(3). $e^{-at}u(-t)$, $a>0$

$$X_b(s) = \int_{-\infty}^{0} e^{-(a+s)t}\,dt = \frac{1}{a+s}e^{-(a+s)t}\Big|_{-\infty}^{0} = \frac{1}{a+s} \quad (\sigma < -a)$$

(6). $e^{-t}u(t-T)$

$\sigma > 1$

$$u(t) \leftrightarrow \frac{1}{s}$$

$$u(t-T) \leftrightarrow e^{-sT_0}\cdot\frac{1}{s}$$

$$e^{-t}u(t-T) \leftrightarrow e^{-(s+1)T_0}\cdot\frac{1}{s+1}$$

36、(7). $x(t)=\begin{cases}1 & 0\le t\le 1\\ 0 & 其他\end{cases}$

$$X_b(s)=\int_0^1 e^{-st}\cdot dt=-\frac{1}{s}e^{-st}\Big|_0^1=\frac{1-e^{-s}}{s}$$

零点：$1-e^{-s}=0$，即 $e^{-s}=1$ $\Rightarrow$ $e^{-s}=e^{j\cdot 2k\pi}$

$\therefore$ $s=j\cdot 2k\pi$，$k=0,\pm1,\pm2\cdots$

极点：$s=0$

其中 $s=0$ 点既为零点，又为极点，相抵消。

收敛域为整个 s 平面

40、(1). $x(t)$的傅里叶变换存在，则收敛域需包含$j\omega$轴，由零极点图可知：

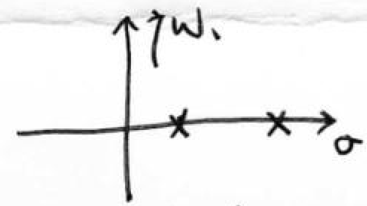
$$-1<\sigma<1$$

(2). $x(t)\cdot e^{2t}$的傅里叶变换存在。

$x(t)$的拉氏变换为 $X_L(s)=\dfrac{k}{(s+1)(s-1)}$（由零极点图得到）

$x(t)\cdot e^{2t}$的拉氏变换为 $\dfrac{k}{(s+1-2)(s-1-2)}=\dfrac{k}{(s-1)(s-3)}$

该拉氏变换的极点为 $s=1$ 与 $s=3$.



若该函数的傅里叶变换存在，则收敛域需包含$j\omega$轴。

∴收敛域为 $\sigma<1$

(3). $x(t)=0$, $t>0$.

该信号为~~右~~左边信号.　　$\sigma<-1$

(4). $x(t)=0$, $x<5$.

该信号为右边信号.　　$\sigma>1$

第三章 离散信号的分析

方璐 2教322
杭州电子科技大学 自动化学院

第三节 快速傅里叶变换（FFT）

（Fast Fourier Transform）

一、基本思路

二、基2FFT算法

三、FFT的应用

一、基本思路

DFT 的运算量：

$$DFT: X(k)=\sum_{n=0}^{N-1} x(n)e^{-j\Omega_0 nk}=\sum_{n=0}^{N-1} x(n)W_N^{\ nk}, \quad 0\le k\le N-1$$

$$\mathrm{W_N=e^{-j\Omega_0}=e^{-j\frac{2\pi}{N}}}$$ （指数因子，旋转因子或加权因子）

- 对每个特定 k，$X(k)$的 **DFT的计算：有(N—1) 个复加和N 个复乘**
- 计算整个**DFT** 共有N (N-1)个复加和N^2 个复乘
- 计算每次复乘，需要**4**次实数乘法和**2**次实数加法

$$X(k)=\sum_{n=0}^{N-1} x(n)W_N^{\ nk}=\sum_{n=0}^{N-1}\{\mathrm{Re}[x(n)]+j\,\mathrm{Im}[x(n)]\}\{\mathrm{Re}[W_N^{nk}]+j\,\mathrm{Im}[W_N^{nk}]\}$$

$$=\sum_{n=0}^{N-1}\{\mathrm{Re}[x(n)]\mathrm{Re}[W_N^{nk}]-\mathrm{Im}[x(n)]\mathrm{Im}[W_N^{nk}]+j(\mathrm{Re}[x(n)]\mathrm{Im}[W_N^{nk}]+\mathrm{Im}[x(n)]\mathrm{Re}[W_N^{nk}])\}$$

一、基本思想

$$X(k)=\sum_{n=0}^{N-1}x(n)W_N^{nk} \Rightarrow W_N^{nk}=e^{-j\Omega_0 nk}=(e^{-j\frac{2\pi}{N}})^{nk}$$

$$W_N^0=e^0=1$$

$$W_N^N=(e^{-j\frac{2\pi}{N}})^N=e^{-j2\pi}=1$$

$$W_N^{N/2}=(e^{-j\frac{2\pi}{N}})^{N/2}=e^{-j\pi}=-1$$

$$W_N^{N/4}=(e^{-j\frac{2\pi}{N}})^{N/4}=e^{-j\pi/2}=-j$$

$$W_{2N}^k=(e^{-j\frac{2\pi}{2N}})^k=e^{-j\frac{2\pi}{N}\cdot k/2}=W_N^{k/2}$$

一、基本思想

$$X(k)=\sum_{n=0}^{N-1}x(n)W_N^{nk} \Rightarrow W_N^{nk}=e^{-j\Omega_0 nk}=(e^{-j\frac{2\pi}{N}})^{nk}$$

1. 周期性

$$W_N^{k}=W_N^{k+lN} \qquad W_N^{nk}=W_N^{(n+mN)(k+lN)}$$

例如：*N=4*， 有 $W_4^6=W_4^2, W_4^9=W_4^1$

2. 对称性

$$W_N^{nk+N/2}=-W_N^{nk}$$

例如：*N=4*， 有 $W_4^3=-W_4^1, W_4^2=-W_4^0$

3. 可约性

$$W_N^{nk}=W_{mN}^{mnk}=W_{N/m}^{nk/m}$$

第三节 快速傅里叶变换（FFT）

（Fast Fourier Transform）

一、基本思路

二、基2FFT算法

三、FFT的应用

二、基2FFT算法

基2FFT：N为2的整数次幂的FFT称为基2FFT。

要求序列点数$N=2^M$，若不满足这个条件，可以人为地加上若干个零值点。

基本思想：利用旋转因子的对称性和周期性，将一个长序列的DFT分解为一些逐次变小的DFT来计算。

◆ 按时间抽取的基2FFT算法（库利—图基（Cooley-Tukey）算法）：对时间进行奇偶分解；

◆ 按频率抽取的基2FFT算法（桑德—图基（Sande-Tukey）算法）：对频率进行前后分解。

二、基2FFT算法

原理: 设序列长度为$N=2^n$,将$x[n]$ 分成 偶和奇 两个部分:

$$0 \le k \le N-1$$

$$
\begin{aligned}
X[k] &= \sum_{n=0}^{N-1} x[n] W_N^{kn} \\
&= \sum_{l=0}^{N/2-1} x[2l] W_N^{k(2l)} + \sum_{l=0}^{N/2-1} x[2l+1] W_N^{k(2l+1)} \\
&= \sum_{l=0}^{N/2-1} x[2l] W_{N/2}^{kl} + W_N^k \sum_{l=0}^{N/2-1} x[2l+1] W_{N/2}^{kl} \\
&= G[k] + W_N^k H[k]
\end{aligned}
$$

