

MATLAB/Simulink仿真数字基带信号与发送滤波器的

数字通信系统中,从信源发出的数字信息可以表示成数字码元序列,数字基带信号 是数字码元序列的脉冲电压或电流表现形式。数字基带信号的波形可以采用方波、三角 波、升余弦波形等。由于数字信道的特性及要求不同,例如,很多信道不能传输信号的直 流分量和频率很低的分量。另外,为了在接收端得到每个码元的起止时刻,需要在发送 的信号中带有码元起止时刻的信息,为此需要将原码元依照一定的规则转换成适合信道 传输要求的传输码(又称作线路码)。

为了保证信号在传输时不出现或少出现码间干扰,基带传输系统的设计必须满足奈 奎斯特(Nyquist)第一准则。满足奈奎斯特第一准则的基带传输系统有很多种,最简单 的一种就是理想低通系统。但实际传输中,不可能有绝对理想的基带传输系统,常常采 用具有奇对称"滚降"特性的低通滤波器(发送滤波器)对信号进行成形滤波。本章将介 绍基带信号波形、码型以及成形滤波器的仿真方法。

3.1 数字基带信号波形仿真

数字基带信号的波形经常采用方波,其中最基本的二进制基带信号波形有单极性归 零波形、单极性不归零波形、双极性归零波形、双极性不归零波形,如图 3-1 所示。



1. 数字基带信号波形的 MATLAB 仿真

下面通过 MATLAB 程序来仿真一串随机消息代码的基带信号波形,首先产生 1000 个随机信号序列,分别用单极性归零码、单极性不归零码、双极性归零码和双极性不归零 码编码,并且求平均功率谱密度。其中基本波形的 MATLAB 仿真流程如图 3-2 所示。

源代码(以双极性为例)如下:

close all	
clear all	
k = 14;	❀采样点数的设置
L = 32;	℅每码元采样数的设置
$N = 2^{k};$	
M = N/L;	% M 为码元个数
dt = 1/L;	8时域采样间隔



<pre>k = round(rand(1,M));</pre>	%产生一个长度为 M 的随机序列 k,0 和 1 等概率出现
<pre>nrz = zeros(L,M);</pre>	%产生一个 L 行 M 列的不归零矩阵,初始化为全 0 矩阵
<pre>rz = zeros(L,M);</pre>	%产生一个 L 行 M 列的归零矩阵,初始化为全 O 矩阵
for i = 1:M	
ifk(i) == 1	
<pre>nrz(:,i) = 1;</pre>	% 使不归零矩阵第 i 列全部元素都为 1
rz(1:L/2,i) = 1;	% 使归零矩阵第 i 列前 L/2 个元素为 1
else	
nrz(:,i) = -1;	% 使不归零矩阵第 i 列全部元素都为 - 1

------ MATLAB/System View通信原理实验与系统仿真(第2版)

```
rz(1:L/2,i) = -1;
                                 % 使归零矩阵第 i 列前 L/2 个元素为-1
     end
     end
     %分别重排 nrz、rz 矩阵为1行 N列的矩阵
    nrz = reshape(nrz, 1, N);
    rz = reshape(rz,1,N);
        %作傅里叶变换并计算功率谱密度
        NRZ = t2f(nrz, dt);
        P1 = NRZ. * conj(NRZ)/T;
        RZ = t2f(rz, dt);
    P2 = RZ. * conj(RZ)/T;
        &求功率谱密度的均值
        EP1 = (EP1 * (x - 1) + P1)/x;
        EP2 = (EP2 * (x - 1) + P2)/x;
end
%程序第2部分:绘制波形图和功率谱密度曲线
                                 8开启一个编号为1的绘图窗口
figure(1)
                                 ≈绘制双极性不归零码的时域图
subplot(2,2,1);plot(t,nrz)
axis([-5,5,min(nrz)-0.1,max(nrz)+0.1])
title('双极性不归零码','fontsize',12)
xlabel('t(ms)', 'fontsize',12)
ylabel('nrz(t)','fontsize',12)
grid on
subplot(2,2,2);plot(t,rz)
                                 8绘制双极性归零码的时域图
axis([-5,5,min(rz)-0.1,max(rz)+0.1])
title('双极性归零码','fontsize',12)
xlabel('t(ms)', 'fontsize',12)
ylabel('rz(t)', 'fontsize',12)
grid on
subplot(2,2,3);plot(f,EP1)
                                 8绘制双极性不归零码的功率谱密度图
axis([-5,5,0,1.2])
title('双极性不归零码功率谱密度图','fontsize',12)
xlabel('f(kHz)','fontsize',12)
ylabel('P1(f)', 'fontsize',12)
grid on
                                 8 绘制双极性归零码的功率谱密度图
subplot(2,2,4);plot(f,EP2)
axis([-5,5,0,0.3])
title('双极性归零码功率谱密度图','fontsize',12)
xlabel('f(kHz)', 'fontsize', 12)
ylabel('P2(f)','fontsize',12)
grid on
```

