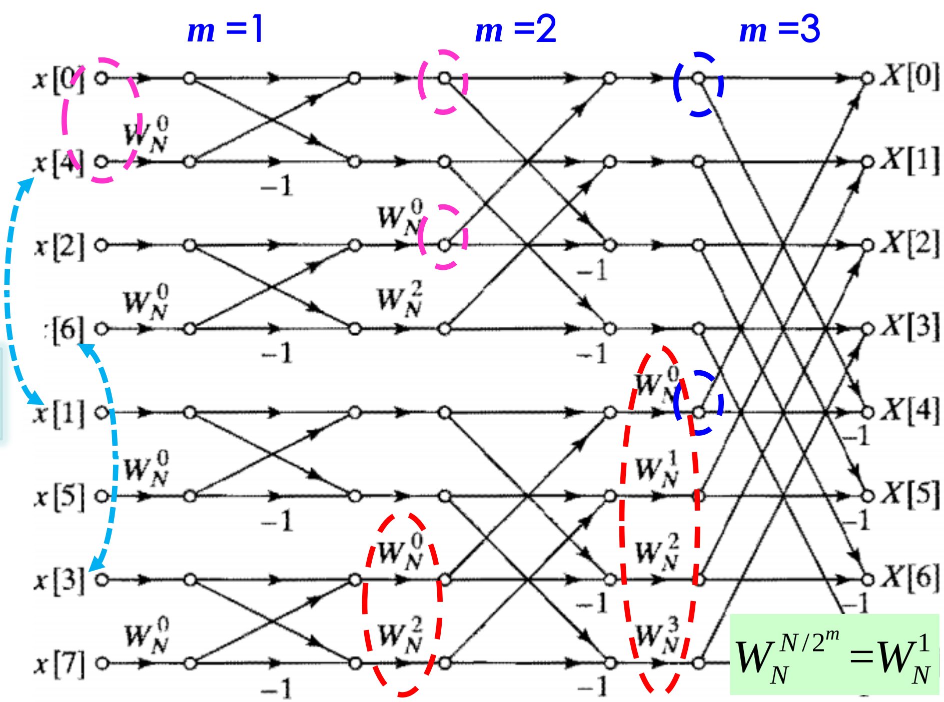
第九章：离散傅里叶变换的计算

- 9.1 离散傅里叶变换的高效计算
- 9.2 Goertzel算法
- 9.3 按时间抽取的FFT算法
- 9.4 按频率抽取的FFT算法
- 9.5 实现问题考虑
- 9.6 用卷积实现DFT
- 9.7 有限寄存器长度的影响

9.5.1 标号排序

某些DFT算法为实现同址运算，输入必须是倒位序，而输入序列通常不是倒位序，因此必须**对序列重排序**



每种算法都有其特有的排序问题

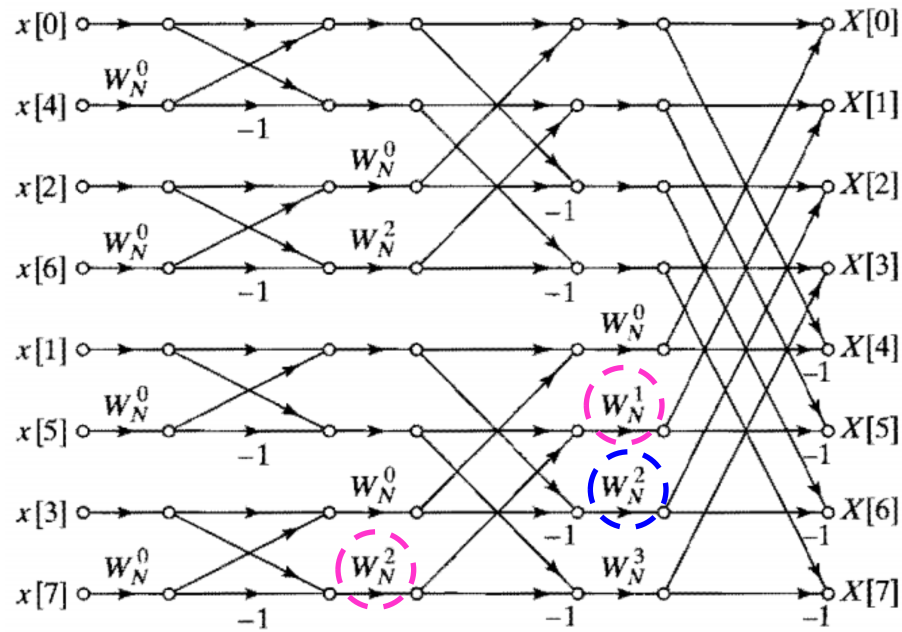
- 输入序列排序
- 中间值排序
- 系数排序
- 输出序列排序

图中输入序列的**顺位序**需重排为**倒位序**——可用两个计数器采用**同址方式**在**成对序号**间进行，即将**顺位序**计数器和**倒位序**计数器所指定的两个位置上的**数据互换**。

每级中的蝶形输入**间隔**不同，第m级的间隔为2^{m-1}，（$m=1,\cdots,v$）。

第m级的系数为 $W_N^{N/2^m}$ 的幂次方，从顶部开始为顺位序增加幂次

9.5.2 系数



- **系数的位序：**

系数 W_N^r 可以要求是倒位序或顺位序

- **系数的生成：**

➢ 查表法

需要存储 $r=0,1,\cdots,N/2-1$ 时的 W_N^r

完整表由 $N/2$ 个复寄存器构成

➢ 实时计算法

每级系数为 W_N^q 的幂次，其中 q 与算法与级数有关

第m级(m=1,2,3)，$q=N/2^m$

对于某级的顺位序系数，第 l 个系数可由第 $(l-1)$ 个系数递推获得

$$\left(W_N^q\right)^l=W_N^{ql}=W_N^q\cdot W_N^{q(l-1)}$$

上图中，第3级（$q=1$）的第 l 个系数为 $W_N^l=W_N^1\cdot W_N^{(l-1)}$

✓ 该式不适用于倒位序系数的算法

✓ 有限精度计算时为避免迭代生成误差，需重置预定点上的值

如 $W_N^{(N/4)}=-j$

9.5.3 更一般N值时的算法

◆ 更一般的复合数N的DFT算法

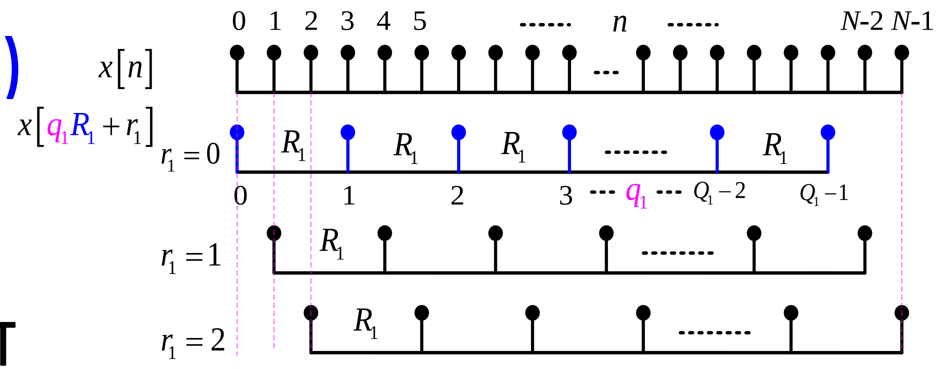
虽然N为2的整数幂可得出高效的DFT算法，但不是减少DFT计算量的唯一N取值情况

当N为复合数（两个或多个整数乘积）时，可采用类似2的整数幂情况的按时间或频率抽取算法

- ◆ 若$N=RQ$，则可将N点DFT表示为R个Q点DFT之和或Q个R点DFT之和
- ◆ 若N为多个因子的积，则对每个因子均可重复上述过程

更一般的复合数N点DFT的快速实现算法需要比2的整数幂情况更复杂的序号排列

9.5.3 更一般N值时的算法



◆ 复合数点DFT的分解(时间抽取)

