第一章　离散时间信号与系统

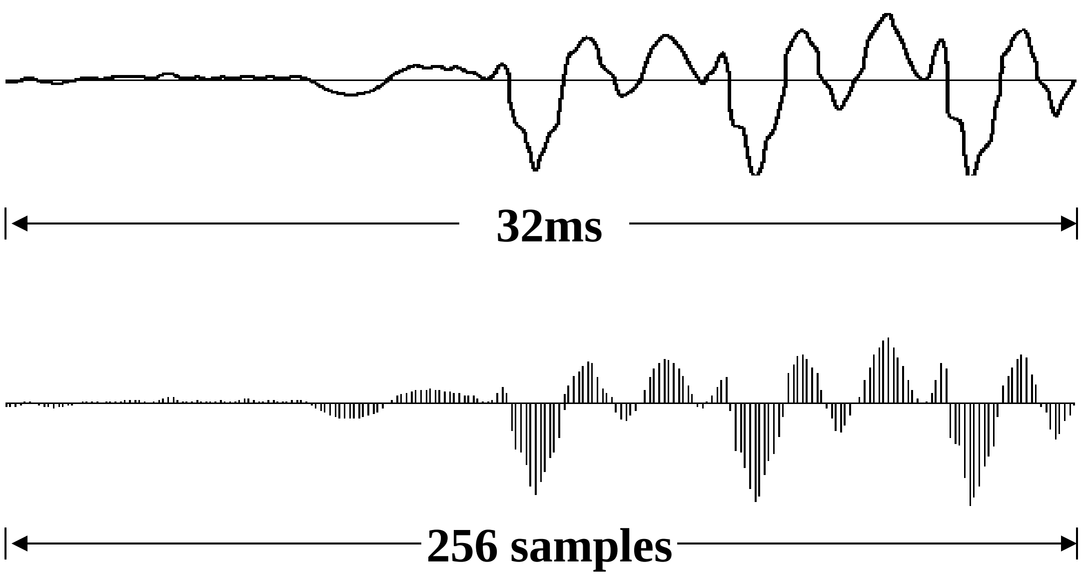
主要内容：

§1.1 离散时间信号(序列)
Discrete-time signals (Sequences)

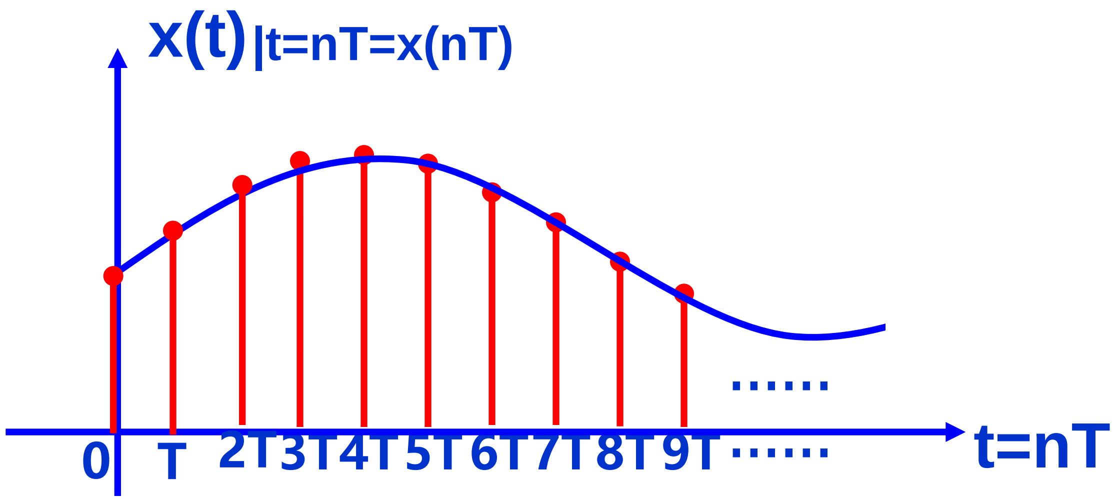
一、离散时间信号的由来

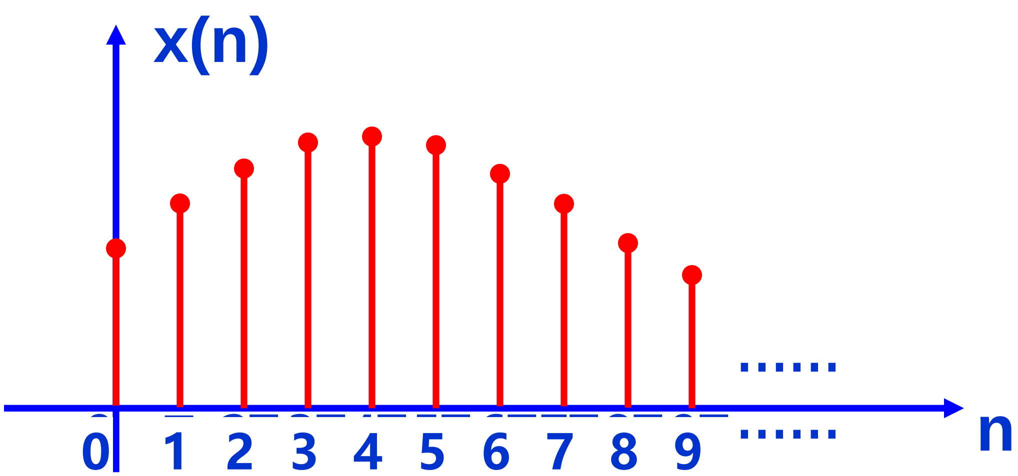
离散时间信号（又称序列），是连续时间信号以时间T等间隔采样得到的,T称为采样间隔（单位:秒）。

$$T=\frac{32\times10^{-3}}{256}$$



❖ 一般，采样间隔是均匀的，用x(nT)表示离散时间信号在nT点上的值，n为整数。由于x(nT)顺序存放在存储器中，我们通常直接用x(n)表示离散时间信号一序列。





二、离散时间信号的表示方法

1、用枚举的方式(数列形式)表示：

x(n) = { 3，4，2，1，0，5，7，8 }

注：用箭头标出n=0在序列中的位置，上面序列的x(0)=1

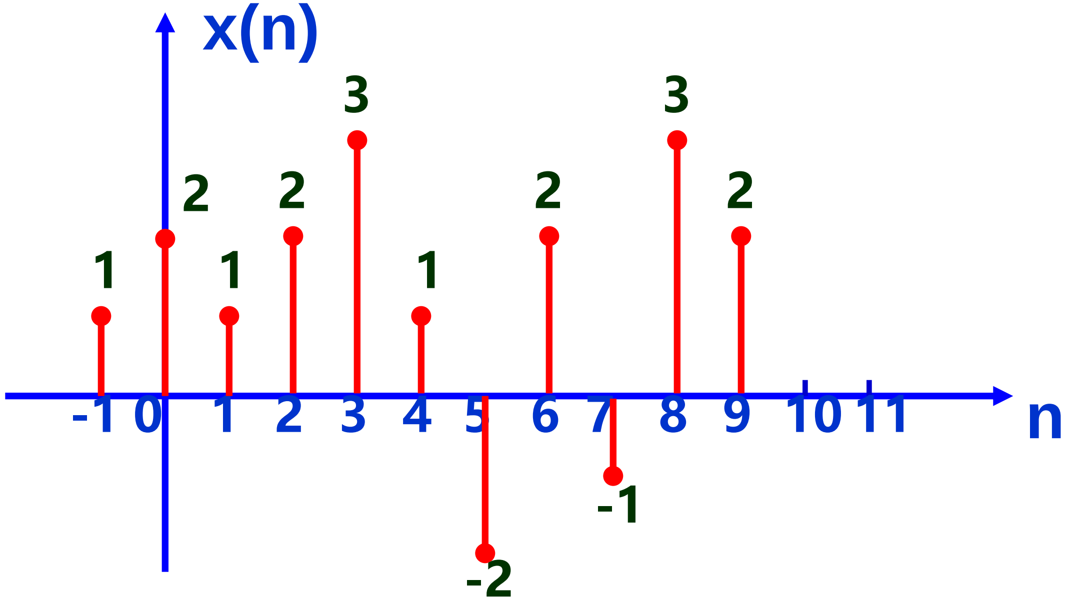
2、用公式表示：

$$x(n) = A\sin(\omega n + \varphi)$$

$$x(n)=\begin{cases} 2^{-n} & n \geq 0 \\ 3^{n} & \boxed{n<0} \rightarrow \boxed{n \leq -1} \end{cases}$$

因为n只能取整数，所以两种写法是一样的。

3、用图形的方式表示：



……

$x(0) = 2$

$x(1) = 1$

$x(2) = 2$

$x(3) = 3$

……

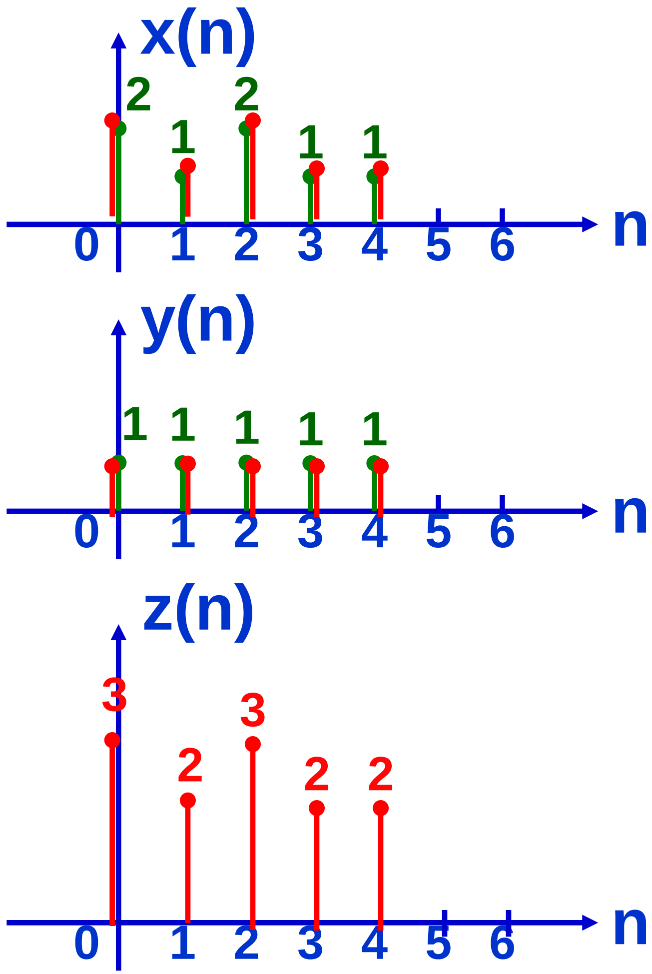
❖ 图中横坐标n表示离散的时间坐标，仅在n为整数时才有意义，纵坐标代表信号点的值。

4、用单位抽样序列表示.