上面程序中需要调用一个傅里叶变换的函数 t2f,该函数定义如下:

%将时域信号作傅里叶变换到频域,x必须是二阶的矩阵,dt是信号的时域分辨率 function X = t2f(x,dt) X = fftshift(fft(x)) * dt;



双极性和单极性二进制信号的波形及功率谱分别如图 3-3、图 3-4 所示。



图 3-4 单极性二进制信号波形的 MATLAB 仿真

第 3 章

数字基带信号与发送滤波器的MATLAB/Simulink仿真

MATLAB/SystemView通信原理实验与系统仿真(第2版)

2. 数字基带信号波形的 Simulink 仿真

用Simulink实现对单极性归零波形、单极性不归零波形、双极性归零波形和双极性 不归零波形的仿真,仿真模型如图 3-5 所示,模型中各模块的主要参数设置见表 3-1。



图 3-5 数字基带信号波形的仿真模型

表 3-1 数字基带信号波形的 Simulink 仿真参数

模 块	参数名称	取 值				
	Probability of a zero	0.3				
Bernoulli Binary Generator	Initial seed	61				
	Sample time	1(s)				
Switch	Criteria for passing first in put	u2>=Threshold				
Switch	Threshold	0.5				
Switch 1	Criteria for passing first in put	u2>=Threshold				
Switch 1	Threshold	0				
Dulas Consector	Period	1(s)				
i uise Generator	Pulse Width	50(% of period)				

Simulink 仿真结果如图 3-6 所示,示波器从上到下依次是原始信号、双极性不归零 信号、单极性不归零信号、双极性归零信号和单极性归零信号。

🛦 Sco	pe		. 21	E 1		u a-	9	0	J	£1	-		×
<u>F</u> ile	<u>T</u> ools <u>'</u>	<u>V</u> iew	S <u>i</u> mulatio	on <u>H</u> e	elp								3
<u>۰</u> (4		\$.	⊕	<u>۰</u>	f	•						Ē
1													
0.5													
0													
1													
0													
-1 -													
1													
0.5													
0				_									
1													
0				_									
-1 -													
1													
0.5													
0	1		2	3	4		5	6		7	8	9	10
Ready												-	T=10.000

图 3-6 数字基带信号波形的 Simulink 仿真结果

3.2 数字基带信号码型仿真

在基带信号传输时,由于不同传输媒介具有不同的传输特性,需要使用不同的接口 线路码型(传输码)。为了匹配于基带信道的传输特性,并考虑到接收端提取时钟方便, 传输码应具备以下的特性:

- (1) 不含直流分量,且低频分量较少;
- (2) 含有分富的定时信息,便于从接收码流中提取位同步时钟信号;
- (3) 功率谱主瓣宽度窄,以节省传输频带;
- (4) 不受信息源统计特性的影响;
- (5) 具有内在的监错能力;
- (6) 编译码简单,以降低通信延时和成本。

满足或部分满足以上特性的传输码型种类很多,下面介绍目前常用的几种传输码的仿真。

3.2.1 数字双相码

双相码,又名曼彻斯特(Manchester)码,其编码规则是:将信息代码"0"编码为线路码"01";信息代码"1"编码为线路码"10"。双相码常用于局域网传输,每一位的中间的跳

