**数**

**字**

**信**

**号**

**处**

**理**

**实**

**验**

**报**

**告**

实验二 [信号的采样与重建](#_实验目的)

实验三 [快速傅里叶变换及其应用](#_实验目的_1)

实验四 IIR数字滤波器的设计

实验五 FIR数字滤波器的设计

姓名：赵拯基

学号：06017419

日期：2020/6/10

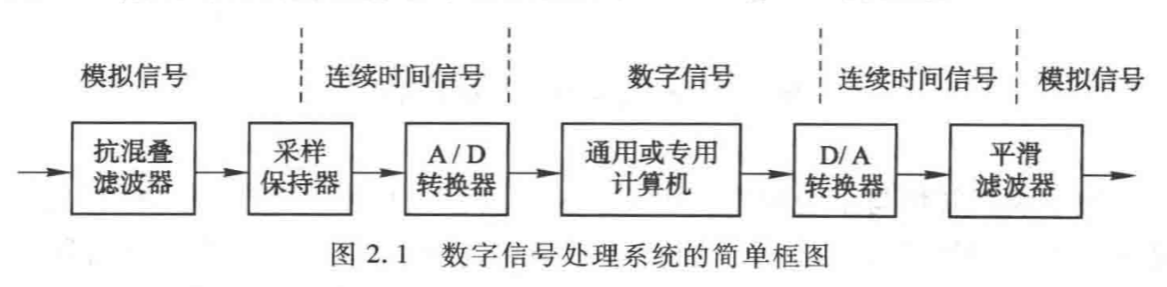
**实验二 信号的采样与重建**

# 实验目的

1. 通过实验加强有关信号采样与重建的基本概念。
2. 通过观察采样信号的混叠现象，进一步理解奈奎斯特采样频率的意义。
3. 通过实验，了解数字信号采样率转换过程中的频谱特征。
4. 对实际的音频文件作内插和抽取操作，体会低通滤波器在内插和抽取中的作用。

# 实验原理

数字信号处理是用数字序列表示信号，再通过数字计算机处理这些序列。一个简单数字信号处理系统的框图如下所示。



整个系统可以大致划分为以下几个部分：**模拟信号的采样和保持**、**模拟数字转换与量化（数字化）**、**信号处理**、**数字模拟转换与平滑滤波。**

信号采样与重建：

对于等间隔采样，采样周期是常数，采样频率记为，对应角频率为f。

理想化的采样函数即为：

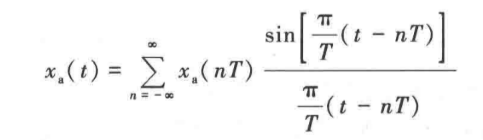
是周期信号，可以用傅里叶级数展开：

任意信号经过调制（采样）后为。得到采样信号的频谱（是的傅里叶变换）。是一个以为周期的连续函数，是对的周期延拓，并除以T。

通过一个带宽为/2的理想低通滤波器，就可以将提取出来。如果的最高频率超过/2，则会出现频谱的混叠，无法恢复。能够恢复出原始信号的最低采样频率称为**奈奎斯特**采样频率。

在满足奈奎斯特采样频率的条件下，通过一个理想低通滤波器：

得到输出。**时域卷积等于频域相乘，**得到。函数是函数的逆傅里叶变换。因此也可以表示为如下所示：



它表明时间函数由它的采样值加权表达。

更多关于插值和抽取的内容请参考书本。

# 实验步骤和实验程序

实验题目一：

采样混叠，对一个模拟信号进行等间采样，采样频率为8kHz。模拟信号由三个正弦函数相加得到。

实验题目三：

对一个采样的信号进行抽取和插值。

实验题目五：

通过MATLAB读取一个.wav音频文件，通过插值和抽取的方式生成不同采样率的音频文件并导出。

实验题目一相关代码：

x = 0:0.00000001:0.03;

y = sin(50 \* 2 \* pi \* x) +  sin(500 \* 2 \* pi \* x) + sin(1000 \* 2 \* pi \* x);

subplot(2,1,1);

plot(x,y);

axis([0 0.03 -3.5 3.5]);

title('original');

