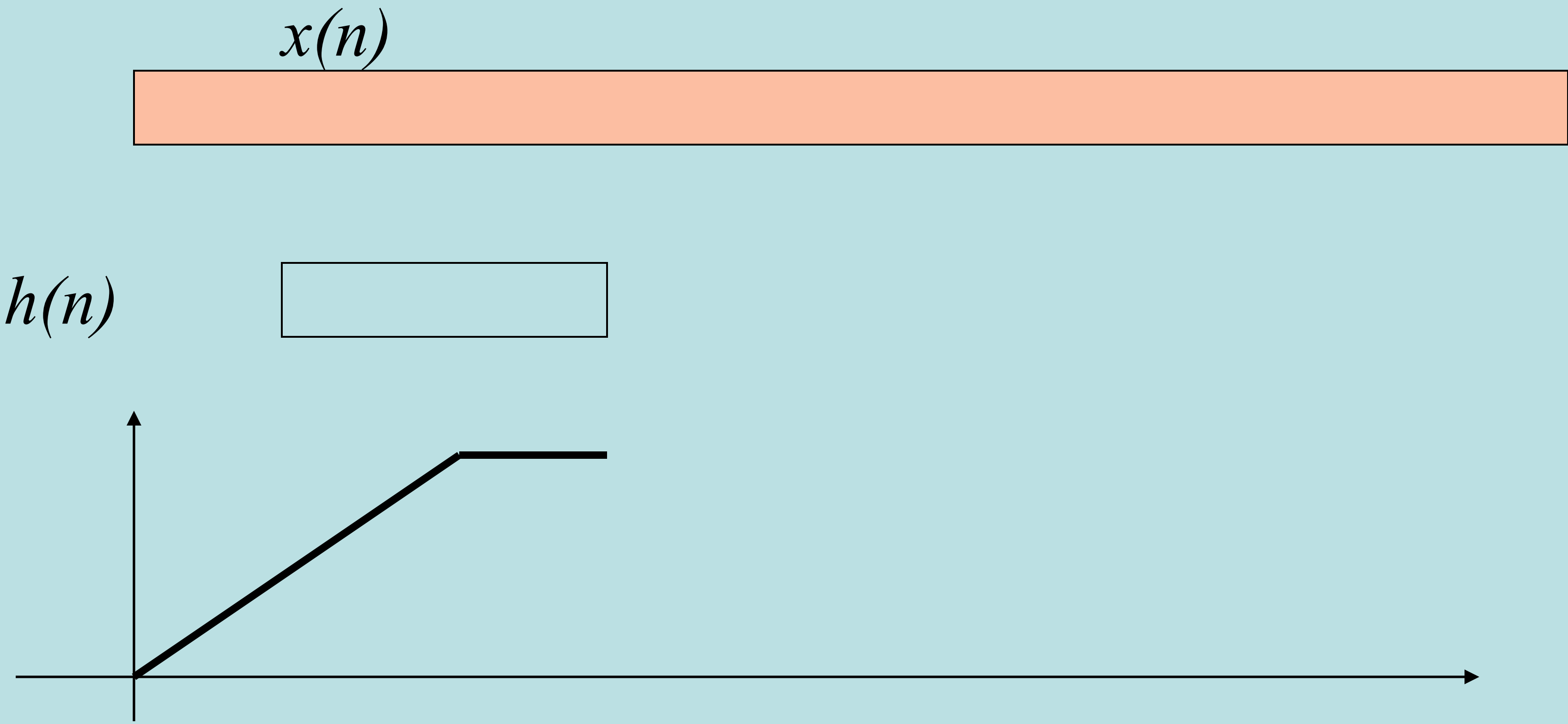
Convolution of long sequences



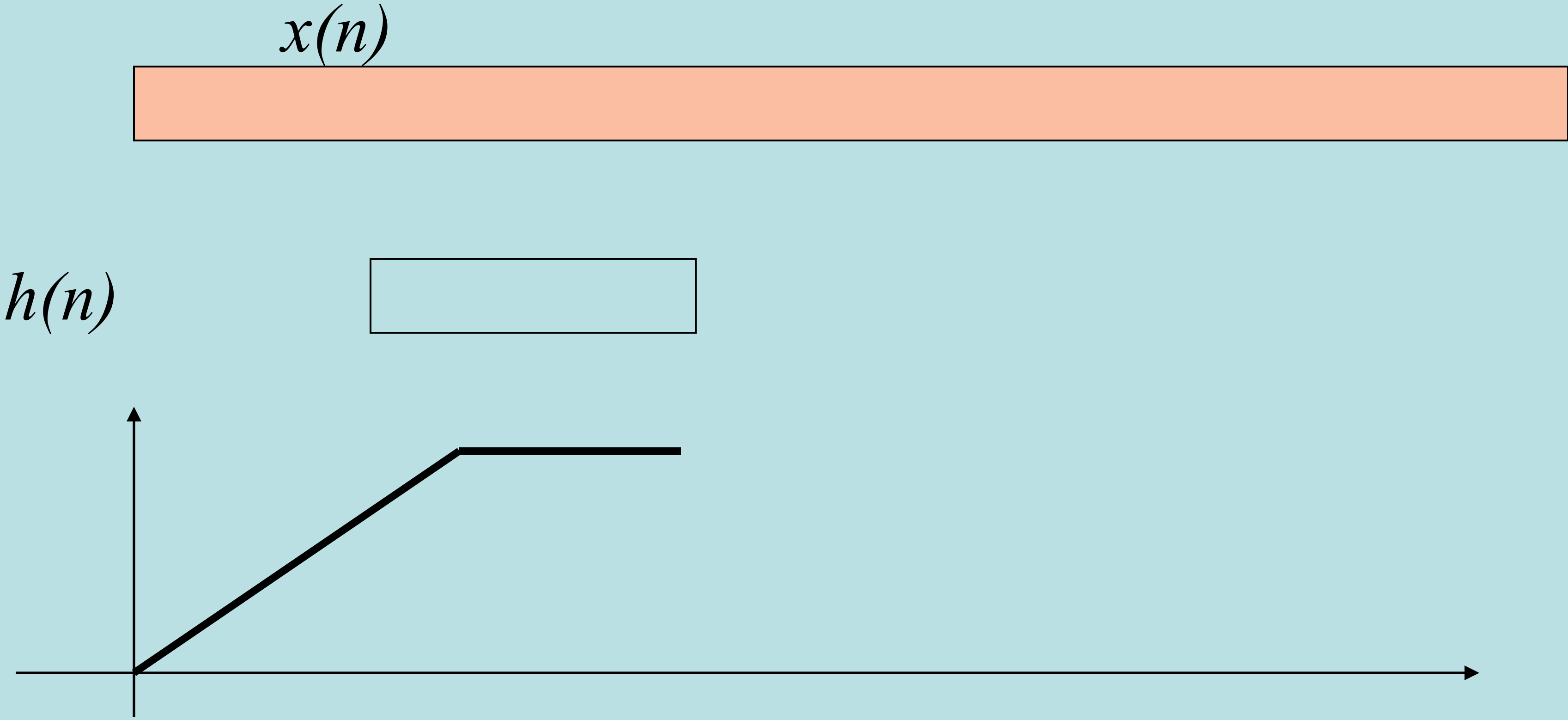
Convolution of long sequences



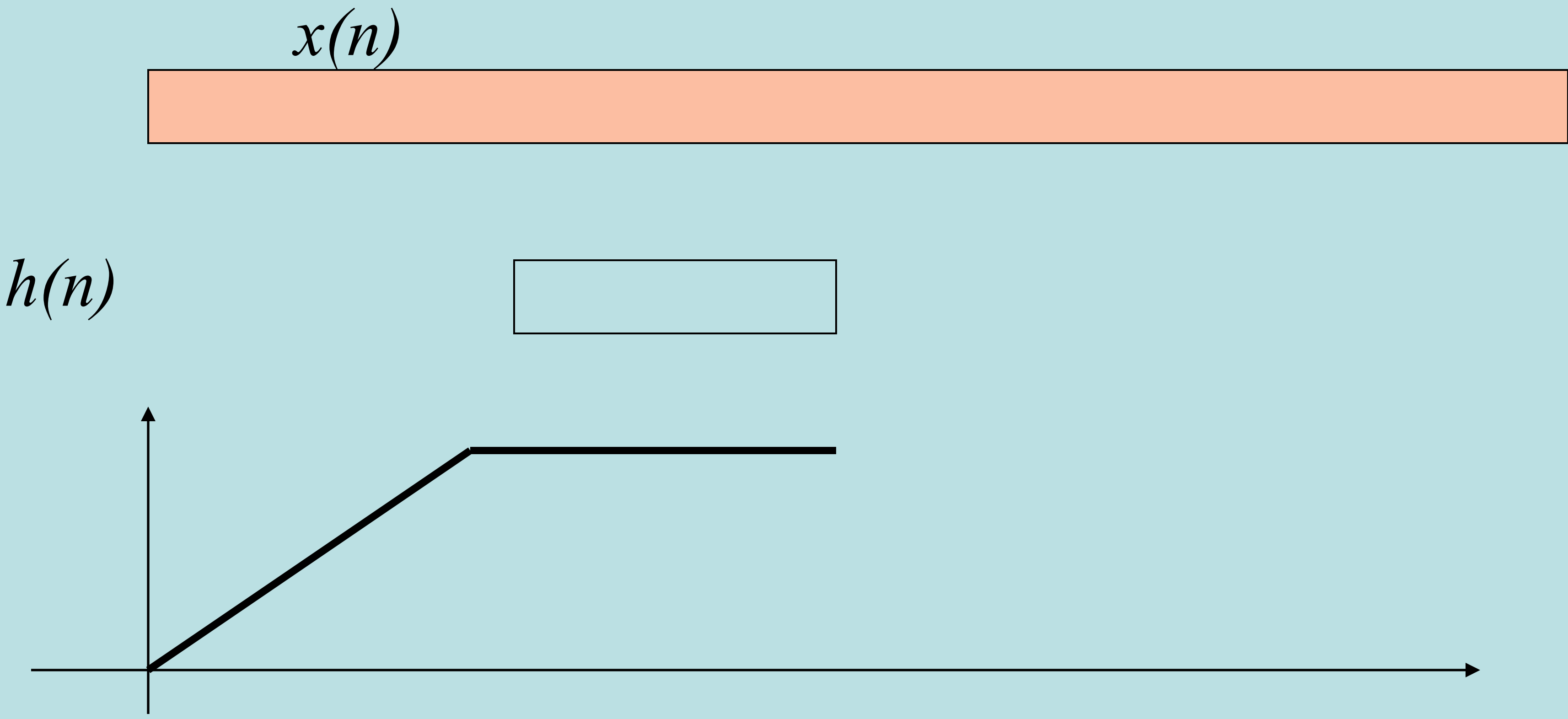
Convolution of long sequences



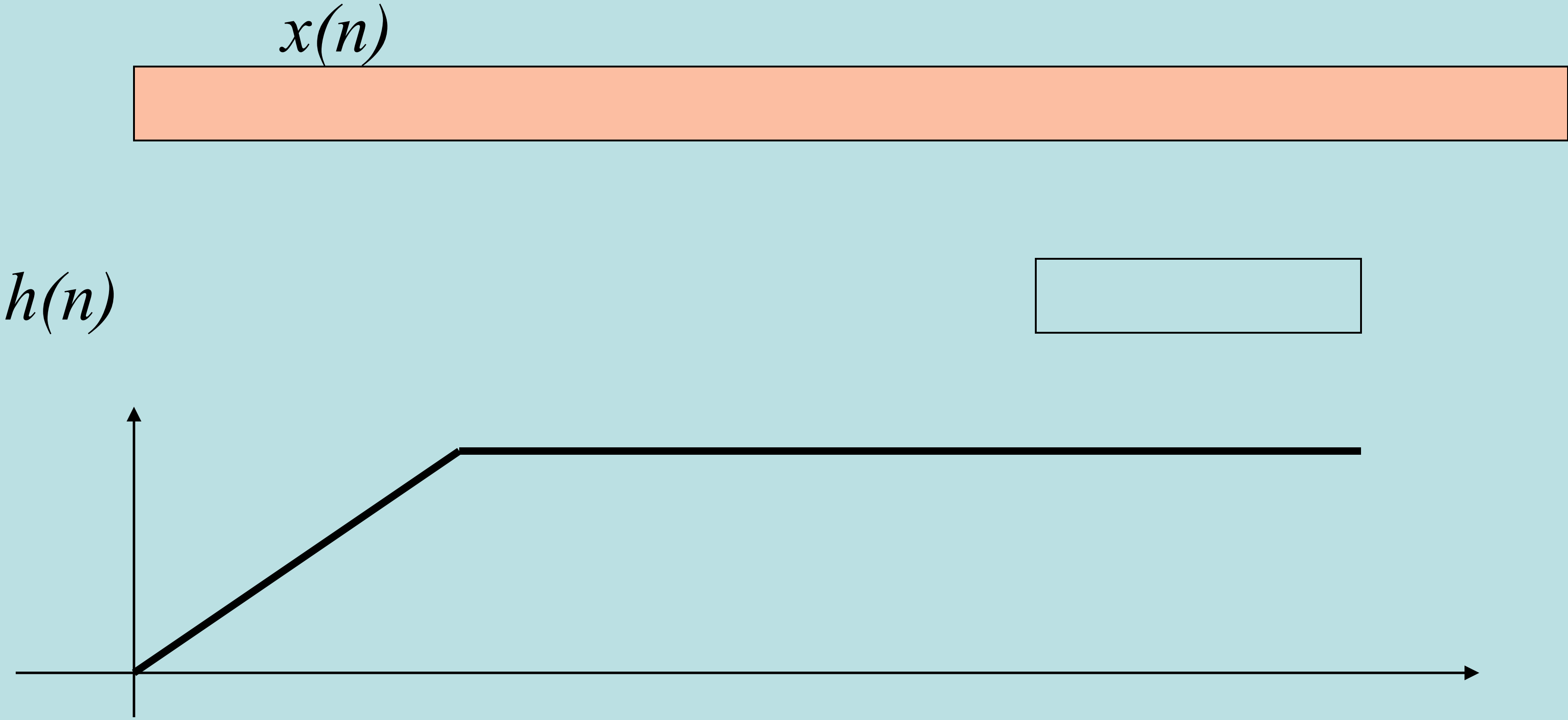
