《数字信号处理》

上机(实验)指导

北京工商大学计算机与信息工程学院

2012.12

实验一：离散时间系统分析　2 学时

1. 实验目的

(1) 回顾连续时间系统的基本概念和表示方法；
(2) 加深对离散时间系统基本概念的理解；
(3) 熟悉离散时间系统的有关特性；
(4) 掌握离散时间系统输出序列的计算方法。

2. 实验任务

若离散时间系统的输入序列为：

$$x(n)=[3 \quad 11 \quad 7 \quad \underset{\uparrow}{0} \quad -1 \quad 4 \quad 2] \qquad -3\le n\le 3$$

系统的单位抽样响应序列为：

$$h(n)=[2 \quad \underset{\uparrow}{3} \quad 0 \quad -5 \quad 2 \quad 1] \qquad -1\le n\le 4$$

试求该系统的输出响应 $y(n)$。

3. 实验原理

(1) 离散时间系统可以如图 1-1 所示的原理

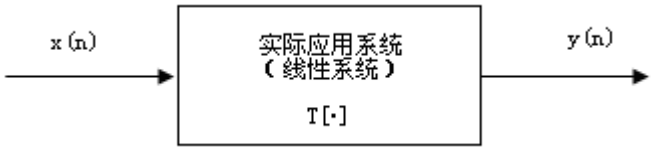


图 1-1　离散时间系统

(2) 单位抽样响应 h(n)是指输入为单位抽样信号 $\delta(n)$ 时，系统的输出 y(n)。即：

$$h(n)=T[\delta(n)] \qquad (1\text{-}1)$$

(3) 对于任意输入 x(n)，系统的输出 y(n)为：

$$y(n)=T[x(n)]$$

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

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

$$= \sum_{m=-\infty}^{\infty} x(m)h(n-m) \qquad (1\text{-}2)$$

4．实验结果

$y(n) = [6 \quad 31 \quad 47 \quad 6 \quad -51 \quad -5 \quad 41 \quad 18 \quad -22 \quad -3 \quad 8 \quad 2]$

其中：$-4 \le n \le 7$ 

实验二：线性差分方程求解　2 学时

1. 实验目的

(1) 加深对差分方程基本概念的理解；
(2) 熟悉线性差分方程的基本形式；
(3) 掌握线性差分方程求解的基本方法；
(4) 巩固高级语言程序设计中的常用算法。

2. 实验任务

根据式(2-1)所示的差分方程：

$$y(n) - y(n-1) + 0.9y(n-2) = x(n) \qquad \forall n \qquad (2\text{-}1)$$

完成以下要求：

(1) 计算并画出抽样响应 h(n)；
(2) 计算并画出阶跃响应 y(n)。

3. 实验原理

线性差分方程的一般形式如下：

$$\sum_{k=0}^{N} a_k y(n-k) = \sum_{m=0}^{M} b_m x(n-m) \qquad \forall n \qquad (2\text{-}2)$$

(1) 如果 $a_N \neq 0$，则差分方程(2-2)为 N 阶差分方程。

(2) 差分方程的解法有序列域求解法和变换域求解法两类。而序列域求解法又包括迭代递推法和卷积和计算方法两种。

(3) 递推法求解差分方程：

差分方程(2-2)描述了根据输入和以前的输出计算输出的一个迭代递推方法，式(2-2)可以化为：

$$y(n) = \sum_{m=0}^{M} b_m x(n-m) - \sum_{k=1}^{N} a_k x(n-k) \qquad (2\text{-}3)$$