Ts = 1/8000;

n = 0:0.03 / Ts;

y = sin(50 \* 2 \* pi \* Ts \* n) +  sin(500 \* 2 \* pi \* Ts \* n) + sin(1000 \* 2 \* pi \* Ts \* n);

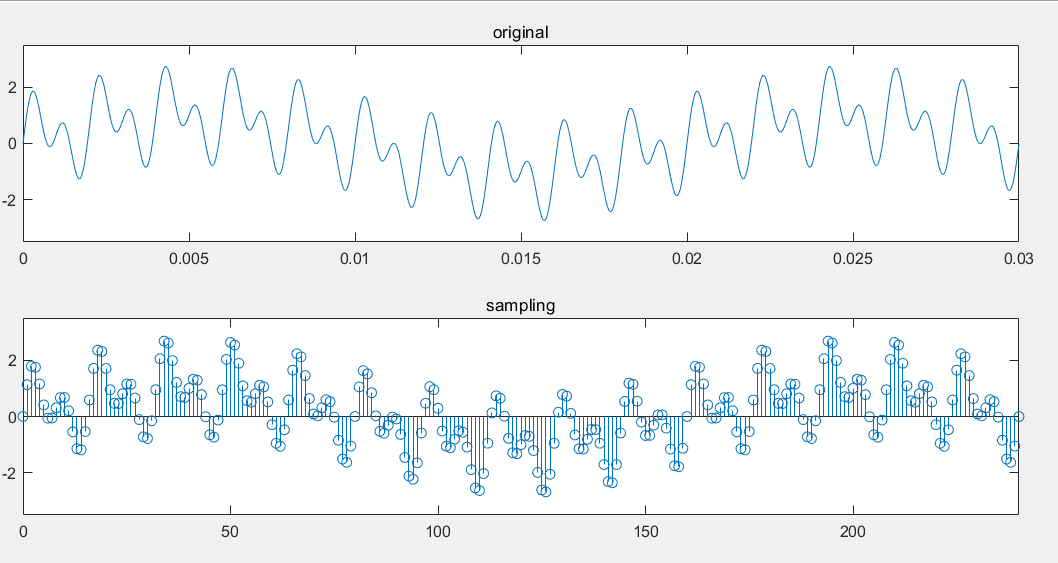
subplot(2,1,2);

stem(n,y);

axis([0 0.03 / Ts -3.5 3.5]);

title('sampling');

实验题目一结果：



实验题目三相关代码：

f = 1;fs = 16;N = 32;M = 2;L = 3;

n = 0:1:N-1;

x = cos(2 \* pi \* f / fs \* n);

y1 = x(1:M:N);            % 直接抽取序列

y2 = decimate(x,M);       % 采样率降低

subplot(3,2,1);

stem(n,x(1:N));

axis([0 N-1 -1 1]);

title('原始输入序列');

xlabel('n');ylabel('幅度');

% 抽取不经过滤波

subplot(3,2,3);

n = 0:N/M -1;

stem(n,y1);

title('抽取不经过滤波的输出');

xlabel('n');ylabel('幅度');

% 抽取经过滤波

subplot(3,2,5);

m = 0:N/M -1;

stem(m,y2);

title('抽取经过滤波的输出');

xlabel('n');ylabel('幅度');

n = 0:1:N-1;

subplot(3,2,2);

stem(n,x(1:N));

axis([0 N-1 -1 1]);

title('原始输入序列');

xlabel('n');ylabel('幅度');

% 不经抗镜像低通滤波

y1 = zeros(1,N\*L);

y1(1:L:N\*L) = x;

m = 0:N\*L-1;

subplot(3,2,4);

stem(m,y1(1:N\*L));

axis([0 N\*L-1 -1 1]);

title('内插不经抗镜像低通滤波的输出序列');

xlabel('n');ylabel('幅度');