Convolution of long sequences



Convolution of long sequences



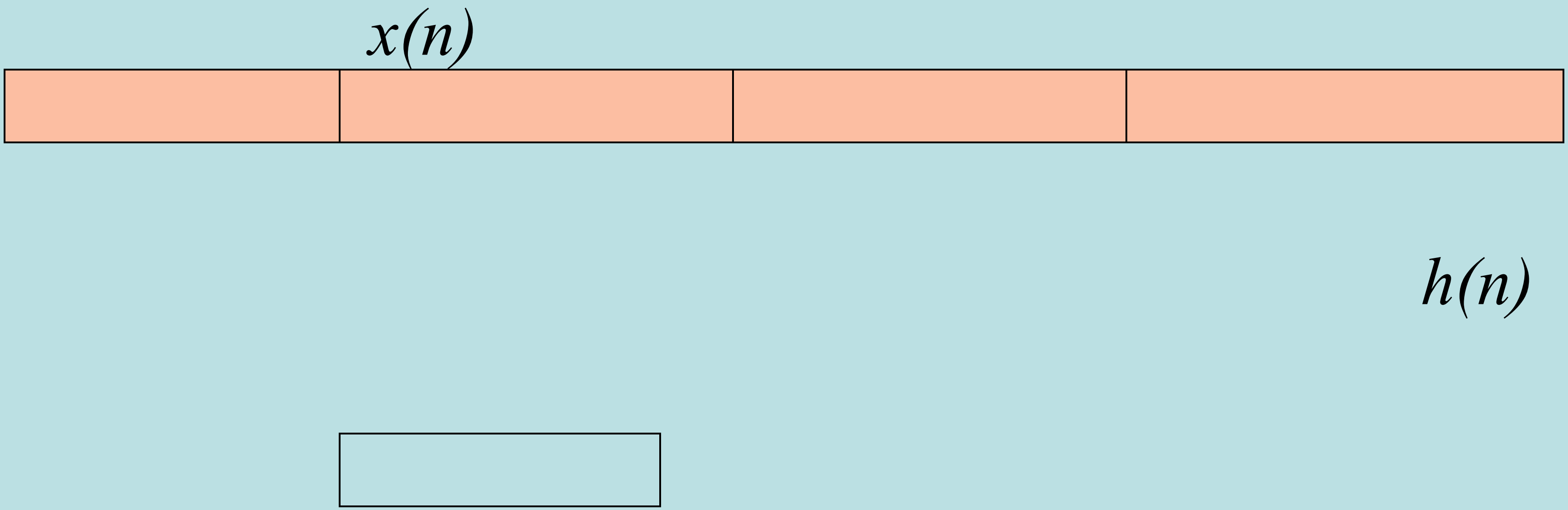
Convolution of long sequences



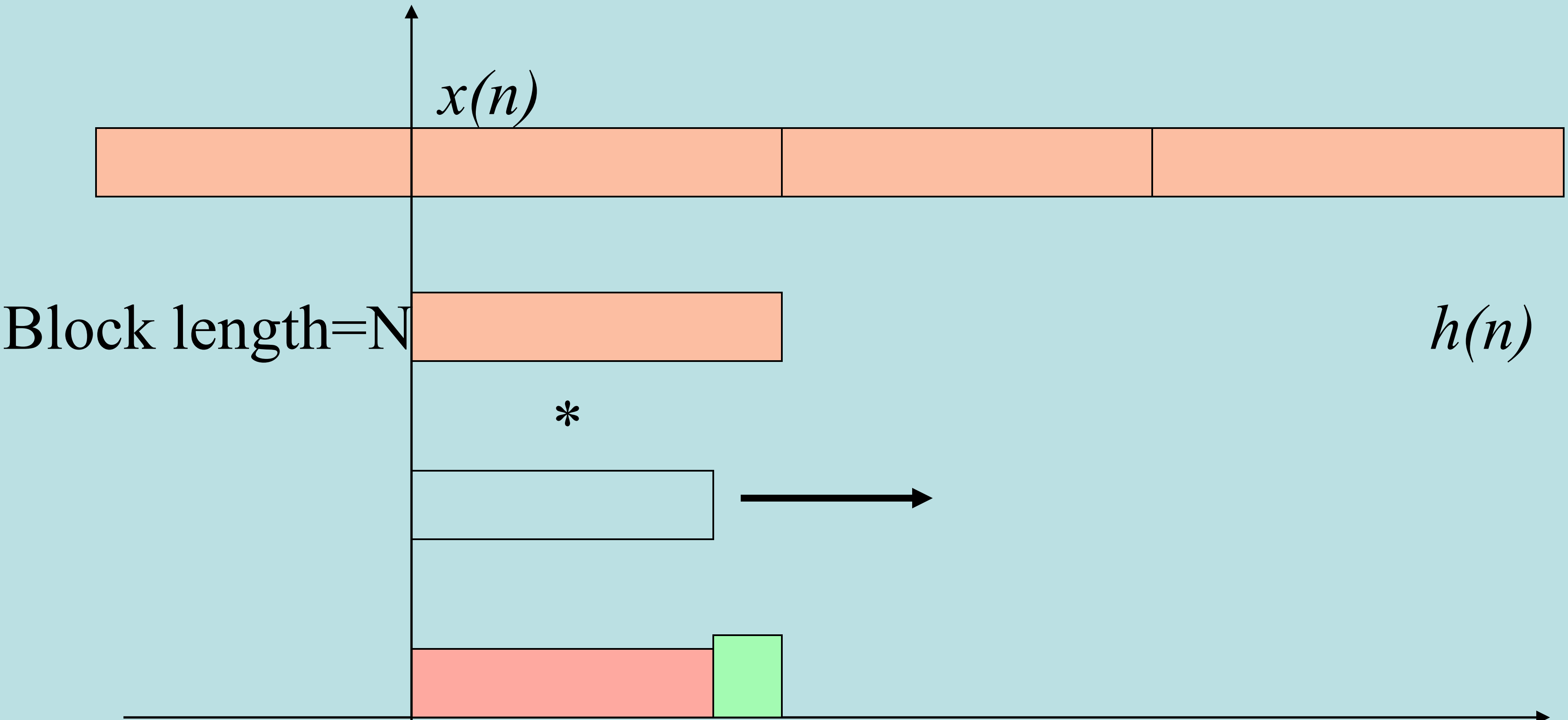
Convolution of Long Sequences --- Block Based approach

- x(n) are divided into blocks;
- convolutions are performed for each block and *h(n)* --- short time convolution;
- construct the output by combining the results of block convolution;
- Issues : how to construct the blocks? How to construct the output?
- Two approaches: overlap-save and overlap-add