方程的解可以表示为通解 $y_G(n)$ 与特解 $y_s(n)$ 之和：

$$y(n) = y_G(n) + y_s(n) \tag{2-4}$$

特解 $y_G(n)$ 为：

$$y_G(n) = \sum_{k=1}^{N} c_k z_k^n \tag{2-5}$$

其中 z_k 是特征方程

$$\sum_{k=1}^{N} a_k z^k = 0$$

的 N 个根。特征方程对于系统的稳定性具有重要意义，若特征根满足如下条件，则系统稳定。

$$|z_k| < 1 \qquad k = 1,2,3,\cdots,N \tag{2-6}$$

4．实验结果

根据实验任务要求，实验结果如图 2-2 和 2-2 所示。

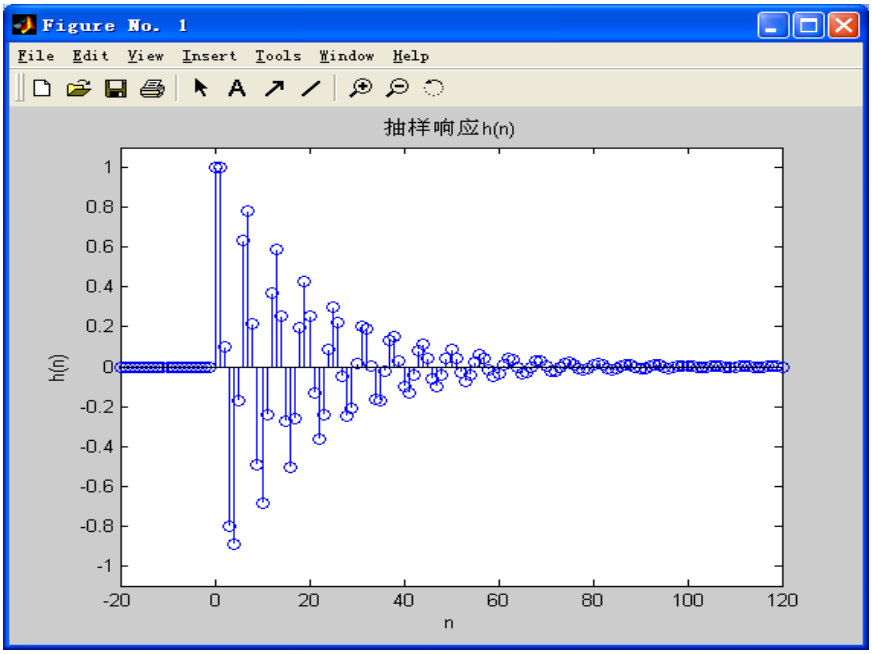


图 2-1 抽样响应 h(n)

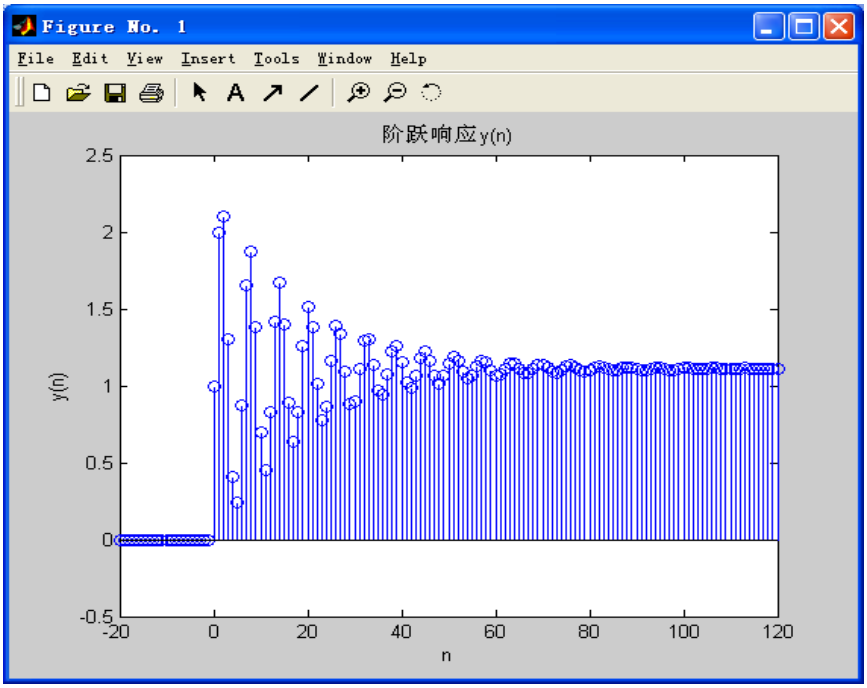


图 2-2 阶跃响应 y(n)