三、序列的基本运算

1、序列的和：

❖ 两序列的和是指同序号n的序列值逐项对应相加而构成的新序列。



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

…… ……

$z(0) = x(0) + y(0) = 3$

$z(1) = x(1) + y(1) = 2$

$z(2) = x(2) + y(2) = 3$

$z(3) = x(3) + y(3) = 2$

$z(4) = x(4) + y(4) = 2$

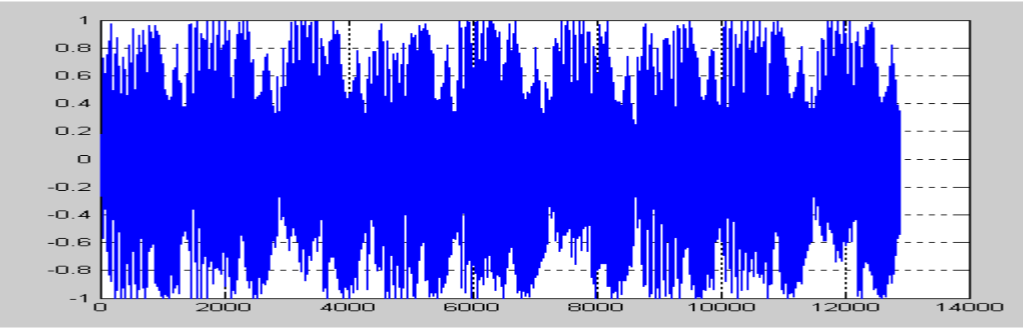
…… ……

仿真实验(Matlab)

```
x1=wavread('w1.wav');

x2=wavread('w2.wav');

y=x1+x2;

figure(1); plot(x1); grid on;

figure(2); plot(x2); grid on;

figure(3); plot(y); grid on;

wavwrite(y,'w3.wav');
```

实验结果 y(n) = x1(n)+ x2(n)

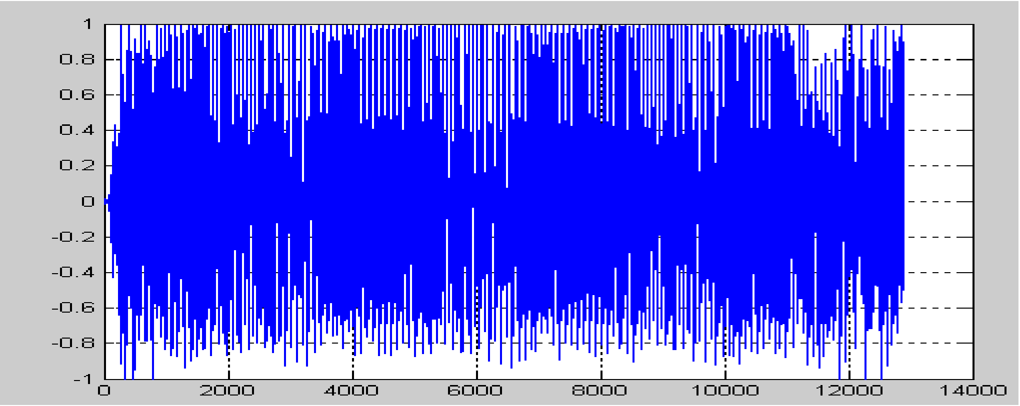
x1(n)



‘w1.wav’

(.wav)

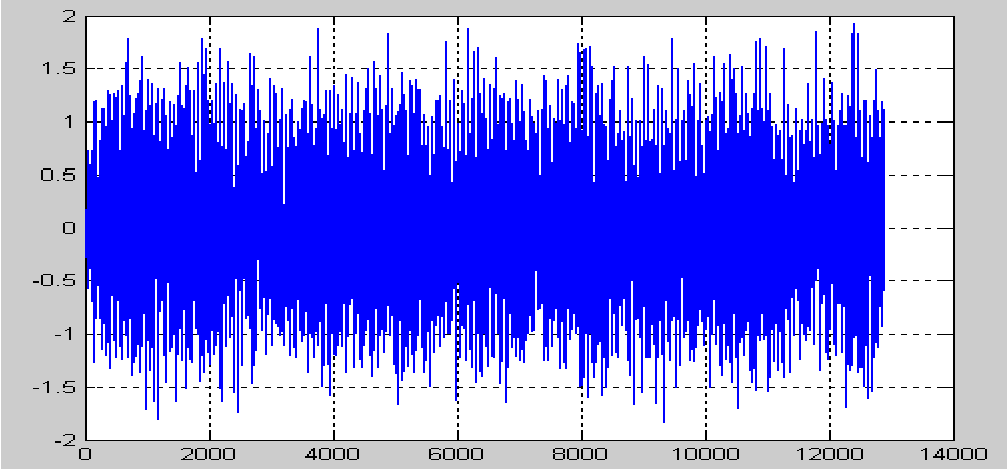
x2(n)



‘w2.wav’

(.wav)

y(n)



‘w3.wav’

2、序列的积：

两序列的积是指同序号n的序列值逐项对应相乘而构成的新序列。

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

… …

$z(0) = x(0) * y(0) = 2$

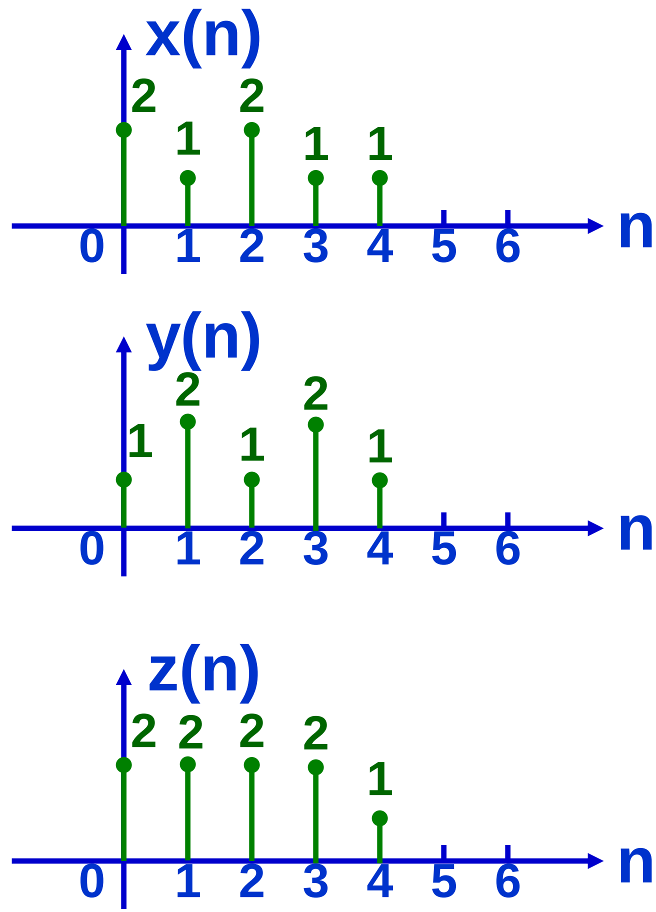
$z(1) = x(1) * y(1) = 2$

$z(2) = x(2) * y(2) = 2$

$z(3) = x(3) * y(3) = 2$

$z(4) = x(4) * y(4) = 1$

… …



3、序列的移位：

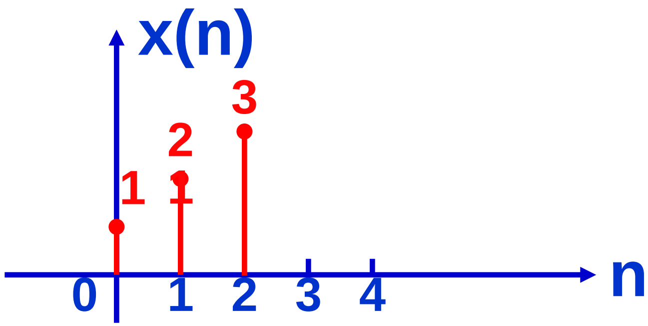
$$y(n) = x(n \pm m)$$

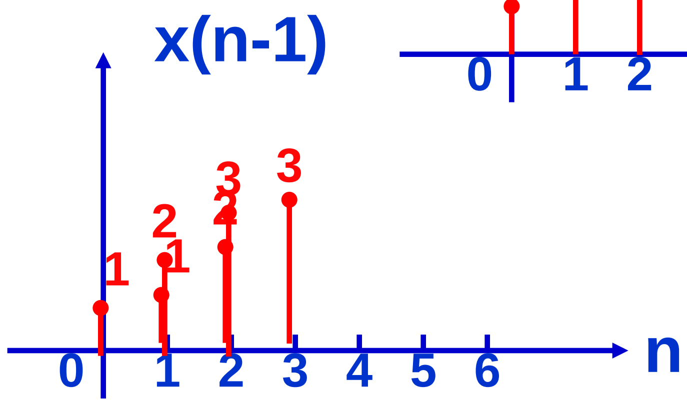
设有一序列x(n)，当m为正时：

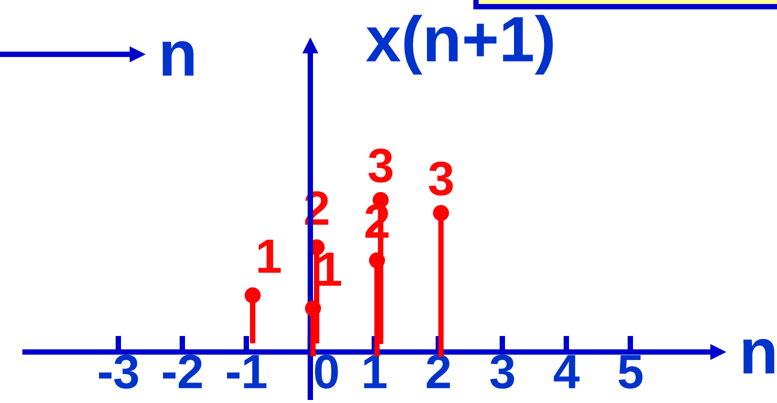
x(n-m)表示序列x(n)逐项依次右移m位后得到的序列。

x(n+m)表示序列x(n)逐项依次左移m位后得到的序列。

x(0)=1
x(1)=2
x(2)=3







❖ 实例： 序列右移（序列延迟）的应用

延时单元可以将以前的某采样时刻的数据暂存起来，参与这个时刻的运算。

回声可以用延迟单元来生成。直接声音和它的延迟了R个周期的单个回声可以用下面的式子来表示（α 为回声的衰减系数）：

$$y(n)=x(n)+\alpha x(n-R) \quad |\alpha|<1$$

为了生成间隔为R个周期的多重回声，可将上式改为：

$$y(n)=x(n)+\alpha x(n-R)+\alpha^{2}x(n-2R)+\cdots+\alpha^{N-1}x(n-(N-1)R) \quad |\alpha|<1$$