Convolution of long sequences

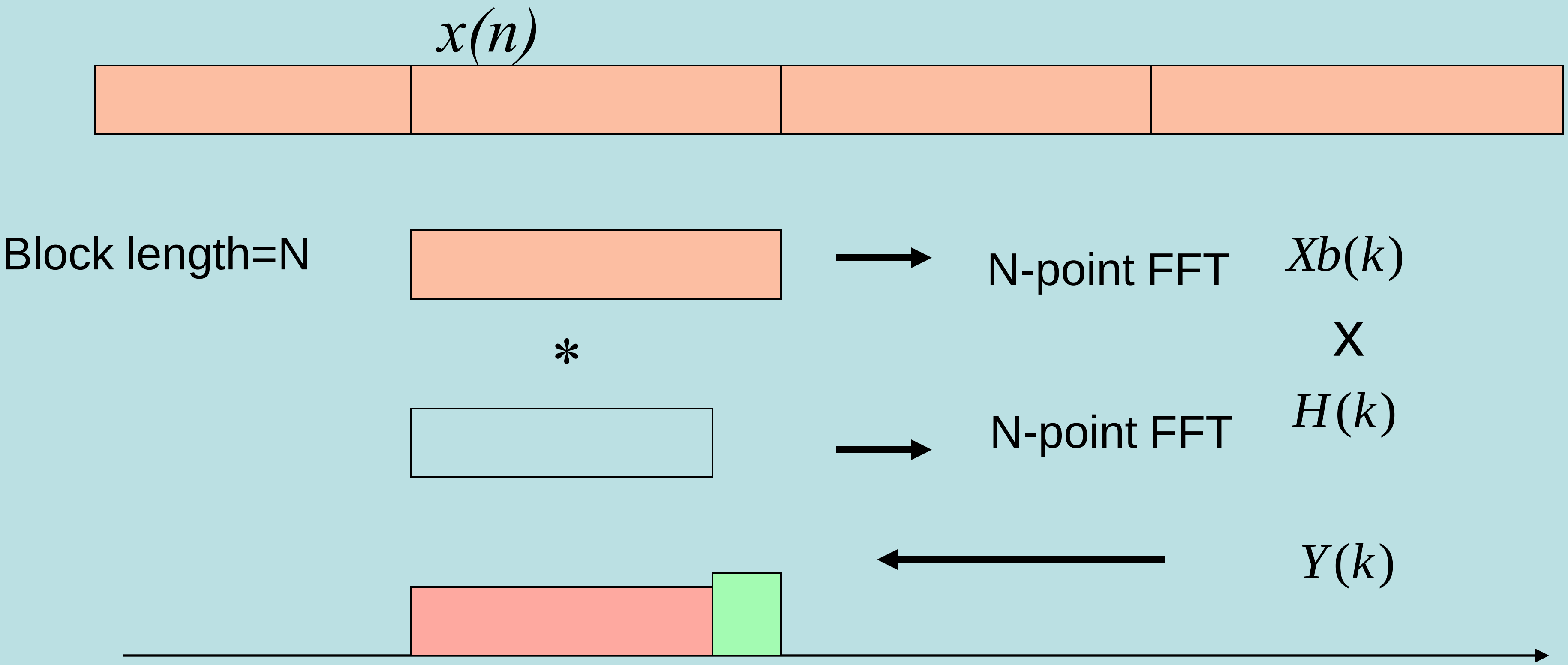


Overlap-save approach



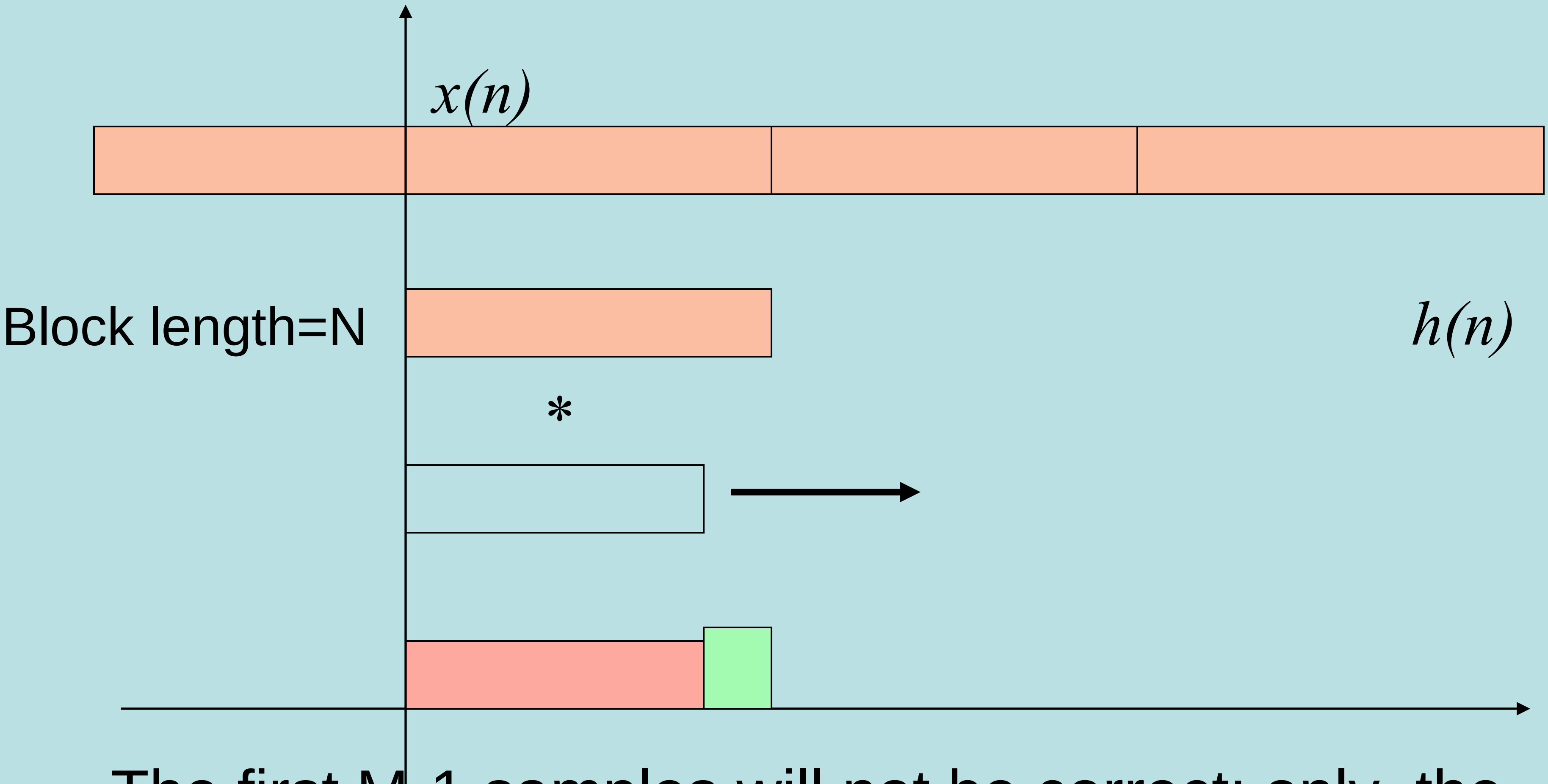
If N-point circular convolution is performed, the first M-1 samples of the result will not be correct; only the last N-M+1 samples are correct;

Overlap-save approach



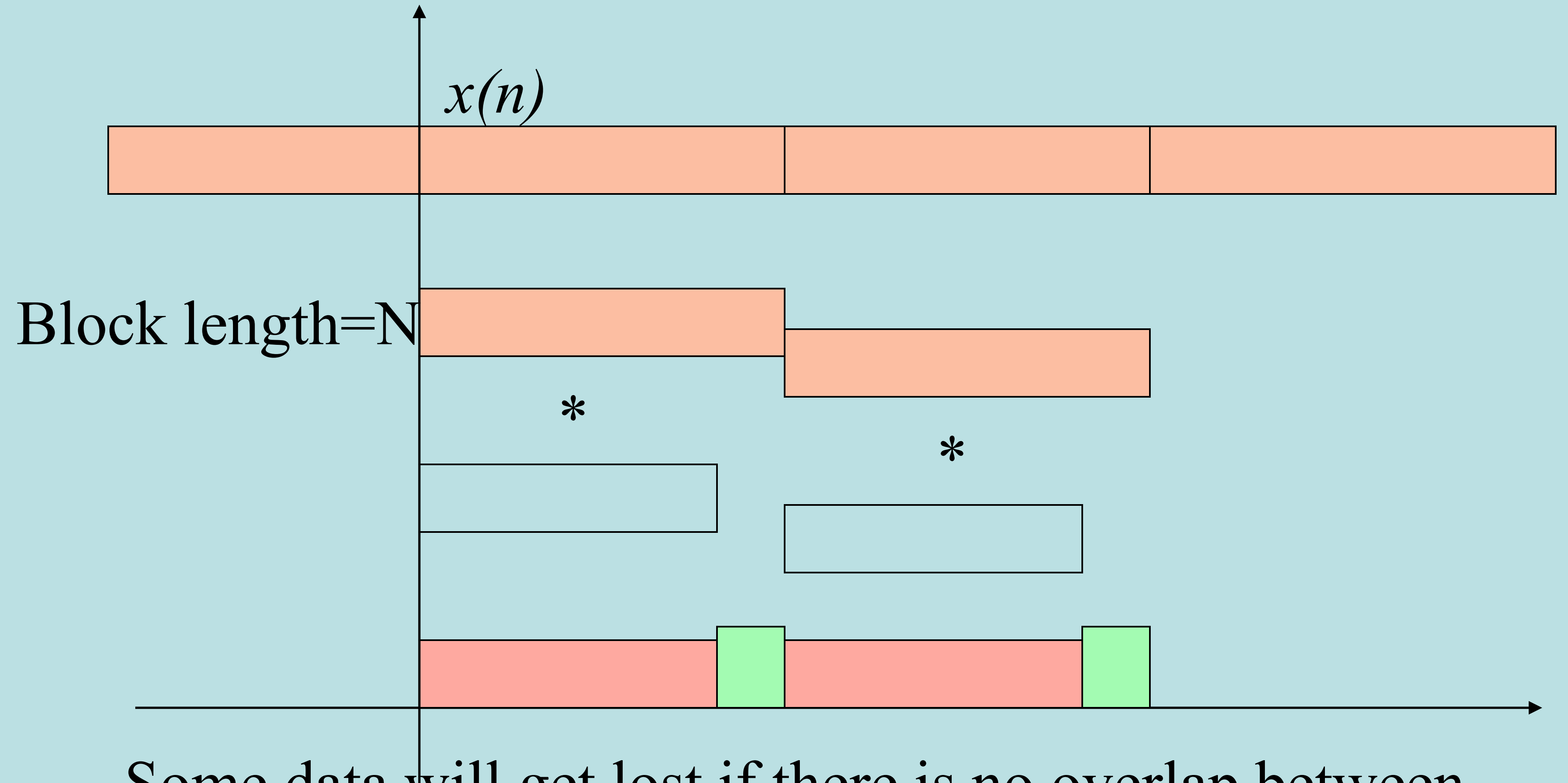
The first (M-1) samples will not be correct; only the (N-M+1) samples are correct;