$$0 \le k \le N-1$$

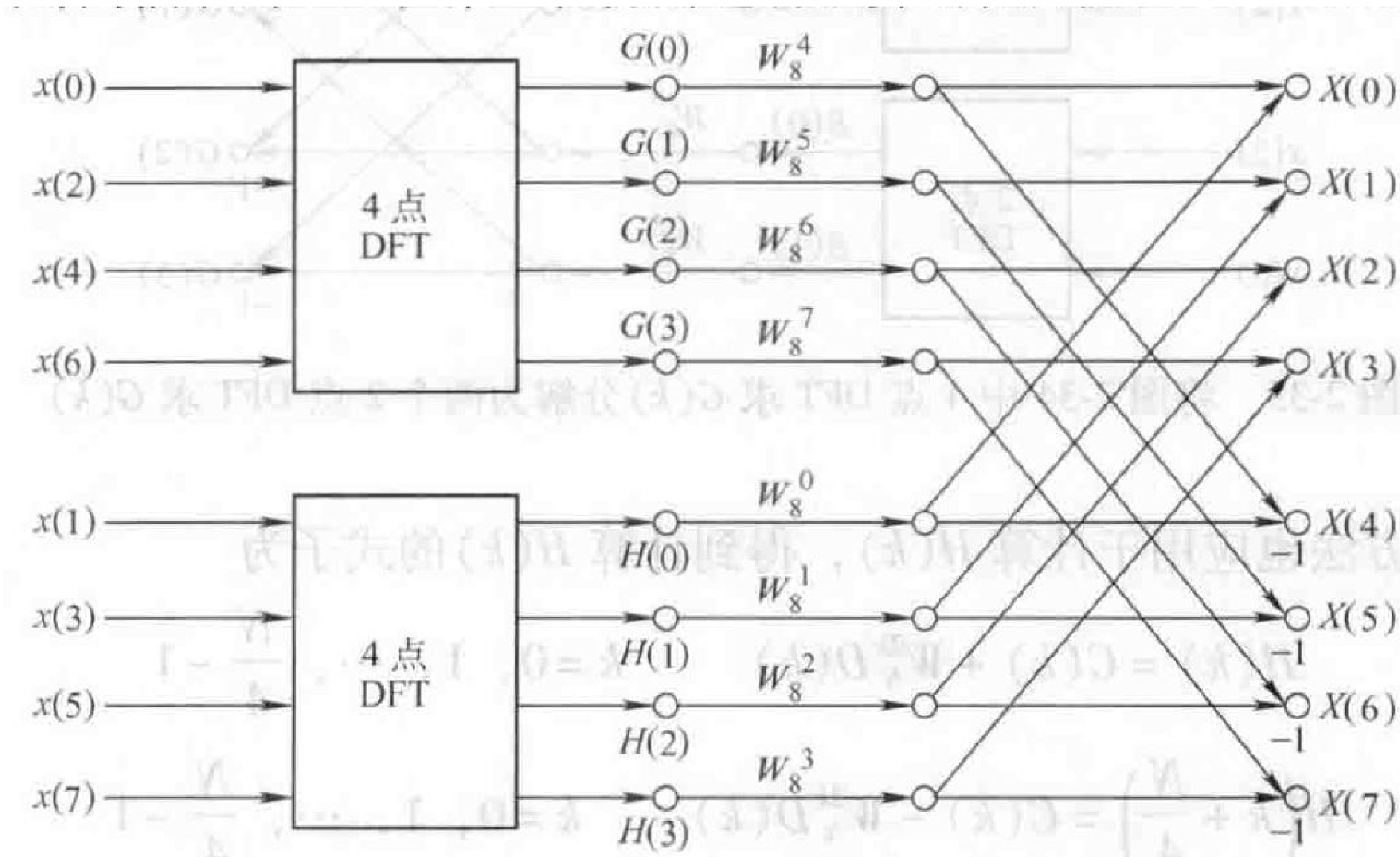
式中,$G(k)$ 和 $H(k)$ 是$N/2$ 点DFT,它们的周期为$N/2$,

但,因W_N^k的周期为N,故$X(k)$的周期为N.

$$\text{Q}\ X(k+N/2) = G(k+N/2) + W_N^{k+N/2}H(k+N/2)$$
$$= G(k) - W_N^k H(k)$$

$$\therefore \begin{cases} X(k) = G(k) + W_N^k H(k), \\ X(k+N/2) = G(k) - W_N^k H(k), \end{cases} \qquad 0 \le k \le N/2\text{-}1$$

例：N=8



二、基2FFT算法

继续将$g[n]$和$h[n]$分成偶与奇两部分，得：

$$G(k) = \sum_{l=0}^{N/2-1} x(2l) W_{N/2}^{kl}$$

$$= \sum_{l偶} x(2l) W_{N/2}^{kl} + \sum_{l奇} x(2l) W_{N/2}^{kl}$$

令$l = 2r$

$l = 2r+1$

$$= \sum_{r=0}^{N/4-1} x(4r) W_{N/2}^{2kr} + \sum_{r=0}^{N/4-1} x(4r+2) W_{N/2}^{k(2r+1)}$$

$$= \sum_{r=0}^{N/4-1} x(4r) W_{N/4}^{kr} + W_{N/2}^{k} \sum_{r=0}^{N/4-1} x(4r+2) W_{N/4}^{kr}$$

$$= A(k) + W_{N/2}^{k} B(k)$$

$$= A(k) + W_{N}^{2k} B(k) \qquad K = 0,1,\ldots,\frac{N}{4}-1$$

$$G(k) = A(k) + W_{N}^{2k} B(k)$$

$$G(k+\frac{N}{4}) = A(k) - W_{N}^{2k} B(k) \qquad K = 0,1,\ldots,\frac{N}{4}-1$$

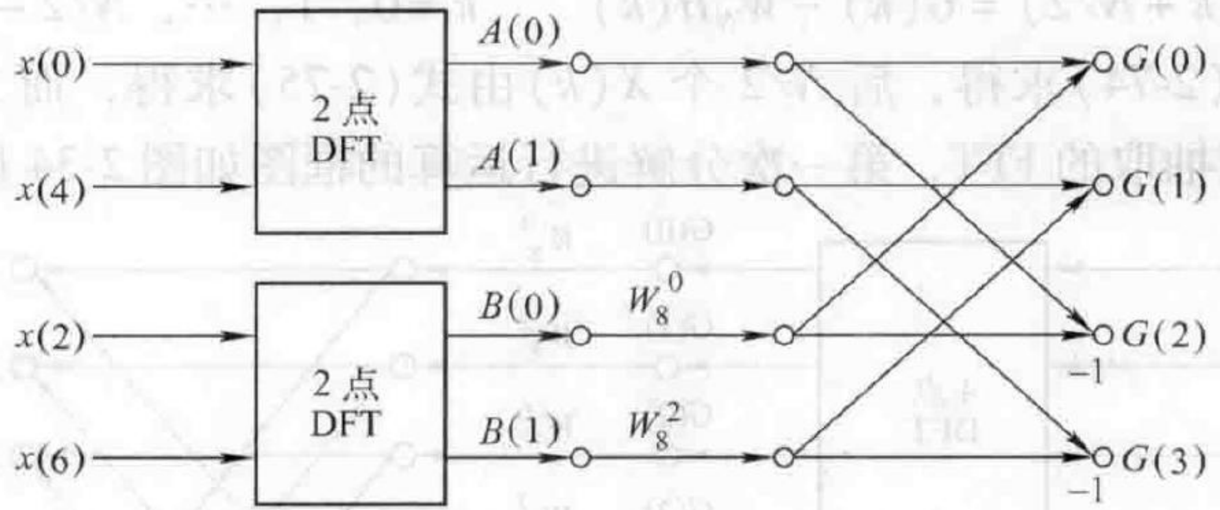


图 2-35　将图 2-34 中 4 点 DFT 求 $G(k)$ 分解为两个 2 点 DFT 求 $G(k)$

$$A(k) = \sum_{r=0}^{N/4-1} x(4r) W_{N/4}^{kr}$$

$$B(k) = \sum_{r=0}^{N/4-1} x(4r+2) W_{N/4}^{kr} \qquad K = 0,1,\ldots,\frac{N}{4}-1$$

$$G(k) = A(k) + W_N^{2k} B(k)$$

$$G(k+\frac{N}{4}) = A(k) - W_N^{2k} B(k) \qquad K = 0,1,\ldots,\frac{N}{4}-1$$

$$C(k) = \sum_{r=0}^{N/4-1} x(4r+1) W_{N/4}^{kr}$$

$$D(k) = \sum_{r=0}^{N/4-1} x(4r+3) W_{N/4}^{kr} \qquad K = 0,1,\ldots,\frac{N}{4}-1$$