% 经过抗镜像滤波

y2 = interp(x,L);

subplot(3,2,6);

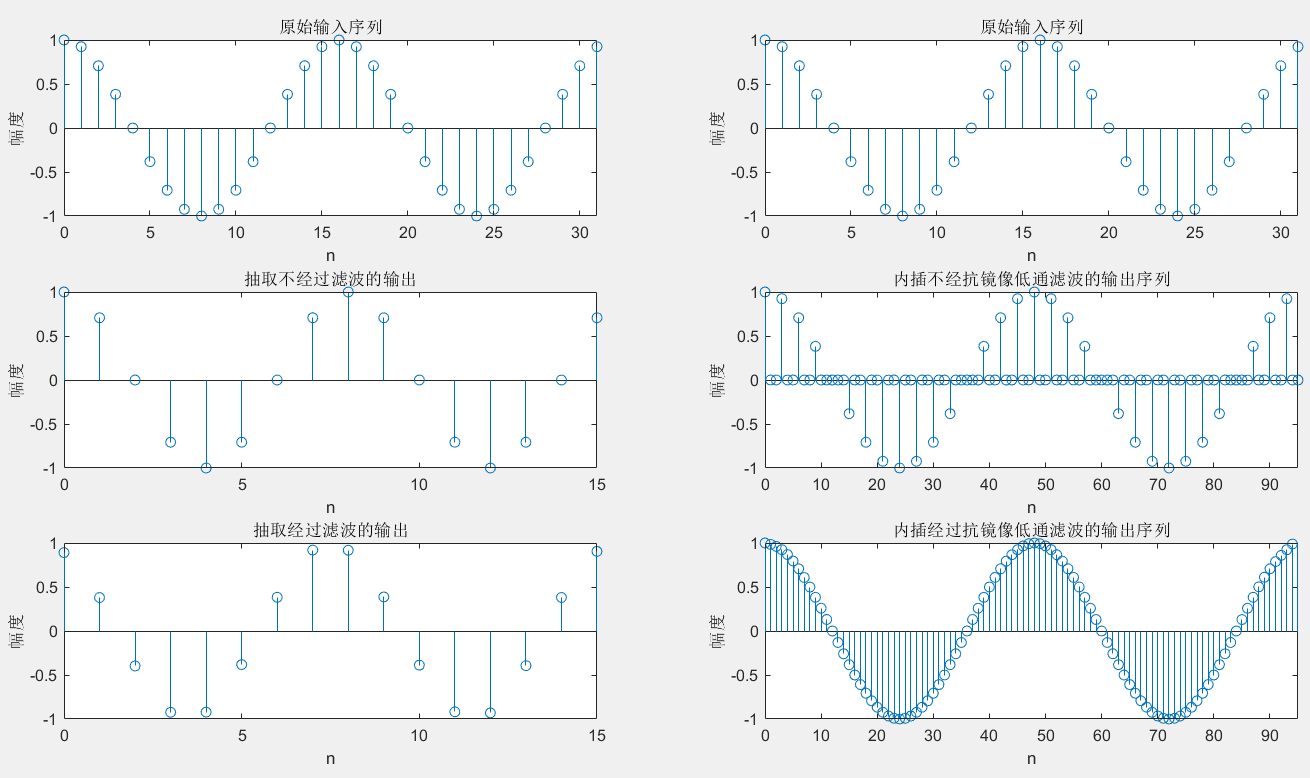
stem(m,y2(1:N\*L));

axis([0 N\*L-1 -1 1]);

title('内插经过抗镜像低通滤波的输出序列');

xlabel('n');ylabel('幅度');

实验题目三结果：



实验题目五相关代码：

[channel, Fs] = audioread('D:\DSP\exp\_2\question\_5\test.wav');

channel\_1 = channel(:,1)';

channel\_2 = channel(:,2)';