Overlap-save approach

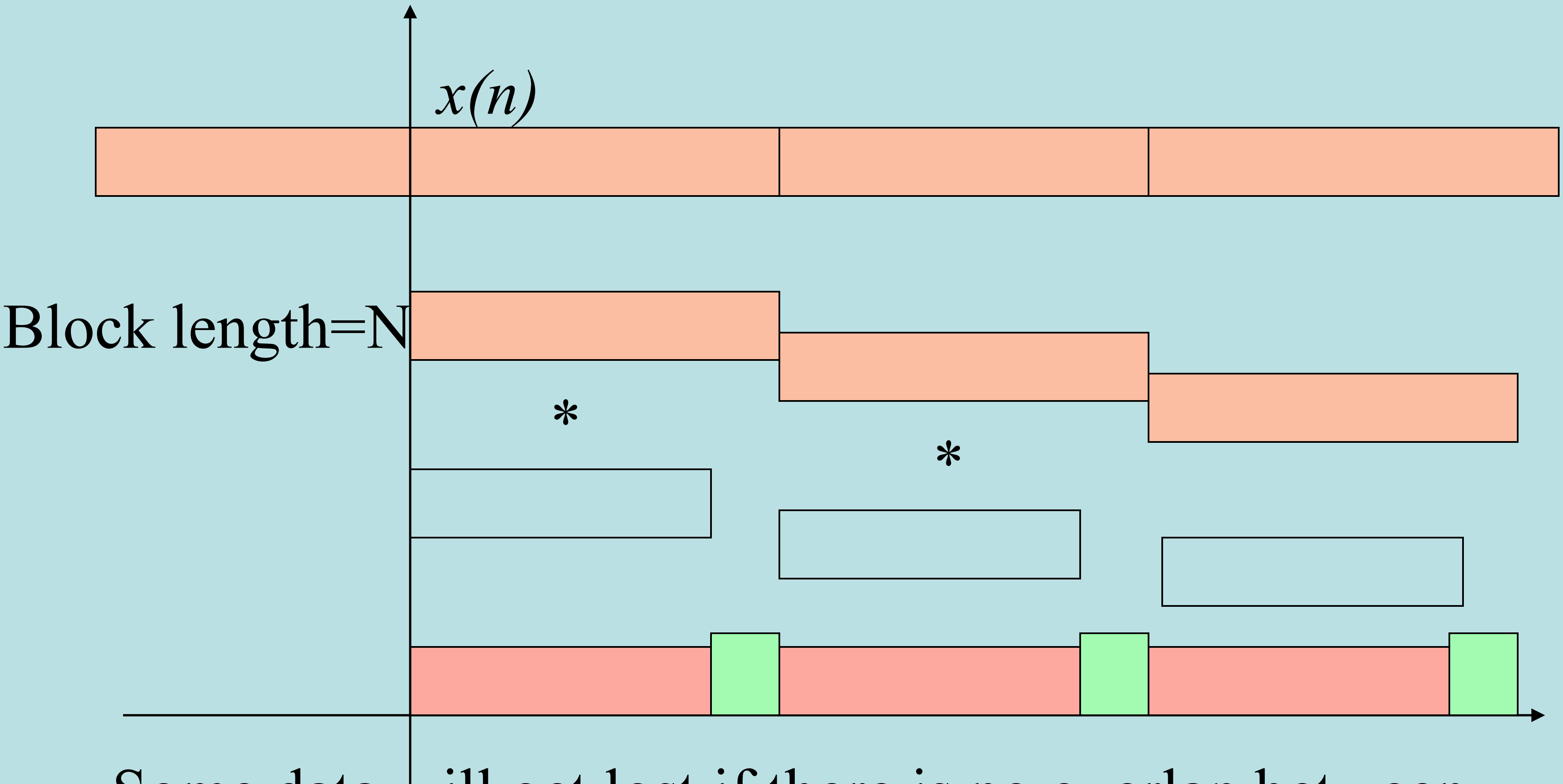


The first M-1 samples will not be correct; only the N-M+1 samples are correct;

Overlap-save approach

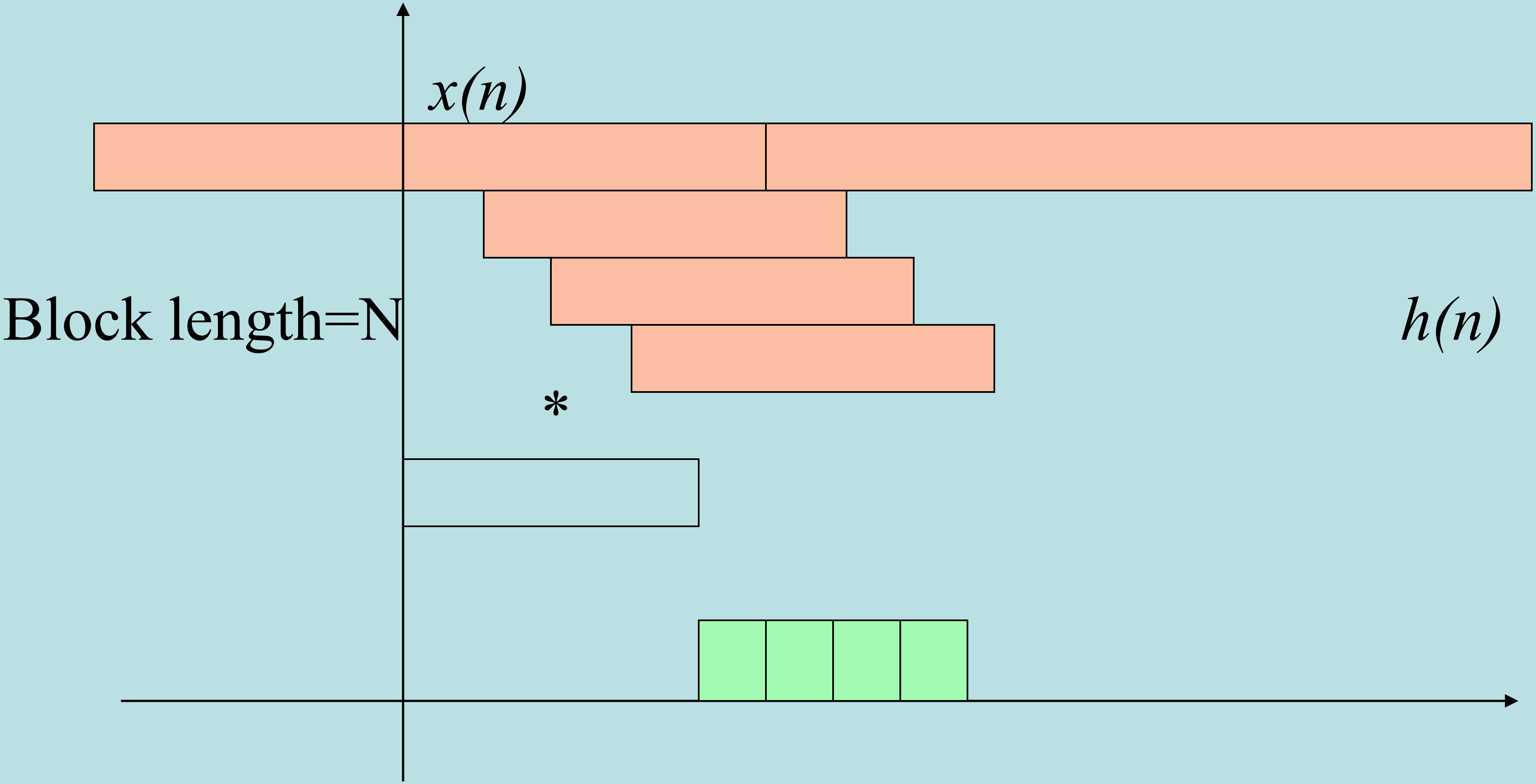


Some data will get lost if there is no overlap between the blocks



Some data will get lost if there is no overlap between the blocks

Overlap-save approach



Overlap-save approach

- The above process is called overlap-save methods:
  - Take N signal samples as a block;
  - do N-point DFT of the block, and N-point DFT of h(n) ($N>M$ the length of h(n));
  - Multiple $X_b(k)$ and $H(k)$;
  - Do the IDFT of $Y(k)$
  - Discard the first $(M-1)$ samples of $y(n)$;

Overlap-save approach

- Get the next block by getting *N-M+1* new samples, and discard (N-M+1) oldest samples
- Repeat the above convolution process.

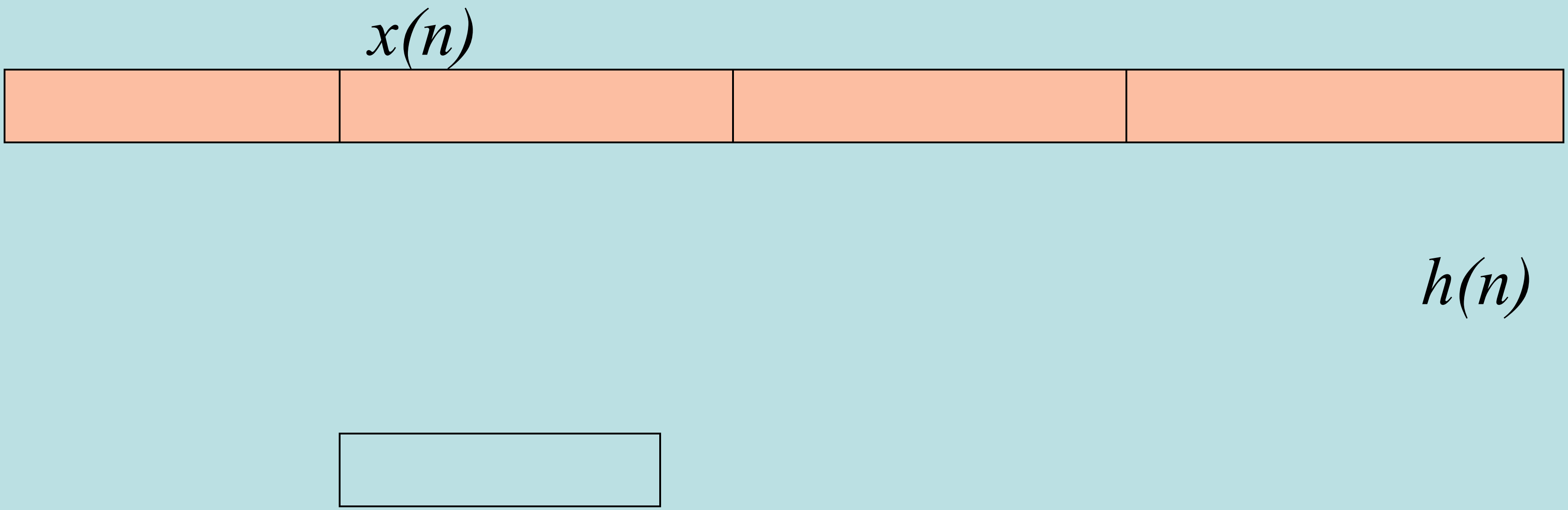
Overlap-save approach--- an example

- Convolve a 50-pint sequence h(n) with a long sequence x(n):
  - 1. Let N=64;
  - 2. taking 64 samples from x(n), perform circular convolution using 64-point FFT. Discard the first 49 samples and keep the last 64-50+1=15 samples;
  - Move the block by getting 15 samples from x(n), repeat step 2 and keep the next 15 samples of the result….
  - Combine all the 15 samples together to get the convolution results