$$H(k) = C(k) + W_N^{2k} D(k)$$

$$H(k+\frac{N}{4}) = C(k) - W_N^{2k} D(k) \qquad K = 0,1,\ldots,\frac{N}{4}-1$$

二、基2FFT算法

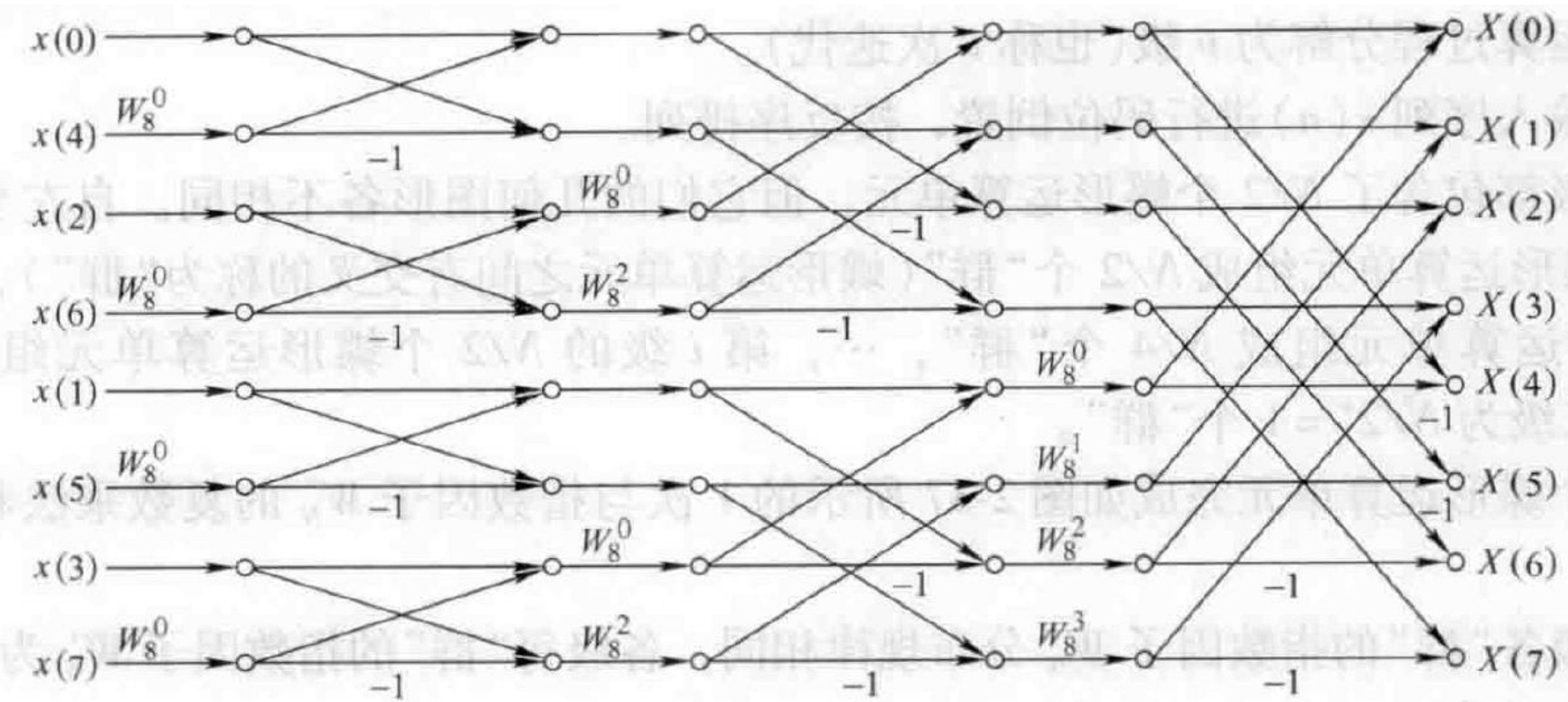
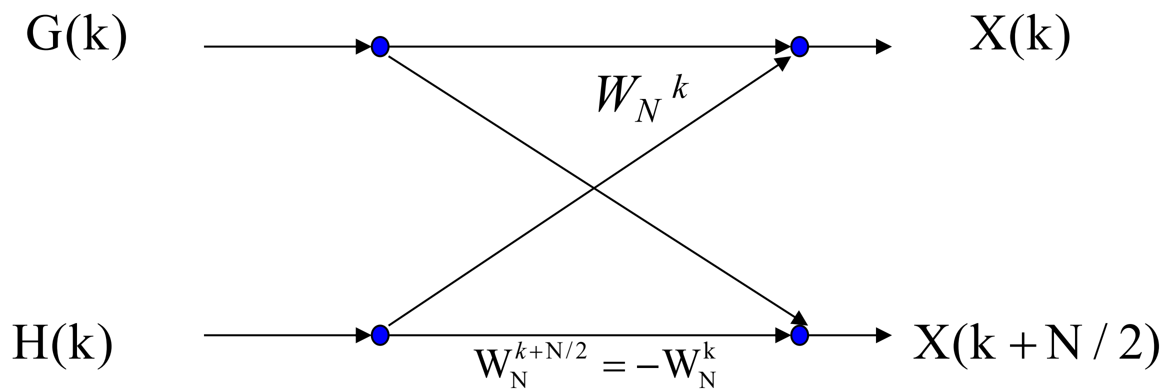


图 2-36　一个完整的8点基2按时间抽取FFT算法流程

二、基2FFT算法

蝶形运算

$$\begin{cases} X(k) = G(k) + W_N^k H(k), \\ X(k+N/2) = G(k) - W_N^k H(k), \end{cases} \quad 0 \le k \le N/2-1$$



1）长度为N=2^{v}的序列x(n)，通过基2按时间分解为$\log_2 N = v$级运算；

2）每级运算由N/2个蝶形运算完成：每个蝶形运算单元由2个复数输入数据和2个复数输出数据，包含了一次复数乘法和二次复数加法；

3）每级：N 个复加 和 N/2 个复乘；整个 DFT共有：$N/2\log_2 N$ 次复乘计算。

二、基2FFT算法

表 4-1　FFT 算法与直接 DFT 算法的比较

N	N^2	$\frac{N}{2}\log_2 N$	$N^2 \Big/ \left(\frac{N}{2}\log_2 N\right)$
2	4	1	4.0
4	16	4	4.0
8	64	12	5.4
16	256	32	8.0
32	1024	80	12.8
64	4096	192	21.4
128	16384	448	36.6
256	65536	1024	64.0
512	262144	2304	113.8
1024	1048576	5120	204.8
2048	4194304	11264	372.4

假如处理一幅N*N（N=1024）点的二维图像，如用每秒可做10万次复数乘法的计算机（不考虑加法运算时间），直接计算DFT所需时间为3000小时，用FFT算法只需要2分钟。

二、基2FFT算法

◆ 同址（原位）运算

蝶形的两个输出值扔放回蝶形的两个输入所在的存储器中。

某一列的**N**个数据送到存储器后，经蝶形运算，其结果为下一列数据，它们以蝶形为单位仍存储在这同一组存储器中，直到最后输出，中间无需其他存储器。即，蝶形的两个输出值仍放回蝶形的两个输入所在的存储器中。

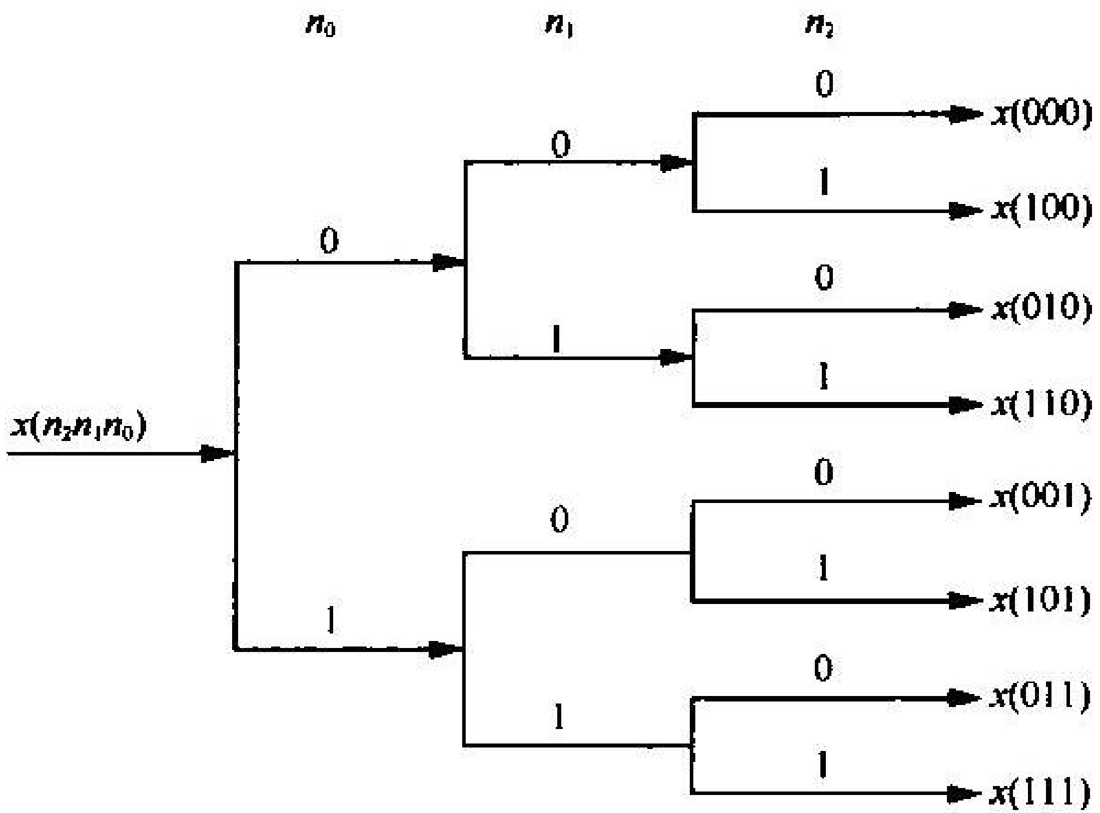
每列的**N/2**个蝶形运算全部完成后，再开始下一列的蝶形运算，这样存储器数据只需**N**个存储单元。下一级的运算仍采用这种原位方式，只不过进入蝶形结的组合关系有所不同