%%

% 44.1khz ת 48 khz

L = 160; M = 147;

channel\_r1 = decimate(interp(channel\_1, L), M);

channel\_r2 = decimate(interp(channel\_2, L), M);

channel\_r(:,1) = channel\_r1';

channel\_r(:,2) = channel\_r2';

audiowrite('test\_48khz.wav',channel\_r,48000);

clear channel\_r channel\_r1 channel\_r2

%%

% 44.1khz ת 32 khz

L = 320; M = 441;

channel\_r1 = decimate(interp(channel\_1, L), M);

channel\_r2 = decimate(interp(channel\_2, L), M);

channel\_r(:,1) = channel\_r1';

channel\_r(:,2) = channel\_r2';

audiowrite('test\_32khz.wav',channel\_r,32000);

clear channel\_r channel\_r1 channel\_r2

%%

% 44.1khz ת 22.05 khz

L = 1;M = 2;

channel\_r1 = decimate(interp(channel\_1, L), M);

channel\_r2 = decimate(interp(channel\_2, L), M);

channel\_r(:,1) = channel\_r1';

channel\_r(:,2) = channel\_r2';

audiowrite('test\_22.05khz.wav',channel\_r,22050);

clear channel\_r channel\_r1 channel\_r2

%%

% 44.1khz ת 16 khz

L = 160; M = 441;

channel\_r1 = decimate(interp(channel\_1, L), M);

channel\_r2 = decimate(interp(channel\_2, L), M);

channel\_r(:,1) = channel\_r1';

channel\_r(:,2) = channel\_r2';

audiowrite('test\_16khz.wav',channel\_r,16000);

clear channel\_r channel\_r1 channel\_r2

%%

% 44.1khz ת 8 khz

L = 80; M = 441;

channel\_r1 = decimate(interp(channel\_1, L), M);

channel\_r2 = decimate(interp(channel\_2, L), M);

channel\_r(:,1) = channel\_r1';

channel\_r(:,2) = channel\_r2';

audiowrite('test\_8khz.wav',channel\_r,8000);

实验五的结果涉及文件输出，相关文件请参考[github](https://github.com/ZZJ34/DSP-homework)

# 思考题回答

1. 对于周期性信号，在进行采样时，其采样周期必须满足采样定理，即采样频率应该大于信号最高频率的两倍，这样才能避免迭混，以便采样后仍能准确的恢复原信号。
2. 需要考虑的指标有：通带波动、最小阻带衰减、阻带下降率、通带边界频率和阻带边界频率等。欠采样，采样频率过低，无法满足奈奎斯特采样定理，会发生信号混叠，必定需要模拟抗混叠滤波器。
3. 会造成频谱成分丢失，但是可以减少信号在数字系统中的存储空间和运算压力。

**实验三 快速傅里叶变换及其应用**

# 实验目的

1. 通过实验加深对FFT的理解，熟悉MATALB中的有关函数。
2. 应用FFT对典型信号进行频谱分析。
3. 了解应用FFT进行信号频谱分析过程中可能出现的问题，以便在实际中正确应用FFT。
4. 应用FFT实现序列的线性卷积和相关。

# 实验原理

在各种信号序列中，有限长序列信号处理占有很重要地位，对有限长序列，我们可以使 用离散傅里叶变换(DFT)。这一变换不但可以很好的反映序列的频谱特性，而且易于用快速算法在计算机上实现，当序列x(n)的长度为N时，它的DFT定义为：

反变换为：

有限长序列的DFT是其Z变换在单位圆上的等距采样，或者说是序列x(n)傅里叶变换的等距采样，因此可以用于序列的谱分析。

FFT并不是与DFT不同的另一种变换，而是为了减少DFT运算次数的一种快速算法。 它是对变换式进行一次次分解，使其成为若干小点数的组合，从而减少运算量。常用的FFT是以2为基数的，其长度 。它的效率高，程序简单，使用非常方便，当要变换的序列长度不等于2的整数次方时，为了使用以2为基数的FFT，可以用末位补零的方法，使其长度延长至2的整数次方。