设 $N=R_1R_2\cdots R_M$，令 $Q_1=R_2\cdots R_M$

则N点DFT第1次分解为 $N=R_1Q_1$

即将N点DFT表示为 R_1 个 Q_1 点DFT之和，需将输入的N点序列分解为 R_1 个长度为 Q_1 的子序列的组合，即输入序列可表示为

$$x[n]=x[q_1R_1+r_1],\quad q_1=0,\cdots,Q_1-1;\quad r_1=0,\cdots,R_1-1$$

第r_1个 Q_1 点子序列由N点序列 $x[n]$ 偏移 r_1 点后间隔 R_1 点抽取而得。

因此 $x[n]$的N点DFT可表示为

$$X[k]=\sum_{n=0}^{N-1}x[n]W_N^{kn}=\sum_{r_1=0}^{R_1-1}\sum_{q_1=0}^{Q_1-1}x[q_1R_1+r_1]W_N^{k(q_1R_1+r_1)}$$

$$=\sum_{r_1=0}^{R_1-1}W_N^{kr_1}\sum_{q_1=0}^{Q_1-1}x[q_1R_1+r_1]W_N^{kq_1R_1}$$

9.5.3 更一般N值时的算法

$$X[k]=\sum_{r_1=0}^{R_1-1}W_N^{kr_1}\sum_{q_1=0}^{Q_1-1}x[q_1R_1+r_1]W_N^{kq_1R_1}$$

◆ 复合数点DFT的分解（续）

由于 $W_N^{kq_1R_1}=W_{N/R_1}^{kq_1}=W_{Q_1}^{kq_1}$

因此 $X[k]=\sum_{r_1=0}^{R_1-1}W_N^{kr_1}\sum_{q_1=0}^{Q_1-1}x[q_1R_1+r_1]W_{Q_1}^{kq_1}=\sum_{r_1=0}^{R_1-1}W_N^{kr_1}X_{Q_1}^{r_1}[k],\quad k=0,\cdots,N-1$

其中 $X_{Q_1}^{r_1}[k]=\sum_{q_1=0}^{Q_1-1}x[q_1R_1+r_1]W_{Q_1}^{kq_1},\quad k=0,\cdots,Q_1-1;\quad r_1=0,\cdots,R_1-1$

为抽取的第 r_1 个子序列 $x[q_1R_1+r_1]$ 的 Q_1 点DFT

这样，N点DFT分解为R_1个Q_1点DFT，分别按周期R_1拓展后，再分别乘以系数 $W_N^{kr_1}$的加权和。

采用相同的方法，Q_1点（$Q_1=R_2Q_2$）DFT可进一步分解为R_2个Q_2点DFT，其中 $Q_2=R_3\cdots R_M$。以此类推，直至不可分解为止。

9.5.3 更一般N值时的算法

◆ 例：N=9点DFT的分解为3个3点DFT时间抽取FFT实现

（1）将输入的9点序列分解为 3个 （R_1=3 ）3点（Q_1=3 ）子序列的组合，即

$$x[n]=x[q_1R_1+r_1]=x[q_1 3+r_1],\quad q_1=0,\cdots,2;\quad r_1=0,\cdots,2$$

（2）9点DFT的分解表达式

$$X[k]=\sum_{r_1=0}^{R_1-1}W_N^{kr_1}\sum_{q_1=0}^{Q_1-1}x[q_1R_1+r_1]W_{Q_1}^{kq_1}$$

$$X[k]=\sum_{n=0}^{8}x[n]W_9^{kn}=\sum_{r_1=0}^{2}W_9^{kr_1}\sum_{q_1=0}^{2}x[3q_1+r_1]W_3^{kq_1}=\sum_{r_1=0}^{2}W_9^{kr_1}X_3^{r_1}[k],\quad k=0,\cdots,8$$

（3）3个3点DFT $X_3^{r_1}[k],\quad r_1=0,\cdots,2;\quad k=0,\cdots,2$ 的表达式

$r_1 \rightarrow$, $k \downarrow$

$r_1=0$

$$X_3^0[0]=x[0]+x[3]+x[6]$$
$$X_3^0[1]=x[0]+x[3]W_3^1+x[6]W_3^2$$
$$X_3^0[2]=x[0]+x[3]W_3^2+x[6]W_3^4$$

$r_1=1$

$$X_3^1[0]=x[1]+x[4]+x[7]$$
$$X_3^1[1]=x[1]+x[4]W_3^1+x[7]W_3^2$$
$$X_3^1[2]=x[1]+x[4]W_3^2+x[7]W_3^4$$

$r_1=2$

$$X_3^2[0]=x[2]+x[5]+x[8]$$
$$X_3^2[1]=x[2]+x[5]W_3^1+x[8]W_3^2$$
$$X_3^2[2]=x[2]+x[5]W_3^2+x[8]W_3^4$$

9.5.3 更一般N值时的算法

$$X_3^0[0]=x[0]+x[3]+x[6]$$
$$X_3^0[1]=x[0]+x[3]W_3^1+x[6]W_3^2$$
$$X_3^0[2]=x[0]+x[3]W_3^2+x[6]W_3^4$$

◆ 例：N=9点FFT实现（续）

（4）9点DFT的各值表达式

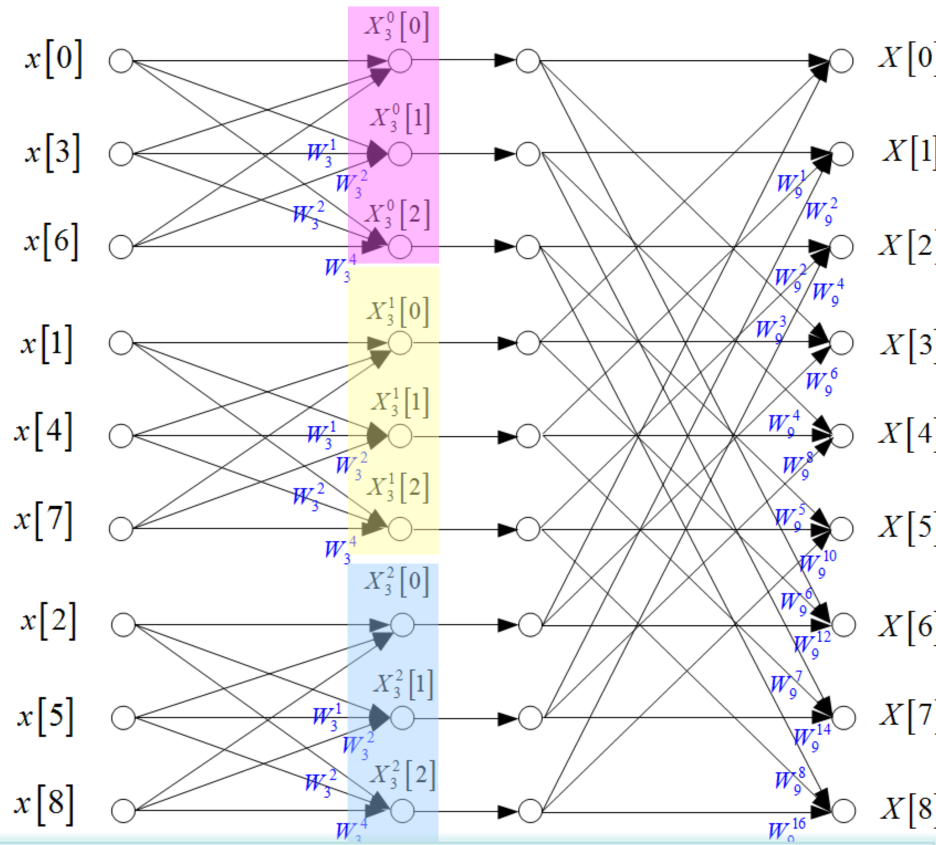
由 $X[k]=\sum_{r_1=0}^{2}W_9^{kr_1}X_3^{r_1}[k],\quad k=0,\cdots,8$

