通信系统仿真
第5章后处理

何晨光
哈尔滨工业大学
电子与信息工程学院

第5章 后处理

4.1 基本图形方法
4.2 估计

第4章 带通的低通仿真模型

后处理器的作用是将仿真产生的数据处理成有用的形式。

后处理器通常是图形密集型的，因为视觉显示比数字列表更容易理解，而后者是仿真程序最常见的数据输出。

例如，不同系统的误比特率曲线比包含同样信息的数值表格能更快地传达信息。

第4章 带通的低通仿真模型

后处理程序可能引入(也可能不引入)相当的计算复杂度。有些后处理器只是简单地取得仿真产生的数据，对其进行适当的格式编排后，产生合适的图形输出。

比如，产生误比特概率或误符号概率P_e，作为E_b/N_0函数的曲线，仿真产生P_e值及其对应的E_b/N_0，并将它们以数据文件形式送入后处理器，后处理器对数据进行简单的格式编排，并画出所需的曲线。

作最少处理来产生图形输出的后处理器的其他例子还包括显示信号波形、眼图和散点图的后处理器。

第4章 带通的低通仿真模型

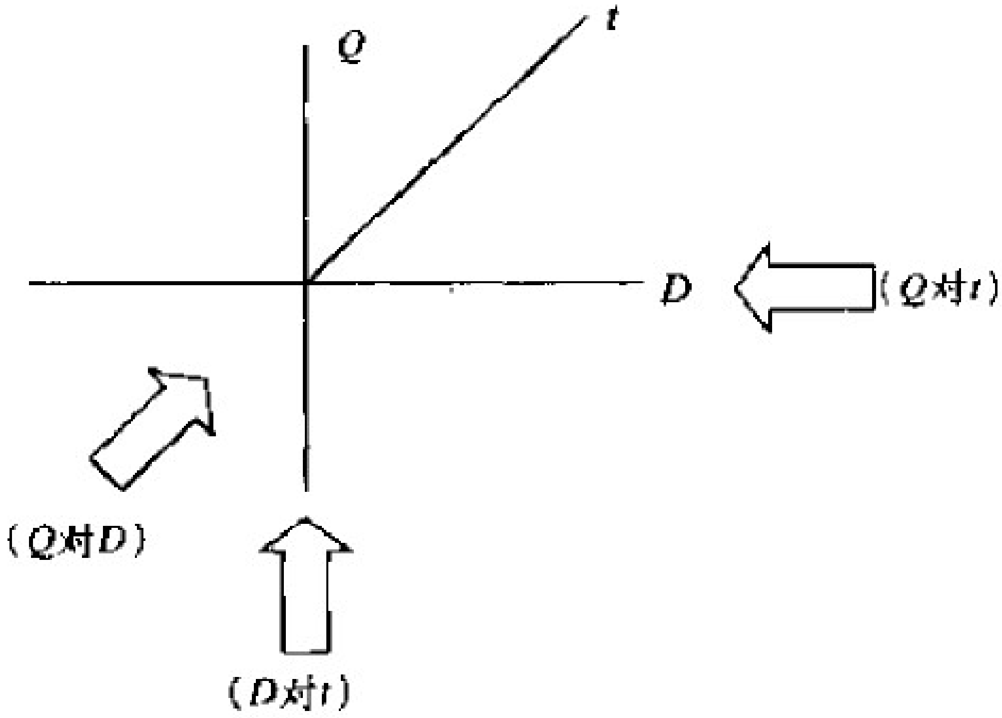
有些后处理子程序会涉及相当多的数据处理，它们中多数要涉及到某种估计。

一个简单的例子就是直方图的产生，它是对概率密度函数的估计。更复杂的例子有延迟、信噪比(SNR)和功率谱密度估计器。

4.1 基本图形方法

波形，眼图和散点图

如图是一个三维坐标系，形成了三个相交平面，每个平面包含两个坐标轴。



这些平面分别由D和t轴、Q和t轴、Q和D轴构成。

同相信道信号$x_d(t)$画在(D, t)平面上，正交信号$x_q(t)$画在(Q, t)平面上，就产生了以t为参数的三维信号。

把这个信号投影到给定子空间(D, t)、(Q, t)或(Q, D),就会产生$x_d(t)$, $x_q(t)$或散点图。

4.1 基本图形方法

眼图

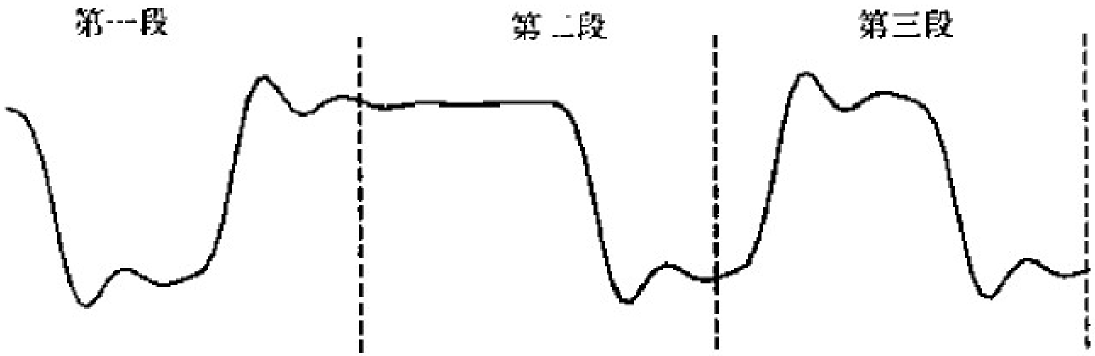
眼图给出系统性能的一种定性量度。

明晰张开的眼孔通常表明好的性能，而模糊的眼孔往往表明差的性能。

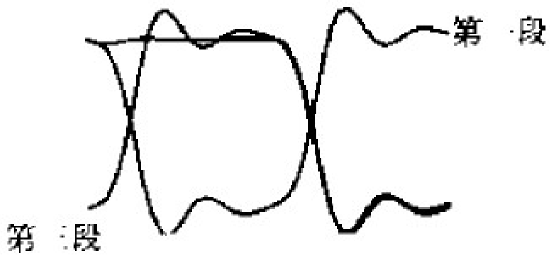
此外，眼孔的大小跟符号同步器所要求的精度有关。尽管眼图不能对系统性能提供定量的量度，但很难想像一个性能优异的系统会有不明确的眼图。。

4.1 基本图形方法

眼图



(a) 某波形的三段（60个样点）

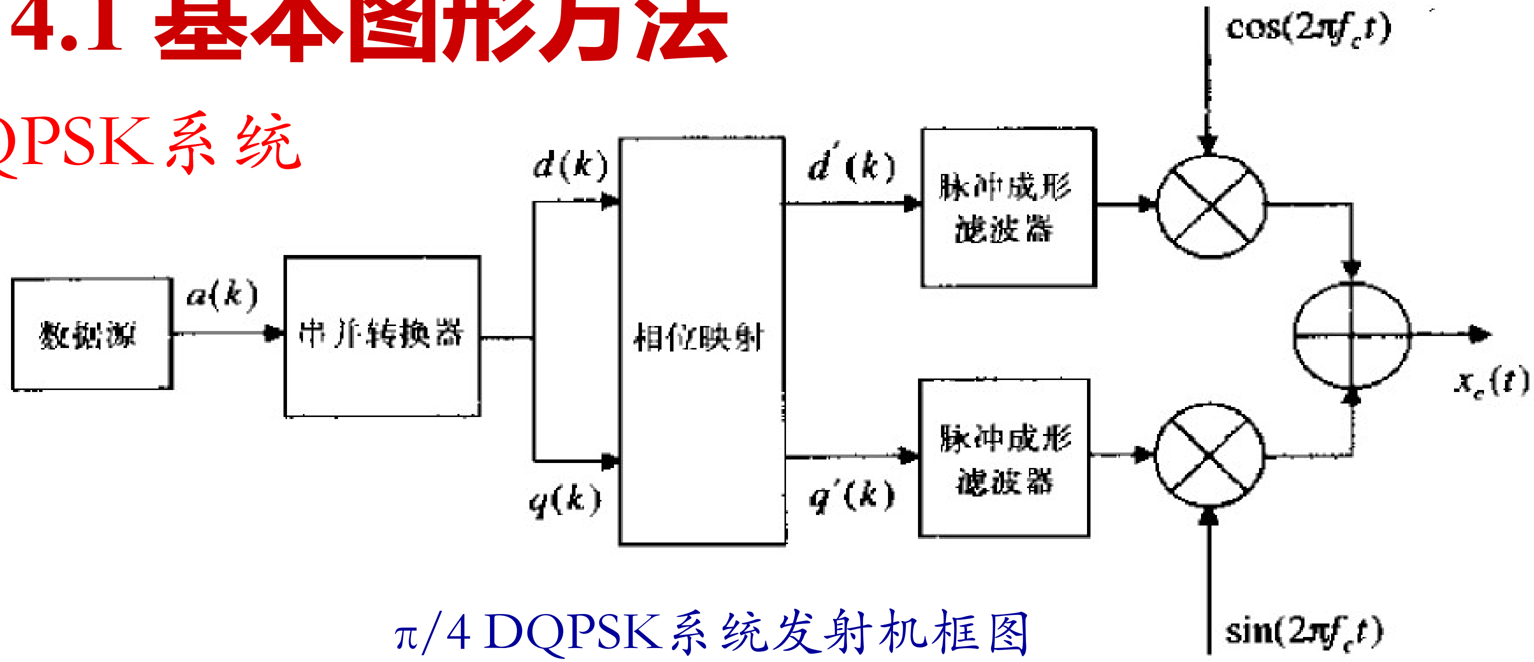


(b) 三段眼图

图显示了一个波形的三段，每段对应一个符号周期。假设用示波器显示波形，并且示波器在垂直虚线处触发，结果就是如图所示的三段眼图。

4.1 基本图形方法

π/4 DQPSK系统



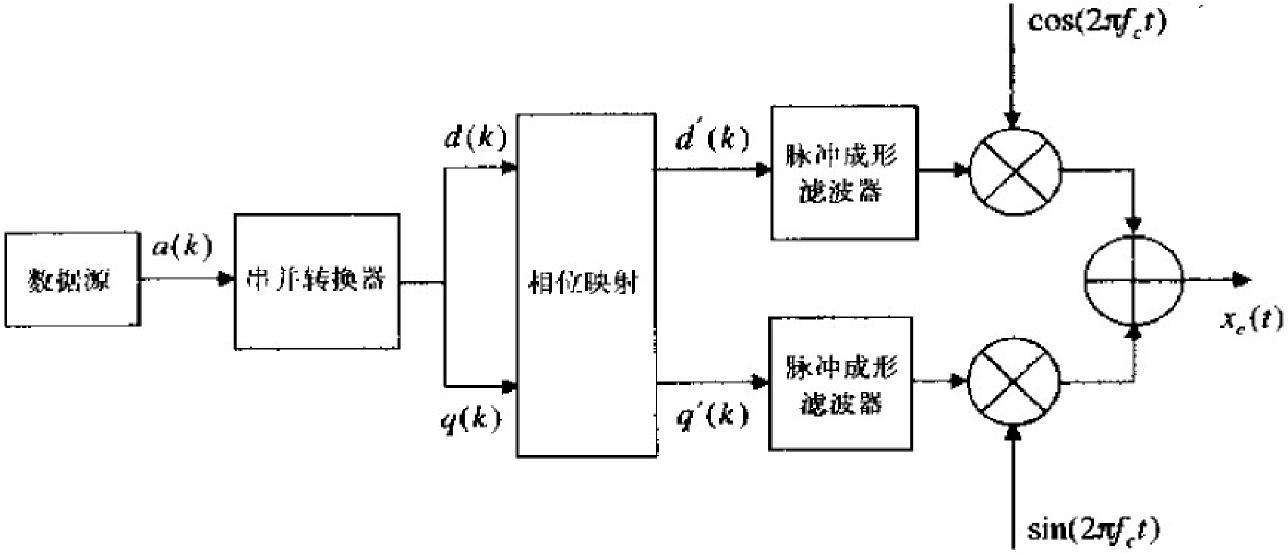
π/4 DQPSK系统发射机框图

假设数据信源的输出是具有如下形式的序列a:

$$a(1)a(2)a(3)a(4)\cdots a(\mathrm{k})\cdots$$

串并转换器交替将(奇序数的)符号分配给同相信道，将其他(偶序数的)符号分配给正交信道，于是有

$$a(1)a(2)a(3)a(4)\cdots a(\mathrm{k})\cdots = d(1)\,\mathrm{q}(1)d(2)\,\mathrm{q}(2)\cdots d(\frac{k+1}{2})\,\mathrm{q}(\frac{k}{2}+1)\cdots$$

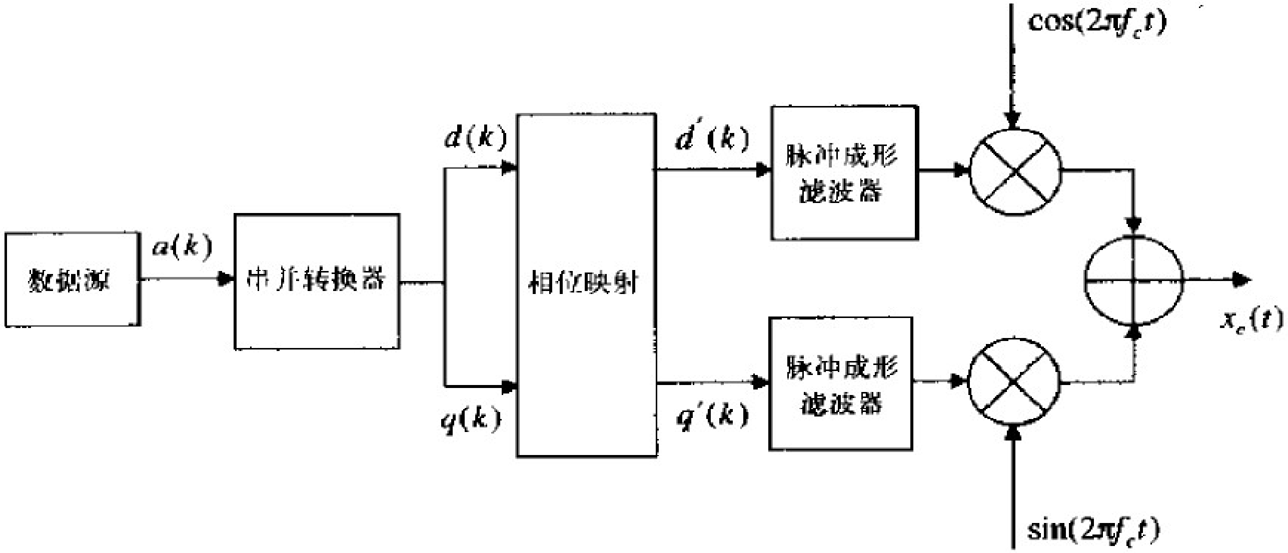


发射信号：

$$x_C(t) = A\cos[2\pi f_C t + \theta(k)] \quad (k-1)T_S < t < kT_S$$

其中，Ts是符号周期，发射信号的相位偏移由d(k)和q(k)的值以及前一符号周期的相位偏移$\theta(k-1)$决定。对前一符号周期的这种依赖关系就使得π/4 DQPSK成为一种差分调制方法。$\theta(k)$与 $\theta(k-1)$之间的关系如下

$$\theta(k) = \theta(k-1) + \phi(k)$$

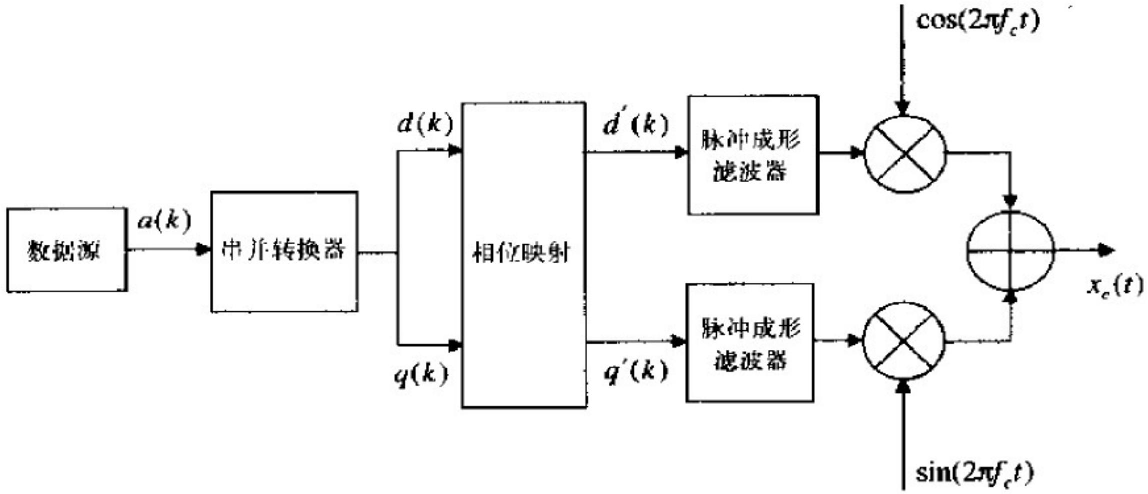


相位映射器产生出所需的发送相位。相位映射器利用d(k)、q(k)和$\theta(k-1)$产生新的d'(k)和q' (k)值，使发射信号有合适的相位，再通过适当的脉冲成型，同相和正交信道信号转换成传输频率fc。

d(k)、q(k)和 $\theta(k-1)$ 的关系如下

信息符号$d(k)$和$q(k)$	差分相移$\phi(k)$
1 1	$\pi/4$
0 1	$3\pi/4$
0 0	$-3\pi/4$
1 0	$-\pi/4$

信息符号$d(k)$和$q(k)$	差分相移$\phi(k)$
1 1	$\pi/4$
0 1	$3\pi/4$
0 0	$-3\pi/4$
1 0	$-\pi/4$



$$\theta(k) = \theta(k-1) + \phi(k)$$

假设数据信源输出的二进制序列为：0010110111……

且初始相位定义为$\theta(0)=0$，因为前两个数据符号是00，根据上表可知

$$\phi(1) = -3\pi/4$$

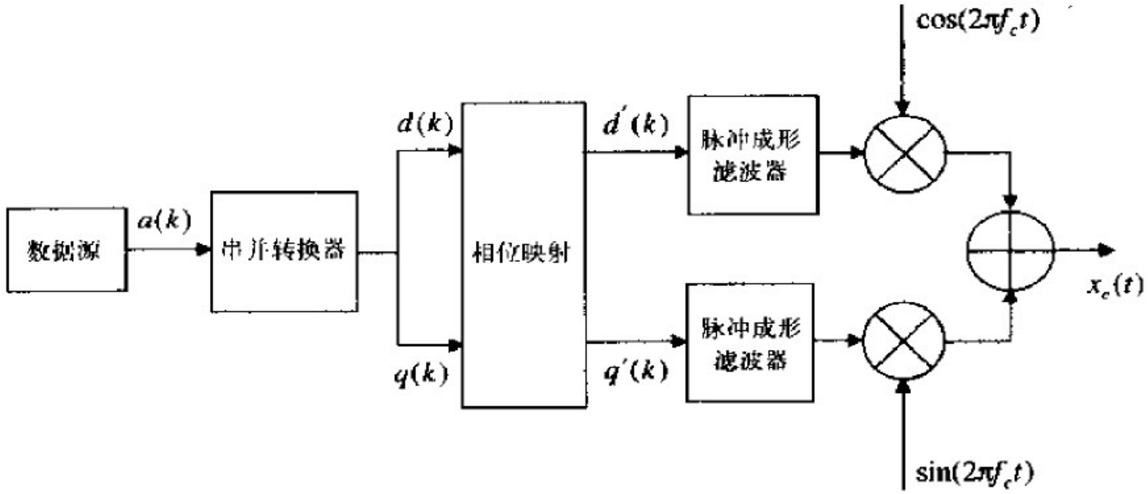
所以：

$$\theta(1) = \theta(0) + \phi(1) = 0 - \frac{3\pi}{4} = -\frac{3\pi}{4}$$

接下来两个数据是10，由表中的映射关系可得：

$$\phi(2) = -\pi/4$$ 得： $$\theta(2) = -\frac{3\pi}{4} - \frac{\pi}{4} = -\pi$$

信息符号$d(k)$和$q(k)$	差分相移$\phi(k)$
1 1	$\pi/4$
0 1	$3\pi/4$
0 0	$-3\pi/4$
1 0	$-\pi/4$