这种原位结构可以节省存储单元，降低设备成本。

二、基2FFT算法

◆ 变址（倒位序）运算

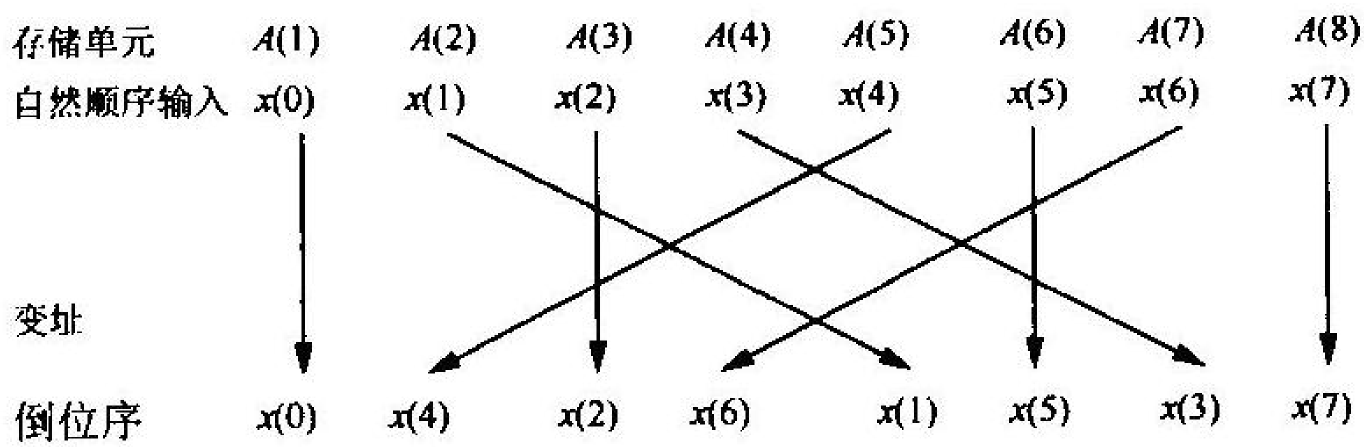
造成倒位序的原因是输入x(n)按标号n的偶奇不断分组。



二、基2FFT算法

自然顺序和码位倒置

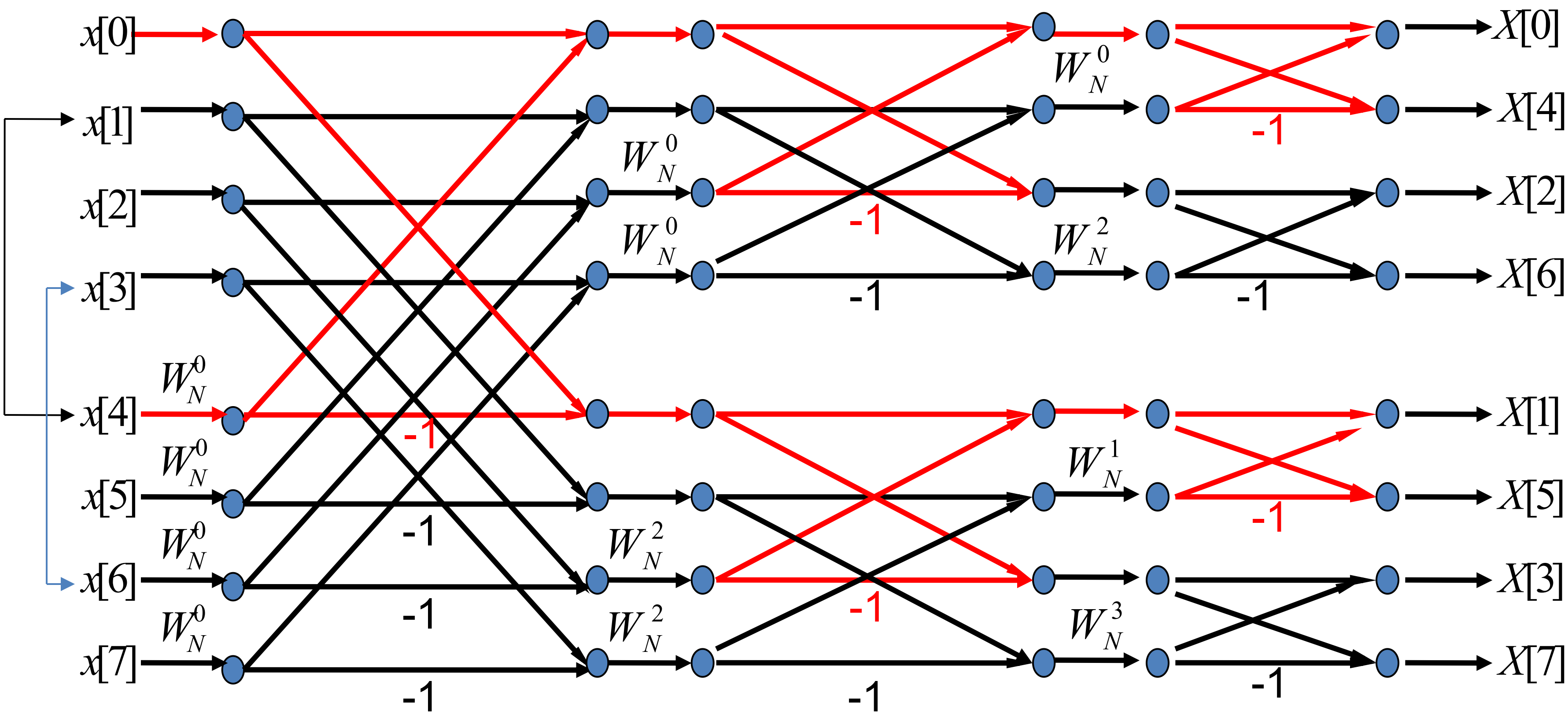
n	二进制	码位倒置二进制	n'
0	000	000	0
1	001	100	4
2	010	010	2
3	011	110	6
4	100	001	1
5	101	101	5
6	110	011	3
7	111	111	7