$r_1=0 \qquad =1 \qquad =2$

k

$$X[0]=X_3^0[0]+X_3^1[0]+X_3^2[0]$$
$$X[1]=X_3^0[1]+W_9^1X_3^1[1]+W_9^2X_3^2[1]$$
$$X[2]=X_3^0[2]+W_9^2X_3^1[2]+W_9^4X_3^2[2]$$
$$X[3]=X_3^0[0]+W_9^3X_3^1[0]+W_9^6X_3^2[0]$$
$$X[4]=X_3^0[1]+W_9^4X_3^1[1]+W_9^8X_3^2[1]$$
$$X[5]=X_3^0[2]+W_9^5X_3^1[2]+W_9^{10}X_3^2[2]$$
$$X[6]=X_3^0[0]+W_9^6X_3^1[0]+W_9^{12}X_3^2[0]$$
$$X[7]=X_3^0[1]+W_9^7X_3^1[1]+W_9^{14}X_3^2[1]$$
$$X[8]=X_3^0[2]+W_9^8X_3^1[2]+W_9^{16}X_3^2[2]$$

（5）9点DFT的FFT实现信流图



9点DFT分解后所需复乘数仅为9x2次，显著低于直接法的81次。

9.5.4 IDFT的快速算法（IFFT）

◆ 由FFT的结构通过改变系数获得IFFT

由IDFT和DFT定义

$$x[n] = IDFT\{X[k]\} = \frac{1}{N}\sum_{k=0}^{N-1} X[k] W_N^{-kn}$$

$$X[k] = DFT\{x[n]\} = \sum_{n=0}^{N-1} x[n] W_N^{kn}$$

IDFT和DFT的计算方法只相差指数 W_N^{-kn} 的幂次符号和系数 $1/N$ 。

将基序列取共轭 $W_N^{kn} \to W_N^{-kn}$ ，再乘以常数 $1/N$ ，即可由 $FFT \to IFFT$。

9.5.4 IDFT的快速算法（IFFT）

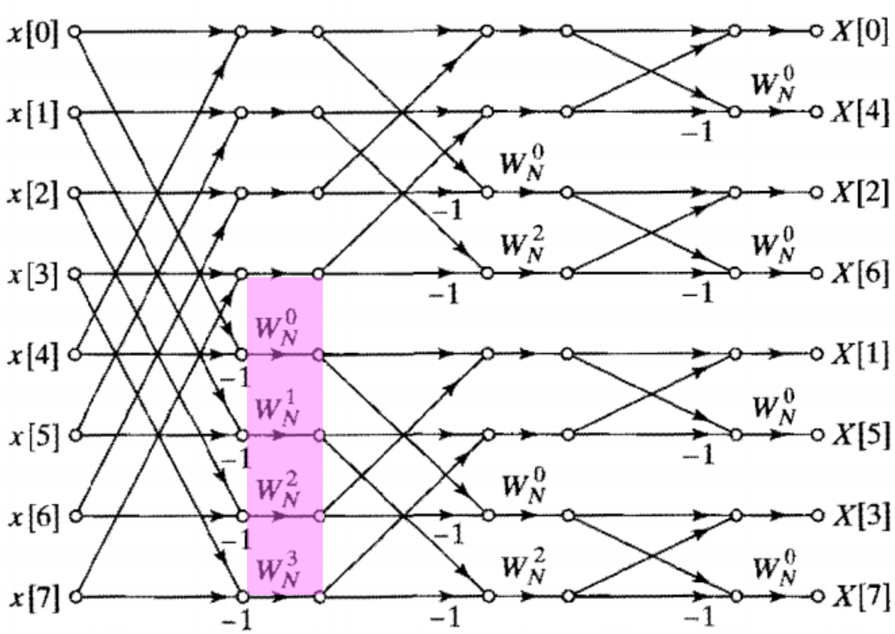
◆ 由FFT的实现流图直接获得IFFT的实现流图

由频率抽取的FFT流图，各蝶形计算变更如下

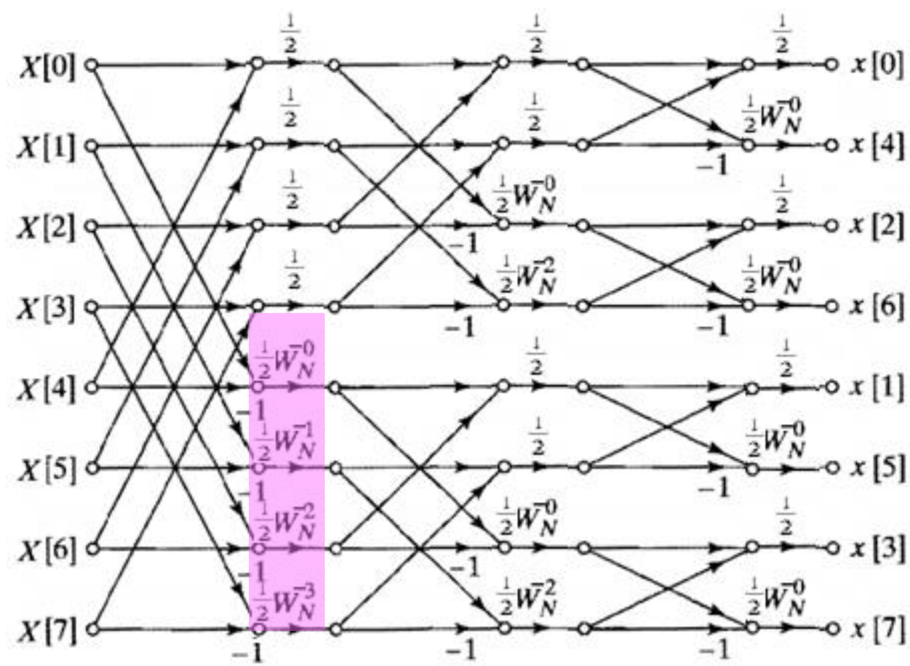
➢所有系数取共轭，$W_N^r \rightarrow W_N^{-r}$

➢各级节点输出再乘上系数1/2

➢输入$x[n]$与输出$X[k]$互换。



按频率抽取8点FFT完整流图



按时间抽取8点IFFT完整流图

IFFT输出时域序列$x[n]$是按奇偶分组计算，等效为时间抽取

上海科技大学
ShanghaiTech University

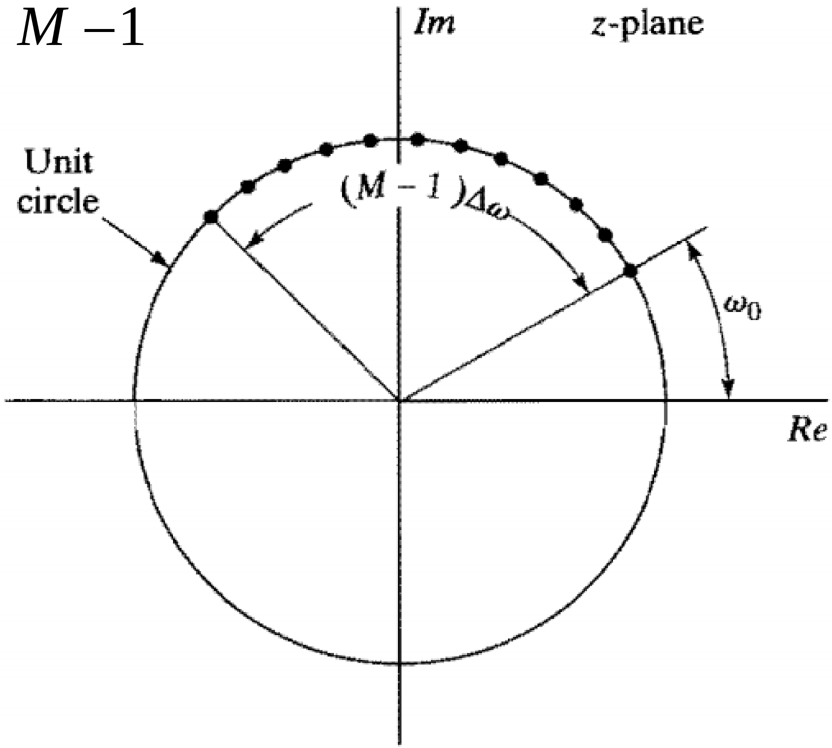
第九章：离散傅里叶变换的计算

- 9.1 离散傅里叶变换的高效计算
- 9.2 Goertzel算法
- 9.3 按时间抽取的FFT算法
- 9.4 按频率抽取的FFT算法
- 9.5 实现问题考虑
- 9.6 用卷积实现DFT
- 9.7 有限寄存器长度的影响