实验三：周期卷积运算　2 学时

1. 实验目的

(1) 加深对周期卷积的理解；

(2) 掌握卷积计算的原理和方法。

2. 实验任务

若两周期序列 $\tilde{x}_1(m)$和$\tilde{x}_2(m)$分别如图 3-1 和 3-2 所示：

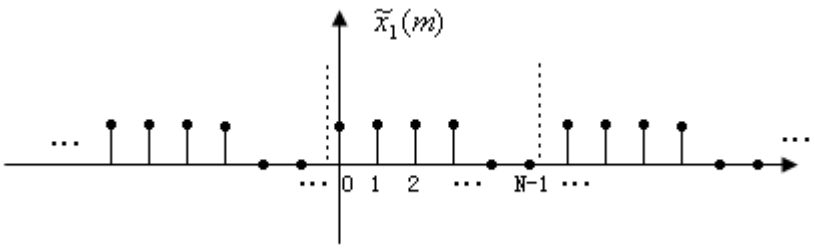


图 3-1 序列 $\tilde{x}_1(m)$ 的波形图

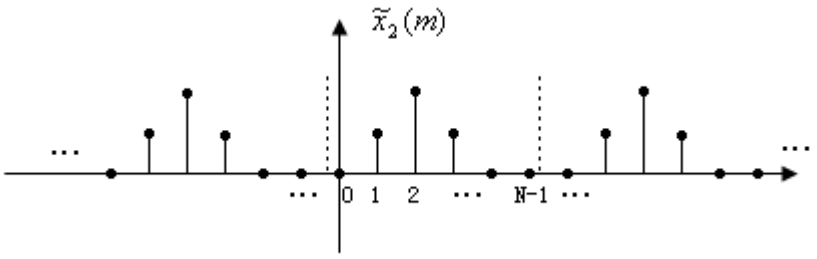


图 3-2 序列 $\tilde{x}_2(m)$ 的波形图

试求序列 $\tilde{x}_1(m)$和$\tilde{x}_2(m)$ 的周期卷积。

3．实验原理

若 $$\tilde{Y}(k) = \tilde{X}_1(k)\,\tilde{X}_2(k) \tag{3-1}$$

则：

$$\tilde{y}(n) = IDFS[\tilde{Y}(k)]$$

$$= \sum_{m=0}^{N-1} \tilde{x}_1(m)\tilde{x}_2(n-m) \tag{3-2}$$

和卷积运算一样，周期卷积也可以表示为：

$$\tilde{y}(n) = \sum_{m=0}^{N-1} \tilde{x}_2(m)\tilde{x}_1(n-m) \tag{3-3}$$

4．实验结果

根据周期卷积的原理，周期卷积如图 3-3 所示：

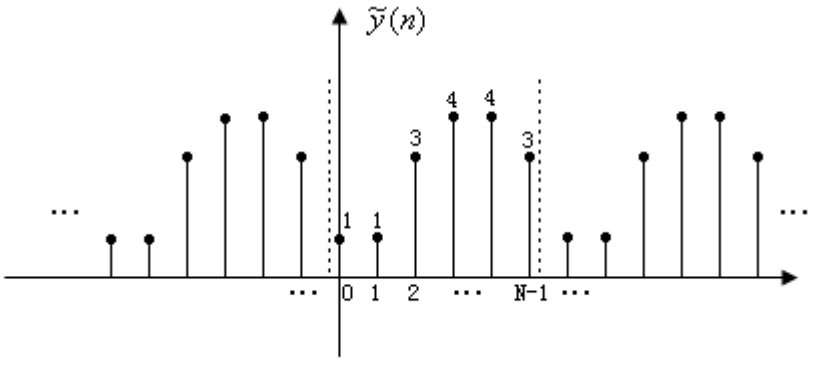


图 3-3　序列 $\tilde{y}(n)$ 的波形图

实验四：DFT/FFT 算法分析(4 学时)

1．实验目的

(1) 掌握离散傅立叶变换（DFT）的基本原理；

(2) 掌握 N 点 DFT 运算的运算量大小；

(3) 熟悉减少 DFT 运算减少运算量的方法；

(4) 熟悉 DFT 快速运算（FFT）的处理过程和计算方法。

2．实验任务

有一被噪声污染的混合信号，该混合信号 50Hz 和 120Hz 信号组成，噪声为零均值随机噪声。设采样频率为 1000Hz，利用离散傅立叶变换分析其频率成分。

3．实验原理

(1) DFT 的运算量

离散傅立叶变换的矩阵形式如下：

$$\begin{bmatrix} X(0) \\ X(1) \\ \vdots \\ X(N-1) \end{bmatrix} = \begin{bmatrix} W_N^{0\times 0} & W_N^{0\times 1} & \cdots & W_N^{0\times(N-1)} \\ W_N^{1\times 0} & W_N^{1\times 1} & \cdots & W_N^{1\times(N-1)} \\ \vdots & \vdots & \vdots & \vdots \\ W_N^{(N-1)\times 0} & W_N^{(N-1)\times 1} & \cdots & W_N^{(N-1)\times(N-1)} \end{bmatrix} \begin{bmatrix} x(0) \\ x(1) \\ \vdots \\ x(N-1) \end{bmatrix} \quad (4\text{-}1)$$