在运用DFT进行频谱分析的过程中可能产生三种误差：

**混叠**，序列的频谱是被采样信号的周期延拓，当采样速率不满足Nyquist定理时，就会发生频谱混叠，使得采样后的信号序列频谱不能真实的反映原信号的频谱。避免混叠现象的唯一方法是保证采样速率足够高，使频谱混叠现象不致出现，即在确定采样频率之前，必须对频谱的性质有所了解，在一般情况下，为了保证高于折叠频率的分量不会出现，在采样前，先用低通模拟滤波器对信号进行滤波。

**泄漏**，实际中我们往往用截短的序列来近似很长的甚至是无限长的序列，这样可以使用较短的DFT来对信号进行频谱分析，这种截短等价于给原信号序列乘以一个矩形窗函数，也相当于在频域将信号的频谱和矩形窗函数的频谱卷积，所得的频谱是原序列频谱的扩展。

泄漏不能与混叠完全分开，因为泄漏导致频谱的扩展，从而造成混叠。为了减少泄漏的影响，可以选择适当的窗函数使频谱的扩散减至最小。

**栅栏效应**，DFT是对单位圆上Z变换的均匀采样，所以它不可能将频谱视为一个连续函数，就一定意义上看，用DFT来观察频谱就好像通过一个栅栏来观看一个图景一样，只能在离散点上看到真实的频谱，这样就有可能发生一些频谱的峰点或谷点被“尖桩的栅栏”所拦住，不能别我们观察到。

减小栅栏效应的一个方法就是借助于在原序列的末端填补一些零值，从而变动DFT的点数，这一方法实际上是人为地改变了对真实频谱采样的点数和位置，相当于搬动了每一根“尖桩栅栏”的位置，从而使得频谱的峰点或谷点暴露出来。

# 实验步骤和实验程序

实验题目一：

观察高斯序列的时域和幅频特性，固定信号中参数p=8，改变q的值，使q分别等于2，4，8，观察它们的时域和幅频特性，了解当q取不同值时，对信号序列的时域幅频特性的影响；固定q=8，改变p，使p分别等于8，13，14，观察参数p变化对信号序列的时域及幅频特性的影响，观察p等于多少时，会发生明显的泄漏现象，混叠是否也随之出现？记录实验中观察到的现象，绘出相应的时域序列和幅频特性曲线。

实验题目三：

观察三角波和反三角波序列的时域和幅频特性，用N=8点FFT分析信号序列和的幅频特性，观察两者的序列形状和频谱曲线有什么异同？绘出两序列及其幅频特性曲线。

在和末尾补零，用N=16点FFT分析这两个信号的幅频特性，观察幅频特性发生了什么变化？两情况的FFT频谱还有相同之处吗？这些变化说明了什么。

实验题目五：

用FFT分别实现（p＝8，q＝2）和 （a＝0.1，f＝0.0625）的16点圆周卷积和线性卷积。

实验题目一相关代码：

N = 20;

n = 0:N-1;

y1 = xa(n,8,2);

y2 = xa(n,8,4);

y3 = xa(n,8,8);

[h1,w1] = freqz(y1);

[h2,w2] = freqz(y2);

[h3,w3] = freqz(y3);

figure(1);

% p=8,q=2

subplot(3,3,1);

stem(n,y1);

title('p=8,q=2');

xlabel('n');ylabel('Xa');

% p=8,q=4

subplot(3,3,2);

stem(n,y2);

title('p=8,q=4');

xlabel('n');ylabel('Xa');

% p=8,q=8

subplot(3,3,3);

stem(n,y3);

title('p=8,q=8');

xlabel('n');ylabel('Xa');

% p=8,q=2 的幅频

subplot(3,3,4);

plot(w1(1:512),abs(h1(1:512)));

title('p=8,q=2 幅频');

xlim([0,w1(512)]);

xlabel('w');ylabel('幅度');

% p=8,q=4 的幅频

subplot(3,3,5);

plot(w2(1:512),abs(h2(1:512)));

title('p=8,q=4 幅频');

xlim([0,w2(512)]);

xlabel('w');ylabel('幅度');