9.6.2 线性调频变换算法

令 $x[n]$ 表示一个N点序列，$X\left(e^{j\omega}\right)$表示其傅里叶变换。则$X\left(e^{j\omega}\right)$在单位圆上等角度间隔排列的M个样本可表示为

$$X\left(e^{j\omega_k}\right)=\sum_{n=0}^{N-1}x[n]e^{-j\omega_k n},\quad k=0,1,\cdots,\ M-1$$



其中

$$\omega_k=\omega_0+k\Delta\omega,\quad k=0,1,\cdots M-1$$

起始频率ω_0 与频率增量 $\Delta\omega$ 可任意选取。

对于傅里叶变换，$\omega_0=0$ ，$\Delta\omega=2\pi/N$ ，$M=N$ 。

若定义 $W=e^{-j\Delta\omega}$，则

M个样本可表示为 $X\left(e^{j\omega_k}\right)=\sum_{n=0}^{N-1}x[n]e^{-j\omega_0 n}W^{kn},\quad k=0,1,\cdots,M-1$

9.6.2 线性调频变换算法

$$X\left(e^{j\omega_k}\right)=\sum_{n=0}^{N-1}x[n]e^{-j\omega_0 n}W^{kn}$$

利用恒等式 $nk\equiv\frac{1}{2}\left[n^2+k^2-(k-n)^2\right]$

M个样本为 $X\left(e^{j\omega_k}\right)=\sum_{n=0}^{N-1}x[n]e^{-j\omega_0 n}W^{n^2/2}W^{k^2/2}W^{-(k-n)^2/2}$

若令 $g[n]=x[n]e^{-j\omega_0 n}W^{n^2/2}$ ，其中 $W^{n^2/2}$为**线性调频信号**（序列）

M个样本简化为

$$X\left(e^{j\omega_k}\right)=W^{k^2/2}\sum_{n=0}^{N-1}g[n]W^{-(k-n)^2/2},\quad k=0,1,\cdots M-1$$

或

$$X\left(e^{j\omega_n}\right)=W^{n^2/2}\sum_{k=0}^{N-1}g[k]W^{-(n-k)^2/2}$$

$$=W^{n^2/2}g[n]*W^{-n^2/2},\quad n=0,1,\cdots M-1$$

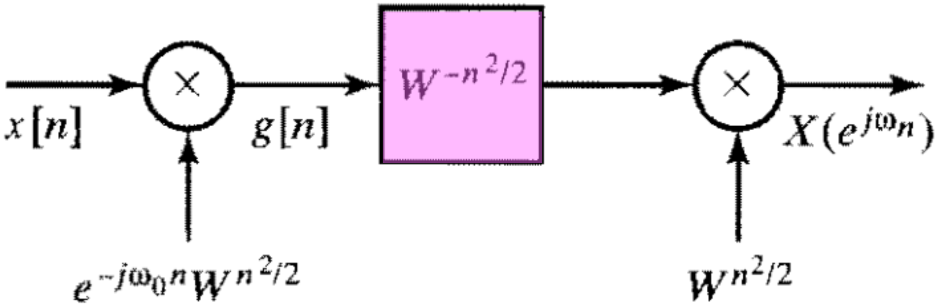
序列$x[n]$的**傅里叶变换的M个采样值** $X\left(e^{j\omega_k}\right)$等效为序列$x[n]$与线性调频序列$e^{-j\omega_0 n}W^{n^2/2}$相乘后，与序列 $W^{-n^2/2}$ **卷积输出的M个值**，再与序列 $W^{n^2/2}$ 矢量乘积。

9.6.2 线性调频变换算法

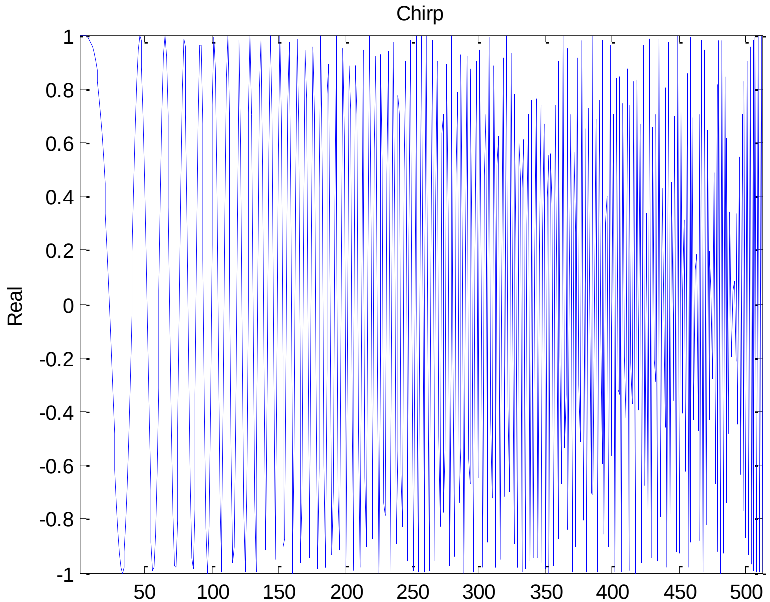
◆ 线性调频变换算法实现模型

由卷积过程 $X\left(e^{j\omega_n}\right)=W^{n^2/2}g[n]*W^{-n^2/2},\quad n=0,1,\cdots M-1$

可得线性调频变换算法实现框图如下

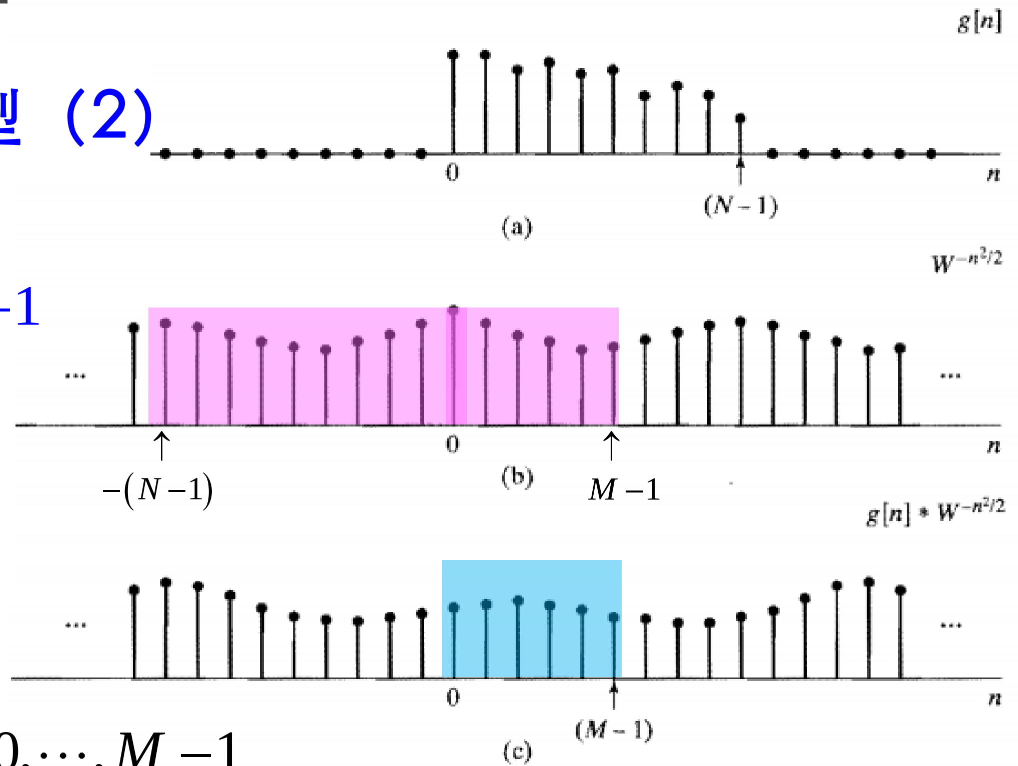


由于序列 $W^{-n^2/2}=e^{j\Delta\omega n^2/2}$ 是频率以间隔 $n\Delta\omega$ 线性增加的复指数序列，在雷达系统中这种信号称为线性调频信号（Chirp），因此该方法称为线性调频变换。