原声：

混响1：

α=0.3, R=5000

混响2：

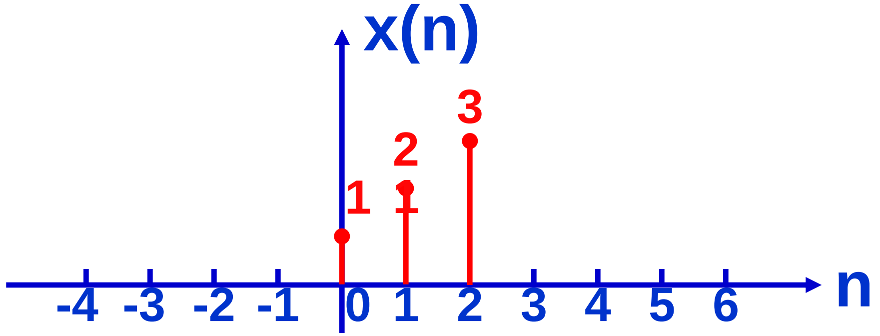
α=0.3, R=10000

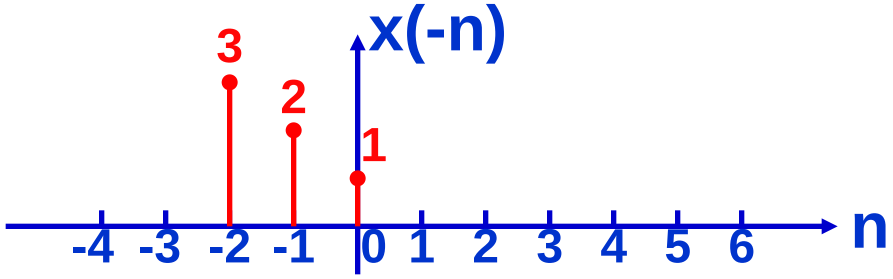
4、序列的反褶：

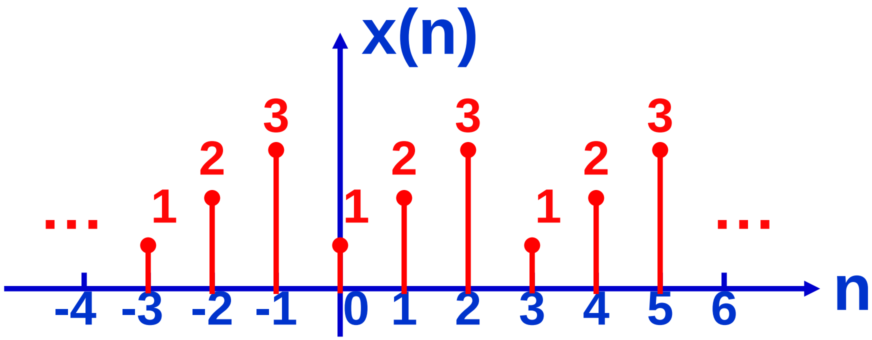
$$y(n) = x(-n)$$

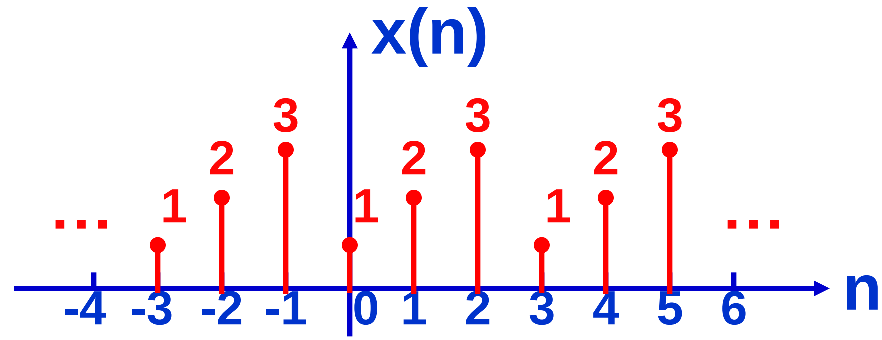
设有序列x(n)，

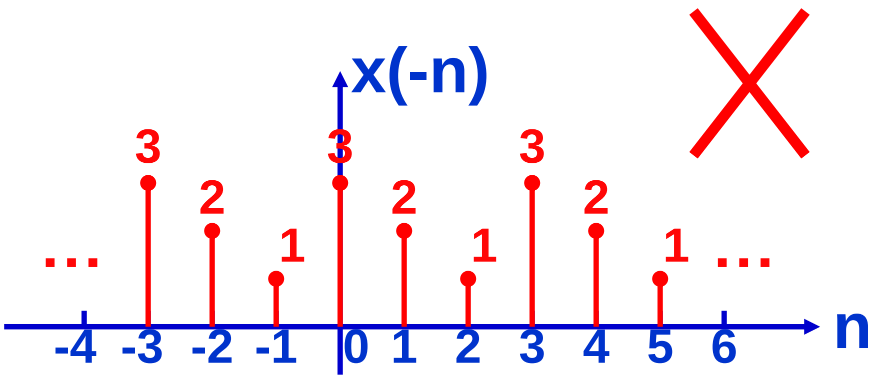
则x(-n)是以n=0为纵轴将x(n)反褶后的序列。

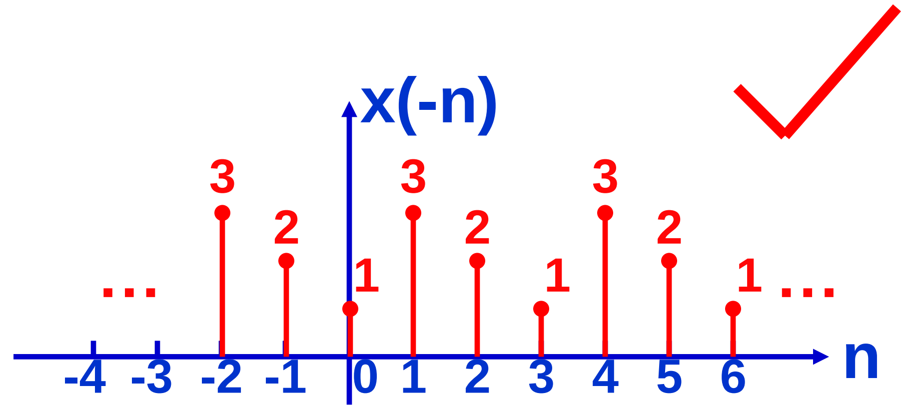




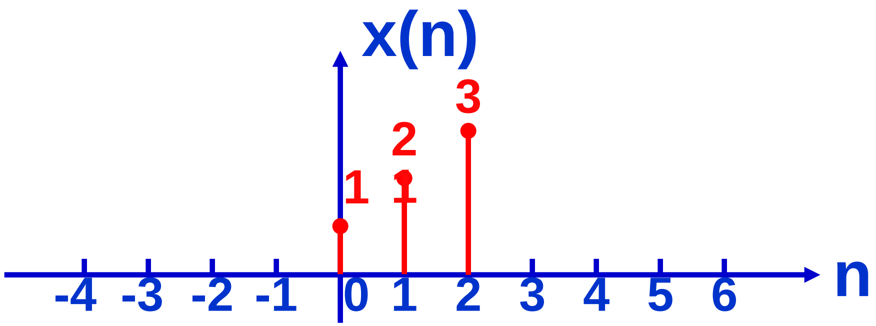

x(n)
n
… 1 2 3 1 2 3 1 2 3 …
-4 -3 -2 -1 0 1 2 3 4 5 6


x(n)
n
… 1 2 3 1 2 3 1 2 3 …
-4 -3 -2 -1 0 1 2 3 4 5 6

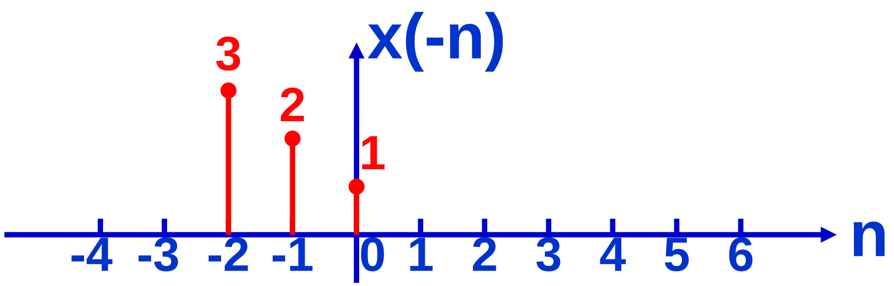

x(-n)
n
… 3 2 1 3 2 1 3 2 1 …
-4 -3 -2 -1 0 1 2 3 4 5 6

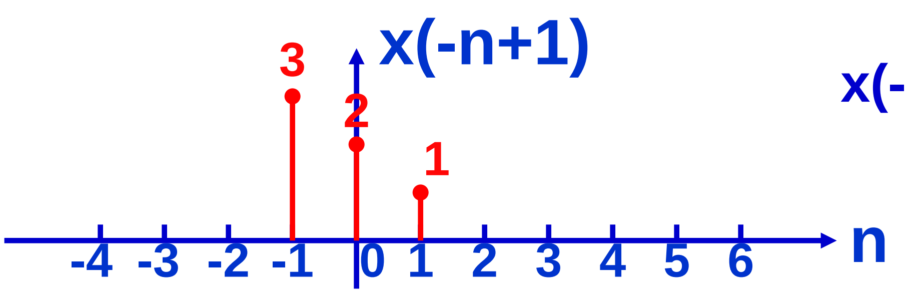

x(-n)
n
… 3 2 1 3 2 1 3 2 1 …
-4 -3 -2 -1 0 1 2 3 4 5 6

❖ 思考：x(-n+1)和x(-n-1)与x(-n)的移位关系？

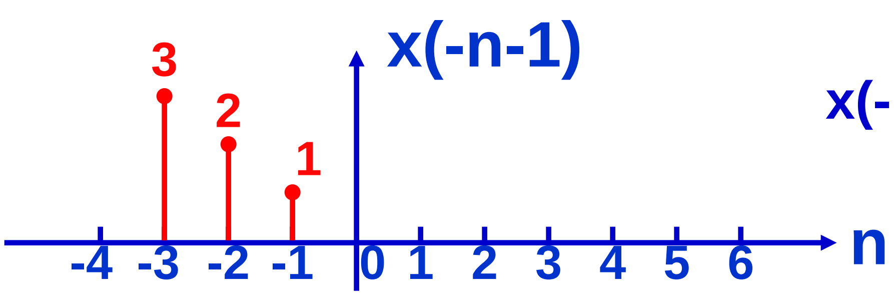


x(0)=1
x(1)=2
x(2)=3





x(-n+1) 是x(-n) 右移一位后的序列

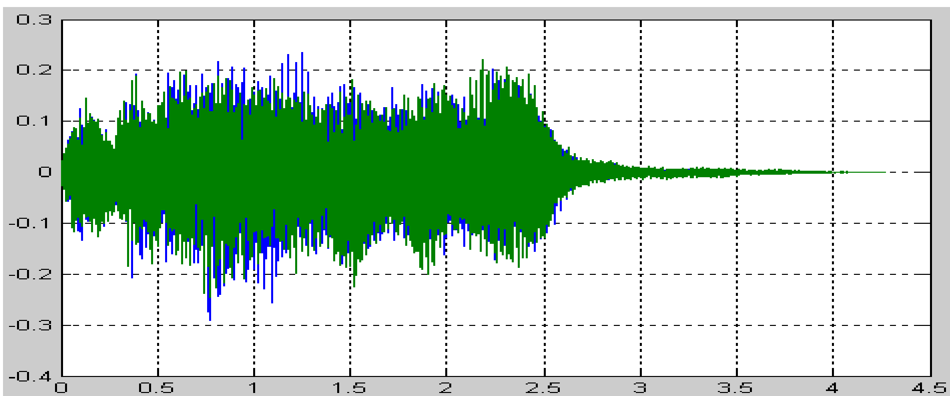


x(-n-1) 是x(-n) 左移一位后的序列

❖ 仿真实验(Matlab)

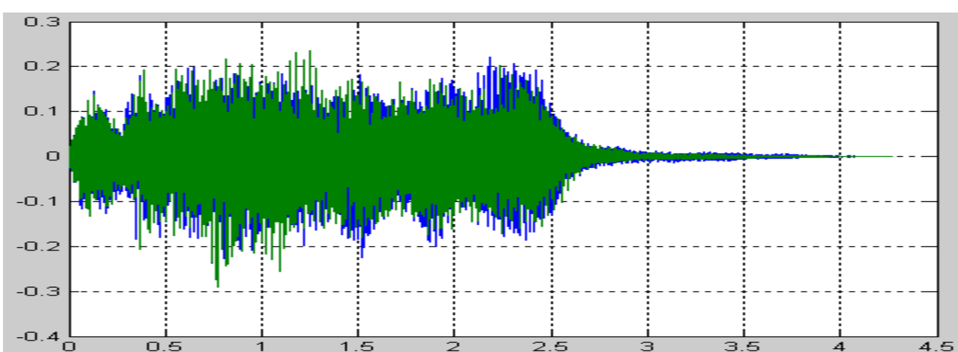
```
x = wavread(‘w2.wav’);          %读入声音文件
y = fliplr(x);                  %反褶
figure(1); plot(x); grid on;    %画图显示结果
figure(2); plot(y); grid on;
wavwrite(y,‘w4.wav’);           %结果保存为声音文件
```

x(n)



‘w av’

y(n)=x(-n)



w4.wav

5、累加

设序列x(n)，则x(n)的累加序列y(n)定义为：

$$y(n)=\sum_{k=-\infty}^{n}x(k)$$

它表示y(n)在某一个n0上的值等于这一个n0上的x(n0)以及n0从前的所有n值上的x(n)值之和。

例如：

$$x(n)=\begin{cases}\frac{1}{2}\left(\frac{1}{2}\right)^n & n\geq -1\\ 0 & n<-1\end{cases} \Longrightarrow y(n)=\begin{cases}\sum_{k=-1}^{n}\frac{1}{2}\left(\frac{1}{2}\right)^n & n\geq -1\\ 0 & n<-1\end{cases}$$