$$\theta(k)=\theta(k-1)+\phi(k)$$

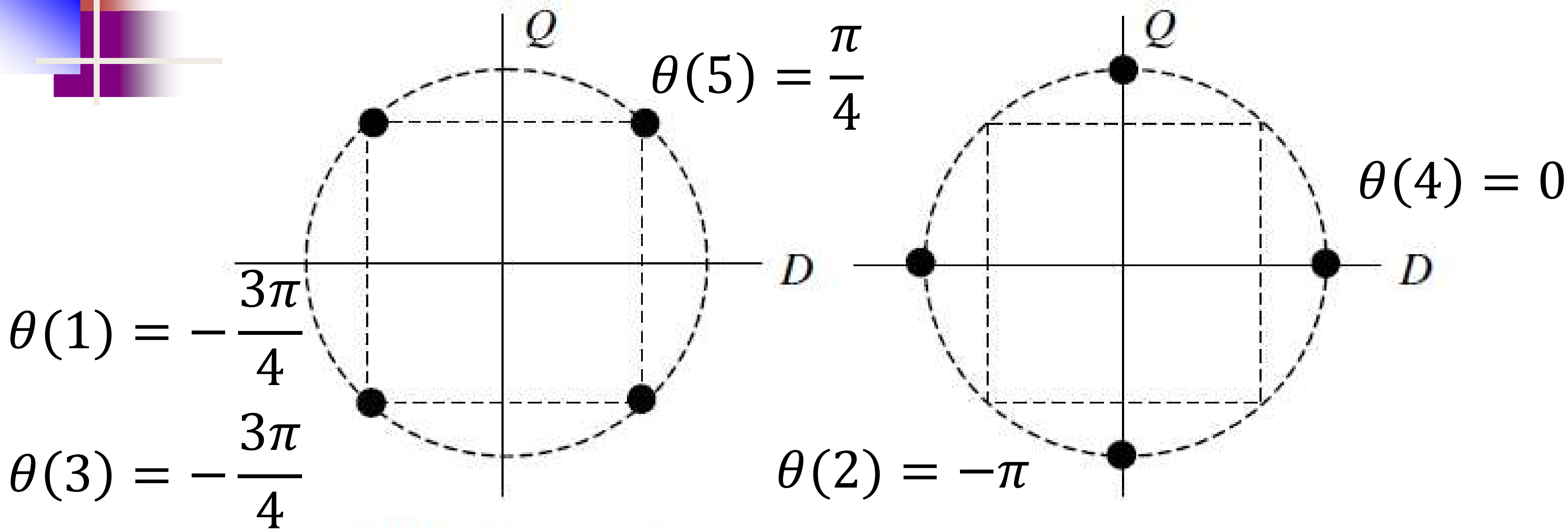
数据是00： $\phi(1)=-3\pi/4$ 得： $\theta(1)=0-\frac{3\pi}{4}=-\frac{3\pi}{4}$

数据是10： $\phi(2)=-\pi/4$ 得： $\theta(2)=-\frac{3\pi}{4}-\frac{\pi}{4}=-\pi$

数据是11： $\phi(3)=\pi/4$ 得： $\theta(3)=-\pi+\frac{\pi}{4}=-\frac{3\pi}{4}$

数据是01： $\phi(4)=3\pi/4$ 得： $\theta(4)=-\frac{3\pi}{4}+\frac{3\pi}{4}=0$

数据是11： $\phi(5)=\pi/4$ 得： $\theta(5)=0+\frac{\pi}{4}=\frac{\pi}{4}$



(a) QPSK: Phase set 1　　(b) QPSK: Phase set 2

π/4 DQPSK通过交替地从两个QPSK号星座发射信号点来实现，这两种QPSK信号星座的相位差为π/4 。

虽然上面采用了一个具体的数据序列来进行演示，但却是普遍适用的，因为差分相位只取± π/4或± 3π/4四个值。

4.1 基本图形方法

例1 对π/4 DQPSK系统发射机进行仿真，对一些重要信号进行后处理，并产生图形输出。

在MATLAB进行程序执行后，会出现一个菜单，用户可通过达个菜单选择以下七个选项中的一个(产生一个图形后，击空格键会显示菜单以便作另一个选择):

1.滤波前的 DQPSK号星座图
2.滤波前的 DQPSK眼图
3.滤波后的 DQPSK信号星座图
4,滤波后的 DQPSK眼图
5.滤波前的同相和正交信号
6.滤波后的同相和正交信号
7.退出程序(返同MATLAB命令提示符)

4.1 基本图形方法 （代码见c501.m）

```
m = 200;     bits = 2*m;                          % number of symbols and bits
sps = 10;                                         % samples per symbol
iphase = 0;                                       % initial phase
order = 5;                                        % filter order
bw = 0.2;                                         % normalized filter bandwidth
%
% initialize vectors
%
data = zeros(1,bits); d = zeros(1,m); q = zeros(1,m);
dd = zeros(1,m); qq = zeros(1,m); theta = zeros(1,m);
thetaout = zeros(1,sps*m);
%
% set direct and quadrature bit streams
%
data = round(rand(1,bits));
dd = data(1:2:bits-1);
qq = data(2:2:bits);
```

串并转换器交替将(奇序数的)符号分配给同相信道，将其他(偶序数的)符号分配给正交信道

$$a(1)a(2)a(3)a(4)\cdots a(\mathrm{k})\cdots = d(1)\,\mathrm{q}(1)d(2)\,\mathrm{q}(2)\cdots d(\frac{k+1}{2})\,\mathrm{q}(\frac{k}{2}+1)\cdots$$

4.1 基本图形方法

```
theta(1) = iphase;                    % set initial phase
thetaout(1:sps) = theta(1)*ones(1,sps);
for k=2:m
    if dd(k) == 1
        phi_k = (2*qq(k)-1)*pi/4;
    else
        phi_k = (2*qq(k)-1)*3*pi/4;
    end
    theta(k) = phi_k + theta(k-1);
    for i=1:sps
        j = (k-1)*sps+i;
        thetaout(j) = theta(k);
    end
end
d = cos(thetaout);
q = sin(thetaout);
[b,a] = butter(order,bw);
df = filter(b,a,d);
qf = filter(b,a,q);
```

信息符号$d(k)$和$q(k)$	差分相移$\phi(k)$
1 1	$\pi/4$
0 1	$3\pi/4$
0 0	$-3\pi/4$
1 0	$-\pi/4$

$$\theta(k) = \theta(k-1) + \phi(k)$$

对每个符号进行采样

4.1 基本图形方法

```
while kk == 0                              % test exit counter
k = menu('pi/4 QPSK Plot Options',...
        'Unfiltered pi/4 QPSK Signal Constellation',...
        'Unfiltered pi/4 QPSK Eye Diagram',...
        'Filtered pi/4 QPSK Signal Constellation',...
        'Filtered pi/4 OQPSK Eye Diagram',...
        'Unfiltered Direct and Quadrature Signals',...
        'Filtered Direct and Quadrature Signals',...
        'Exit Program');
        if k == 1
                sigcon(d,q)                % plot unfiltered signal con.
                pause
        elseif k ==2
                dqeye(d,q,4*sps)           % plot unfiltered eye diagram
                pause
        elseif k == 3
                sigcon(df,qf)              % plot filtered signal con.
                pause
        elseif k == 4
                dqeye(df,qf,4*sps)         % plot filtered eye diagram
                pause
```

4.1 基本图形方法

```
    elseif k == 5
        numbsym = 10;                  % number of symbols plotted
        dt = d(1:numbsym*sps);         % truncate d vector
        qt = q(1:numbsym*sps);         % truncate q vector
        dqplot(dt,qt)                  % plot truncated d and q signals
        pause
    elseif k == 6
        numbsym = 10;                  % number of symbols to be plotted
        dft=df(1:numbsym*sps);         % truncate df to desired value
        qft=qf(1:numbsym*sps);         % truncate qf to desired value
        dqplot(dft,qft)                % plot truncated signals
        pause
    elseif k == 7
        kk = 1;                        % set exit counter to exit value
    end
end
```

4.1 基本图形方法

1.滤波前的 DQPSK号星座图
2.滤波前的 DQPSK眼图
3.滤波后的 DQPSK信号星座图
4,滤波后的 DQPSK眼图
5.滤波前的同相和正交信号
6.滤波后的同相和正交信号
7.退出程序(返同MATLAB命令提示符)

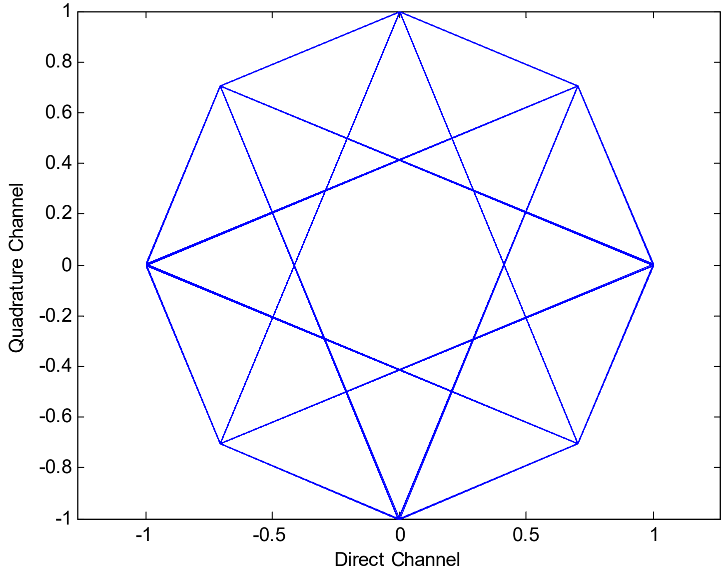
4.1 基本图形方法

```
if k == 1
        sigcon(d,q)                    % plot unfiltered signal con.
        pause
```

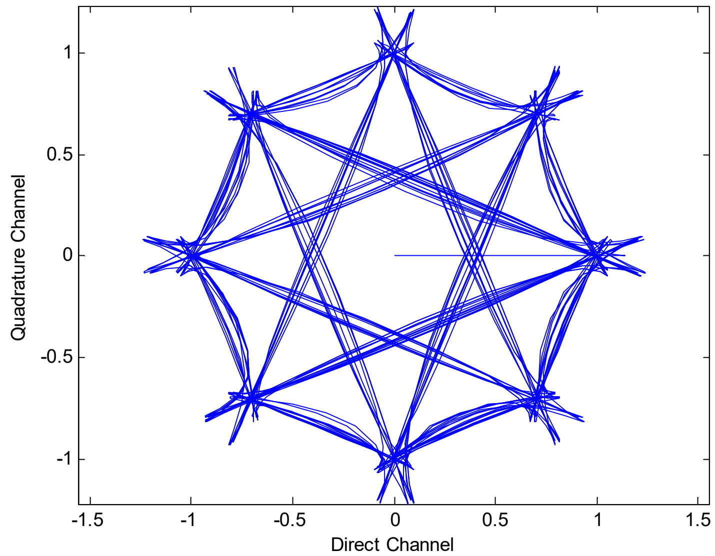
星座图

```
function []=sigcon(x,y)
plot(x,y)
axis('square')
axis('equal')
xlabel('Direct Channel')
ylabel('Quadrature Channel')
```

4.1 基本图形方法



1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
Quadrature Channel
-1
-0.5
0
0.5
1
Direct Channel