9.6.2 线性调频变换算法

◆ 线性调频变换算法实现模型（2）



线性调频变换算法涉及的序列

由于 $g[n]$ 为有限长序列，获得 $g[n]*W^{-n^2/2}$ 在区间 $n=0,\cdots,M-1$ 内的输出仅需用序列 $W^{-n^2/2}$ 在区间 $n=-(N-1),\cdots 0,\cdots,M-1$ 上的样本。

定义一个有限脉冲响应

$$h[n]=\begin{cases} W^{-n^2/2}, & n=-(N-1),\cdots 0,\cdots,M-1 \\ 0, & \text{其它} \end{cases}$$

$h[n]$ 非因果

因此

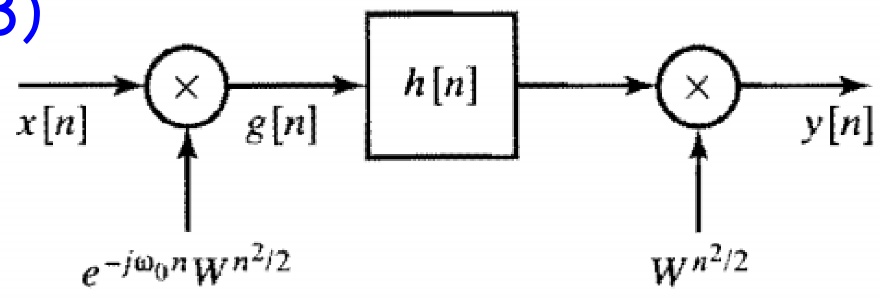
$$g[n]*W^{-n^2/2}=g[n]*h[n],\quad n=0,\cdots,M-1$$

9.6.2 线性调频变换算法

◆ 线性调频变换算法实现模型（3）

用有限长脉冲响应 $h[n]$ 代替无限脉冲响应 $W^{-n^2/2}$，线性调频变换可简化如右图，傅里叶变换输出采样值可表示为

$$X\left(e^{j\omega_n}\right)=y[n],\quad n=0,1,\cdots M-1$$



有限长脉冲响应线性调频变换系统框图

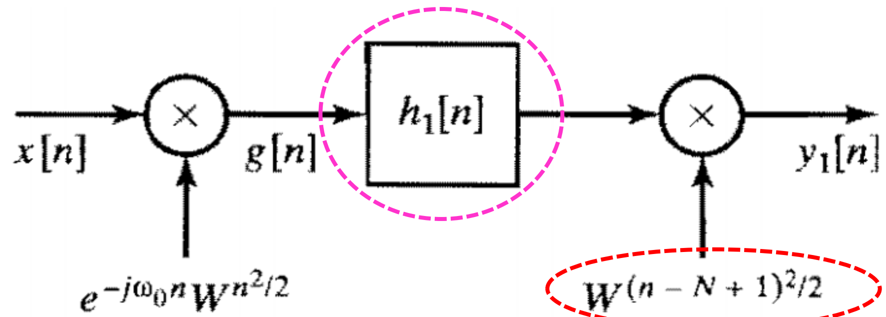
通过延迟非因果脉冲响应 $h[n]$ 可得因果脉冲响应

$$h_1[n]=h[n-(N-1)]=\begin{cases}W^{-(n-N+1)^2/2}, & n=0,1,\cdots,M+N-2\\ 0, & \text{其它}\end{cases}$$

傅里叶变换的采样值可表示为

$$X\left(e^{j\omega_n}\right)=y[n]=y_1[n+N-1]$$

$$,\quad n=0,1,\cdots M-1$$



因果有限长脉冲响应线性调频变换系统框图

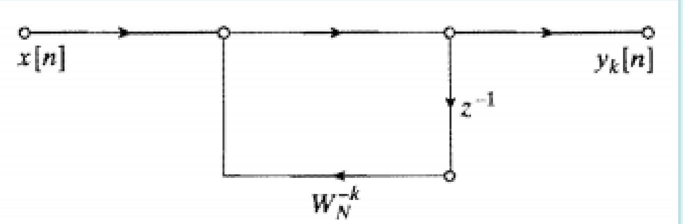
◆ 线性调频变换与DFT的关系

当要计算的频率样本与DFT一致时，线性调频变换算法可进行简化

令 $\omega_0 = 0$ ，$M = N$ ，$\Delta\omega = 2\pi / N$，$W = e^{-j2\pi/N} = W_N$ ，

若N为偶数，且由 $W_N^N = e^{-j2\pi} = 1$，则有

$$W_N^{-(n-N)^2/2} = W_N^{-n^2/2}$$



因果脉冲响应 $h_1[n]$附加额外单位延迟后，

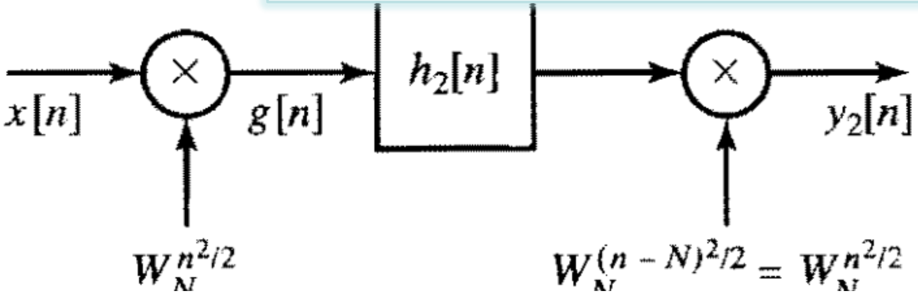
$$h_2[n] = h[n-N] = \begin{cases} W_N^{-(n-N)^2/2} = W^{-n^2/2}, & n = \\ 0, & \end{cases}$$

Goertzel算法对应的系统脉冲响应为 $W_N^{-kn}u[n]$ ，每个系统的输出为单个DFT输出值

此时傅里叶变换输出采样值为

$$X\left(e^{j2\pi n/N}\right) = y[n] = y_2[n+N]$$

$$, \quad n = 0, 1, \cdots M-1$$



可得出DFT样本的线性调频变换系统框图

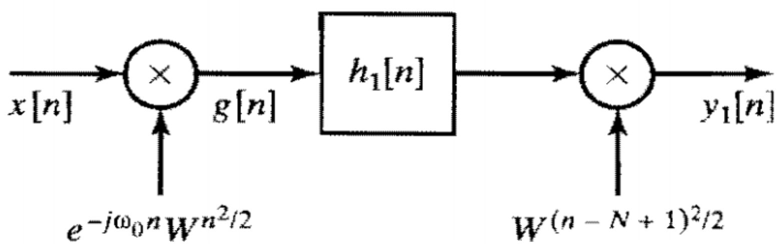
9.6.2 线性调频变换算法

$$h_1[n]=W^{-(n-N+1)^2/2}, n=0,1,\cdots,M+N-2$$
$$W=e^{-j\Delta\omega}$$

◆ 示例：线性调频变换的参数选择

假设有限长序列 $x[n]$ 仅在区间 $n=0,\cdots,25$ 上为非零，且要计算频点 $\omega_k=2\pi/27+2\pi k/1024$， $k=0,\cdots,15$ 上DTFT $X\left(e^{j\omega}\right)$ 处16个样本