6、差分运算

- **前向差分：** $\Delta x(n) = x(n+1) - x(n)$
- **后向差分：** $\nabla x(n) = x(n) - x(n-1)$

$$\nabla x(n) = \Delta x(n-1)$$

差分运算反映了序列**x(n)**的幅值变化规律。

7、序列的时间尺度（比例）变换

设某序列为**x(n)**，则其时间尺度变换序列为**x(mn)**或 **x(n/m)**，**m**为正整数。

- **x(mn)** 为抽取序列
- **x(n/m)**为插值序列

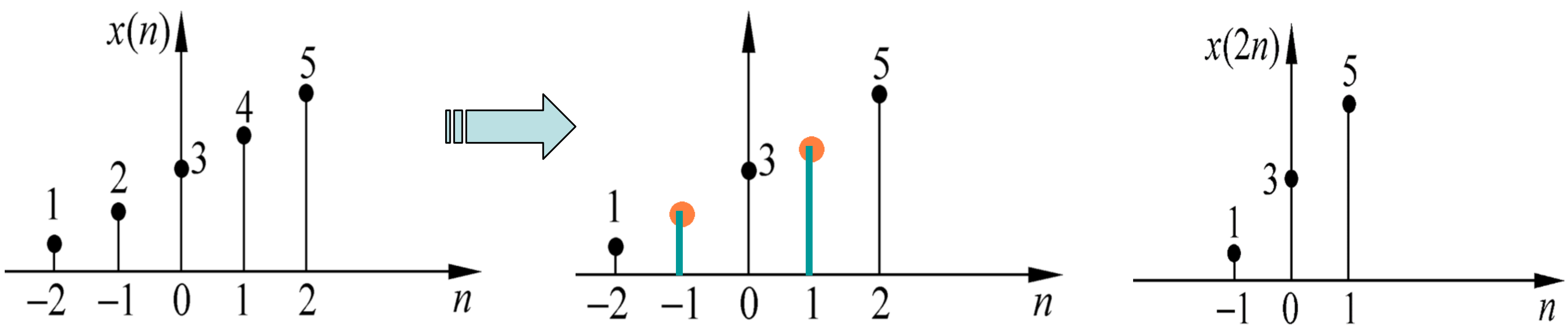
抽取序列：

$x(mn)$：对$x(n)$进行抽取运算

不是简单在时间轴上按比例增加到m倍

以$1/m$倍的取样频率每隔m-1个点抽取1点。

保留 $x(0)$



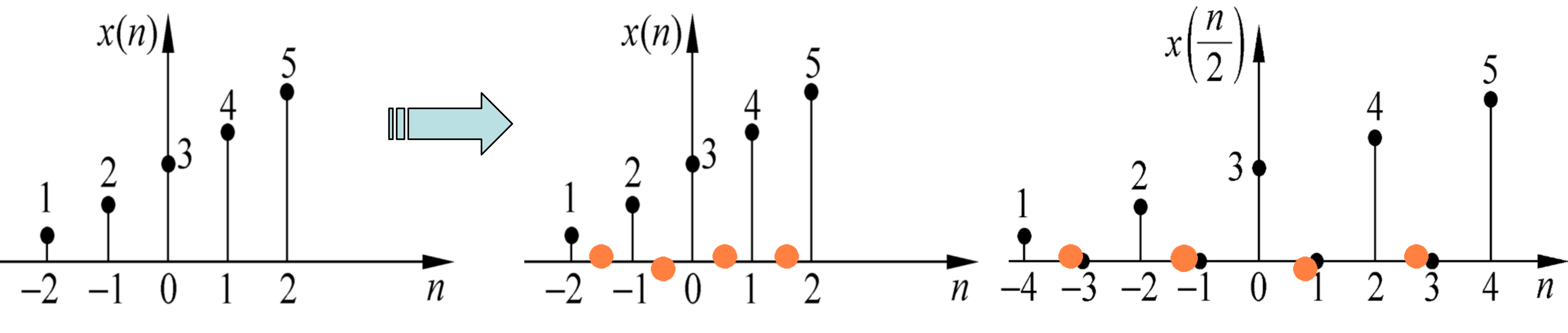
插值序列：

$x(n/m)$ ： 对$x(n)$进行插值运算

表示在原序列$x(n)$相邻两点之间插入

m-1个零值点

保留 $x(0)$



8、卷积和

- 卷积积分是求连续线性时不变系统输出响应的主要方法。

$$x(t) \longrightarrow \boxed{h(t)} \longrightarrow y(t) = x(t) * h(t) = \int_{-\infty}^{\infty} x(m)h(t-m)dm$$

- 卷积和是求离散线性时不变系统输出响应的主要方法。

$$x(n) \longrightarrow \boxed{h(n)} \longrightarrow y(n) = x(n) * h(n) = \sum_{m=-\infty}^{\infty} x(m)h(n-m)$$

多项式乘法

$$\left(x+1\right)\left(x^2+2x+5\right)=\left(x^3+2x^2+5x\right)+\left(x^2+2x+5\right)$$
$$=x^3+3x^2+7x+5$$

❖上式结果多项式中系数通过**先逐项相乘再合并同类项**的方法得到，需**两步操作**才行。

有没有办法一次操作就可以得到这些系数呢？

多项式相乘，相当于系数卷积

$$\begin{array}{r} x+1 \\ \underline{5+2x+x^2} \\ x^3 \end{array} \quad \rightarrow x^3$$

$$\begin{array}{r} x+1 \\ \underline{5+2x+x^2} \\ 2x^2+x^2=3x^2 \end{array} \quad \rightarrow 3x^2$$

$$\begin{array}{r} x+1 \\ \underline{5+2x+x^2} \\ 5x+2x=7x \end{array} \quad \rightarrow 7x$$

$$\begin{array}{r} x+1 \\ \underline{5+2x+x^2} \\ 5 \end{array} \quad \rightarrow 5$$

这种计算方法总结如下：

反褶：一般多项式都是按x的降幂排列，这里将其中一个多项式的各项按x的升幂排列。

平移：将按x的升幂排列的多项式每次向右平移一个项。

相乘：垂直对齐的项分别相乘

求和：相乘的各结果相加

反褶、平移、相乘、求和 - 这是 “卷积”的计算过程。

卷积的表达式

多项式 $x+1$ 的系数 $\left[a(1)a(0)\right]=[1\ 1]$

多项式 x^2+2x+5 的系数 $\left[b(2)b(1)b(0)\right]=[1\ 2\ 5]$

二者想乘所得的多项式 x^3+3x^2+7x+5 的系数

$$\left[c(3)c(2)c(1)c(0)\right]=[1\ 3\ 7\ 5]$$

利用上面的计算方法，我们很容易得到：

$$c(0)=a(0)b(0)$$
$$c(1)=a(0)b(1)+a(1)b(0)$$
$$c(2)=a(0)b(2)+a(1)b(1)+a(2)b(0)$$
$$c(3)=a(0)b(3)+a(1)b(2)+a(2)b(1)+a(3)b(0)$$

卷积公式推广

在上面的基础上推广一下：

$$c(4)=a(0)b(4)+a(1)b(3)+a(2)b(2)+a(3)b(1)+a(4)b(0)$$

假定两个多项式的系数分别为 $a(n),n=0\sim n1,b(n),n=0\sim n2$

这两个多项式相乘所得的多项式系数为才c（n），则；

$$c(n)=\sum_{k=0} a(k)b(n-k),n=0\sim(na+n2)$$

上面这个式子就是的$a(n),b(n)$**卷积的表达式**

通常我们把 $a(n),b(n)$ **的卷积记为：**$a(n)*b(n)$

其中*表示卷积运算符。

多项式乘法给了我们启发：如果信号可以分解为类似多项式的这种形式:

$a_2x^2+a_1x+a_0$, **同时满足** $x^n = f\left(n\omega_0\right)$

则两个信号相乘的结果就可以通过卷积计算

注：之所以强调， $x^n = f\left(n\omega_0\right)$ **是因为频谱分析通常关心各频率成分的大小（任何一个周期信号都可以表示为多个频率分量之和；直流分量，基波分量（角频率** $\omega_0 = 2\pi f_0$**），2次谐波分量（角频率为** $2\omega_0$ **），3次谐波分量（角频率为** $3\omega_0$ **）等等，所以我们希望多项式中的各项是** $n\omega_0$ **的函数。**

时域相乘，相当于频域卷积

上面这种把信号表示成形式类似于多项式的方法，本质上就是傅里叶级数展开，多项式中各项的系数实际就是傅里叶系数：

$$f(t)=\sum_{k=-\infty}^{+\infty} c_k e^{jk\omega_0 t}$$

信号的傅立叶级数展开

卷积和计算的四个步骤:

翻转：$x(m)$，$h(m) \rightarrow h(-m)$

移位：$h(-m) \rightarrow h(n-m)$

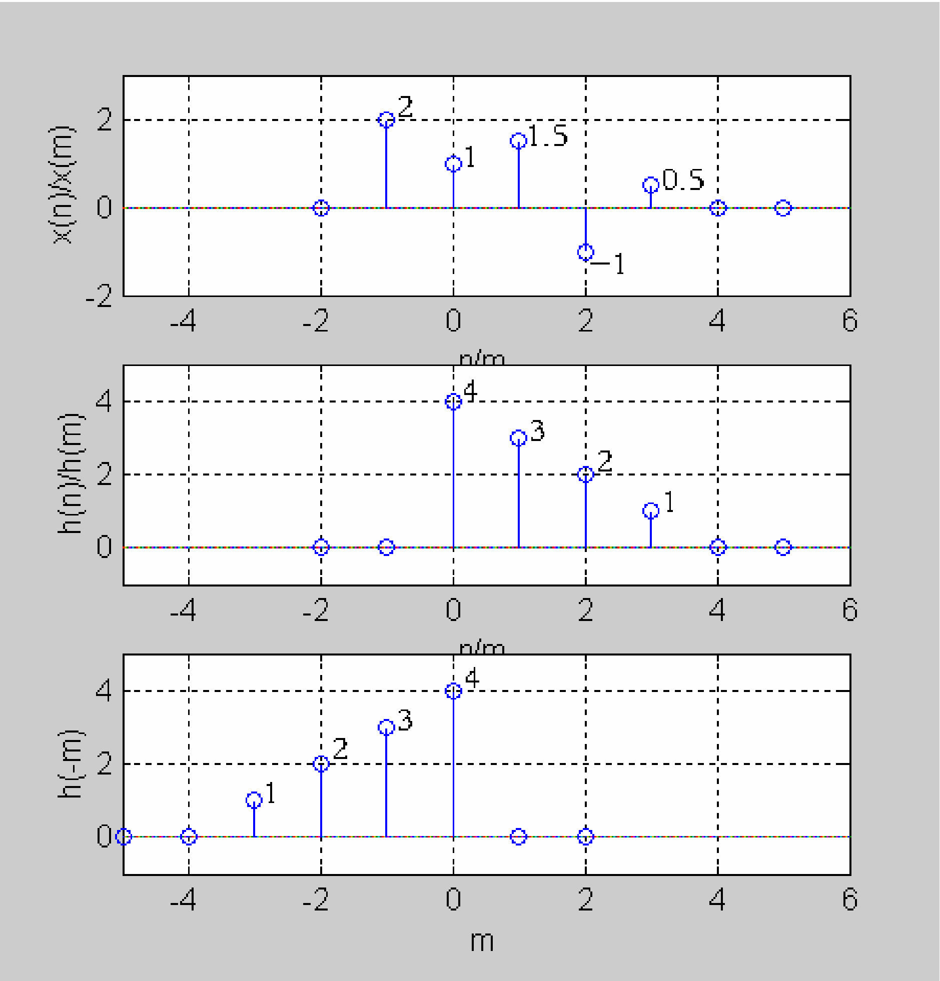
n为正数时，右移n位

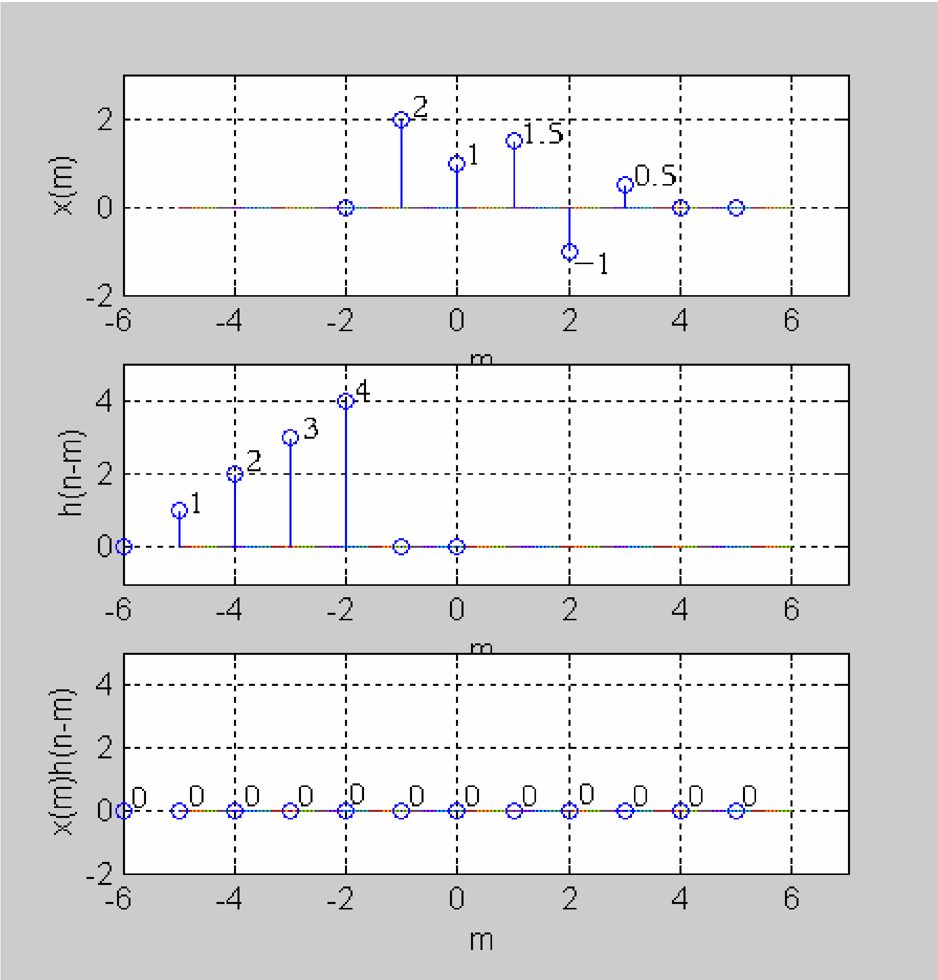
n为负数时，左移n位

相乘：$h(n-m) \cdot x(m)$（m值相同）

相加：$y(n) = \sum \{h(n-m) \cdot x(m)\}$

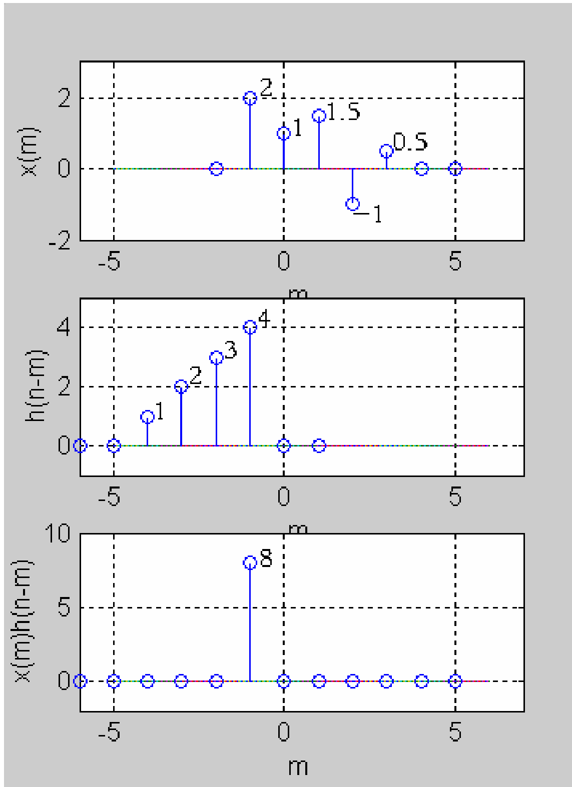
举例说明卷积过程





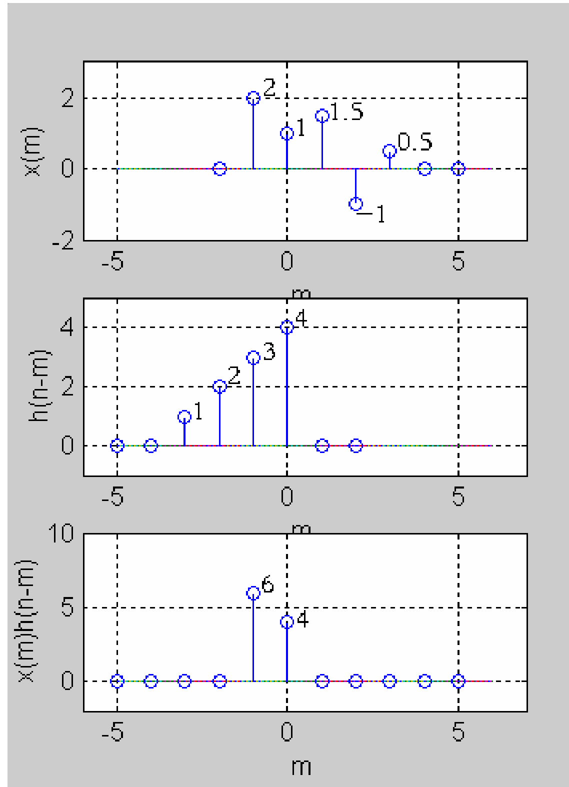
$n \leq -2,\ y(n)=0$

n=-1



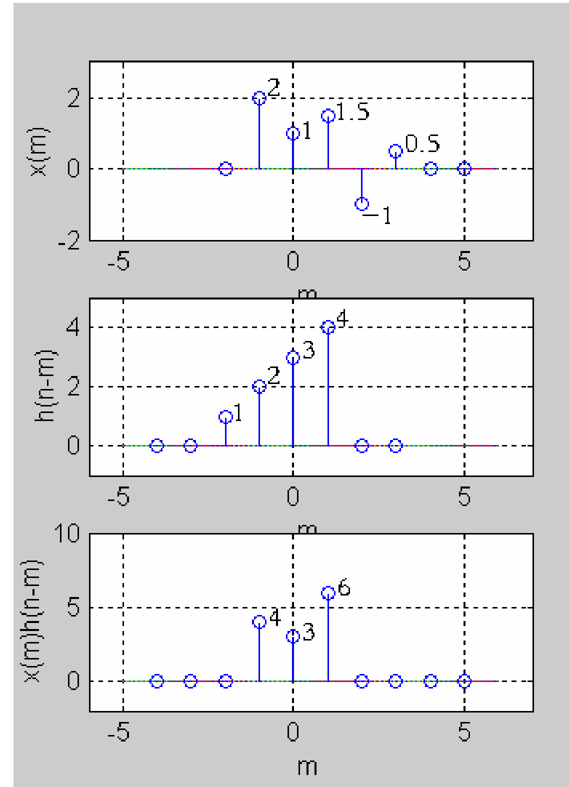
y(-1)=8

n=0



y(0)=6+4=10

n=1

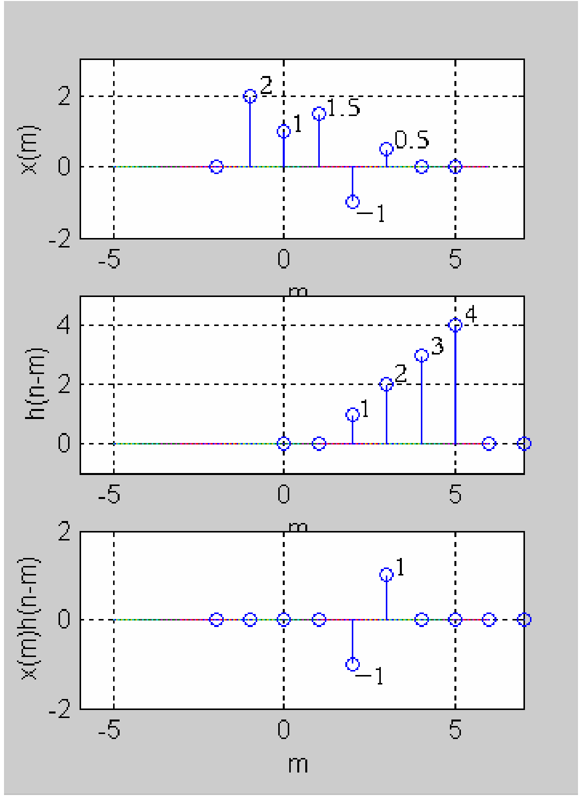


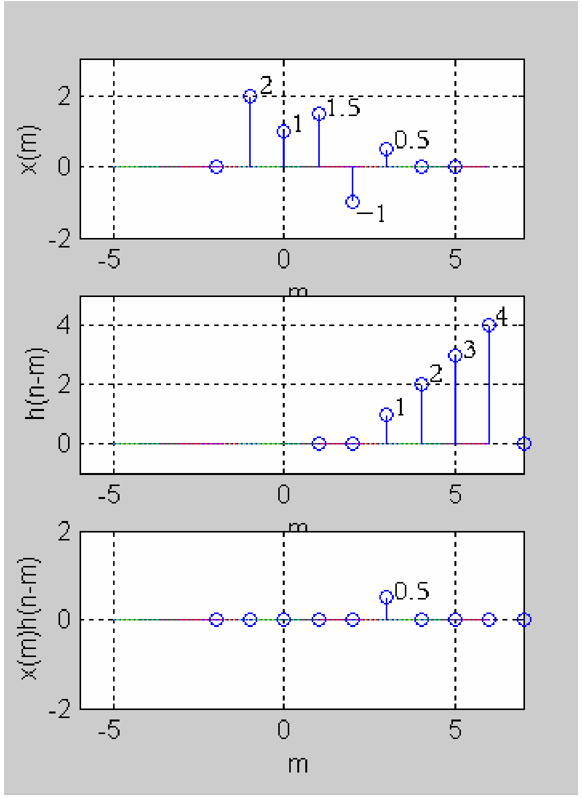
y(1)=4+3+6=13

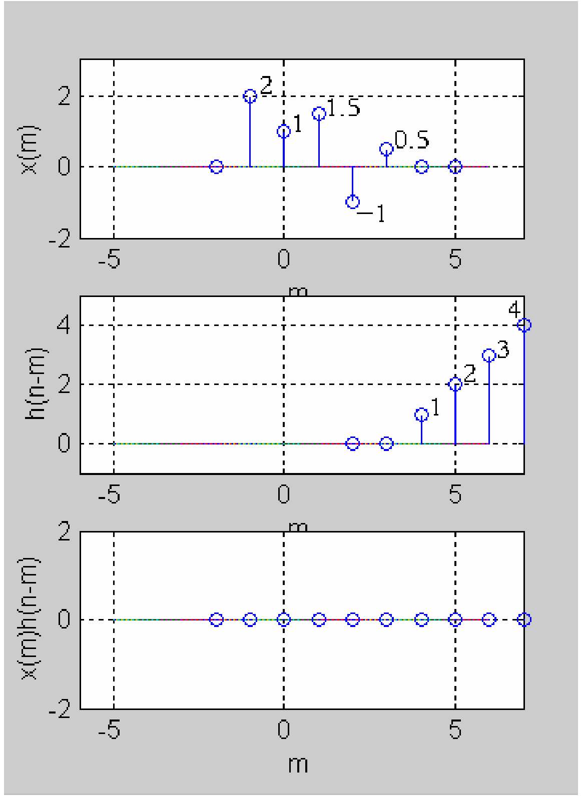
n=5

n=6

n=7



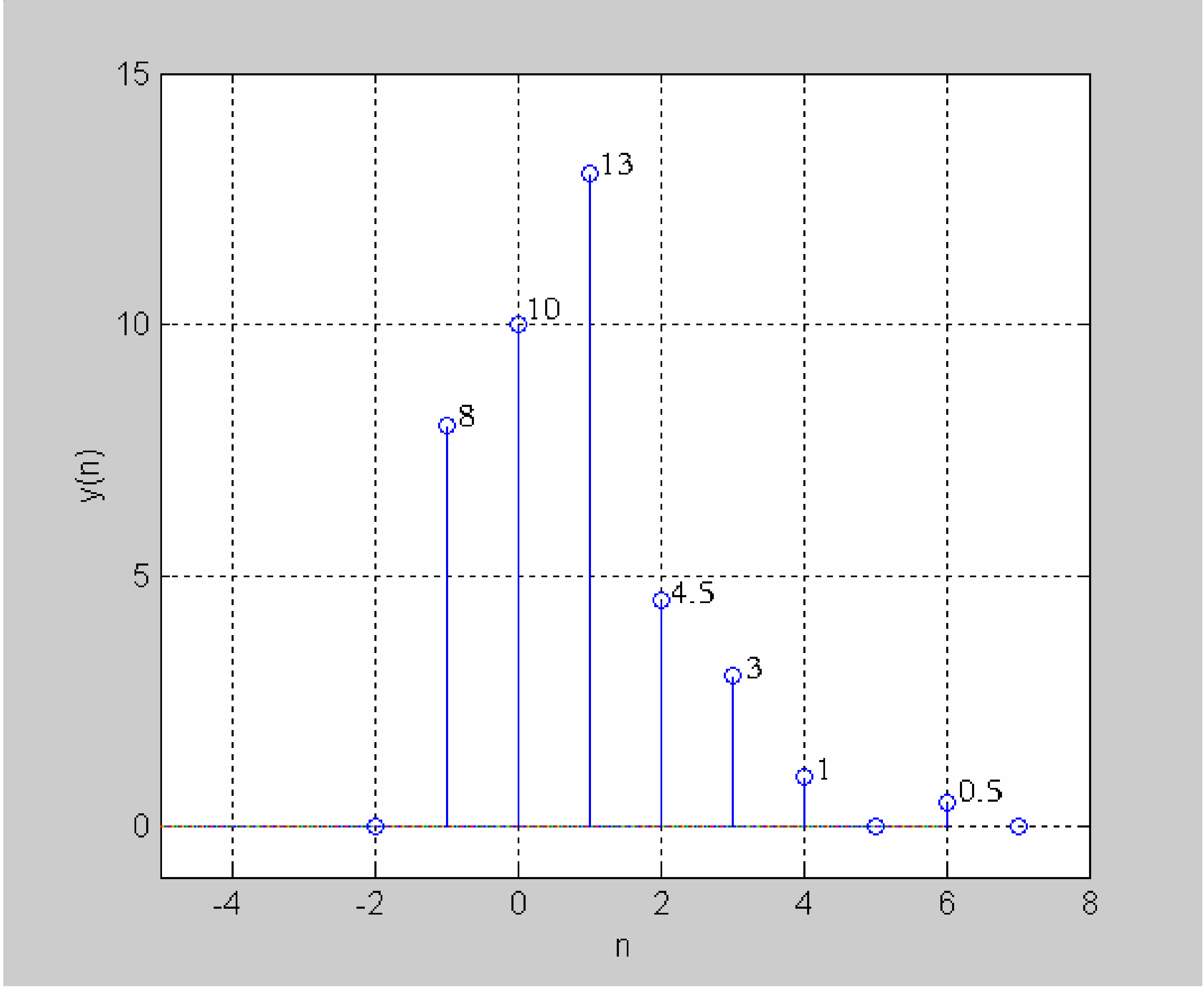




y(5)=-1+1=0

y(6)=0.5

y(n)=0, n ≥ 7


15
10
5
0
y(n)
13
10
8
4.5
3
1
0.5
-4
-2
0
2
4
6
8
n

卷积和与两序列的前后次序无关

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

$$= \sum_{n-k=-\infty}^{\infty} x(n-k)h(k)$$

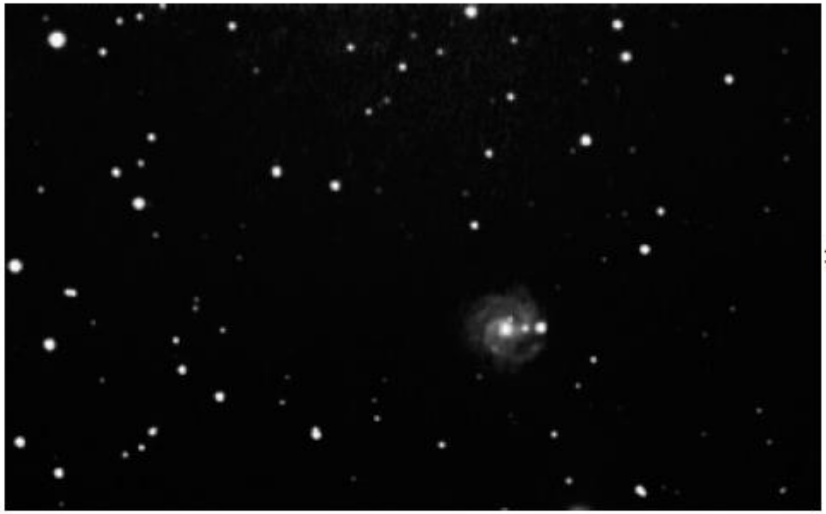
$$n - m = k$$

$$m = n - k$$

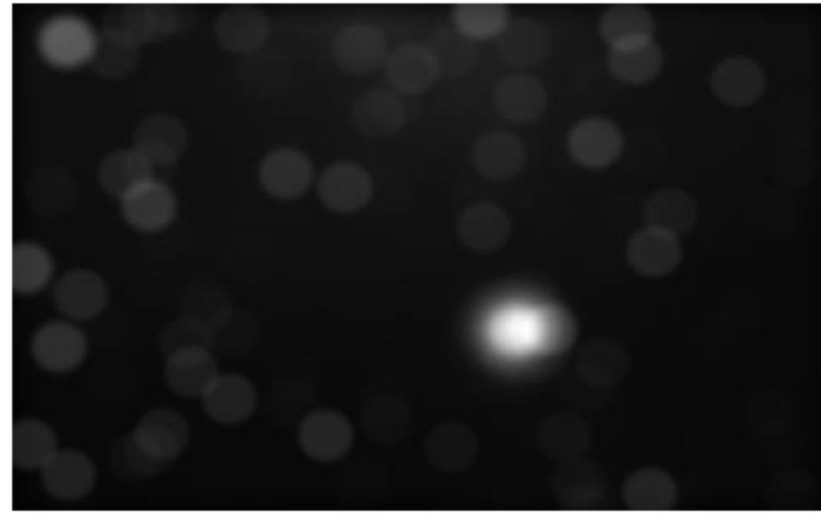
$$= \sum_{k=-\infty}^{\infty} h(k)x(n-k) = h(n) * x(n)$$

Convolution: optics example

If a projective lens is out of focus, the blurred image is equal to the original image convolved with the aperture shape (e.g., a filled circle):