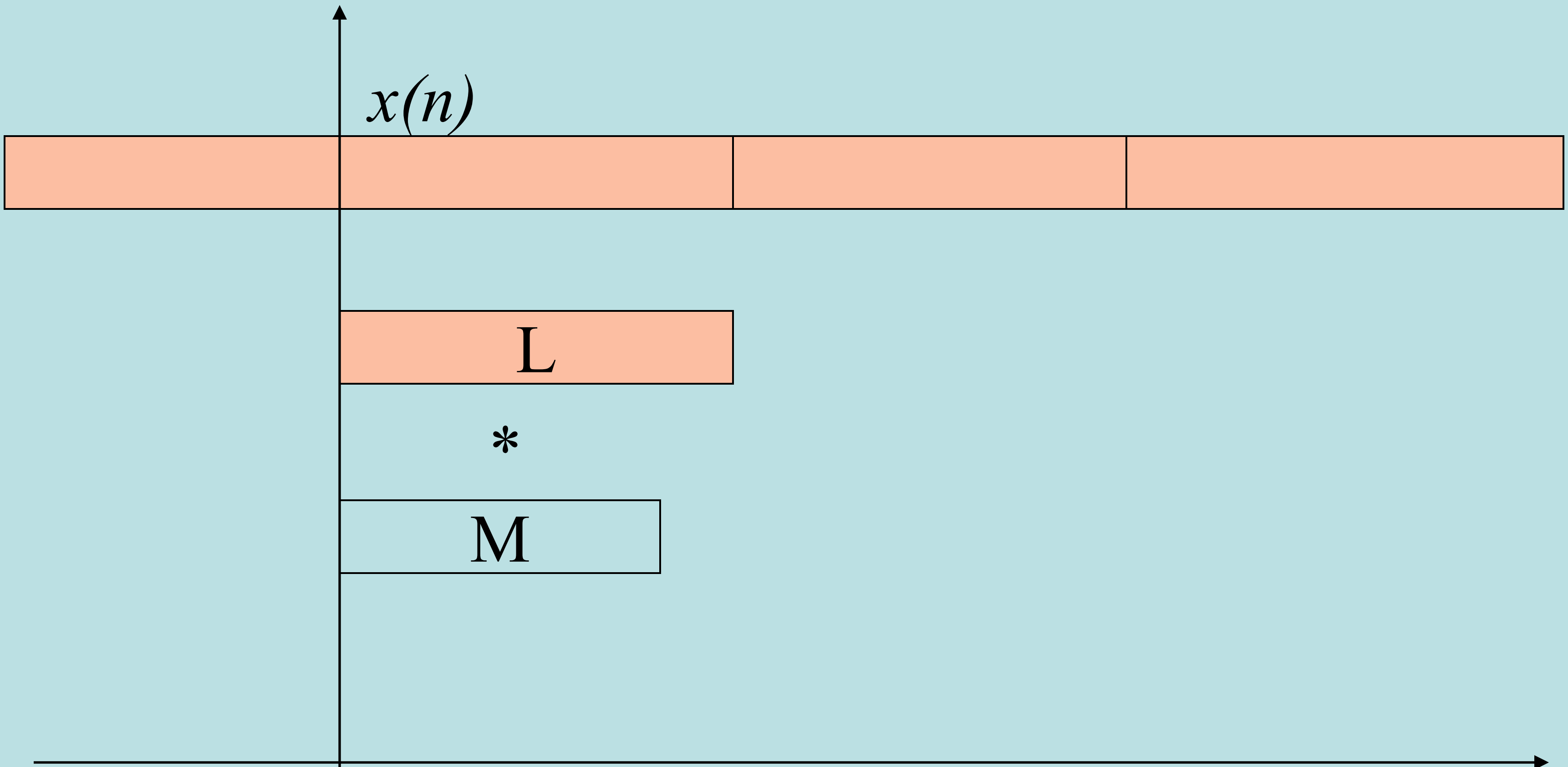
Convolution of Long Sequences --- Overlap-Add Method

- Here we try to use linear convolution instead of circular convolution:
  - Take a block $x_b(n)$ of length L;
  - H(n) is of length M;
  - Take the N-point DFT of them, where $N=L+M-1$
  - Calculate Y(k)=X(k)H(k), k=0, 1, …, N-1
  - Calculate IDFT of Y(k) yield y(n), n=0, 1, …, N-1