1
0.5
0
-0.5
-1
Quadrature Channel
-1.5
-1
-0.5
0
0.5
1
1.5
Direct Channel

4.1 基本图形方法

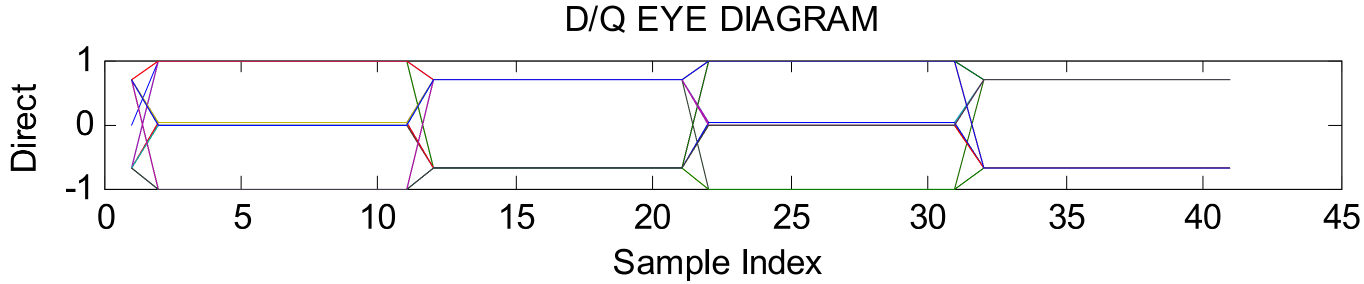
```
elseif k ==2
        dqeye(d, q, 4*sps)              % plot unfiltered eye diagram
        pause
```

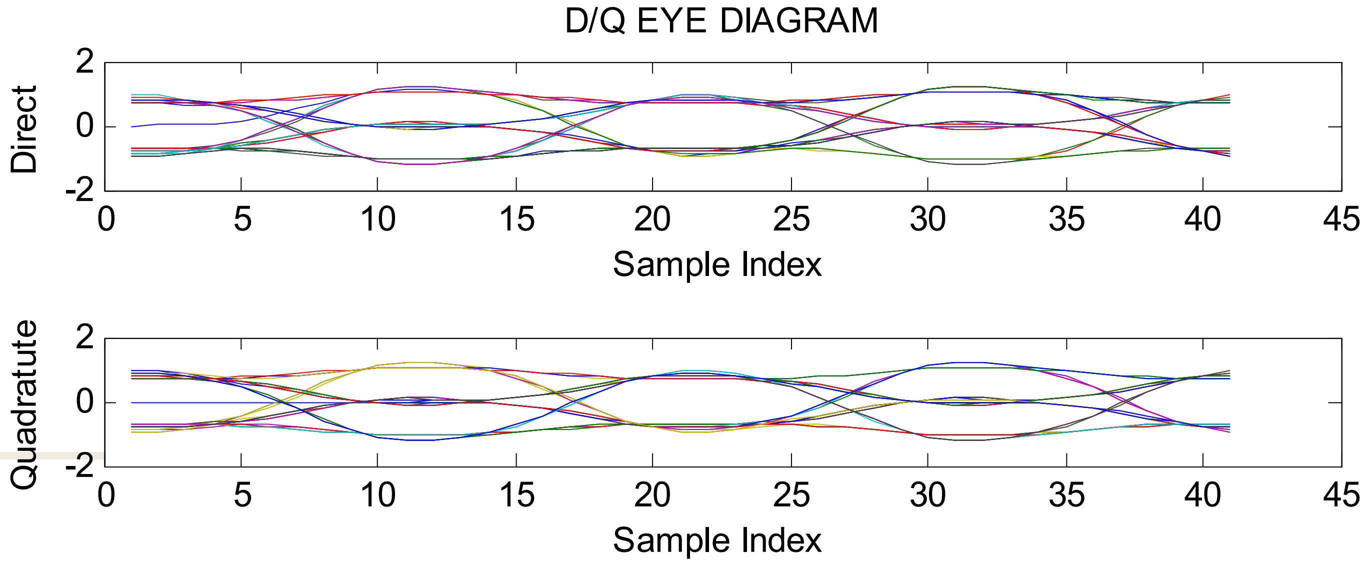
眼图

```
function [] = dqeye(xd,xq,m)
lx = length(xd);                      % samples in data segment
kcol = floor(lx/m);                   % number of columns
xda = [0,xd]; xqa = [0,xq];           % append zeros
for j = 1:kcol                        % column index
    for i = 1:(m+1)                   % row index
        kk = (j-1)*m+i;               % sample index
        y1(i,j) = xda(kk);
        y2(i,j) = xqa(kk);
    end
end
```

```
subplot(211)                          % direct channel
plot(y1);
title('D/Q EYE DIAGRAM');
xlabel('Sample Index');
ylabel('Direct');
subplot(212)                          % quadrature channel
plot(y2);
xlabel('Sample Index');
ylabel('Quadratute');
subplot(111)                          % restore
```

4.1 基本图形方法


D/Q EYE DIAGRAM
Direct
Quadratute
Sample Index
Sample Index


D/Q EYE DIAGRAM
Direct
Quadratute
Sample Index
Sample Index

4.1 基本图形方法

```
elseif k == 5
        numbsym = 10;                % number of symbols plotted
        dt = d(1:numbsym*sps);       % truncate d vector
        qt = q(1:numbsym*sps);       % truncate q vector
        dqplot(dt,qt)                % plot truncated d and q signals
        pause
```

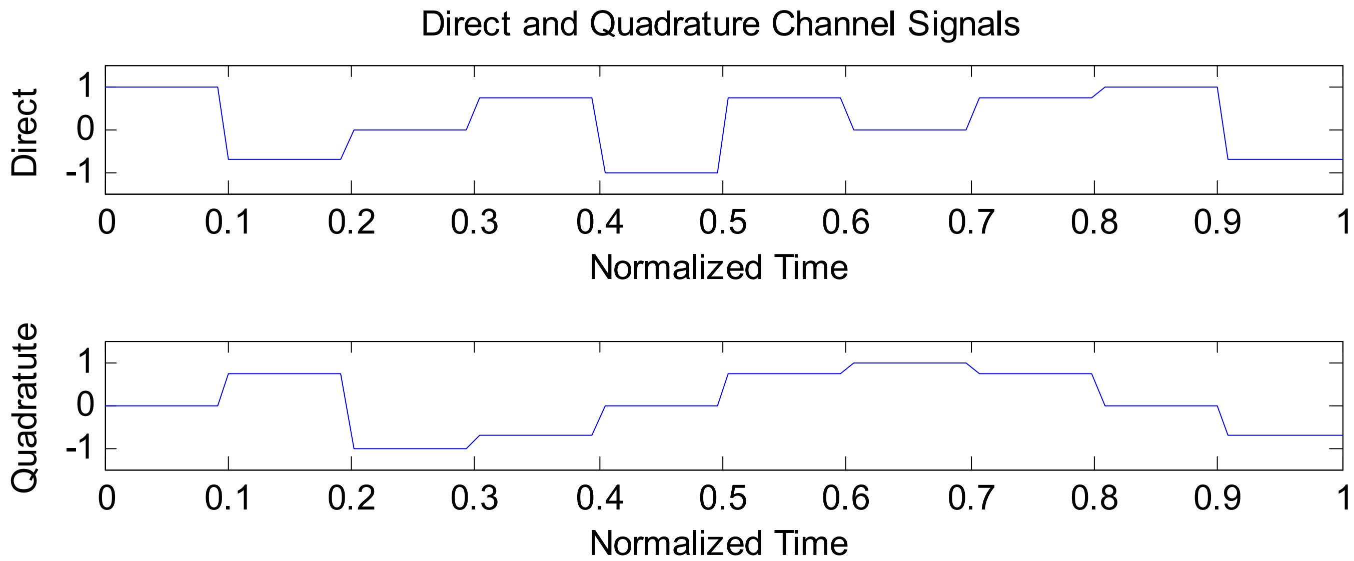
波形图

```
function [] = dqplot(xd,xq)
lx = length(xd);
t = 0:lx-1;
nt = t/(lx-1);
nxd = xd(1,1:lx);
nxq = xq(1,1:lx);
```

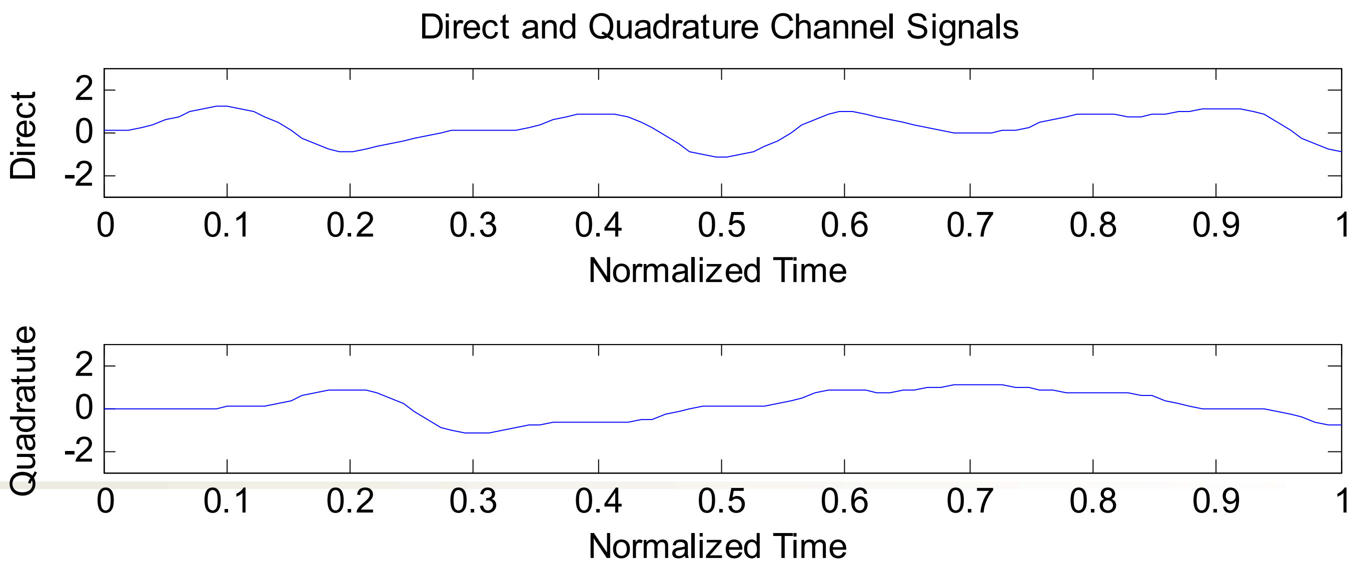
归一化操作

```
subplot(211)
plot(nt,nxd);
a = axis;
axis([a(1) a(2) 1.5*a(3) 1.5*a(4)]);
title('Direct and Quadrature Channel Signals');
xlabel('Normalized Time');
ylabel('Direct');
subplot(212)
plot(nt,nxq);
a = axis;
axis([a(1) a(2) 1.5*a(3) 1.5*a(4)]);
xlabel('Normalized Time');
ylabel('Quadratute');
subplot(111)
```