根据离散傅立叶变换公式及其逆变换，对于 N 点的序列 x(n)，无论是计算 X(k)（u=0,1,2,…,N-1）或 x(n) （u=0,1,2,…,N-1），以正变换为例（逆变换运算次数与正变换相同),每计算一个 X(k)的值,都需要计算 N 次复数乘法,即 $x(n)W_N^{ux}$ 及（N-1）次加法，考虑到计算机中乘法运算所需时间是加法运算所需时间要高一个数量级左右，本书讨论运算次数时均略去加法运算时间。因此，计算全部 N 个 X(k)需要 N^2 次复数乘法。

(2) DFT 的快速运算

为了减少 DFT 的运算量，除可以利用的 W_N 周期性适当减少一些计算量之外，可以采用如下快速运算方法。

FFT 算法根据算法原理可以分为基于时间分解（DIT）和基于频率分解（DIF）2 类算法，例如典型的 DIT 基 2 算法和 DIF 基 2 算法等。这里仅以 DIT 基 2 算法为例，介绍 FFT 的基本形式及运算流程。

FFT 算法要求离散序列的点数满足关系，即 2 的整数幂次方。如果条件不满足，可以在序列后面补 0，直至满足 $N=2^L$ 为止。这种 N 为 2 的整数幂次方的 FFT 算法称为基 2 FFT 算法。

将具有 $N=2^L$ 个点的离散序列 x(n)按 x 的奇偶分为如下两个子序列：

$$\begin{cases} f_1(n) = x(2n) & n = 0,1,2\cdots\dfrac{N}{2}-1 \\ f_2(n) = x(2n+1) & n = 0,1,2\cdots\dfrac{N}{2}-1 \end{cases} \tag{4-2}$$

于是序列 x(n)的 DFT 变换可以化为

$$X(k) = \sum_{x=0}^{N-1} f(x) W_N^{ux}$$

$$= \sum_{x=0}^{N-1} f(x) W_N^{ux} + \sum_{x=0}^{N-1} f(x) W_N^{ux}$$

x为偶数　　x为奇数

$$= \sum_{x=0}^{\frac{N}{2}-1} f(2r) W_N^{2ru} + \sum_{r=0}^{\frac{N}{2}-1} f(2r+1) W_N^{2ru}$$

$$= \sum_{x=0}^{\frac{N}{2}-1} f_1(r) W_N^{2ru} + W_N^u \sum_{r=0}^{\frac{N}{2}-1} f_2(r) W_N^{2ru} \tag{4-3}$$

由于 $f_1(x)$和 $f_2(x)$分别是两个 N/2 点的离散序列，该两序列的 DFT 变换形式如下：

$$\begin{cases} X_1(k) = \sum_{x=0}^{\frac{N}{2}-1} f_1(r) W_{\frac{N}{2}}^{kx} \\ X_2(u) = \sum_{x=0}^{\frac{N}{2}-1} f_2(r) W_{\frac{N}{2}}^{kx} \end{cases} \tag{4-4}$$

因此有

$$X(k) = X_1(k) + W_N^k X_2(k) \qquad k = 0,1,2,\cdots,\frac{N}{2}-1 \tag{4-5}$$

这样 X(k)的前 N/2 个点可以通过式(4-5)求出，同样，可以推导出后一半点的计算方法如下：

$$X(\frac{N}{2}+k) = X_1(k) - W_N^k X_2(k) \qquad k = 0,1,2,\cdots,\frac{N}{2}-1 \tag{4-6}$$