Convolution of long sequences

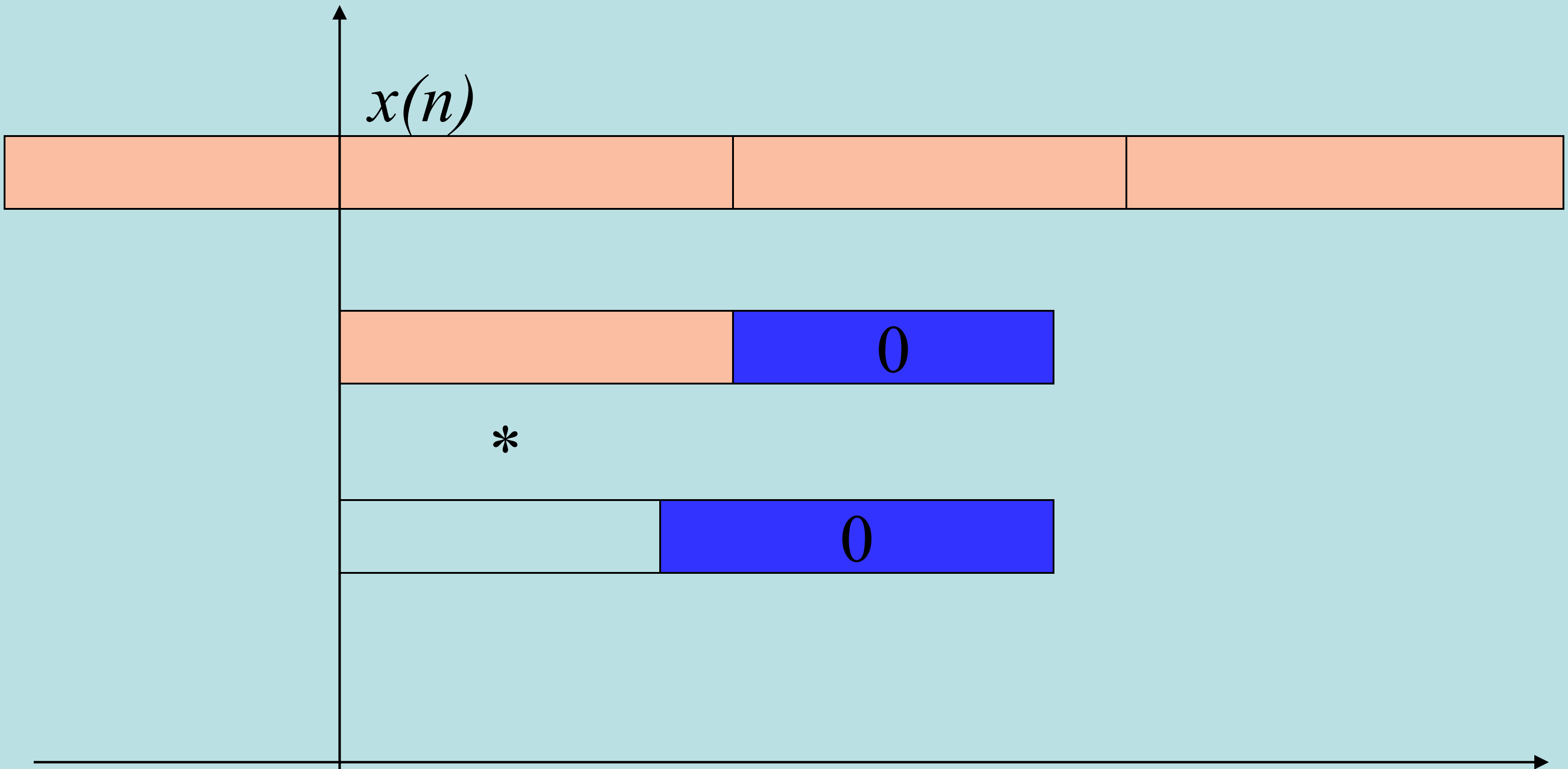


Convolution of long sequences



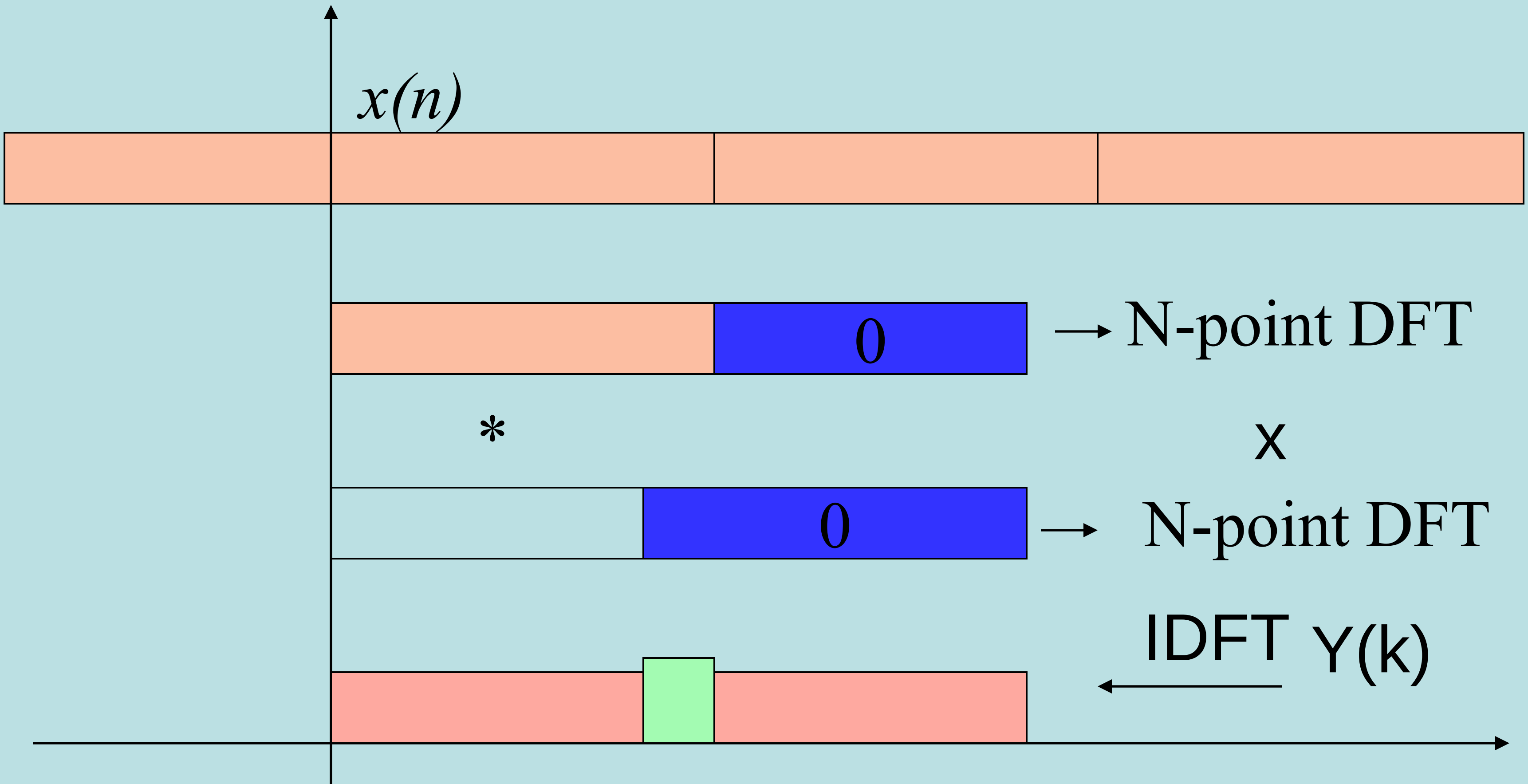
The first M and the last M samples will not be correct; only the N-M samples are correct;

Convolution of long sequences



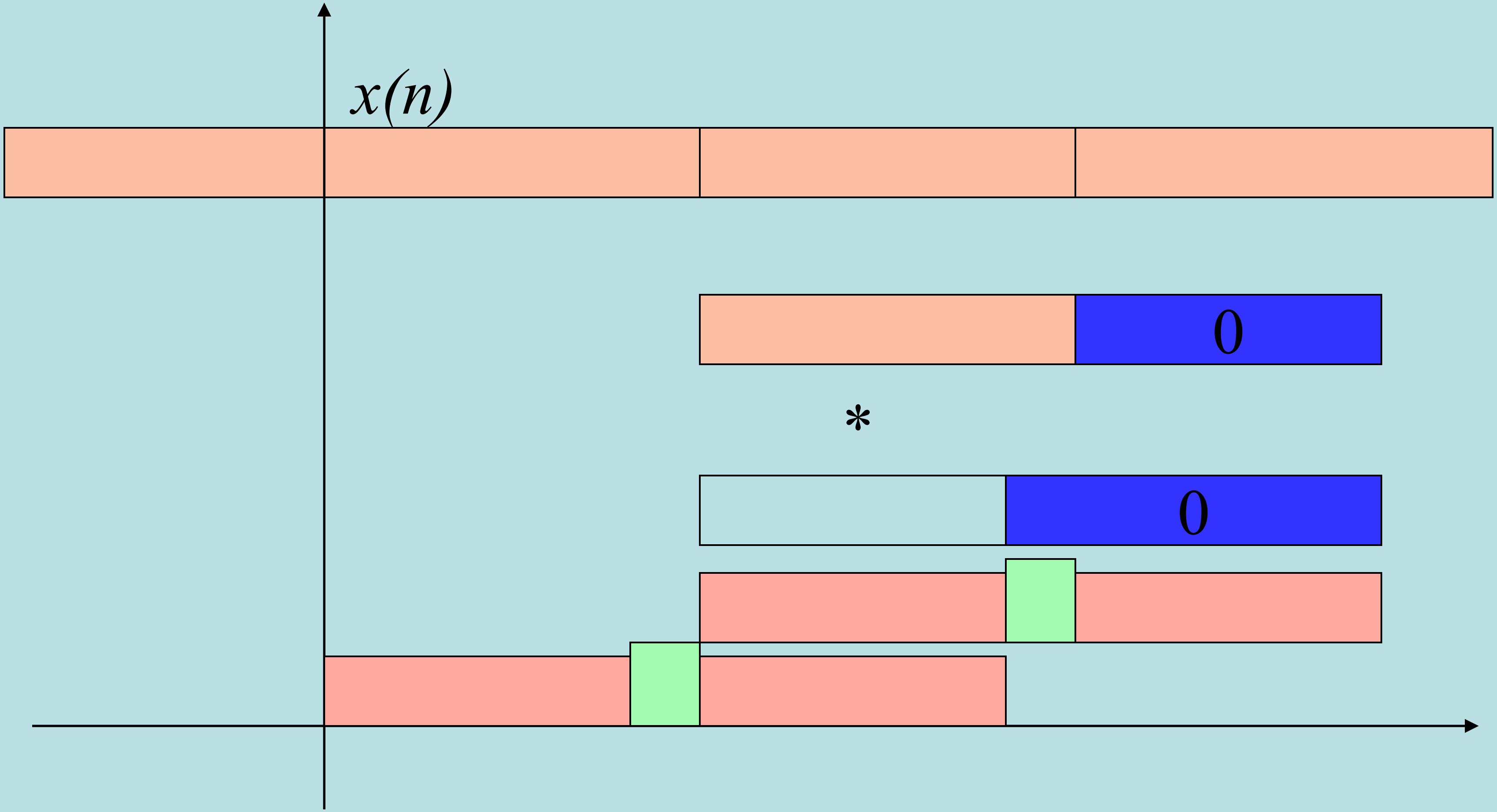
The first M-1 and the last M-1 samples will not be correct; only the N-M samples are correct;