4.1 基本图形方法



Direct and Quadrature Channel Signals
Direct
1
0
-1
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalized Time
Quadratute
1
0
-1
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalized Time



Direct and Quadrature Channel Signals
Direct
2
0
-2
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalized Time
Quadratute
2
0
-2
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Normalized Time

4.2 估计 直方图

如果可以获得随机过程的一组样本（就像在仿真环境下那样），由这组样本得出的直方图经常可用于估计内在的概率密度函数(pdf)。

直方图将所有N个样本的数据分组成B个直方，假定每个直方的宽度为W，其中心记为b_i。那么给定的样本x[n]落在第i个直方，可表示为

$$b_i - \frac{W}{2} < x[n] \le b_i + \frac{W}{2}$$

我们感兴趣的一个量是N_i，它表示落在第i个直方中的样本数目。显然有

$$N = \sum_{i=1}^{B} N_i$$

4.2 估计直方图

采用记号Count{N: R}来表示全部N个样本中进入由R所定义的直方中的样本数目。于是有

$$N_i = \text{Count}\left\{N : b_i - \frac{W}{2} < x[n] \leq b_i + \frac{W}{2}\right\}$$

然后画出条状图，每条的高度和N_i成正比，中心在b_i。为了用作pdf估计器，我们对直方图进行缩放使其总面积为1，将N_i除以NW即可实现这一点。于是，每条的高度为N_i/NW。代表第i个直方的条的面积A_i，等于条的高度乘以宽度W。因此，

$$A_i = \left(\frac{N_i}{NW}\right) W = \frac{N_i}{N}$$

注意到A_i代表第i个直方的相对频率。按要求，如果直方图用于表征概率密度函数，总面积为

$$\sum_{i=1}^{B} \frac{N_i}{N} = 1$$

4.2 估计 直方图

每个直方是用一个常数来表示在有限宽度W上的pdf。但是所有的样本个数N和直方的宽度W直接影响着pdf估计的一致性和无偏性。

例2 假设我们产生了一个具有零均值和单位方差的高斯随机变量的N个样本。通过构造有B个直方的直方图来估计pdf。在这个例子里，我们希望考察不同N和B值的影响。

4.2 估计 直方图

```
subplot(2,2,1)
x = randn(1,100); hist(x,20)
ylabel('N_i'); xlabel('(a)')
subplot(2,2,2)
x = randn(1,100); hist(x,5)
ylabel('N_i'); xlabel('(b)')
subplot(2,2,3)
x = randn(1,1000); hist(x,50)
ylabel('N_i'); xlabel('(c)')
subplot(2,2,4)
x = randn(1,100000); hist(x,50)
ylabel('N_i'); xlabel('(d)')
% End of script file.
```

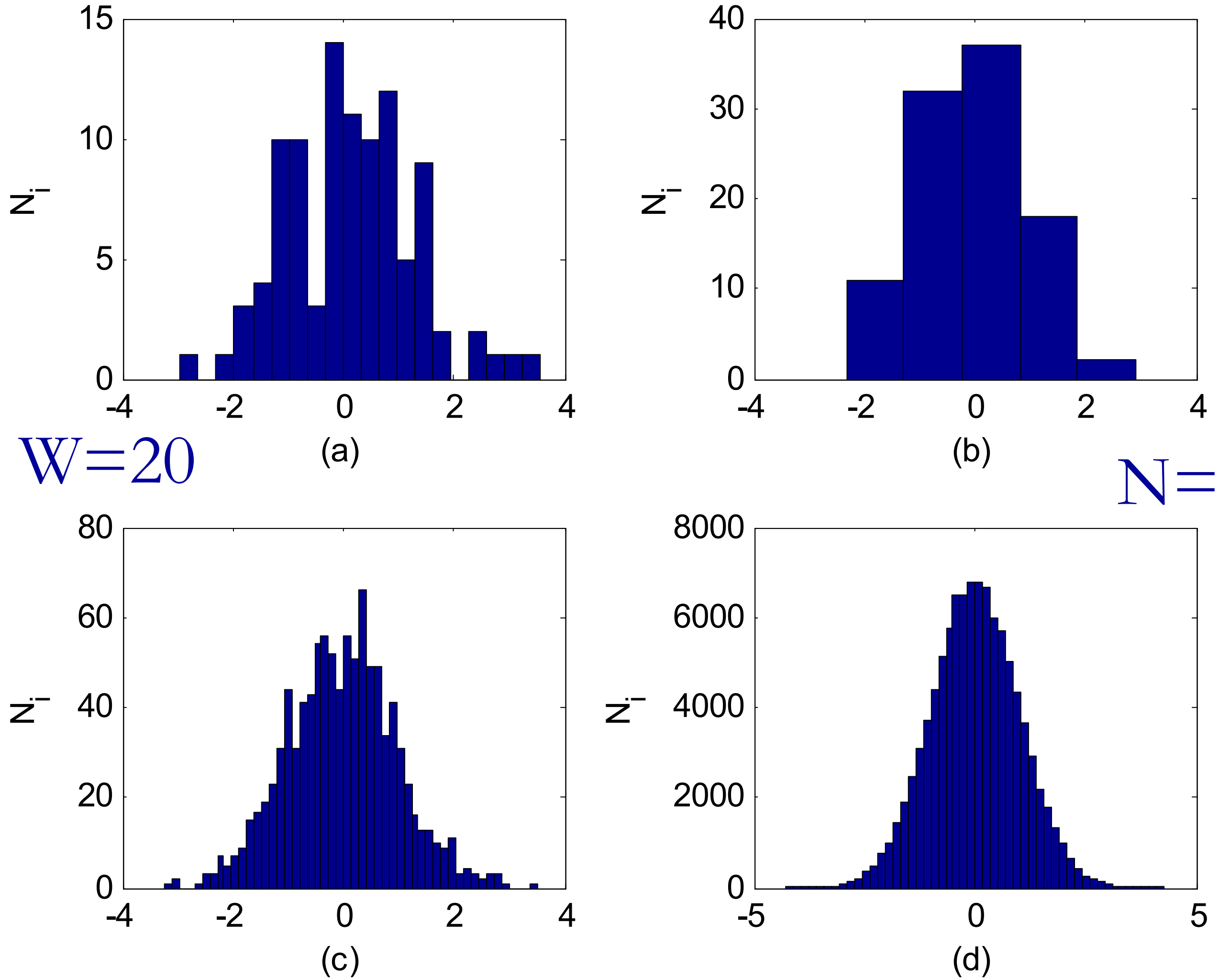
N=100, W=20

N=100, W=5

N=1000, W=50

N=100000, W=50

4.2 估计 直方图



N=100, W=20

N=100, W=5

N=1000, W=50

N=100000, W=50

4.2 估计 功率谱密度的估计

由于随机信号不满足绝对可积的条件，因此是不存在频谱的。但工程中的随机信号一般为功率型信号，可以用功率谱密度来描述它的频域特性。功率谱估计就是利用给定的样本数据来估计一个平稳随机信号的功率谱密度。

功率谱密度可以分为经典功率谱估计和现代功率谱估计。

经典功率谱估计包括直接法和间接法

4.2 估计 功率谱密度的估计

直接法:

随机序列X(n)的某个样本函数x(n)的观测数据长度是有限的，若序列长度为N,则可以认为是一个能量有限的序列。若x(n)的离散时间傅里叶变化 $X(e^{j\omega})$ 存在，则

$$\hat{S}_X(e^{j\omega}) = \frac{1}{N}\left|\sum_{n=0}^{N-1} x(n)e^{-jwn}\right|^2 = \frac{1}{N}|X(e^{j\omega})|^2$$

为序列X(n)的功率谱估计。由于 $X(e^{j\omega})$ 是周期谱，所估计的功率谱也是周期谱，所以这种方法成为周期图法。

周期图法是比较简单的一种功率谱估计方法，如果直接利用数据样本做离散傅里叶变换，可得到的 $X(e^{j\omega})$ 离散值X(k)，稍加预算即可得到离散功率谱

$$\hat{S}_X(K) = \frac{1}{N}|X(K)|^2$$

4.2 估计 功率谱密度的估计

例3：已知随机信号

$$X(n) = \cos(2\pi f_1 t + \varphi_1) + 3\cos(2\pi f_2 t + \varphi_2) + N(n)$$

其中 f_1=30Hz, f_2=100Hz, φ_1 和 φ_2 为在[0,2pi]内均匀分布的随机变量，N(n)是数学期望为零、方差为1的高斯白噪声。

仿真X(n)的一个样本序列，用周期图法估计功率谱。

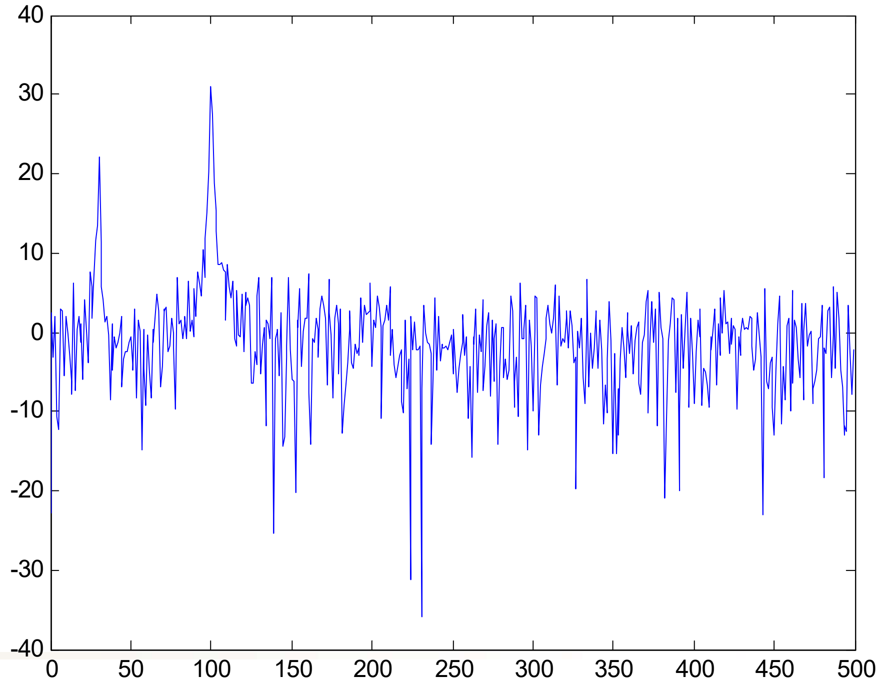
解：由于两个随机相位余弦信号和N(n)均为平稳、各态历经过程，因此可以利用一个样本序列来估计功率谱。