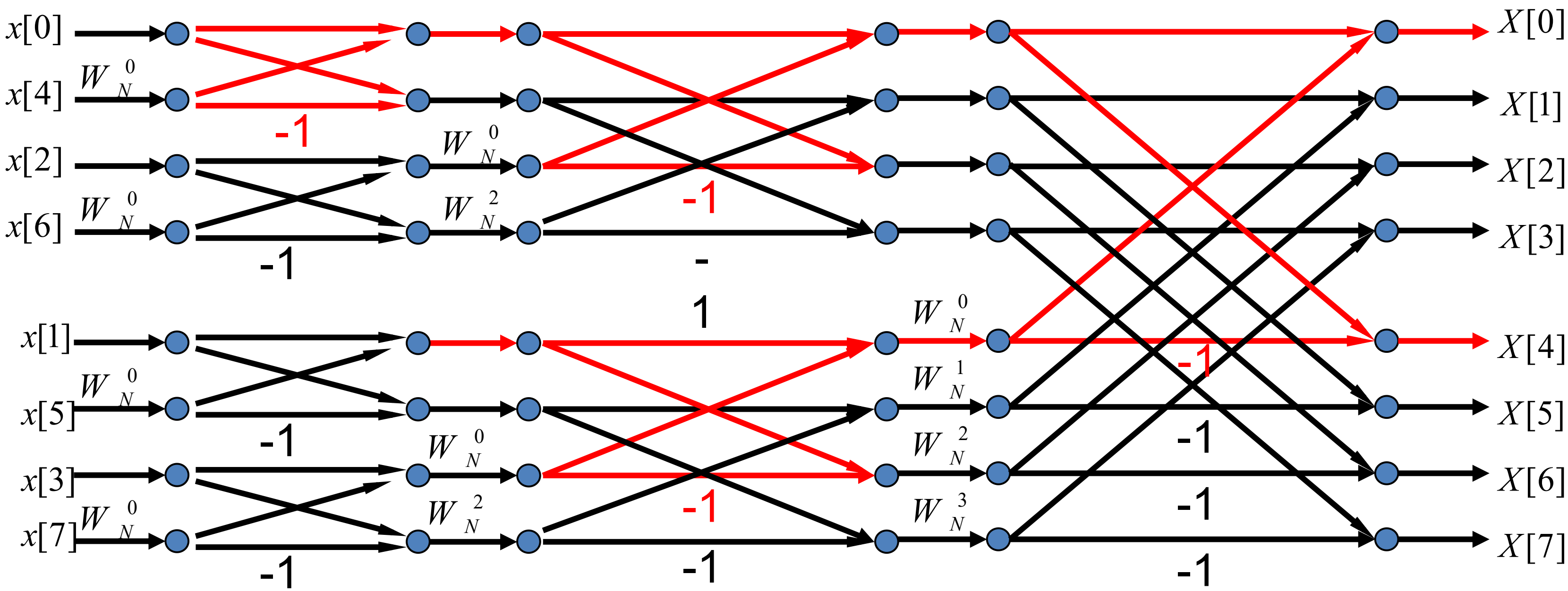
二、基2FFT算法

替代形式: 不管流图如何重新安排，只要各个支路的连接关系不变，

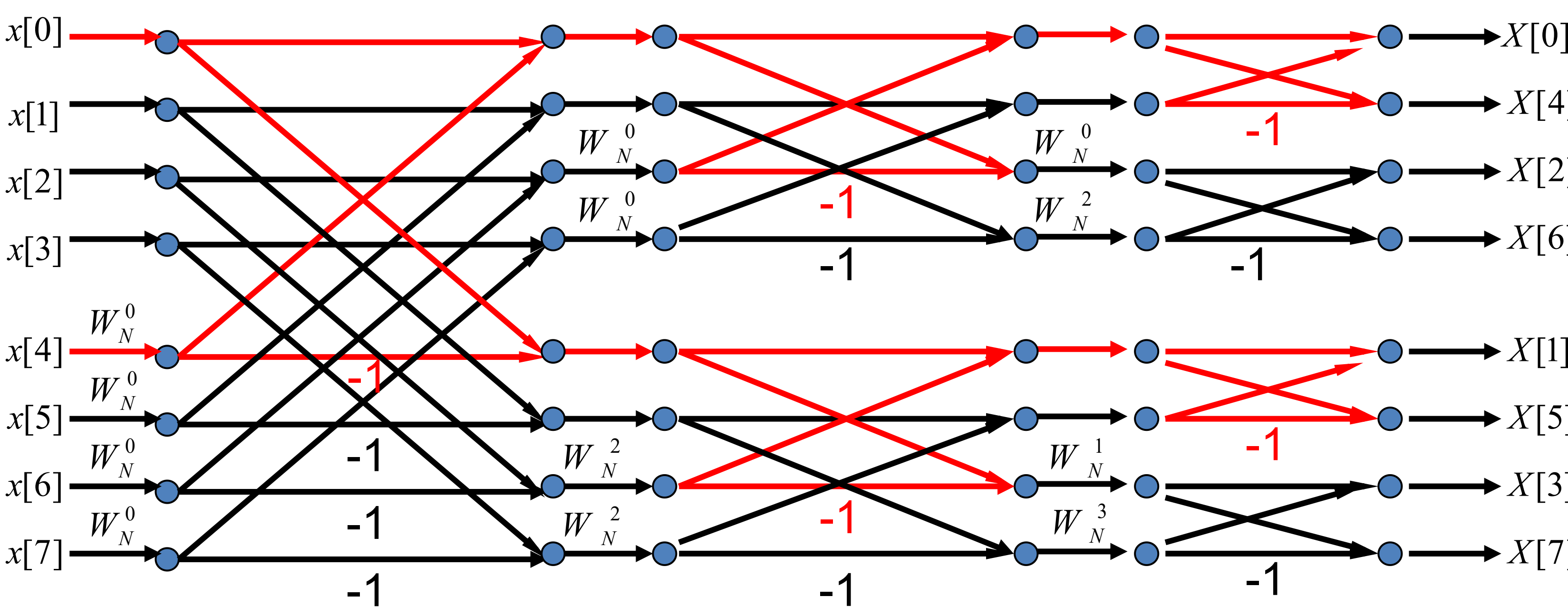
则其结果总是 $x[n]$的DFT有效算法，仅数据存取将不变。



二、基2FFT算法



x[0]
x[4]
x[2]
x[6]
x[1]
x[5]
x[3]
x[7]
W_N^0
W_N^2
W_N^1
W_N^3
-1
X[0]
X[1]
X[2]
X[3]
X[4]
X[5]
X[6]
X[7]



x[0]
x[1]
x[2]
x[3]
x[4]
x[5]
x[6]
x[7]
W_N^0
W_N^2
W_N^1
W_N^3
-1
X[0]
X[4]
X[2]
X[6]
X[1]
X[5]
X[3]
X[7]

二、基2FFT算法

归纳上面的推导过程，对于 $N=2^{\nu}$（ν 为整数），输入反序、输出正序的 FFT 的运算流程可表示如下：

1）全部计算分解为 ν 级（也称 ν 次迭代）。

2）把输入序列 $x(n)$进行码位倒置，按反序排列。

3）每级都包含了 $N/2$ 个碟形运算单元，但它们的几何图形各不相同。自左至右第 1 级的 $N/2$ 个碟形运算单元组成 $N/2$ 个“群”（碟形运算单元之间有交叉的称为“群”），第 2 级的 $N/2$ 个碟形运算单元组成 $N/4$ 个“群”，…，第 i 级的 $N/2$ 个碟形运算单元组成 $N/2^i$ 个“群”，最末级为 $N/2^{\nu}=1$ 个“群”。

4）每个碟形运算单元完成如图 3-36 所示的 1 次与指数因子 W_N^r 的复数乘法和 2 次复数加（减）法。

5）同级各“群”的指数因子 W_N^r 分布规律相同；各级每“群”的指数因子 W_N^r 为：

第 1 级：W_N^0

第 2 级：W_N^0，$W_N^{N/4}$

…

第 i 级：W_N^0，$W_N^{N/2^i}$，$W_N^{2N/2^i}$，…，$W_N^{(2^{i-1}-1)N/2^i}$

…

第三节 快速傅里叶变换（FFT）

（Fast Fourier Transform）

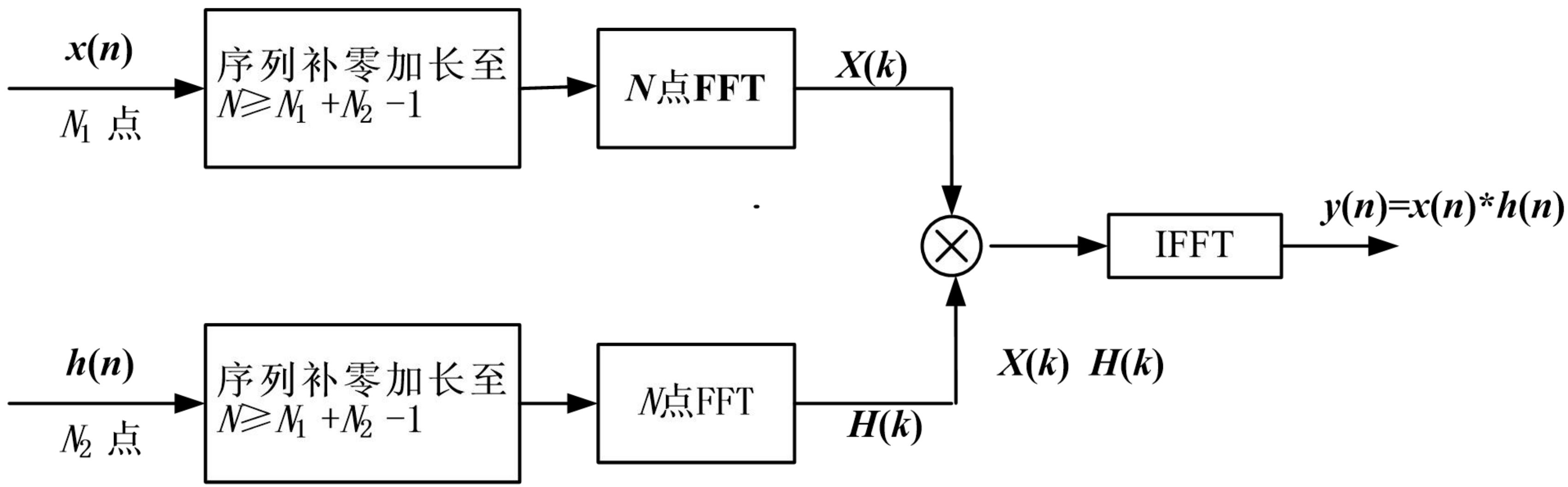
一、基本思路

二、基2FFT算法

三、FFT的应用

三、FFT应用

1. 利用FFT求线性卷积



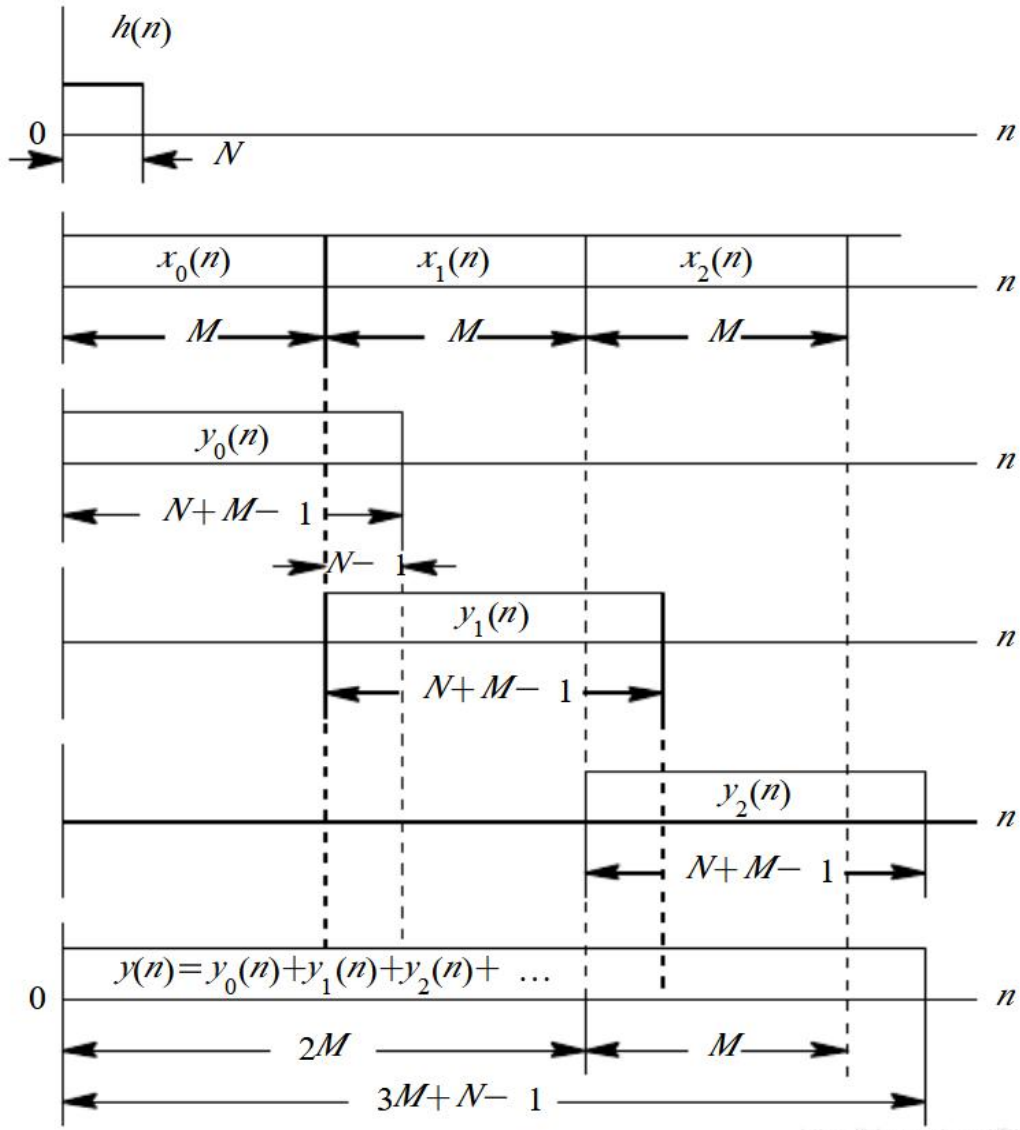
◆ 在MATLAB中直接实现线卷积计算的函数有 conv， conv2, convn。其中conv2和 convn分别用于2维、*n*维的卷积运算。conv则用于向量卷积与多项式乘的计算，调用的格式为 *c*＝conv（*a*，*b*）。式中，a、b表示两个序列，*c*＝*a***b*。在MATLAB中，序列可用向量来表示，若向量*a*的长度为*na*，向量*b*的长度为*nb*，则向量*c*的长度为*na*＋*nb*－1。