% p=8,q=8 的幅频

subplot(3,3,6);

plot(w3(1:512),abs(h3(1:512)));

title('p=8,q=8 幅频');

xlim([0,w3(512)]);

xlabel('w');ylabel('幅度');

% p=8,q=2 的相频

subplot(3,3,7);

plot(w1(1:512),angle(h1(1:512)));

xlim([0,w1(512)]);

title('p=8,q=2 相频');

xlabel('w');ylabel('角度');

% p=8,q=4 的相频

subplot(3,3,8);

plot(w2(1:512),angle(h2(1:512)));

xlim([0,w2(512)]);

title('p=8,q=4 相频');

xlabel('w');ylabel('角度');

% p=8,q=8 的相频

subplot(3,3,9);

plot(w3(1:512),angle(h3(1:512)));

xlim([0,w3(512)]);

title('p=8,q=8 相频');

xlabel('w');ylabel('角度');

%%

y1 = xa(n,8,8);

y2 = xa(n,13,8);

y3 = xa(n,14,8);

[h1,w1] = freqz(y1);

[h2,w2] = freqz(y2);

[h3,w3] = freqz(y3);

figure(2);

% p=8,q=8

subplot(3,3,1);

stem(n,y1);

title('p=8,q=8');

xlabel('n');ylabel('Xa');

% p=13,q=8

subplot(3,3,2);

stem(n,y2);

title('p=13,q=8');

xlabel('n');ylabel('Xa');

% p=14,q=8

subplot(3,3,3);

stem(n,y3);

title('p=14,q=8');

xlabel('n');ylabel('Xa');

% p=8,q=8 的幅频

subplot(3,3,4);

plot(w1(1:512),abs(h1(1:512)));

title('p=8,q=8 幅频');

xlim([0,w1(512)]);

xlabel('w');ylabel('幅度');

% p=13,q=8 的幅频

subplot(3,3,5);

plot(w2(1:512),abs(h2(1:512)));

title('p=13,q=8 幅频');

xlim([0,w2(512)]);

xlabel('w');ylabel('幅度');

% p=14,q=8 的幅频

subplot(3,3,6);

plot(w3(1:512),abs(h3(1:512)));

title('p=14,q=8 幅频');

xlim([0,w3(512)]);

xlabel('w');ylabel('幅度');

% p=8,q=8 的相频

subplot(3,3,7);

plot(w1(1:512),angle(h1(1:512)));

xlim([0,w1(512)]);

title('p=8,q=8 相频');

xlabel('w');ylabel('角度');

% p=13,q=8 的相频

subplot(3,3,8);

plot(w2(1:512),angle(h2(1:512)));

xlim([0,w2(512)]);

title('p=13,q=8 相频');

xlabel('w');ylabel('角度');

% p=14,q=8 的相频

subplot(3,3,9);

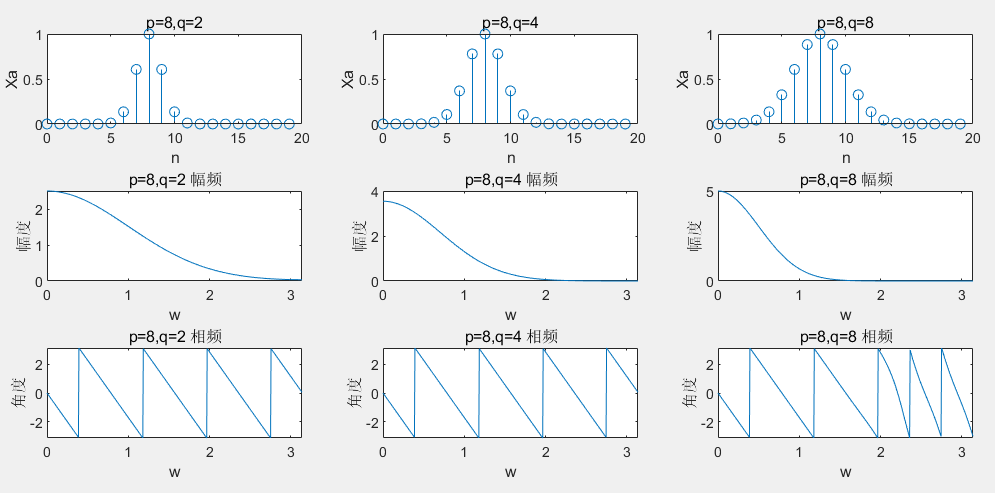
plot(w3(1:512),angle(h3(1:512)));