Convolution of long sequences

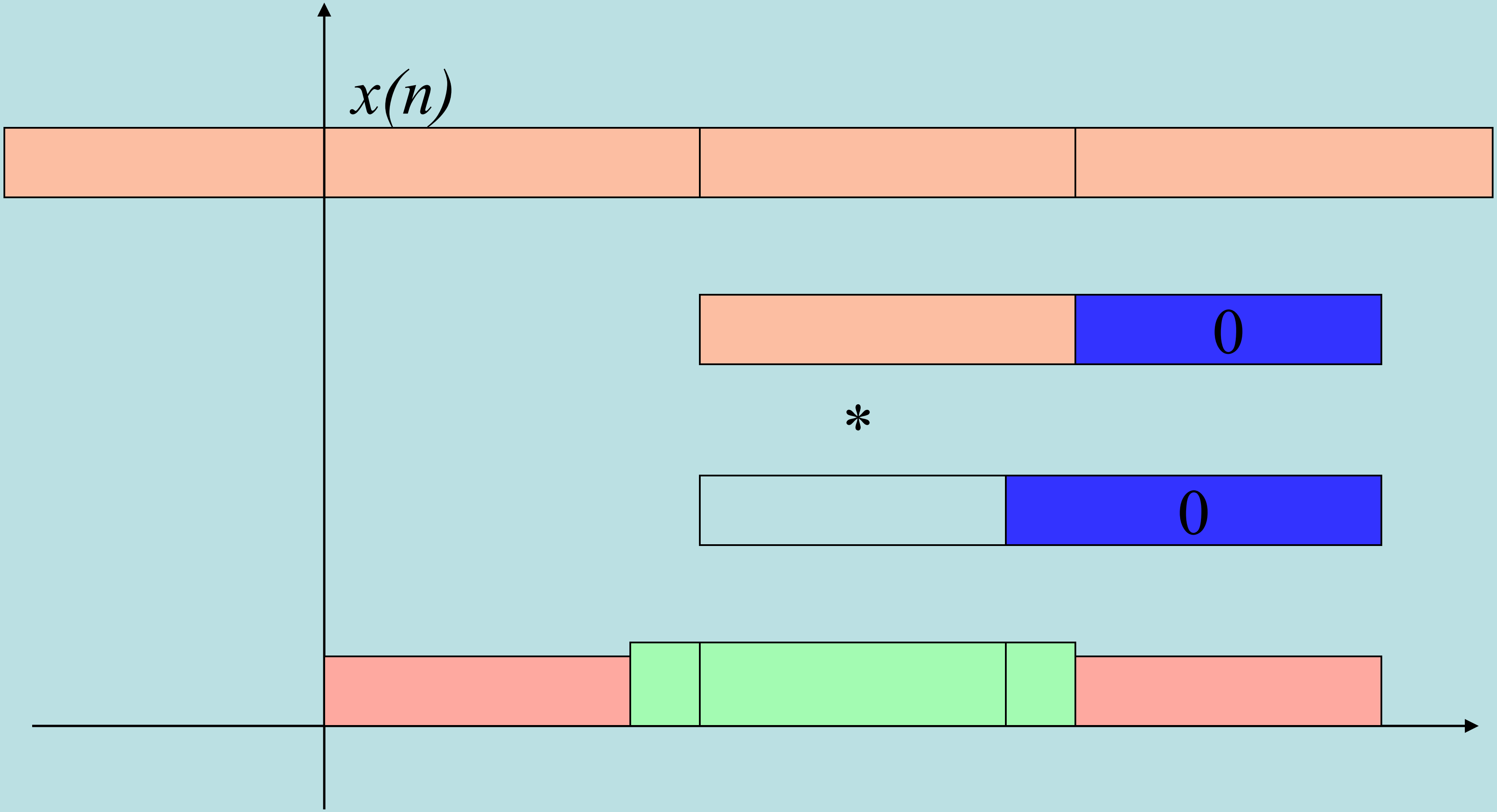


The first M-1 and the last M-1 samples will not be correct; only the N-M+1 samples are correct;

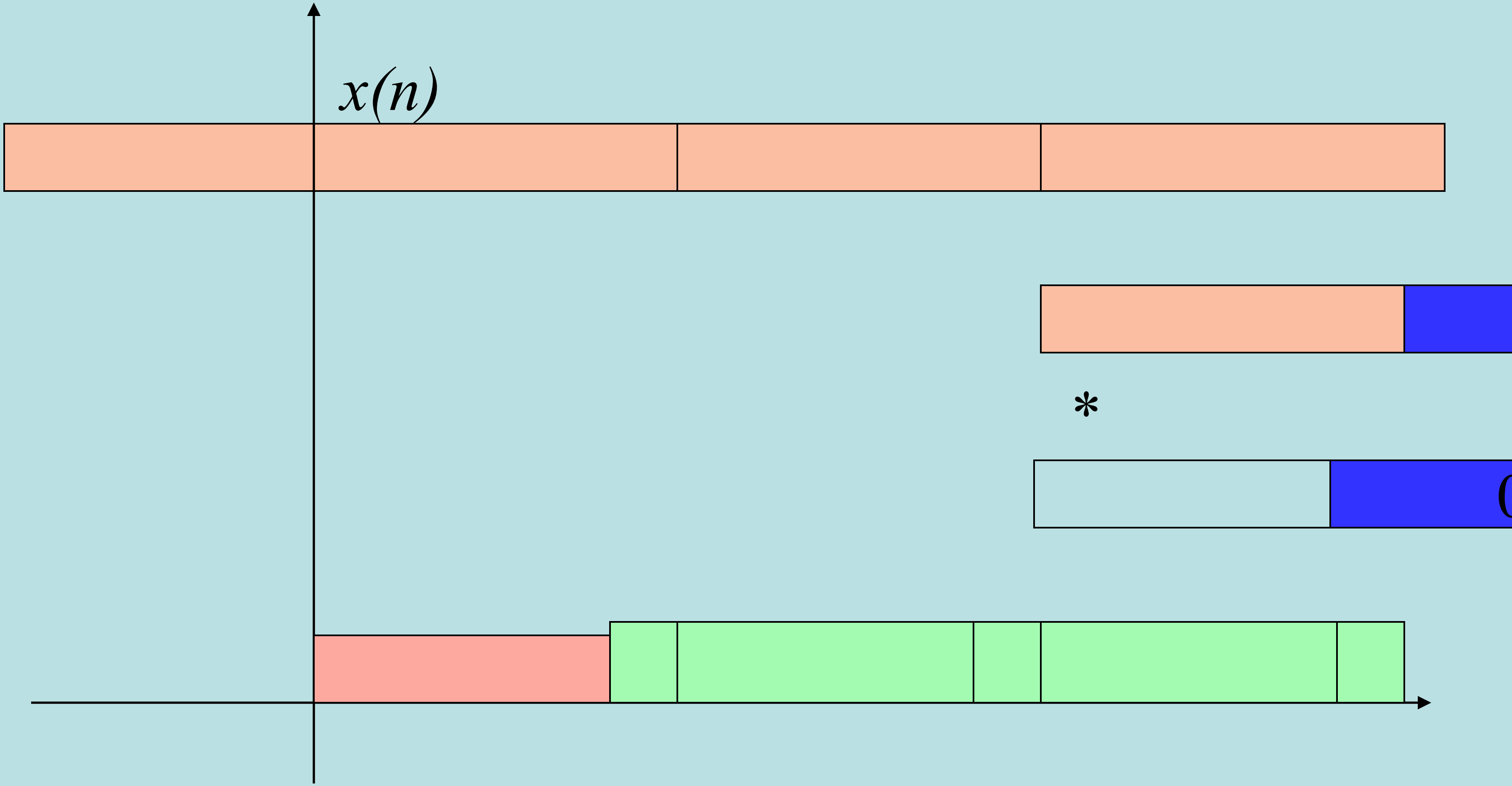
Convolution of long sequences



Convolution of long sequences



Convolution of long sequences



Convolution of Long Sequences --- Overlap-Add Method

- Construct the mth block $x_b(n)$ as:

  $\{x(mL), x(mL+1), \ldots x(mL+L-1), 0, \ldots, 0\} \rightarrow$ Length N
- Take the N-point DFTs of $x_b(n)$ and $h(n)$;
- Multiplication $Y_m(k)=X_b(k)H(k)$
- IDFT: $y(n)=\mathrm{IDFT}(Y(k))$
- Repeat the operation for next block

  $\{x((m+1)L), x((m+1)L+1), \ldots x((m+1)L+L-1), 0, \ldots, 0\}$

  ……….

Convolution of Long Sequences --- Overlap-Add Method

- The last (M-1) points for the first y(n) are overlapped and added to the first (M-1) points of the second y(n);
- The last (M-1) points for the second y(n) are overlapped and added to the first (M-1) points of the third y(n);
- ……
- The above process will result in the convolution of h(n) and x(n)

Summary

- Fast convolution of short sequences
  - Linear convolution
  - Circular convolution
  - When they can be equal?
- Fast convolution of short sequences
  - Overlap-saving (block overlapping, discard some results)
  - Overlap-adding(block separate, overlap and add some results)