三、FFT应用

1. 利用FFT求线性卷积

- **分段快速卷积的方法（两序列长度相差太多）:**将长序列分成若干小段，每小段分别与短序列作卷积运算，然后将所有的分段卷积结果相叠加，就是线卷积的最后结果，这种方法又称为重叠相加法。

1. 利用FFT求线性卷积



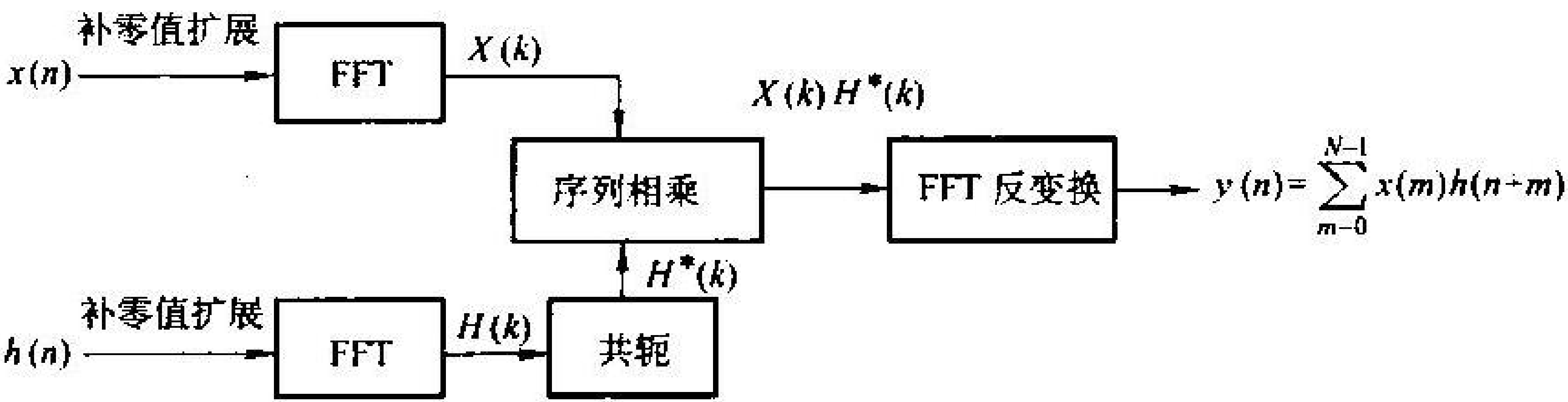
三、FFT应用

2. 利用FFT求线性相关

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

$$R_{xx}(m)=\sum_{n=-\infty}^{\infty}x(n)x(n+m)=x(m)*x(-m)$$

若 $y(t)$ 是实函数，有 $Y(-\omega)=Y^{*}(\omega) \Longrightarrow R_{yx}(\tau)\leftrightarrow X(\omega)Y^{*}(\omega)$



三、FFT应用

3. 利用FFT作连续时间信号的频谱分析

◆ 工程上所遇到的信号，包括传感器的输出信号，大多是连续非周期信号，这种信号无论在时域或频域都是连续的。

◆ 对连续非周期信号的数字谱分析实质上是用有限长抽样序列的DFT（离散谱）来近似无限长连续信号的频谱（连续谱），其结果必然会产生误差，主要的误差包括：混叠效应、频谱泄漏和栅栏效应三种。

3. 利用FFT作连续时间信号的频谱分析

（1）误差产生原因及解决办法

（一）混叠效应

时域信号的离散化是通过抽样实现的，当采样频率 $f_s = 1/T$ 不够高时，采样信号相对原信号就会产生频谱的混叠，引起频谱失真。频谱混叠效应是由于时域的离散化引起的，克服的办法是提高采样频率。

3. 利用FFT作连续时间信号的频谱分析

（1）误差产生原因及解决办法

（二）频谱泄漏

直流信号截断，频谱泄露

频谱泄漏又称截断误差，是由于对信号进行截断，把无限长的信号限定为有限长，即令有限区间外的函数值均为零值，相当于用一个矩形（窗）信号乘相应的信号。

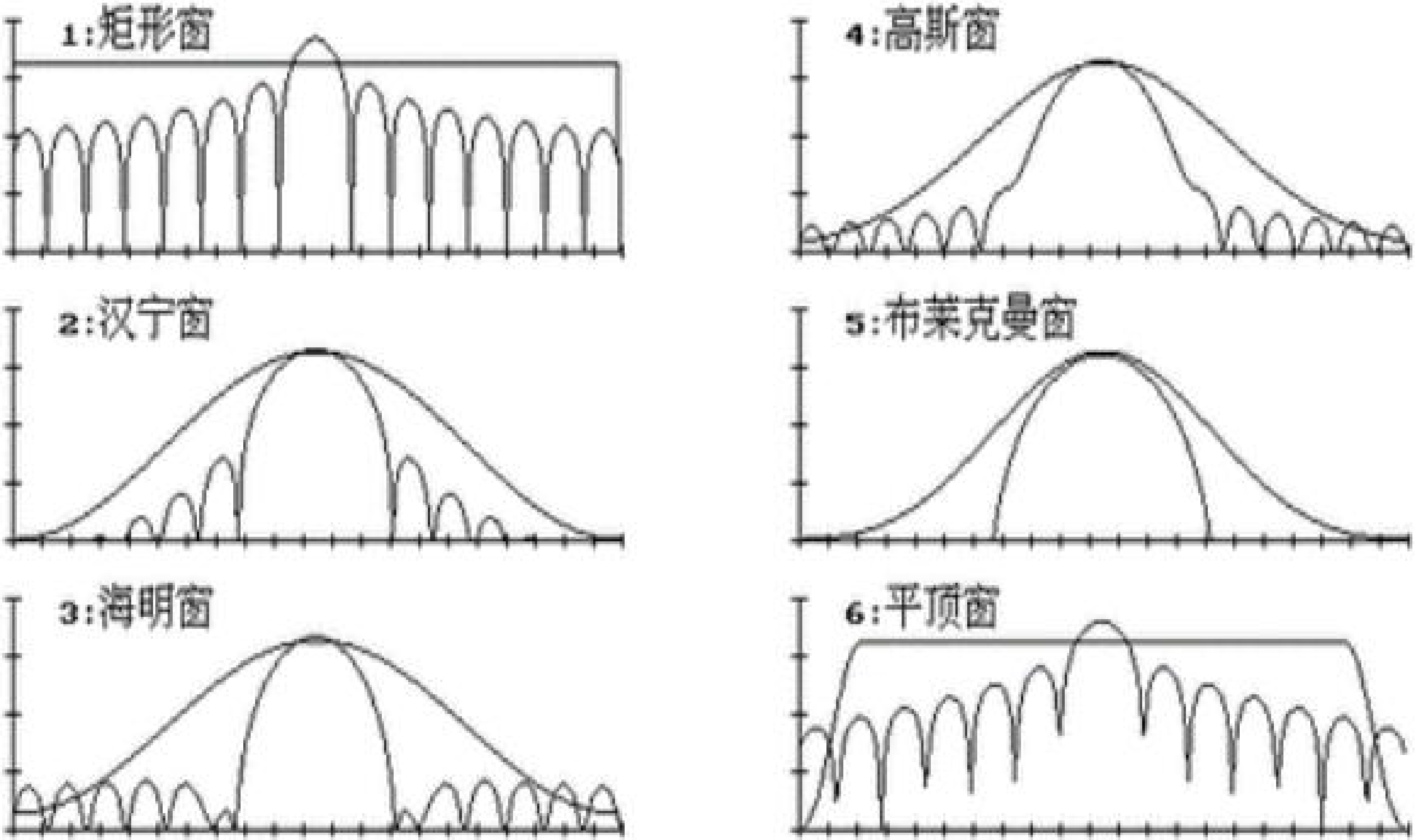
频谱泄漏是由时域信号的截断引起的，减小频谱泄漏的方法一般有两种：

1）增加截断长度；

2）改变窗口形状。

3. 利用FFT作连续时间信号的频谱分析

- 减少截断误差，应使主瓣宽度或者旁瓣幅值减少，使实际频谱接近原频谱。但根据能量守恒定理：旁瓣幅值减少，则主瓣宽度将加大；旁瓣幅值增大，主瓣宽度将缩小，显然，增加旁瓣的幅值，造成旁瓣、主瓣不清，引起两个主瓣的误解。
- 也就是宁可增加主瓣宽度，缩小旁瓣幅值，使能量集中于主瓣。
- 实质：旁瓣是高频分量，缩小旁瓣，就是减少高频分量，适当增加低频分量。
- 什么样的截断窗口？ 幂窗、三角函数窗、指数窗等。他们的高频分量衰减增快，旁瓣明显受到抑制，相对减少了频谱泄露。
- 只能减弱、不能消除。