选择右图线性调频变换系统的参数，即可通过卷积获得要求频点上的值



因果有限长脉冲响应线性调频变换系统框图

设所求样本个数 $M=16$，
序列长度 $N=26$，
样本的起始频率 $\omega_0=2\pi/27$，
频率增量 $\Delta\omega=2\pi/1024$

DTFT采样值可表示为

$$X\left(e^{j\omega_n}\right)\Big|_{\omega_n=2\pi/27+2\pi n/1024}=y_1[n+25], \quad n=0,1,\cdots 15$$

由 (9.48)式可得因果脉冲响应为

$$h_1[n]=\begin{cases}\left(e^{-j2\pi/1024}\right)^{-(n-25)^2/2}, & n=0,1,\cdots,40\\ 0, & \text{其它}\end{cases}$$

线性调频算法对输入序列长度远小于DFT点数和存在起始频偏的DFT计算非常有效。

ShanghaiTech University

第九章：离散傅里叶变换的计算

- 9.1 离散傅里叶变换的高效计算
- 9.2 Goertzel算法
- 9.3 按时间抽取的FFT算法
- 9.4 按频率抽取的FFT算法
- 9.5 实现问题考虑
- 9.6 用卷积实现DFT
- 9.7 有限寄存器长度的影响

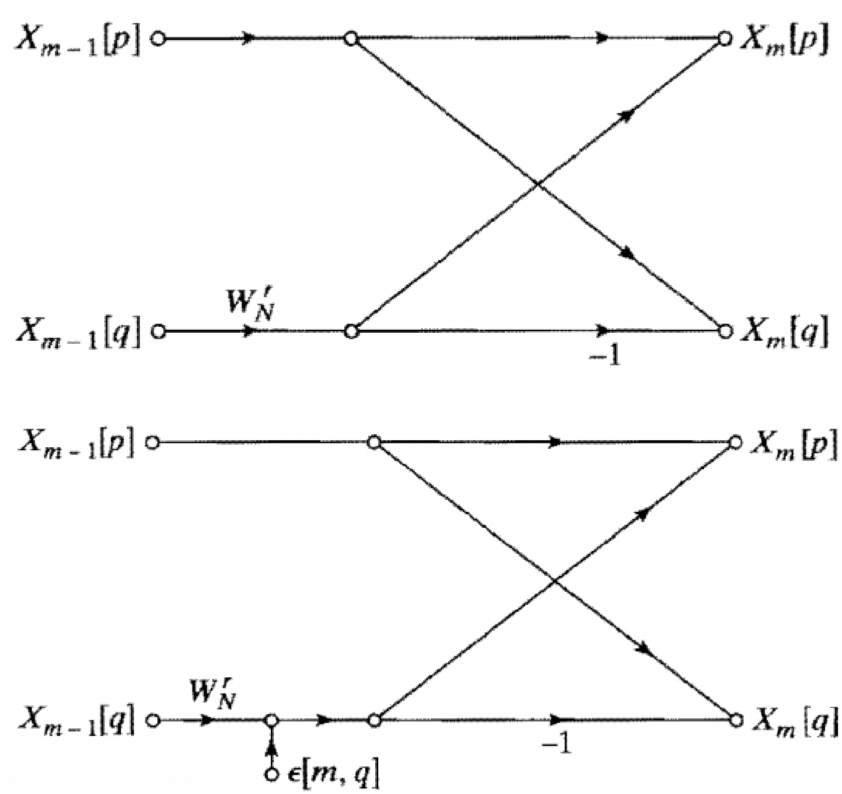
◆ DFT计算舍入量化线性噪声模型

对于基2按时间抽取的N点DFT算法，第m级蝶形计算可表示为

$$X_m[p]=X_{m-1}[p]+W_N^r X_{m-1}[q]$$
$$X_m[q]=X_{m-1}[p]-W_N^r X_{m-1}[q]$$

其中 $X_{m-1}[l]$和 $X_m[l]$ 分别为第m级计算的输入和输出数列，

$$l=0,1,\cdots,N-1;\quad m=1,2,\cdots,v$$



把定点乘法运算与加性噪声发生器联系在一起建立舍入噪声模型

$\varepsilon(m,q)$表示在由第(m-1)级数列计算第m级数列的第q个元素与复系数相乘的结果进行量化时所产生的误差。

9.7 有限寄存器长度的影响

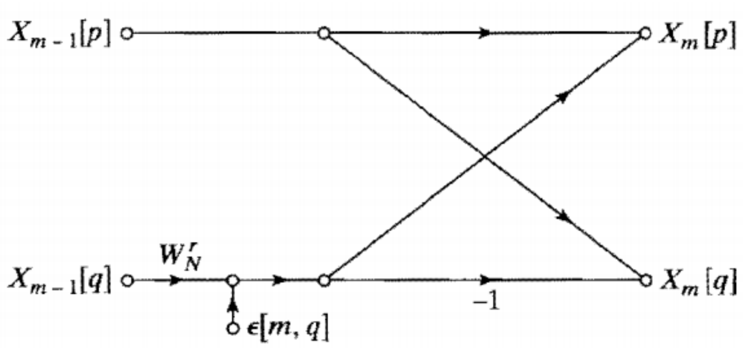
◆ DFT计算舍入量化线性噪声模型（续）

假设FFT输入为复序列，每次复乘由4个实乘组成，每个实数乘法器舍入量化（B+1比特带符号）产生的误差有下列性质：

1、误差是在 $\pm(1/2)\times 2^{-B}$ 区间上均匀分布的随机变量，每个实噪声源的方差为 $2^{-2B}/12$ 。

2、误差相互之间不相关。

3、误差与输入和输出序列不相关



由于4个误差序列为互不相关零均值白噪声，且方差相同，则复乘法器量化误差方差为

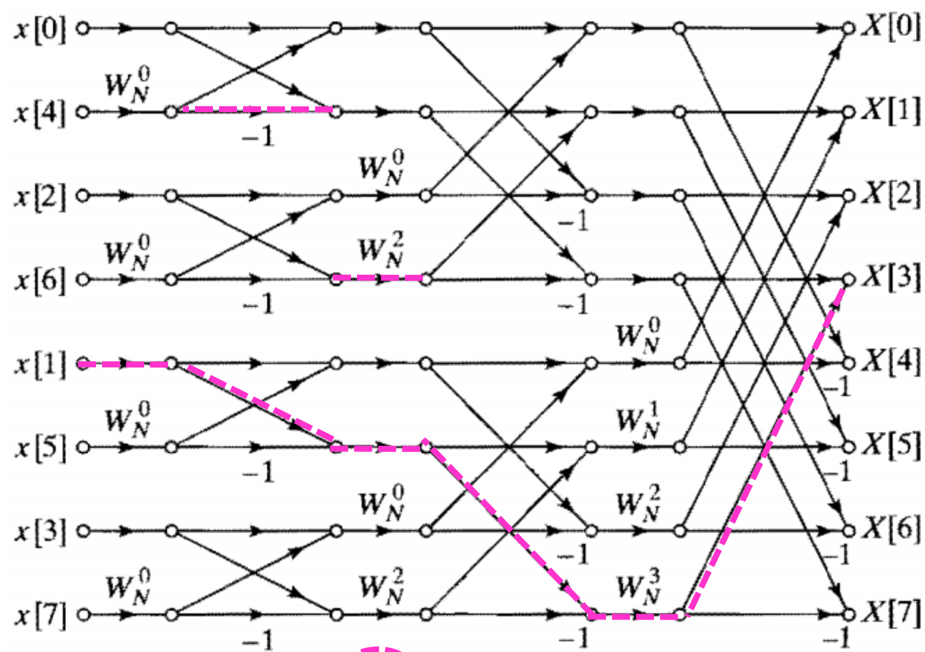
$$\mathscr{E}\left\{\left|\varepsilon(m,q)\right|^2\right\}=4\cdot\frac{2^{-2B}}{12}=\frac{1}{3}\cdot 2^{-2B}=\sigma_B^2$$