$*$ ● $=$

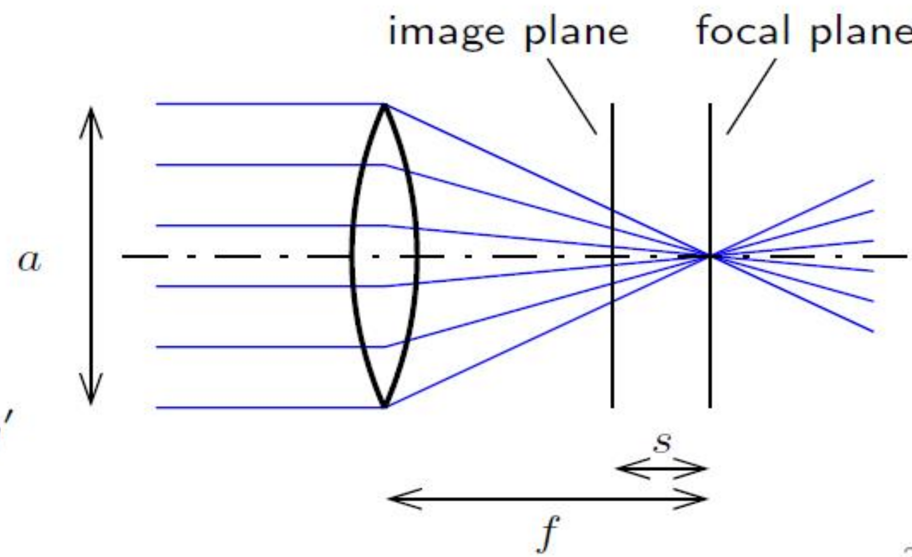


Point-spread function h (disk, $r = \frac{as}{2f}$):

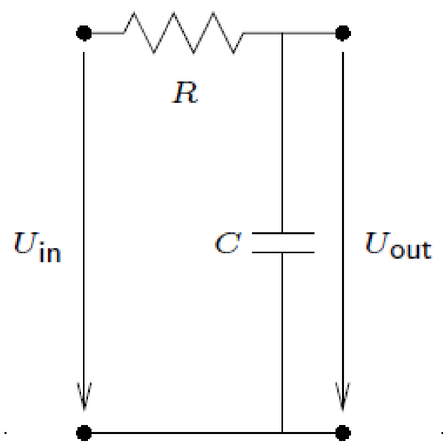
$$h(x, y) = \begin{cases} \frac{1}{r^2\pi}, & x^2 + y^2 \leq r^2 \\ 0, & x^2 + y^2 > r^2 \end{cases}$$

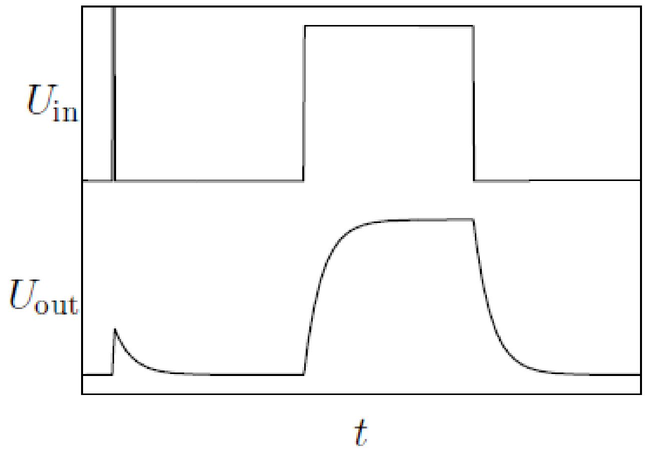
Original image I, blurred image $B = I * h$, i.e.

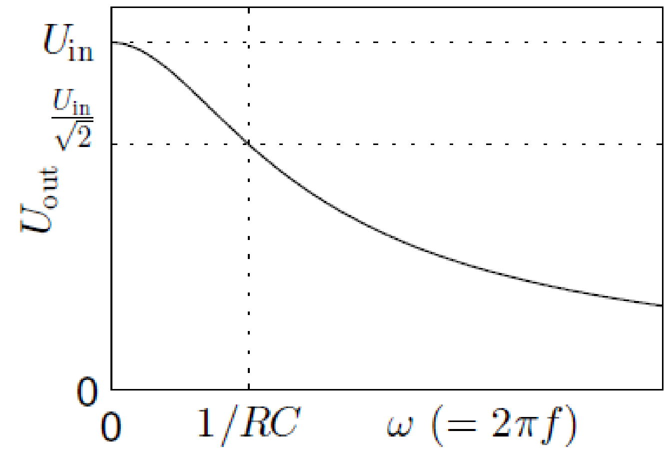
$$B(x, y) = \iint I(x - x', y - y') \cdot h(x', y') \cdot \mathrm{d}x' \mathrm{d}y'$$



Convolution: electronics example







Any passive network (R, L, C) convolves its input voltage U_{in} with an *impulse response function* h, leading to $U_{\text{out}} = U_{\text{in}} * h$, that is

$$U_{\text{out}}(t) = \int_{-\infty}^{\infty} U_{\text{in}}(t - \tau) \cdot h(\tau) \cdot \mathrm{d}\tau$$

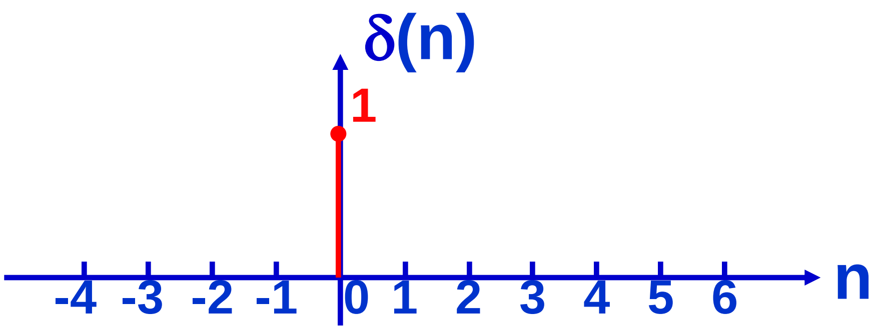
In the above example:

$$\frac{U_{\text{in}} - U_{\text{out}}}{R} = C \cdot \frac{\mathrm{d}U_{\text{out}}}{\mathrm{d}t}\,, \qquad h(t) = \begin{cases} \frac{1}{RC} \cdot \mathrm{e}^{\frac{-t}{RC}}, & t \geq 0 \\ 0, & t < 0 \end{cases}$$