1:矩形窗
4:高斯窗
2:汉宁窗
5:布莱克曼窗
3:海明窗
6:平顶窗

3. 利用FFT作连续时间信号的频谱分析

（1）误差产生原因及解决办法

（三）栅栏效应

非周期信号具有连续谱，但用DFT来计算非周期信号的频谱时，只能观察到有限个（N个）离散频谱值，而频谱间隔中的值就观察不到了，比喻为通过栅栏观察景物，一部分景物被阻挡了，这种现象称为栅栏效应。将能够感受到的频谱最小间隔值称为频谱分辨率，一般用F表示。频谱分辨率反映了谱分析算法能将信号中两个靠得很近的谱保持分开的能力。若时域抽样周期为Ts，抽样点数为N，则有

3. 利用FFT作连续时间信号的频谱分析

$$\Omega_0 = \frac{2\pi}{\mathrm{N}} = \omega_0 T_s = 2\pi f_0 \frac{1}{f_s}$$

$$f_0 = \frac{f_s}{\mathrm{N}} = \frac{1}{T_0}$$

f_0:频率分辨率

- 栅栏效应是由于频域的离散化引起的，使得在频谱抽样间隔之间的频谱无法反映出来，因此是不可避免的。为了改善栅栏效应，提高频率分辨率，应当增加信号的有效数据长度N，也可以采用频谱细化技术，使谱线变密，从而看到原来看不到的“频谱景象”。

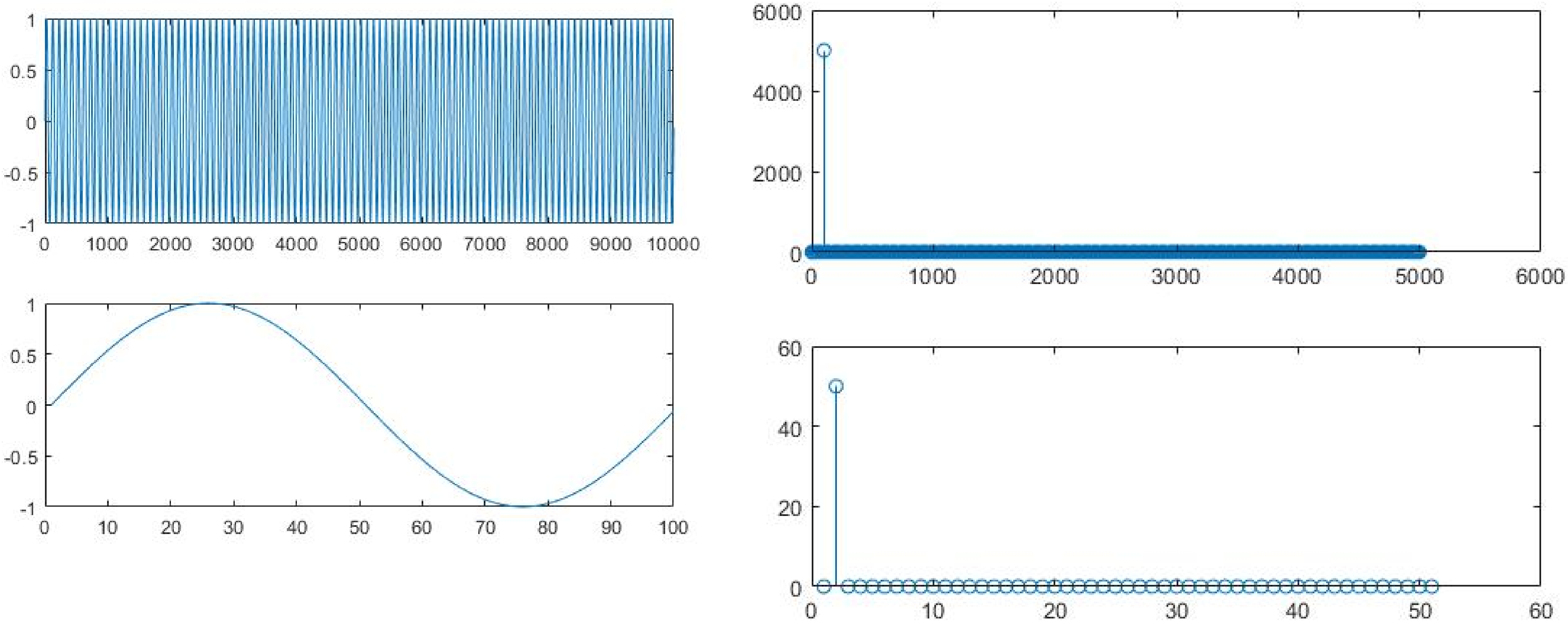
3. 利用FFT作连续时间信号的频谱分析

（2）连续周期信号的频谱分析

- 连续周期信号是非时限信号，若要用FFT作数字谱分析，必须在时域进行有限化（截断）和离散化（抽样）处理。对于一个带限（频谱为有限区间）的周期信号，若抽样频率满足抽样条件，并且作整周期截断，不会产生频谱的混叠。
- 整个周期截断是困难的，则会产生频谱的泄露误差，通过加窗的方法来减少泄露。