9.7 有限寄存器长度的影响

◆ DFT计算流图结构特点

1. 流图中任一节点到相连的其它任一节点的传递函数均与单位幅度的(复)常数相乘。

2. 每个输出节点均与$(N-1)$个蝶形相连接。

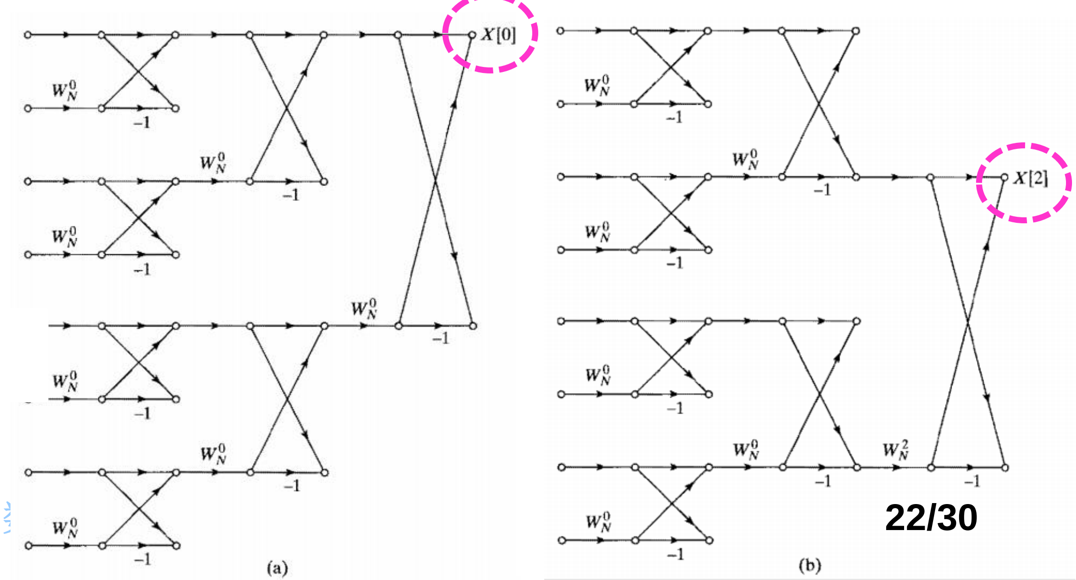


由于每个蝶形至多引入1个复噪声源，且噪声源相互独立，则第k个DFT值输出量化噪声均方值上限为

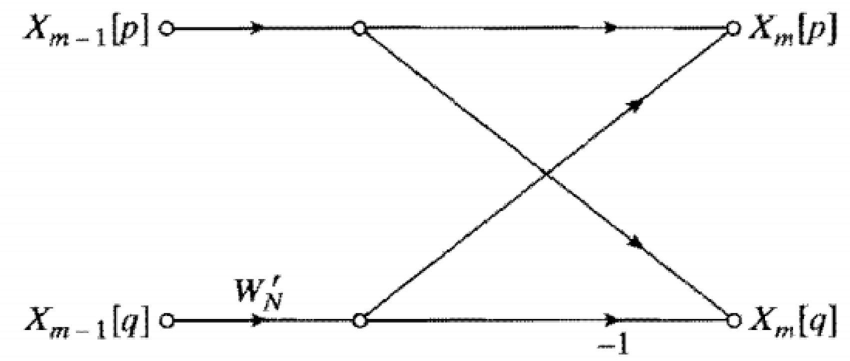
$$\mathscr{E}\left\{\left|F[k]\right|^2\right\}=(N-1)\sigma_B^2$$

当N较大时，可近似为

$$\mathscr{E}\left\{\left|F[k]\right|^2\right\}\approx N\sigma_B^2$$



◆ DFT计算节点避免溢出条件



由第m级蝶形计算关系

$$X_m[p] = X_{m-1}[p] + W_N^r X_{m-1}[q]$$

$$X_m[q] = X_{m-1}[p] - W_N^r X_{m-1}[q]$$

可得

$$\max\left(\left|X_{m-1}[p]\right|, \left|X_{m-1}[q]\right|\right) \le \max\left(\left|X_m[p]\right|, \left|X_m[q]\right|\right)$$

$$\max\left(\left|X_m[p]\right|, \left|X_m[q]\right|\right) \le 2\max\left(\left|X_{m-1}[p]\right|, \left|X_{m-1}[q]\right|\right)$$

由于蝶形计算输出的最大幅度逐级非递减（后级不低于前级），若要FFT最终输出幅度小于1，则各节点幅度必小于1，即任一数列均不会溢出。

9.7 有限寄存器长度的影响

◆ DFT计算节点避免溢出条件（续）

由DFT的定义可得

$$\left|X[k]\right| = \left|\sum_{n=0}^{N-1} x[n] W_N^{kn}\right| \le \sum_{n=0}^{N-1} \left|x[n]\right|, \quad 0 \le k \le N-1$$

因此，保证FFT输出不溢出，即

$$\left|X[k]\right| < 1, \quad 0 \le k \le N-1$$

的充分必要条件是输入序列幅度满足

$$\left|x[n]\right| < 1/N, \quad 0 \le n \le N-1$$

$$|x[n]| < 1/N \to |x[n]|^2 < 1/N^2$$
$$|\mathrm{Re}\{x[n]\}|^2 = |\mathrm{Im}\{x[n]\}|^2 < 1/2N^2$$
$$|\mathrm{Re}\{x[n]\}| = |\mathrm{Im}\{x[n]\}| < 1/\sqrt{2}N$$

◆ DFT算法（量化）输出噪声—信号比

假设输入序列值为互不相关复序列，序列值的实部和虚部也不相关，且实部和虚部的幅度密度在 $\pm 1/\left(\sqrt{2}N\right)$ 之间均匀分布

则复输入序列的均方值为

$$\mathscr{E}\left\{|x[n]|^2\right\} = \frac{1}{3N^2} = \sigma_x^2$$

$$\sigma_x^2 = 2 \cdot \int_{-1/\sqrt{2}N}^{1/\sqrt{2}N} x_r^2 \frac{N}{\sqrt{2}} dx_r$$

幅度密度

由DFT的定义式

$$X[k] = \sum_{n=0}^{N-1} x[n] W_N^{kn}$$

则FFT输出信号的均方值为

$$\mathscr{E}\left\{|X[k]|^2\right\} = \sum_{n=0}^{N-1} \mathscr{E}\left\{|x[n]|^2\right\} \left|W_N^{kn}\right|^2 = N\sigma_x^2 = \frac{1}{3N}$$