4.2 估计 功率谱密度的估计

（代码见c502_Sx.m）

```
N=1024;
fs=1000;
t=(0:N-1)/fs;
fai=random('unif',0,1,1,2)*2*pi;
xn=cos(2*pi*30*t+fai(1))+3*cos(2*pi*100*t+fai(2))+randn(1,N);
Sx=abs(fft(xn)).^2/N;
f=(0:N/2-1)*fs/N
plot(f,10*log10(Sx(1:N/2)));
```

功率谱在30Hz和100Hz处有两个谱峰，分辨率(区分两个临近频率分量的能力)与观测数据的长度有关，当观测数据长度增加时，谱峰更为尖锐。



4.2 估计 功率谱密度的估计

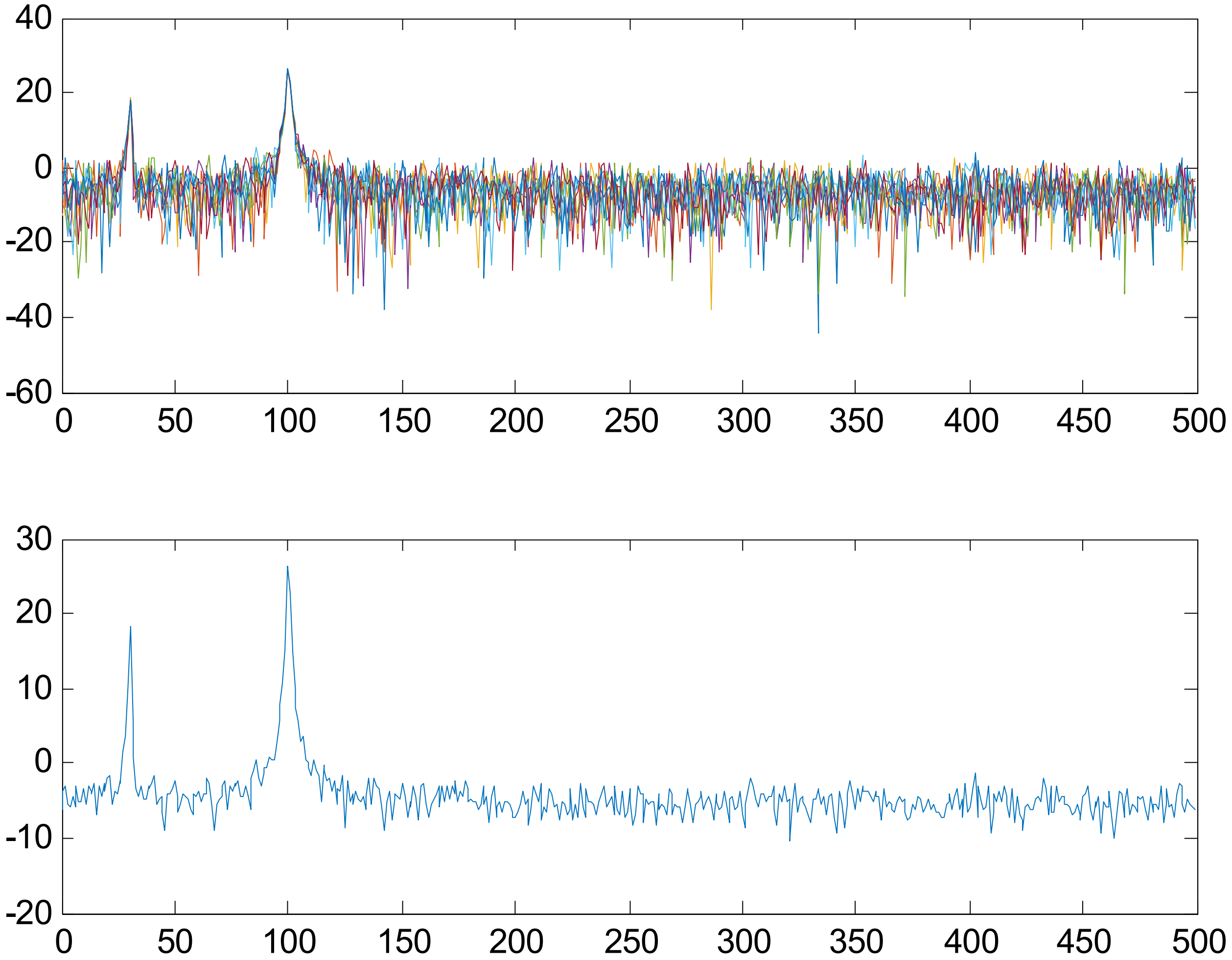
例3：已知随机信号

$$X(n) = \cos(2\pi f_1 t + \varphi_1) + 3\cos(2\pi f_2 t + \varphi_2) + N(n)$$

其中f_1=30Hz, f_2=100Hz, φ_1和φ_2为在[0,2pi]内均匀分布的随机变量，N(n)是数学期望为零、方差为1的高斯白噪声。

仿真用M个样本来估计序列X(n)的功率谱。

4.2 估计 功率谱密度的估计



4.2 估计 功率谱密度的估计

间接法:

由维纳辛钦定理可知，功率谱和相关函数是已对傅里叶变换对，因此，先用序列x(n)估计出其自相关函数R(m)，然后对R(m)进行傅里叶变换，就可以得到x(n)的功率谱估计值

例4：已知随机信号

$$X(n) = \cos(2\pi f_1 t + \varphi_1) + 3\cos(2\pi f_2 t + \varphi_2) + N(n)$$

其中f_1=30Hz, f_2=100Hz, φ_1 和 φ_2 为在[0,2pi]内均匀分布的随机变量，N(n)是数学期望为零、方差为1的高斯白噪声。

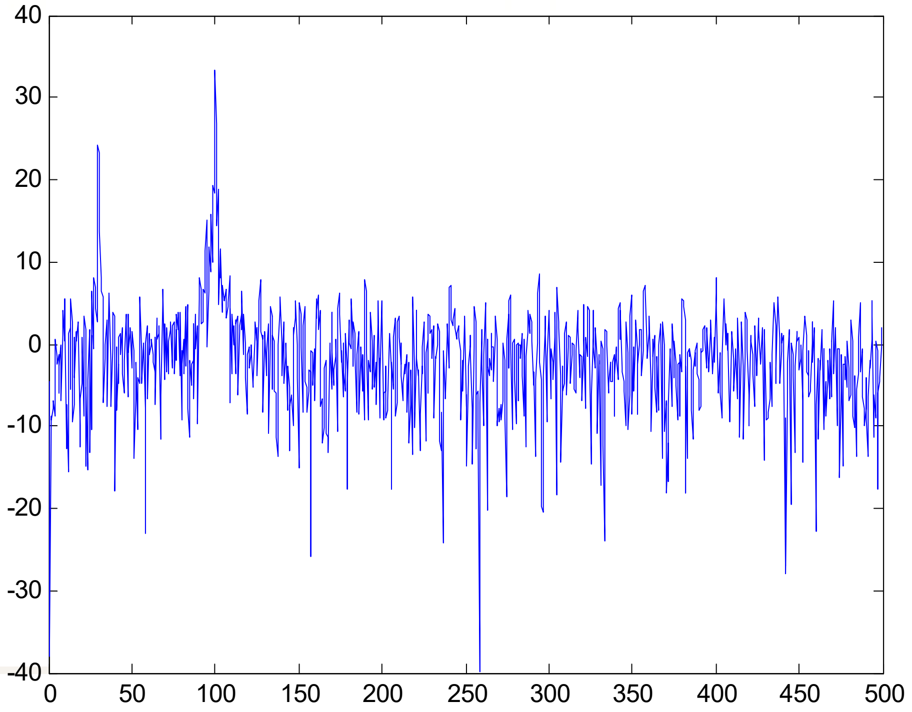
用相关函数法估计随机信号的功率谱密度。

解：先用xcorr函数求自相关，再用fft函数求功率谱

4.2 估计 功率谱密度的估计

（代码见c503_Rxx.m）

```
N=1024;
fs=1000;
t=(0:N-1)/fs;
fai=random('unif',0,1,1,2)*2*pi;
xn=cos(2*pi*30*t+fai(1))+3*cos(2*pi*100*t+fai(2))+randn(1,N);
Rxx=xcorr(xn,'biased')
Sx=abs(fft(Rxx));
f=(0:N-1)*fs/N/2
plot(f,10*log10(Sx(1:N)));
```