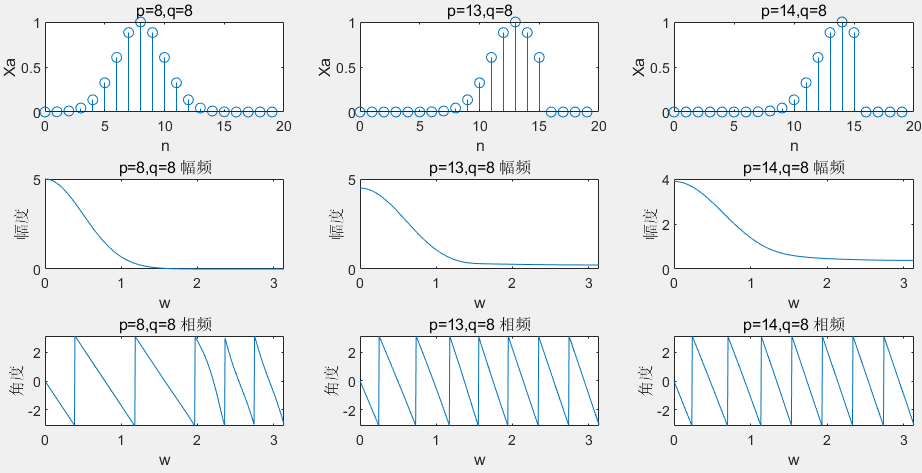
xlim([0,w3(512)]);

title('p=14,q=8 相频');

xlabel('w');ylabel('角度');

实验题目一结果：





实验题目三相关代码：

%%

N = 8;

n = 0:N-1;

yc = xc(n);

yd = xd(n);

figure(1);

subplot(2,2,1);

stem(n,yc);

title('xc');

xlabel('n');ylabel('xc');

subplot(2,2,3);

stem(n,abs(fft(yc)));

title('Xc FFT 分析');

xlabel('n');ylabel('Xc');

subplot(2,2,2);

stem(n,yd);

title('xd');

xlabel('n');ylabel('xd');

subplot(2,2,4);

stem(n,abs(fft(yd)));

title('Xd FFT 分析');

xlabel('n');ylabel('Xd');

%%

N = 32;

n = 0:N-1;

yc = xc(n);

yd = xd(n);

figure(2);

subplot(2,2,1);

stem(n,yc);

title('xc');

xlabel('n');ylabel('xc');

subplot(2,2,3);

stem(n,abs(fft(yc)));

title('Xc FFT 分析');

xlabel('n');ylabel('Xc');

subplot(2,2,2);

stem(n,yd);

title('xd');

xlabel('n');ylabel('xd');