四、常用的典型序列

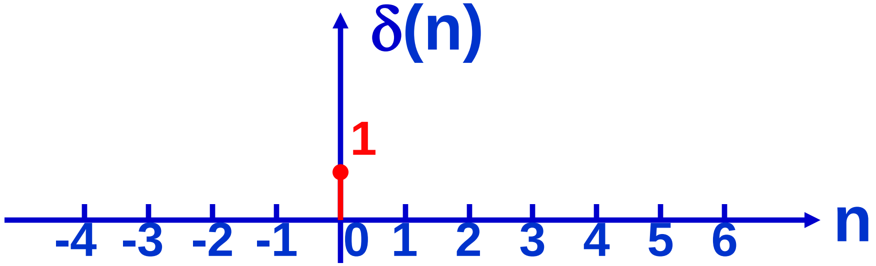
1、单位取样序列δ(n)　　-Unit sample sequence

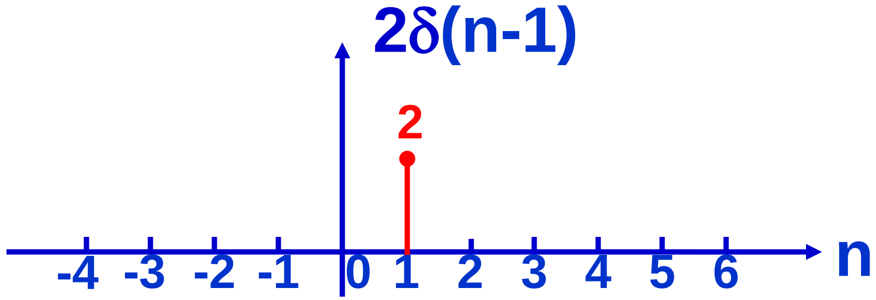
$$\delta(n)=\begin{cases}1 & n=0\\ 0 & n\neq 0\end{cases}$$

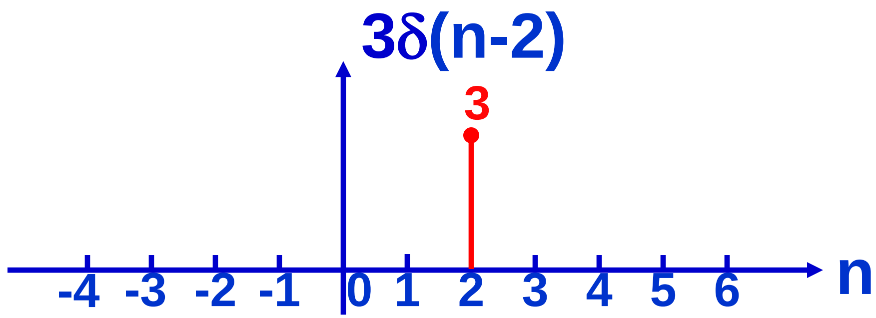


- δ(n)是一个脉冲幅度为1的现实序列。
- δ(t)是脉宽为零，幅度为 ∞ 的一种数学极限，是非现实信号。
- 单位取样序列亦称单位脉冲序列，或时域离散冲激。

❖ 用单位取样序列δ(n)表示任意序列



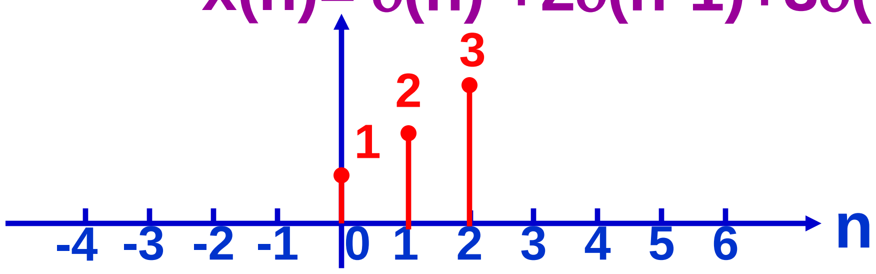




可以将任意序列表示成单位抽样序列的移位加权和

$$x(n)=\sum_{m=-\infty}^{\infty} x(m)\delta(n-m)$$

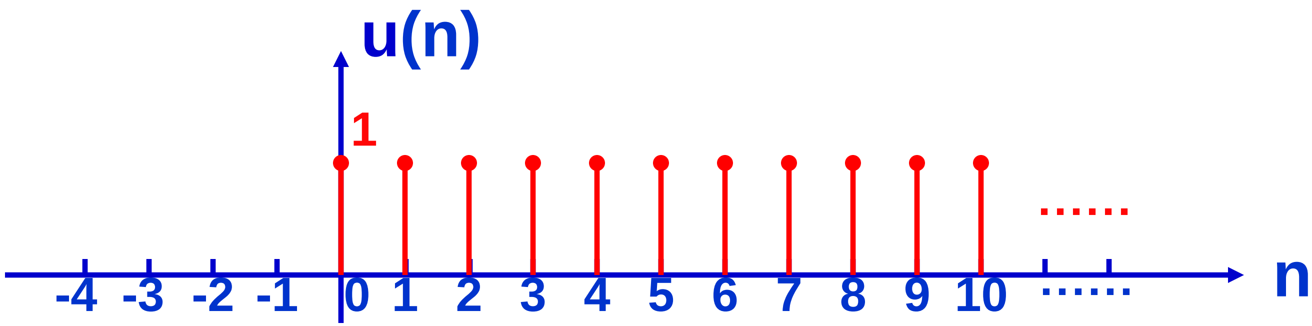
$x(n)=\delta(n)+2\delta(n-1)+3\delta(n-2)=\sum_{m=0}^{2} x(m)\delta(n-m)$



(其中，x(0)=1, x(1)=2, x(2)=3)

2、单位阶跃序列u(n) -Unit step sequence

$$u(n) = \begin{cases} 1 & n \geq 0 \\ 0 & n < 0 \end{cases}$$



❖ **用单位阶跃序列u(n)表示单位取样序列δ(n):**

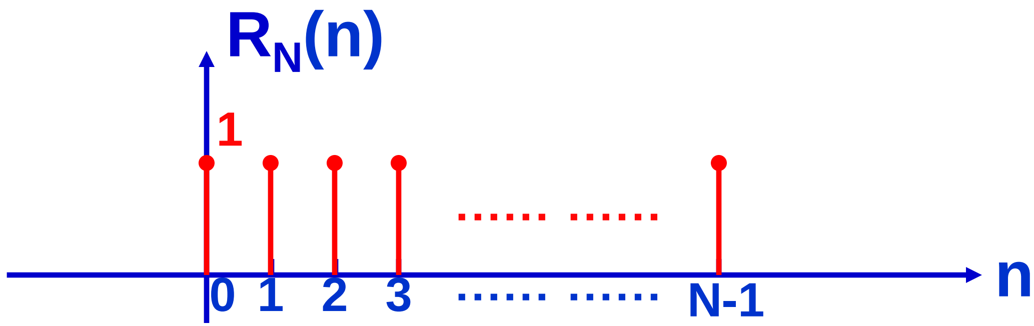
$$\delta(n) = u(n) - u(n-1)$$

❖ **用单位取样序列δ(n)表示单位阶跃序列u(n):**

$$u(n) = \sum_{m=0}^{\infty} \delta(n-m)$$

3、矩形序列$R_N(n)$ - Rectangular sequence

$$R_N(n)=\begin{cases}1 & 0\le n\le N-1\\ 0 & n\end{cases}$$



❖ 用单位阶跃序列u(n)表示矩形序列$R_N(n)$：

$$R_N(n)=u(n)-u(n-N)$$

❖ 用单位取样序列δ(n)表示矩形序列$R_N(n)$：

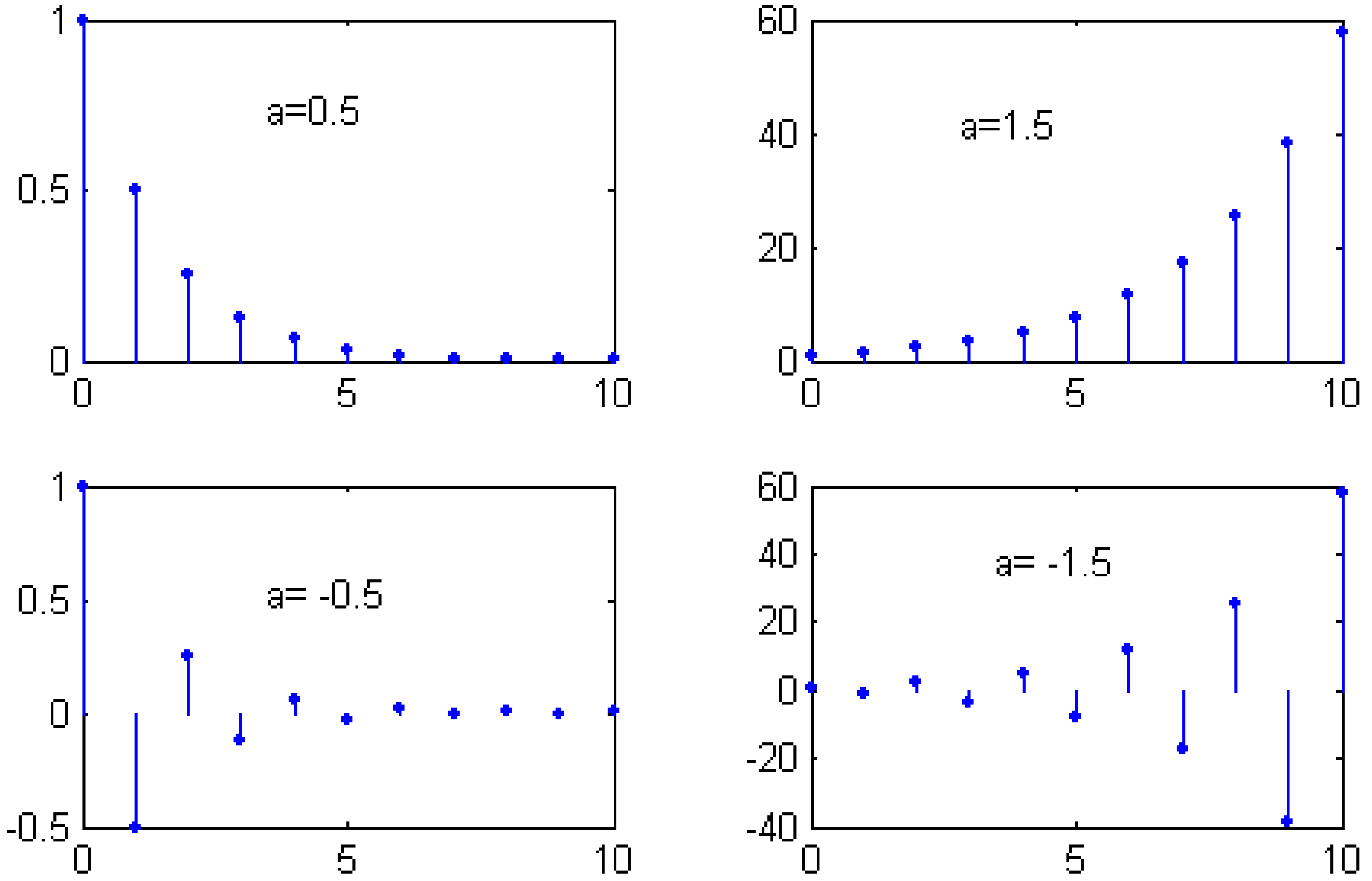
$$R_N(n)=\sum_{m=0}^{N-1}\delta(n-m)$$

4、实指数序列 Real-valued exponential sequence

$$\boxed{x(n) = a^n u(n) = \begin{cases} a^n & n \geq 0 \\ 0 & n < 0 \end{cases}}$$

- 当|a|≥1时，序列发散。
- 当|a|< 1时，序列收敛。
- 当|a|< 1，且a<0时，序列是摇动的

实指数序列



Why are exponential functions useful?