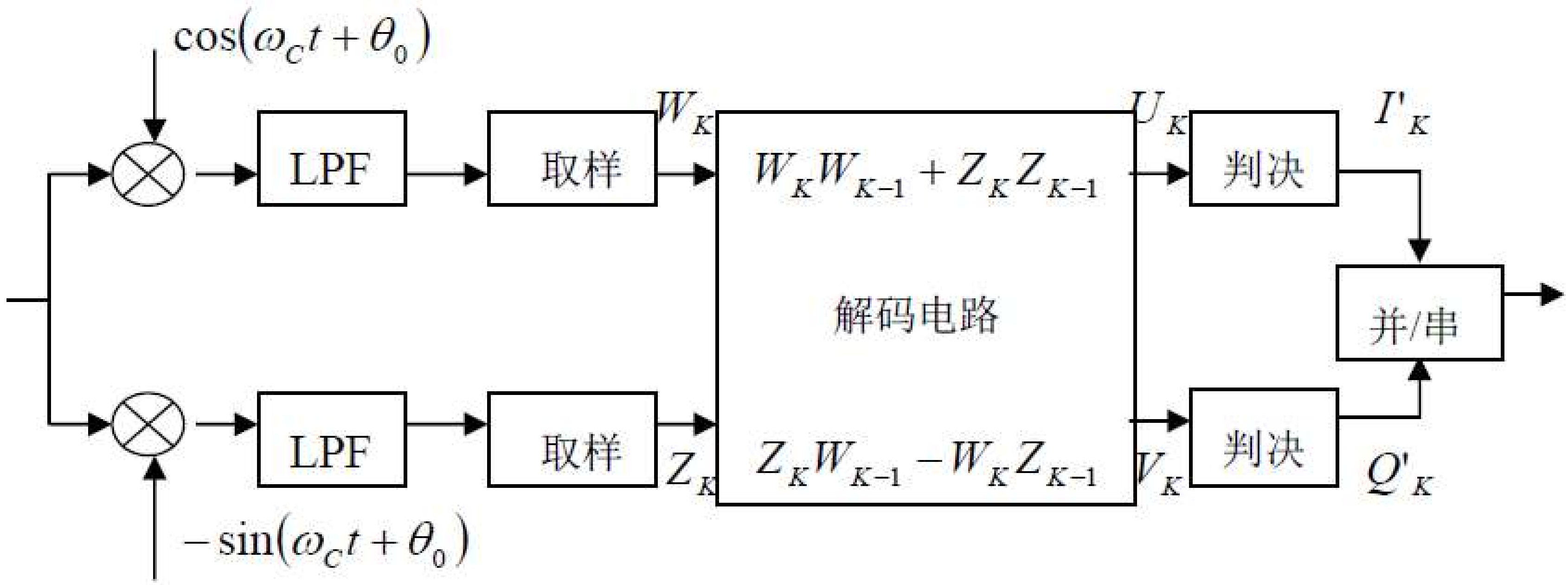


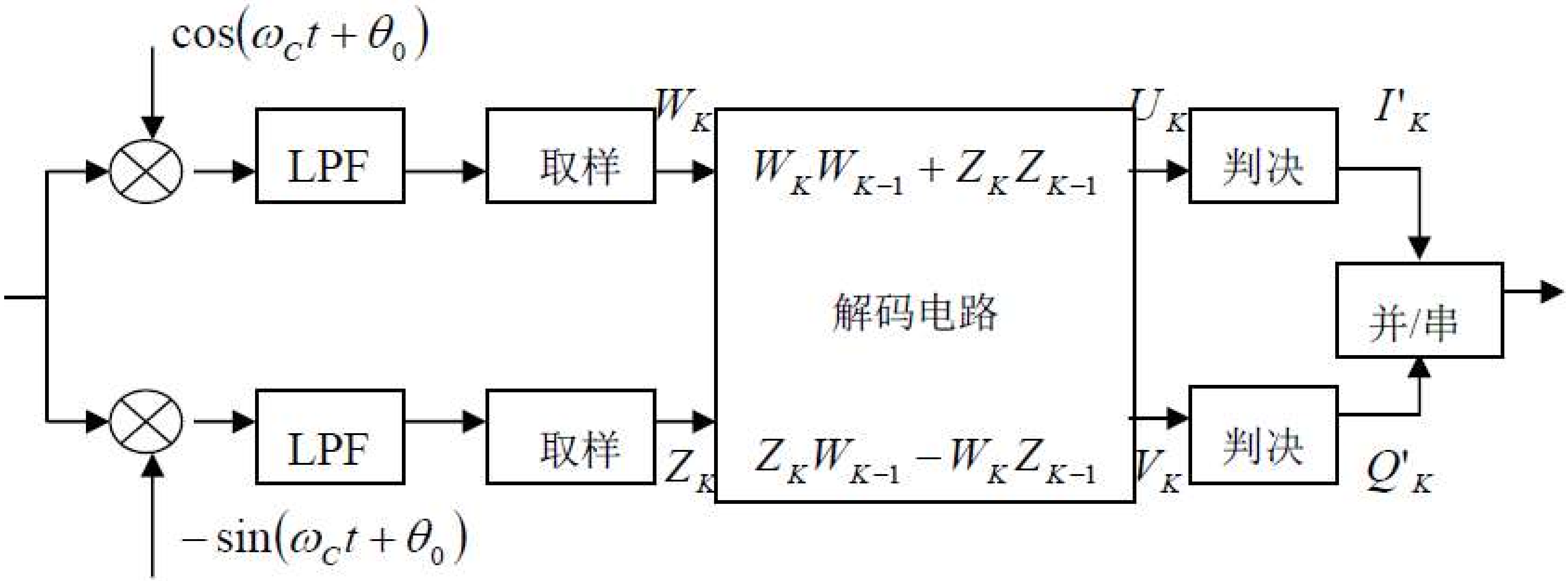
4.2 估计 功率谱密度的估计

周期图法和自相关函数法估计功率谱虽然具有计算量小，与信号本身特征无关等主要优点，计算时相当于对无限长序列加了一个长度为N的矩形窗，不是真实功率谱的一致估计，且存在旁瓣泄露的问题，将导致弱信号可能被强信号的旁瓣淹没。

习题1：试编写Matlab程序对π/4 DQPSK接收机进行仿真。通过对接收机和前面发射机进行联合仿真，证明接收机仿真可以正常工作。

π/4 DQPSK接收机差分解调原理





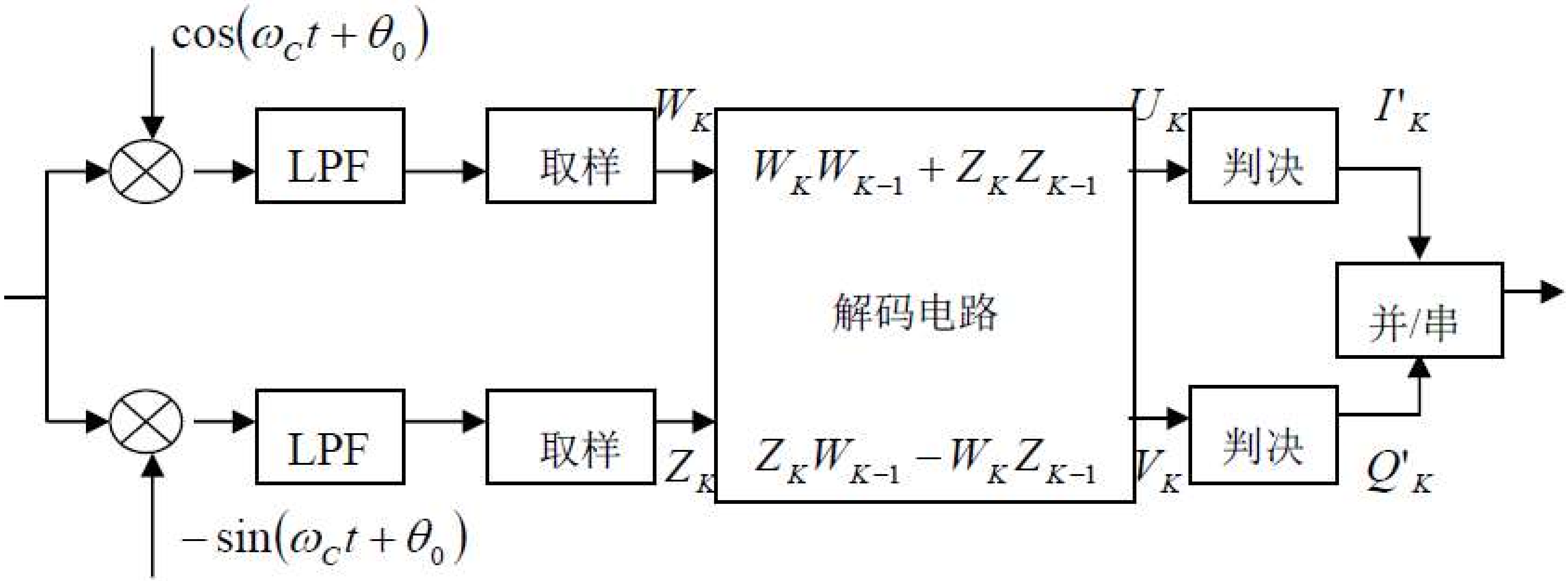
设输入信号为 $S_K(t)=\cos(\omega_C t+\varphi_K)$ ，他在同相支路与本地载波 $\cos(\omega_C t+\theta_0)$ 相乘，滤出的低频分量经取样后，得

$$W_K=\frac{1}{2}\cos(\varphi_K-\theta_0)$$

在正交支路与本地载波 $-\sin(\omega_C+\theta_0)$ 相乘，滤出的低频分量经取样后，得

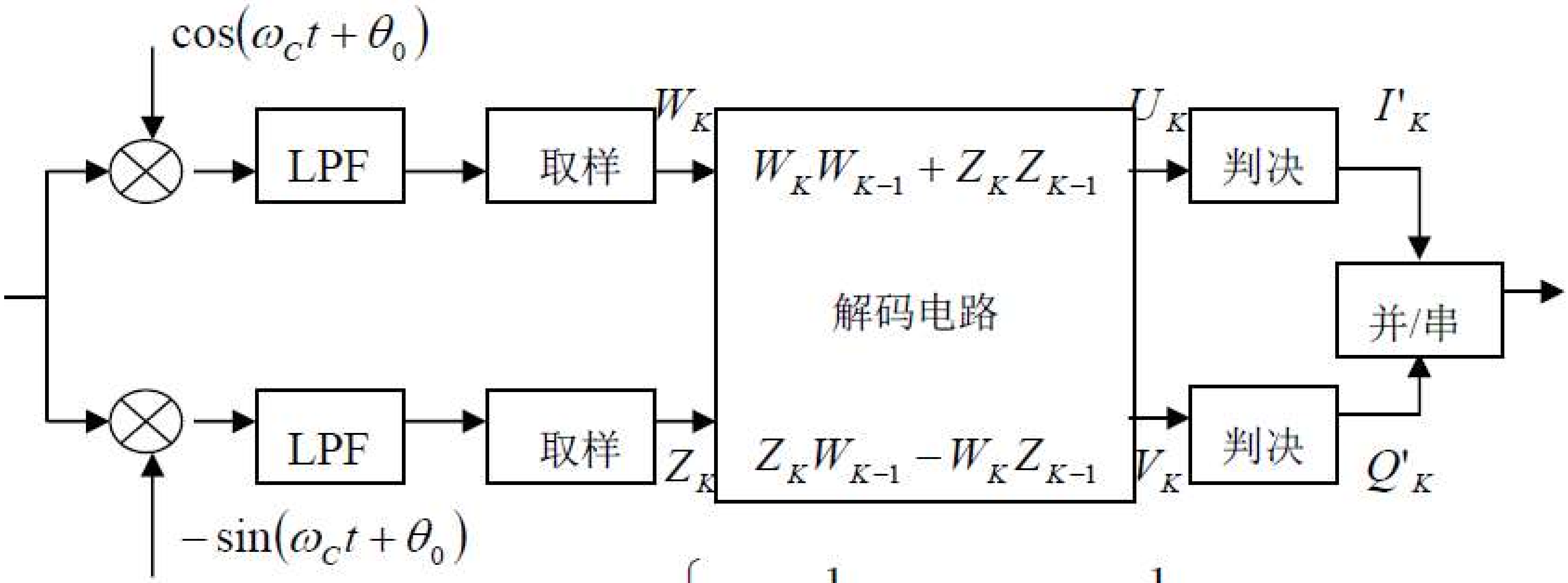
$$Z_K=\frac{1}{2}\sin(\varphi_K-\theta_0)$$