要保持噪信比不变，若N增加1倍(N增为2 N)，需要位宽B增加1位

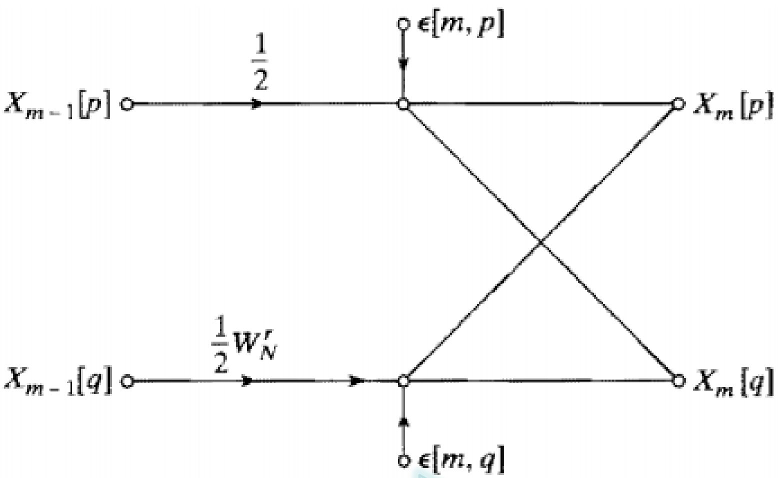
最终FFT输出噪声-信号比为

$$\mathscr{E}\left\{|F[k]|^2\right\} \Big/ \mathscr{E}\left\{|X[k]|^2\right\} = N\sigma_B^2 \Big/ \left(N\sigma_x^2\right) = 3N^2\sigma_B^2 = N^2 2^{-2B}$$

◆ DFT算法输入信号比例改变方法

由于FFT计算从上一级到下一级最大幅度至多增加一倍，只要 $|x[n]|<1$ 并且在每一级的输入加入1/2的衰减就可防止溢出

若每个蝶形输入端加入1/2的比例因子，舍入噪声模型如右图



假设噪声源实部和虚部不相关，且各部分在 $\pm(1/2)2^{-B}$ 之间均匀分布

则复噪声的方差为

$$\mathscr{E}\left\{|\varepsilon(m,q)|^2\right\}=\mathscr{E}\left\{|\varepsilon(m,p)|^2\right\}=\sigma_B^2=\frac{1}{3}\cdot 2^{-2B}$$

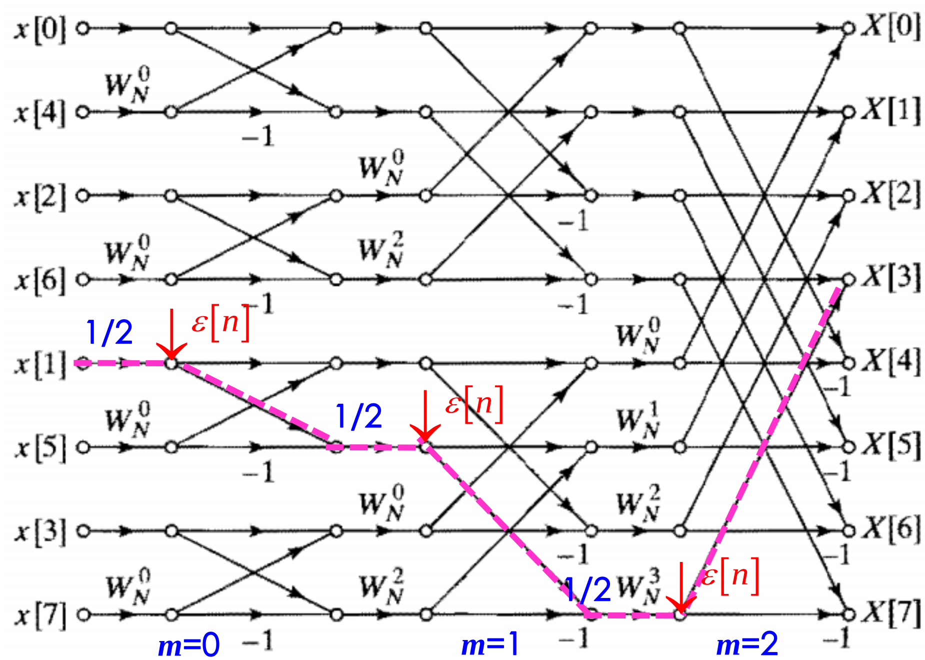
噪声信号叠加于1/2衰减之后，因此会降低SNR

9.7 有限寄存器长度的影响

◆ DFT算法蝶形输入加比例因子的噪声模型特性

每个噪声源经历整个流图时所受到的衰减与该噪声源最初所在的数列有关

起始于第m列的噪声源传播至输出端时所乘常数为 $(1/2)^{v-m-1}$

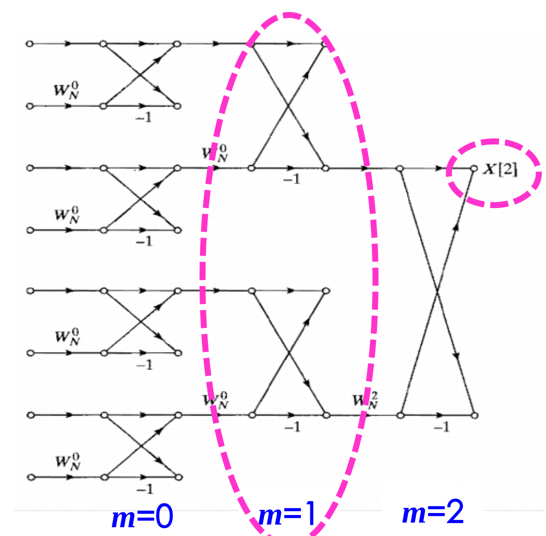


对于N=8的FFT，每个输出节点连接的蝶形：

✓起始于第（v-1）列的1个蝶形

✓起始于第（v-2）列的2个蝶形

✓起始于第（v-3）列的4个蝶形



对于N=2^v的一般情况，每个输出节点与第m列的2^{v-m-1}个蝶形相连。

9.7 有限寄存器长度的影响

◆ DFT算法蝶形输入加比例因子噪声模型特性（续）

由于对于$N=2^v$的一般情况，各输出节点与第m列的2^{v-m-1}个蝶形相连，

即第m列输出节点与$2\times 2^{v-m-1}$个噪声源相连，因此，DFT的各输出的噪声方差为

$$\mathscr{E}\left\{|F[k]|^2\right\}=\sum_{m=0}^{v-1}\left((1/2)^{v-m-1}\right)^2\times 2\times 2^{v-m-1}\sigma_B^2$$

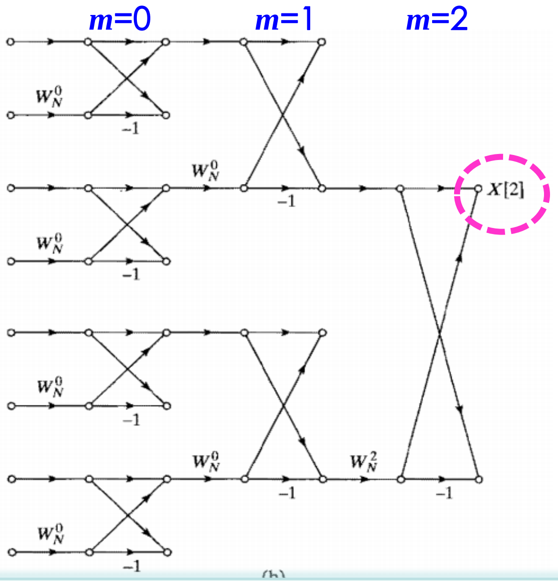
$$=\sigma_B^2\sum_{m=0}^{v-1}(1/2)^{v-m-2}=4\sigma_B^2\left(1-0.5^v\right)$$

N较大时，噪声方差可近似为

$$\mathscr{E}\left\{|F[k]|^2\right\}\approx 4\sigma_B^2=\frac{4}{3}\cdot 2^{-2B}$$

则FFT输出噪声-信号比为

$$\mathscr{E}\left\{|F[k]|^2\right\}\Big/\mathscr{E}\left\{|X[k]|^2\right\}=4\sigma_B^2\Big/\left(N\sigma_x^2\right)=\frac{4}{3}\cdot 2^{-2B}\Big/\left(1/3N\right)=4N\cdot 2^{-2B}$$



对于各输出节点，总共要经历v级蝶形；对于第m级，共有2^{v-m-1}个蝶形；每个蝶形有2个噪声源；各噪声源到输出端需经历$v-m-1$次1/2衰减

要保持噪信比不变，若N增加1倍，需位宽B增加1/2位