Adding together two exponential functions with the same base z, but different scale factor and offset, results in another exponential function with the same base:

$$\begin{aligned} A_1 \cdot z^{t+\varphi_1} + A_2 \cdot z^{t+\varphi_2} &= A_1 \cdot z^t \cdot z^{\varphi_1} + A_2 \cdot z^t \cdot z^{\varphi_2} \\ &= (A_1 \cdot z^{\varphi_1} + A_2 \cdot z^{\varphi_2}) \cdot z^t = A \cdot z^t \end{aligned}$$

Likewise, if we convolve a sequence $\{x_n\}$ of values

$$\ldots, z^{-3}, z^{-2}, z^{-1}, 1, z, z^2, z^3, \ldots$$

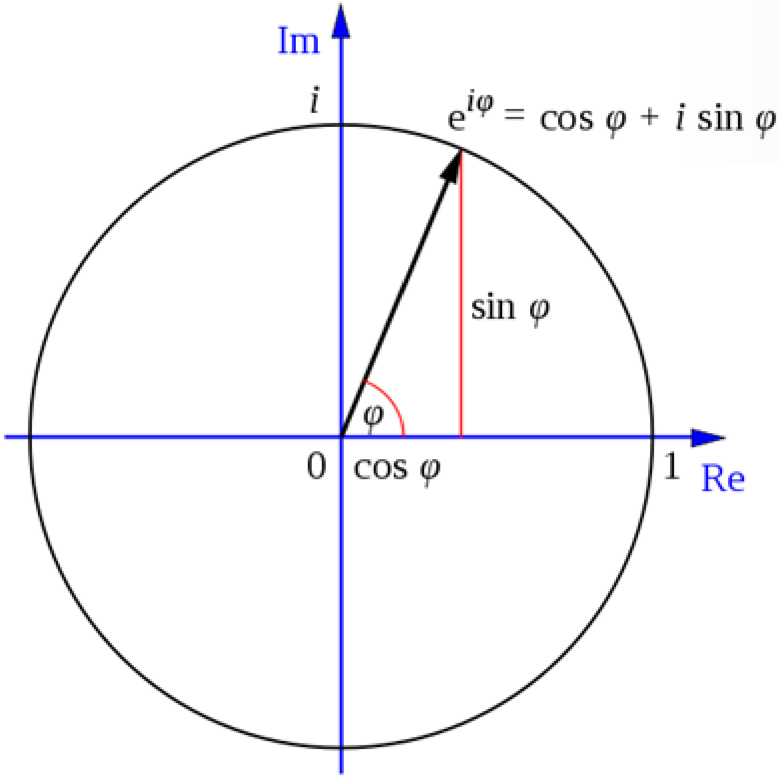
$x_n = z^n$ with an arbitrary sequence $\{h_n\}$, we get $\{y_n\} = \{z^n\} * \{h_n\}$,

$$y_n = \sum_{k=-\infty}^{\infty} x_{n-k} \cdot h_k = \sum_{k=-\infty}^{\infty} z^{n-k} \cdot h_k = z^n \cdot \sum_{k=-\infty}^{\infty} z^{-k} \cdot h_k = z^n \cdot H(z)$$

where $H(z)$ is independent of n.

Exponential sequences are closed under convolution with arbitrary sequences.

The same applies in the continuous case.



高斯曾经说：“一个人第一次看到欧拉公式而不感到它的魅力，他不可能成为数学家。” 虽然不敢肯定她是世界上“最伟大公式”，但是可以肯定它是最完美的数学公式之一。

为什么欧拉公式被称为世界上最完美的公式了？

1、自然数的“e”含于其中。　自然对数的底，大到飞船的速度，小至蜗牛的螺线，谁能够离开它？**有了 e 就有了微积分。**

2、最重要的常数 π 含于其中。　世界上最完美的平面对称图形是圆。**有了 π 就有了圆函数，也就是三角函数**（π和e是两个最重要的无理数！）

3、最重要的运算符号 + 含于其中。**有了加号，可以得到其余运算符号**。减号是加法的逆逆运算，乘法是累计的加法……

4、最重要的关系符号 = 含于其中。　从你一开始学算术，最先遇见它，相信你也会同意这句话。

5、最重要的两个元在里面。零元0，单位1，是构造群，环，域的基本元素，**有了0，1，就可以得到其他的数字**。

6、最重要的虚单位 i 也在其中。　虚单位 i 使数轴上的问题扩展到了平面，**有了 i 就有了虚数，平面向量与其对应，也就有了哈密尔的 4 元数，现实的空间与其对应。**

5、正弦序列 -Sinusoidal sequence

$$x(n) = \sin n\omega_0, \quad x(n) = \cos n\omega_0$$

- **正弦序列的由来**

对连续时间正弦信号取样可以得到正弦序列。

$$\boxed{\sin \Omega_0 t} \xrightarrow[\text{取样}]{t = nT} \boxed{\sin \Omega_0 nT = \sin n\omega_0}$$

其中， $\omega_0 = \Omega_0 T$ **，T是取样间隔(取样周期)。**

ω_0**称为数字域频率，** Ω_0**称为模拟域频率。**

Why are sine waves useful?

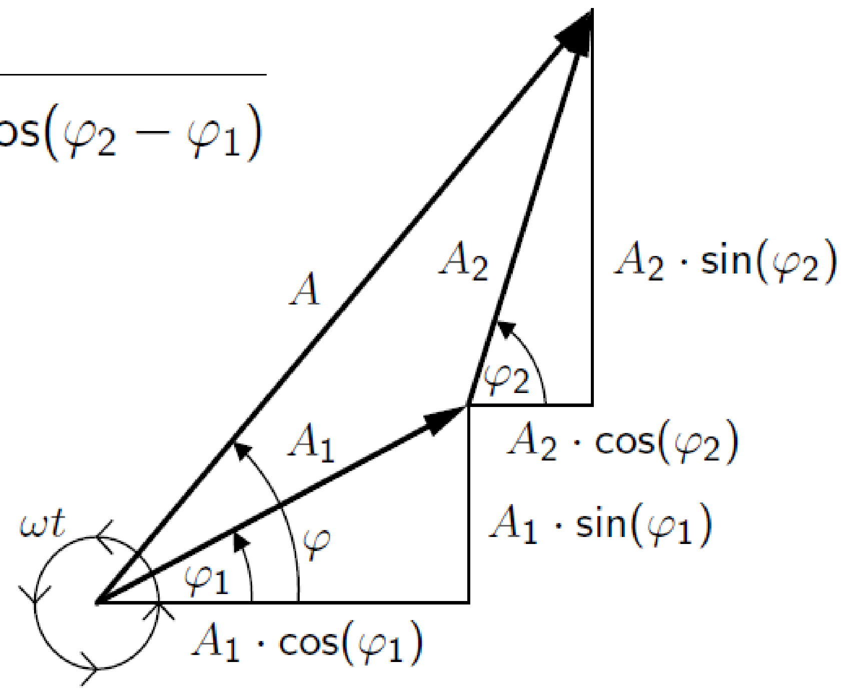
1) Adding together sine waves of equal frequency, but arbitrary amplitude and phase, results in another sine wave of the same frequency:

$$A_1 \cdot \sin(\omega t + \varphi_1) + A_2 \cdot \sin(\omega t + \varphi_2) \; = \; A \cdot \sin(\omega t + \varphi)$$

with

$$A = \sqrt{A_1^2 + A_2^2 + 2A_1A_2\cos(\varphi_2 - \varphi_1)}$$

$$\tan\varphi = \frac{A_1\sin\varphi_1 + A_2\sin\varphi_2}{A_1\cos\varphi_1 + A_2\cos\varphi_2}$$



Sine waves of any phase can be formed from sin and cos alone:

$$A \cdot \sin(\omega t + \varphi) = a \cdot \sin(\omega t) + b \cdot \cos(\omega t)$$

with $a = A \cdot \cos(\varphi)$, $b = A \cdot \sin(\varphi)$ and $A = \sqrt{a^2 + b^2}$, $\tan\varphi = \frac{b}{a}$.

➢ 数字域频率和模拟域频率

❖ 数字域频率是模拟域频率的T倍，以后我们就以ω表示数字域频率，Ω表示模拟域频率（Ω也表示模拟域角频率，$\Omega = 2\pi f$，f表示模拟域线频率）。

❖ 当序列是周期的时，表示正弦序列的序列值重复变化的快慢。

例：$\omega = 0.01\pi$，则序列值每200个重复一次正弦循环

$\omega = 0.1\pi$，则序列值每20个重复一次正弦循环

❖ Ω的量纲为弧/秒，ω的量纲为弧。

5、复指数序列 Complex-valued exponential sequence

$$\boxed{x(n) = e^{j\omega n} = \cos\omega n + j\sin\omega n}$$

❖ **复指数序列$e^{j\omega n}$作为序列分解的基单元，在序列的傅里叶分析中起着重要的作用。**

- 一个系统输入复指数信号，输出必定也是复指数信号。
- 利用复指数信号可描述各种基本信号，如直流信号、指数信号、正弦或余弦信号以及增长或衰减的正弦与余弦信号。利用复指数信号可使许多运算和分析得以简化。

五、序列的周期性

1、定义

如果对于所有n存在一个最小的正整数N，使得：

$$x(n) = x(n+N)$$

成立，则称x(n)为周期序列，周期为N。

2、正弦序列的周期性

正弦信号： $x(n) = A\sin(n\omega_0 + \varphi)$

$$\Longrightarrow x(n+N) = A\sin[(n+N)\omega_0 + \varphi]$$

$$= A\sin[N\omega_0 + n\omega_0 + \varphi]$$

若 $N\omega_0 = 2k\pi$，当k为整数时(即$N\omega_0$为2π的整数倍)

，则有：x(n)=x(n+N)，x(n)为周期信号。

➢ 观察 $N\omega_0 = 2k\pi$： (即 $N = \frac{2\pi}{\omega_0} \cdot k$)

(1) 当 $2\pi/\omega_0$ 为整数时：

k=1，则N= $2\pi/\omega_0$ 为最小整数，且保证x(n)=x(n+N)。

例 序列 $x(n) = 5\sin(\frac{\pi}{4}n + 3)$ 因为$2\pi/\omega = 8$，所以是一个周期序列，其周期$N = 8$。

(2)2π/ω为有理数而非整数时，仍然是周期序列，周期大于2π/ω

例 序列 $x(n) = 2\cos(\frac{3\pi}{4}n + 7)$ $2\pi/\omega = 8/3$是有理数，所以是周期序列，取$k = 3$，得到周期$N = 8$。

(3) 当 $2\pi/\omega_0$ 为无理数时：

任何k都不能使N为整数，此时x(n)不是周期性的。

3、讨论

一个正弦序列若由一个连续正弦信号抽样而得，那么抽样时间间隔 T 和连续正弦信号的周期 T_0 之间应该是什么关系才能使所得到的抽样序列仍为周期序列？

设连续正弦信号为x(t)：

$$x(t) = A\sin(\Omega_0 t + \phi)$$

连续信号x(t)的角频率为 $\Omega_0 = 2\pi f_0$

连续信号x(t)的周期为 $T_0 = \frac{1}{f_0} = \frac{2\pi}{\Omega_0}$

若对x(t)抽样，设抽样时间间隔为T，有：

$$x(n) = x(t)|_{t=nT} = A\sin(\Omega_0 nT + \phi)$$

若令ω_0为数字频率，它满足：

$$\omega_0 = \Omega_0 T = \Omega_0 \frac{1}{f_s} = 2\pi \frac{f_0}{f_s} = 2\pi \frac{T}{T_0}$$

其中f_s是抽样频率， ω_0是相对频率，是连续信号角频率Ω_0相对抽样频率f_s的频率。

$$\Longrightarrow \quad x(n) = x(t)\big|_{t=nT} = A\sin(\Omega_0 nT + \phi)$$

$$= A\sin(\omega_0 n + \phi)$$

在分析一个序列的周期性时，是通过分析$2\pi/\omega_0$的值来实现的。

$$\frac{2\pi}{\omega_0}=\frac{2\pi}{\Omega_0 T}=\frac{T_0}{T}$$

(1) 当 $2\pi/\omega_0$ 为整数时：

$$\frac{2\pi}{\omega_0}=\frac{T_0}{T}=N$$

说明:连续正弦信号x(t)的周期T_0是抽样间隔的整数倍，或者说，是在一个连续信号的周期T_0内以T为采样间隔采样了N个点。

(2) 当 $2\pi/\omega_0$ 为有理数时：

$$\frac{2\pi}{\omega_0}=\frac{T_0}{T}=\frac{N}{K} \quad\Longrightarrow\quad NT=KT_0$$

说明：在K个连续正弦信号x(t)的周期T_0内以T为采样间隔采样了N个点。

例如：序列

$$x(n)=\sin\left(\frac{3}{7}\pi n\right) \qquad \frac{2\pi}{\omega_0}=\frac{N}{K}=\frac{T_0}{T}=\frac{14}{3}$$

x(n)的周期是14，在3个连续信号周期T_0内采样了14个点。