式中： θ_0 为固定相位差。



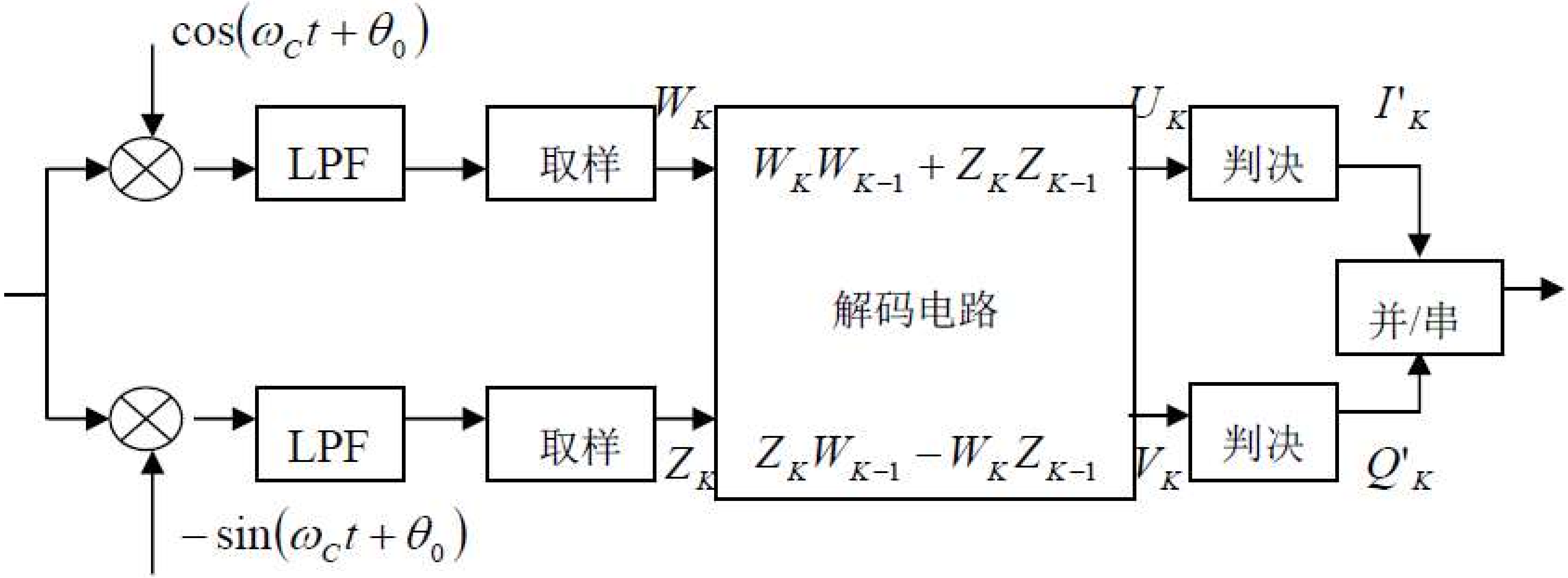
所用的低通滤波器应与发端调制器所用的低通滤波器相匹配，以消除码间干扰。

解码规则为：$\begin{cases} U_K = W_K W_{K-1} + Z_K Z_{K-1} \\ V_K = Z_K W_{K-1} - W_K Z_{K-1} \end{cases}$



$$\begin{cases} U_K = W_K W_{K-1} + Z_K Z_{K-1} \\ V_K = Z_K W_{K-1} - W_K Z_{K-1} \end{cases}$$

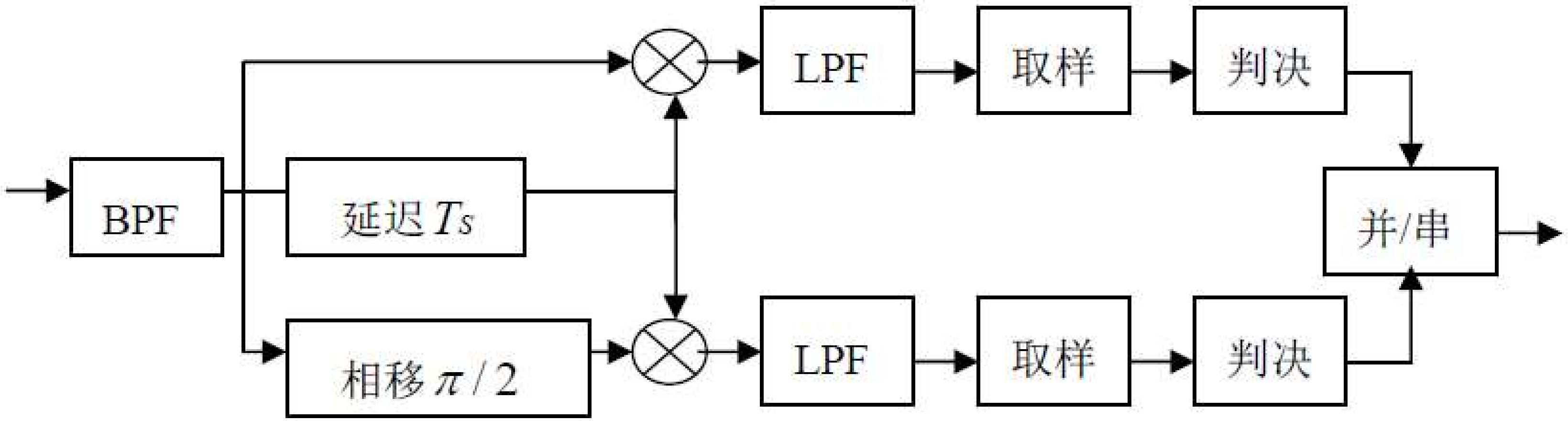
$$\begin{cases} U_K = \dfrac{1}{2}\cos(\varphi_K - \theta_0) \times \dfrac{1}{2}\cos(\varphi_{K-1} - \theta_0) \\ + \dfrac{1}{2}\sin(\varphi_K - \theta_0) \times \dfrac{1}{2}\sin(\varphi_{K-1} - \theta_0) \\ = \dfrac{1}{4}\cos(\varphi_K - \varphi_{K-1}) = \dfrac{1}{4}\cos\Delta\varphi_K \\ V_k = \dfrac{1}{2}\sin(\varphi_K - \theta_0) \times \dfrac{1}{2}\cos(\varphi_{K-1} - \theta_0) \\ - \dfrac{1}{2}\cos(\varphi_K - \theta_0) \times \dfrac{1}{2}\sin(\varphi_{K-1} - \theta_0) \\ = \dfrac{1}{4}\sin(\varphi_K - \varphi_{K-1}) = \dfrac{1}{4}\sin\Delta\varphi_K \end{cases}$$



$$
\begin{cases}
U_K=\dfrac{1}{2}\cos(\varphi_K-\theta_0)\times\dfrac{1}{2}\cos(\varphi_{K-1}-\theta_0) \\
+\dfrac{1}{2}\sin(\varphi_K-\theta_0)\times\dfrac{1}{2}\sin(\varphi_{K-1}-\theta_0) \\
=\dfrac{1}{4}\cos(\varphi_K-\varphi_{K-1})=\dfrac{1}{4}\cos\Delta\varphi_K \\
V_k=\dfrac{1}{2}\sin(\varphi_K-\theta_0)\times\dfrac{1}{2}\cos(\varphi_{K-1}-\theta_0) \\
-\dfrac{1}{2}\cos(\varphi_K-\theta_0)\times\dfrac{1}{2}\sin(\varphi_{K-1}-\theta_0) \\
=\dfrac{1}{4}\sin(\varphi_K-\varphi_{K-1})=\dfrac{1}{4}\sin\Delta\varphi_K
\end{cases}
$$

$$
\begin{cases}
U_K>0 & I'_K\text{判为}1 \\
U_K<0 & I'_K\text{判为}-1 \\
V_K>0 & Q'_K\text{判为}1 \\
V_K<0 & Q'_K\text{判为}-1
\end{cases}
$$

π/4 DQPSK接收机中频差分解调原理

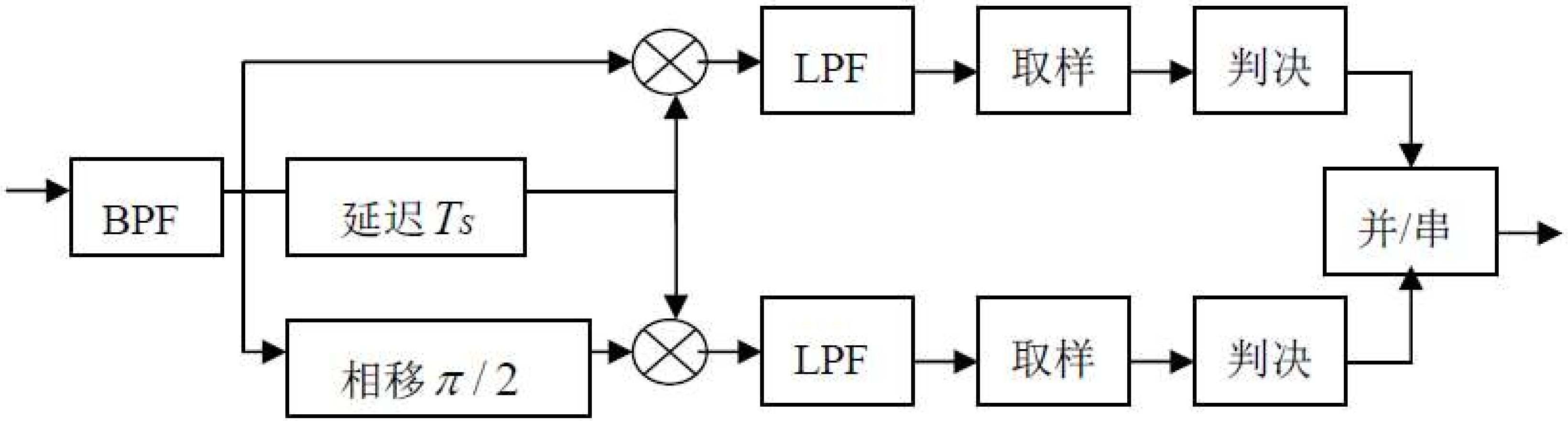


经过延迟的信号 $S_{K-1}(t)=\cos(\omega_C t+\varphi_{K-1})$

育支路信号相乘 $\cos(\omega_C t+\varphi_K)\cos(\omega_C t+\varphi_{K-1})$

$$\sin(\omega_C t+\varphi_K)\cos(\omega_C t+\varphi_{K-1})$$

π/4 DOPSK接收机中频差分解调原理



经滤波和取样

$$\begin{cases} U_K = \frac{1}{2}\cos(\varphi_K - \varphi_{K-1}) = \frac{1}{2}\cos\Delta\varphi_K \\ \\ V_K = \frac{1}{2}\sin(\varphi_K - \varphi_{K-1}) = \frac{1}{2}\sin\Delta\varphi_K \end{cases}$$