于是离散序列 x(n)经过第一次分解以后，通过(4-5)和 (4-6)可以计算出 X(k)，以 8 点序列为例，上述分解过程如图 4-1 所示。复数乘法的运算次数为：

$$2\times(\frac{N}{2})^2 + \frac{N}{2} = \frac{N^2}{2} + \frac{N}{2}$$

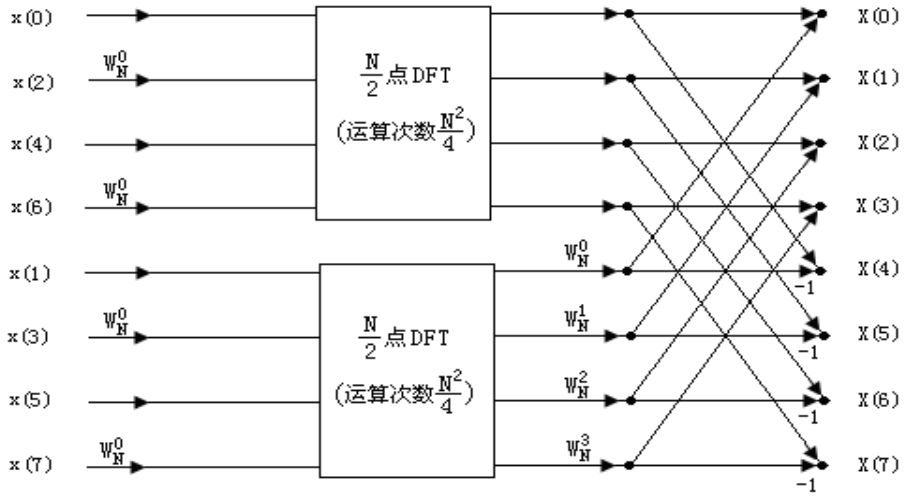


图 4-1　DFT 经过第一次分解后的运算流程图

比直接采用DFT变换公式计算大约减少了一半的运算量，同样的办法对 $x_1(n)$ 和 $x_2(n)$按序号的奇偶分别再分解为两个子序列，直到每一个子序列只有 2 点为止，这时计算量可以最大限度的减少。以 8 点的序列 x(n)为例，需要经过 3 级分解，就可以实现每一个子序列为 2 点，如图 4-2 所示。根据数字信号处理理论，由于每一级有 8/2=4 个基本碟形运算，因此，全部分解以后总的运算次数（即 8 点 FFT）C_{F8} 为

$$C_{F8} = \frac{8}{2} \times \log_2^{\ 8} = 12$$

而直接采用 DFT 计算需要 64 次复数乘法，FFT 的运算量不足 DFT 的 20%，运算次数已大大减少。

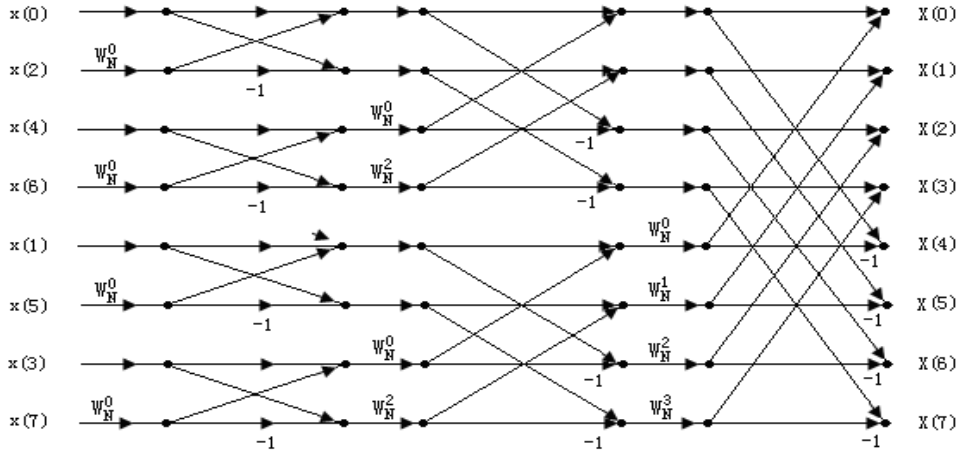


图 4-2　FFT 蝶式运算流程图

FFT 快速算法另一类典型的算法是 DIF 基 2 算法，即按频率抽取法，其基本思路与 DIT 基 2 算法类似，也是将序列 x(n)分解，直到每个子序列两点为止；但分解的方法不同，DIF 基 2 算法的分解为 2 个子序列的方法是直接把序列分为前一半点（第一次分解为 N/2 点）和后一半点（第一次分解为 N/2 点）分解为 2 个序列，然后对 DFT 变换公式进行改进从而实现 FFT 快速算法。DIF 基 2 算法与 DIF 基 2 算法的运算次数相等。