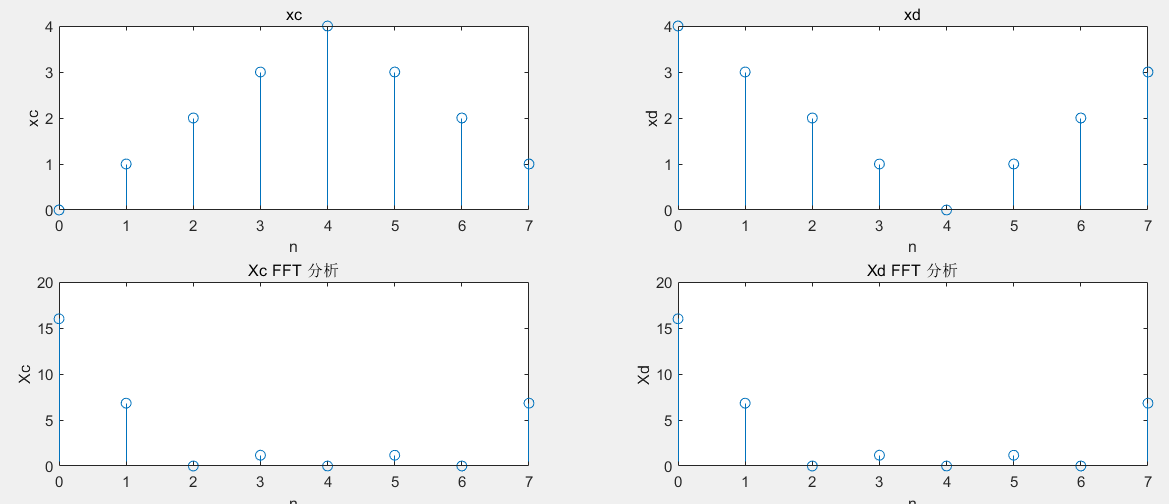
subplot(2,2,4);

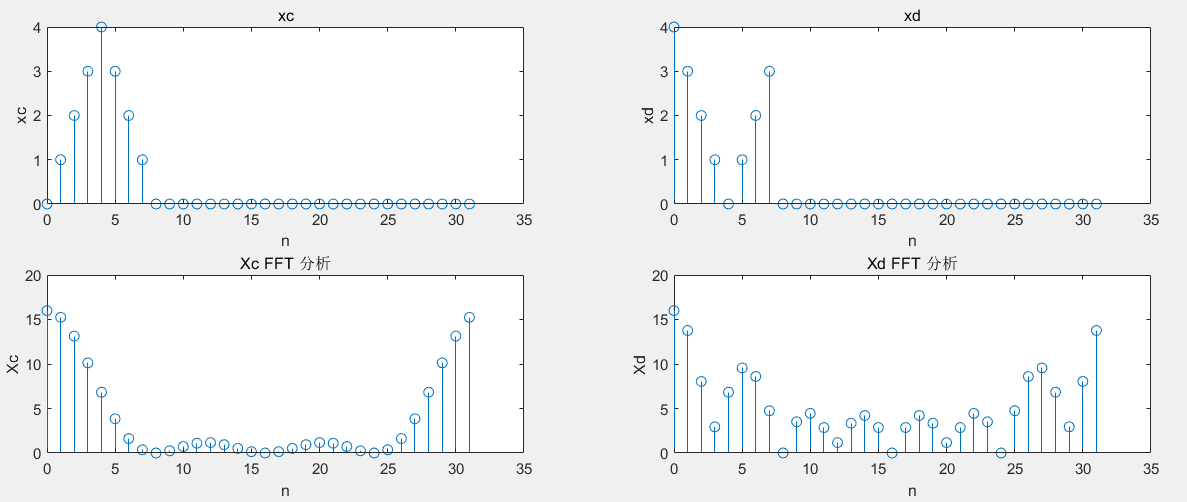
stem(n,abs(fft(yd)));

title('Xd FFT 分析');

xlabel('n');ylabel('Xd');

实验题目三结果：





实验题目五相关代码：

clear

N = 16;

n = 0:N-1;

% 线性卷积

xa = xa(n,8,2);

xb = xb(n,0.1,0.0625);

r1 = conv(xa,xb);

% 循环卷积

Xa = fft(xa,16);

Xb = fft(xb,16);

Xr = Xa .\* Xb;

r2 = ifft(Xr,16);

# 思考题回答

1. 不一样。反向三角波的低频多一些。
2. 可以。正弦信号，当时，正弦函数的周期L = 10，N = 16点的FFT满足，可以恢复出一个周期内的正弦信号。

**实验四 IIR数字滤波器的设计**