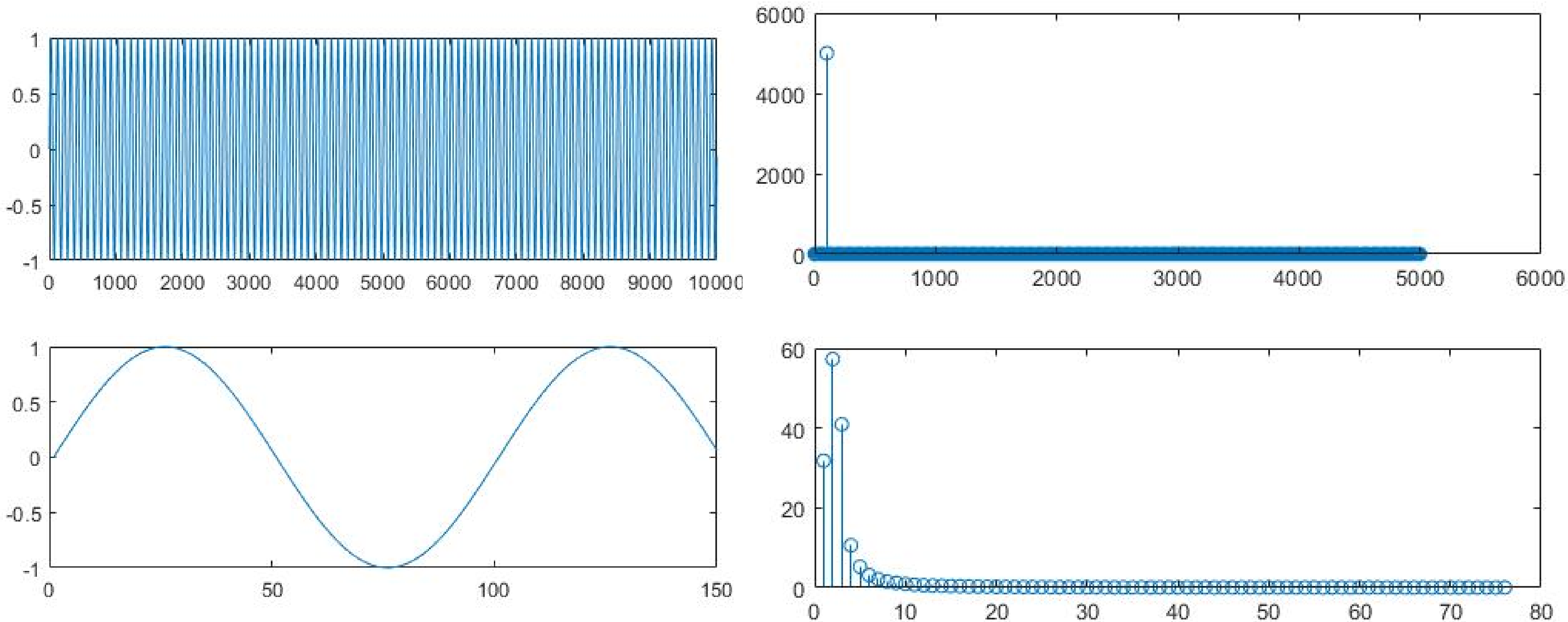
3. 利用FFT作连续时间信号的频谱分析

周期截断



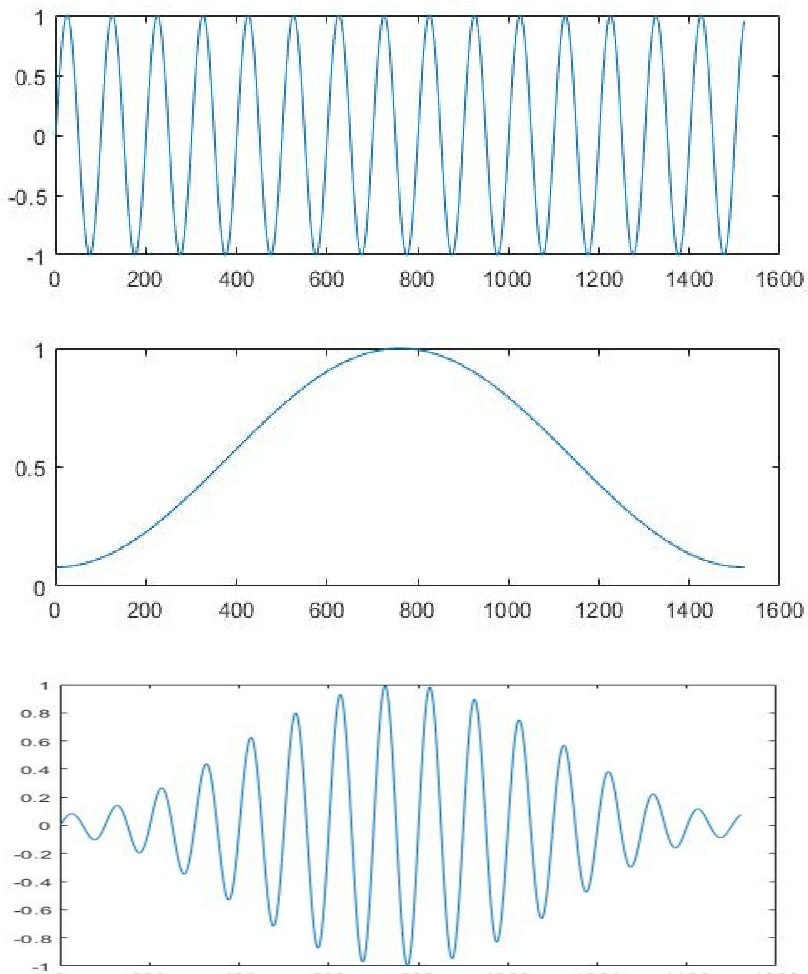
3. 利用FFT作连续时间信号的频谱分析

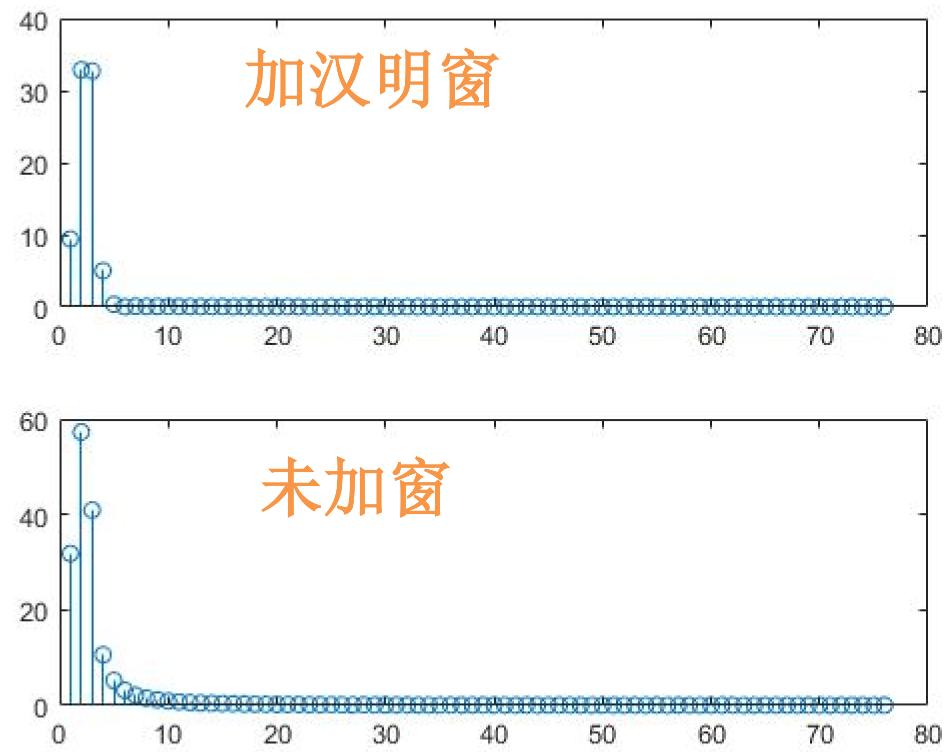
非周期截断



3. 利用FFT作连续时间信号的频谱分析

加汉明窗





3. 利用FFT作连续时间信号的频谱分析

- 应用DFT(或FFT)进行信号的频谱分析时，要根据给定的要求，确定DFT的参数。一般情况下，已知（或先估计）：信号的最高频率 F 、频谱分辨率 f_m、抽样时能够达到的最高抽样频率 $\omega_{sm}(f_{sm})$。需要确定的参数通常包括：截取的信号长度（数据长度）T_1 、抽样频率 f_s （或采样间隔 T ）、点数N及选择什么样的窗口函数等。

- 选择参数的总原则：**尽可能减少混叠、频谱泄漏和栅栏效应**等项误差，保证信号处理的精度和可靠性。

3. 利用FFT作连续时间信号的频谱分析

基本步骤：

1）估计待分析信号中频率范围和频率上限 f_m。

2）选定抽样频率 f_s。

3）根据分析精度，确定数据有效长度 T_1。

4）确定点数N。

5）选窗口。

3. 利用FFT作连续时间信号的频谱分析

例：有一频谱分析用的FFT处理器，其抽样点数必须是2的整数幂，假设没有采用任何的数据处理措施，已给条件为：

1）频率分辨率 $\le 10\,Hz$

2）信号最高频率 $\le 4\,kHz$

试确定以下参量：

1）最小记录长度T_0(频率分辨率倒数)

2）抽样点间的最大时间间隔T（即最小抽样频率）

3）在一个记录中最少点数N

$$\Omega_0 = \frac{2\pi}{\mathrm{N}} = \omega_0 T_s = 2\pi f_0 \frac{1}{f_s}$$

解： 1）最小记录长度：

$$f_0 = \frac{f_s}{\mathrm{N}} = \frac{1}{T_0}$$

$$T_0 \geq \frac{1}{F_0} = \frac{1}{10} = 0.1s$$

2）最大抽样间隔 （$f_s > 2f_h \quad f_s = 1/T$）

$$T < \frac{1}{2f_h} = \frac{1}{2 \times 4 \times 10^3} = 0.125ms$$

3）最小记录点数

$$N > \frac{2f_h}{F_0} = \frac{2 \times 4 \times 10^3}{10} = 800$$

$$取 N = 2^m = 2^{10} = 1024 > 800$$

3. 利用FFT作连续时间信号的频谱分析

例：有一调幅信号

$$x_a(t) = [1 + \cos(2\pi * 100t)]\cos(2\pi * 600t)$$

用DFT做频谱分析，要求能分辨$x_a(t)$的所有频率分量，问

(1)抽样频率应为多少赫兹?

(2)抽样时间间隔应为多少秒?

(3)抽样点数应为多少点？

3. 利用FFT作连续时间信号的频谱分析

解：

$$x_a(t)=\left[1+\cos(2\pi\times100t)\right]\cos(2\pi\times600t)$$

$$=\cos(2\pi\times600t)+\frac{1}{2}\cos(2\pi\times700t)+\frac{1}{2}\cos(2\pi\times500t)$$

（1）抽样频率应为 $f_s\geq2\times700=1400Hz$

（2）抽样时间间隔应为

$$T\leq\frac{1}{f_s}=\frac{1}{1400}=0.00072Sec=0.72ms$$

3. 利用FFT作连续时间信号的频谱分析

（3）$x(n)=x_a(t)\big|_{t=nT}$

$$=\cos\left(2\pi\times\frac{6}{14}n\right)+\frac{1}{2}\cos\left(2\pi\times\frac{7}{14}n\right)+\frac{1}{2}\cos\left(2\pi\times\frac{5}{14}n\right)$$

$x(n)$为周期序列，周期$N=14(N=\dfrac{2\pi}{\Omega})$

$\therefore$抽样点数至少为14点

或者

Q 频率分量分别为500、600、700Hz

$\therefore \mathrm{f}_0=100\mathrm{Hz}$(基础频率即频率分辨率)

$\therefore N=f_s/f_0=1400/100=14$

$\therefore$最小记录点数$N=14$