(3)　FFT 的运算量

推广到 $N=2^L$ 点的一般情况，这时，需要经过 L 级分解才可以实现每个子序

列为 2 点。而每一级总共有 N/2 个基本碟形预算，因此，总的运算量为：

$$C_F = \frac{N}{2}L = \frac{N}{2} \times \log_2^N \tag{4-7}$$

所以 DFT 算法与 FFT 算法的运算量之比为：

$$r_{DF} = \frac{C_F}{C_D} = \frac{N^2}{\frac{N}{2} \times \log_2^N} = \frac{2N}{\log_2^N} \tag{4-8}$$

根据式(4-8)，当 N 越大，比值就越大，因而 FFT 算法计算量的优越性体现就越明显。表 4-1 是 N 取不同值时，FFT 与 DFT 运算量之比值。

表 4-1　FFT 与 DFT 算法比较

N	N^2	$\frac{N}{2} \times \log_2^N$	r_{DF}
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	1048578	5120	204.8
2048	4194304	11264	372.4

（部分有兴趣的同学可以选择 DFT 快速算法，编写 FFT 算法源代码并进行 2^N 点验证）

4．实验结果

实验结果如图 4-3 所示，通过对信号进行离散傅立叶变换，可以明显看出混合信号中的 2 个频率成分，即 50Hz 和 100Hz 频率成分。

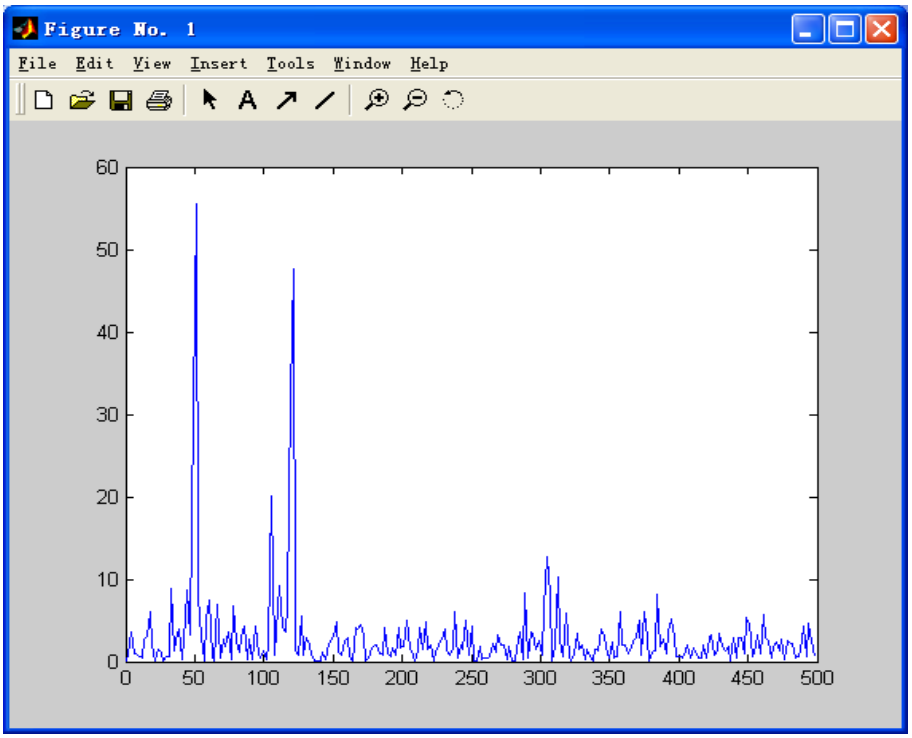


图 4-3　信号频谱图

实验五：IIR 滤波器设计　4 学时

1. 实验目的

(1) 加深对无限长冲击相应（IIR）滤波器基本概念的理解；
(2) 掌握无限长冲击相应滤波器的设计方法；
(3) 了解无限长冲击相应滤波器设计方法的类型；
(4) 加深对滤波器相关指标的物理意义的理解。

2. 实验任务

设计一个低通巴特沃思低通数字滤波器。滤波器指标如下：

(1) 通带截止频率为 300Hz；
(2) 采用频率为 300HZ；
(3) 巴特沃思滤波器的阶数为 3 阶；
(4) 绘出巴特沃思滤波器的频率响应特性曲线。

3. 实验原理

(1) 先设计模拟低通滤波器，然后再转化为低通数字滤波器。模拟低通数字滤波器是由指标参数$\Omega_p, R_p, \Omega_s A_s$确定的，设计巴特沃思滤波器的

思想就是根据这些指标计算滤波器的阶次 N 和截止频率$\Omega_p, R_p, \Omega_s A_s$

(2) 设计依据如下：

N 阶低通低通巴特沃思滤波器的附幅值平方响应为：

$$|H_a(j\Omega)|^2 = \frac{1}{1+\left(\frac{\Omega}{\Omega_c}\right)^{2N}} \tag{5-1}$$

$$R_p = -10\log_{10}\left(\frac{1}{1+\left(\frac{\Omega_p}{\Omega_c}\right)^{2N}}\right) \tag{5-2}$$

$$A_s = -10\log_{10}\left(\frac{1}{1+\left(\frac{\Omega_s}{\Omega_c}\right)^{2N}}\right) \tag{5-3}$$

$$N = \frac{\log_{10}[(10^{\frac{R_p}{10}}-1)(10^{\frac{A_s}{10}}-1)]}{2\log_{10}(\frac{\Omega_p}{\Omega_s})} \tag{5-4}$$

若 N 为非整数，须对 N 按（5-4）方式进行取整

$$N = \left\lceil \frac{\log_{10}[(10^{\frac{R_p}{10}}-1)(10^{\frac{A_s}{10}}-1)]}{2\log_{10}(\frac{\Omega_p}{\Omega_s})} \right\rceil \tag{5-5}$$

即选择比小数数据大且最靠近 N 的整数值作为 N 的数值。

4. 实验参考结果

3 阶巴特沃思滤波器的频率响应特性曲线如图 5-1 所示。

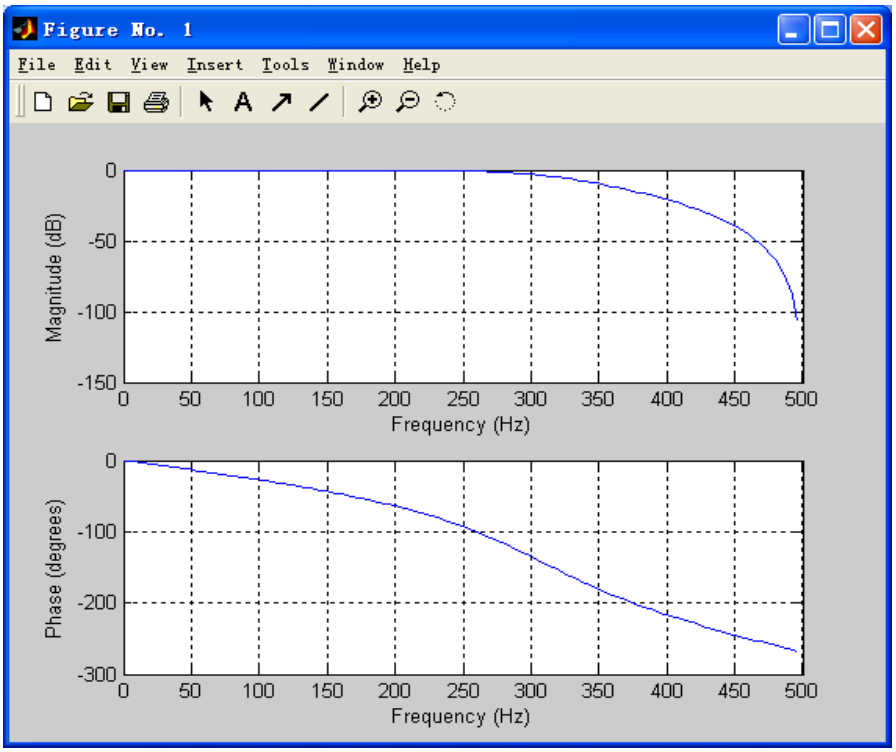


图 5-1　巴特沃思低通滤波器性能

对实验指导书的说明：

本实验指导共有 5 个实验，任课教师可根据上课的实际情况和课时要求进行选择或适当调整