- MATLAB/System View 通信原理实验与系统仿真(第2版)

变既作时钟信号,又作数据信号。

1. 数字双相码的 MATLAB 仿真

下面通过 MATLAB 程序来仿真一串随机消息代码的基带信号波形。首先产生 1000 个随机信号序列,用双相码的编码规则进行编码,并且求平均功率谱密度。其中编 码部分的 MATLAB 仿真流程如图 3-7 所示。



源代码如下:

close all clear all % 采样点数的设置 k = 14; % 每码元采样数的设置 L = 128; N = 2^k; M = N/L; % M 为码元个数 dt = 1/L; % 时域采样间隔 T = N*dt; % 时域截断区间 df = 1.0/T; % 频域采样间隔

```
Bs = N * df/2;
                               8 版域截断区间
t = linspace(-T/2, T/2, N);
                               8产生时域采样点
f = linspace(-Bs, Bs, N);
                               8产生频域采样点
EP1 = zeros(size(f));
EP2 = zeros(size(f));
EP3 = zeros(size(f));
                               8采样 1000 次
for x = 1:1000
                               %产生一个长度为 M 的随机序列 K,0 和 1 等概率出现
 K = round(rand(1, M));
                               %产生一个L行M列的原码矩阵,初始化为全0矩阵
  original = zeros(L,M);
 Manchester = zeros(L,M);
                               %产生一个 L 行 M 列的曼彻斯特码矩阵, 初始化为全 0 矩阵
  for i = 1:M
     if K(i) == 1
         original (:,i) = 1;
                               8原码
         Manchester (1:L/2, i) = 1; % 使曼彻斯特码矩阵第 i 列前 L/2 个元素为 1
     else
         original (:,i) = 0;
                               8原码
                              % 使曼彻斯特码矩阵第 i 列为 1
         Manchester (:, i) = 1;
         Manchester (1:L/2, i) = 0; % 使曼彻斯特码矩阵第 i 列前 L/2 个元素为 0
     end
  end
  %% ------分别重排 nrz、曼彻斯特码矩阵为1行 N列的矩阵 ------ %%
   original = reshape(original,1,N);
   Manchester = reshape(Manchester, 1, N);
     %作傅里叶变换并算出功率谱密度
   ORIGINAL = t2f(original,dt);
   P1 = ORIGINAL. * conj(ORIGINAL)/T;
   MANCHESTER = t2f(Manchester, dt);
   P2 = MANCHESTER. * conj(MANCHESTER)/T;
    ∗求功率谱密度的均值
   EP1 = (EP1 * (x - 1) + P1)/x;
   EP2 = (EP2 * (x - 1) + P2)/x;
end
                               %开启一个编号为1的绘图窗口
figure(1)
subplot(2,2,1);
plot(t, original);
                               8绘制原码的时域图
axis([-5,5,min(original) - 0.1,max(original) + 0.1]);
title('原码','fontsize',12);
xlabel('t(ms)', 'fontsize',12);
ylabel('original(t)', 'fontsize', 12);
grid on
subplot(2,2,2);
                               8绘制数字双相码的时域图
plot(t,Manchester) ;
axis([ -5,5,min(Manchester) - 0.1,max(Manchester) + 0.1]);
title('数字双相码','fontsize',12);
xlabel('t(ms)', 'fontsize',12);
ylabel('Manchester (t)', 'fontsize', 12);
grid on
subplot(2,2,3);plot(f,EP1);
                              8绘制原码的功率谱密度图
axis([-5,5,0,0.3]);
title('原码功率谱密度图','fontsize',12);
xlabel('f(kHz)', 'fontsize',12);
ylabel('P1(f)','fontsize',12);
```

第3章数

字基带信号与发送滤波器的MATLAB/Simulink仿

真

MATLAB/SystemView通信原理实验与系统仿真(第2版)

```
grid on
subplot(2,2,4);plot(f,EP2); %会制数字双相码的功率谱密度图
axis([-5,5,0,0.15]);
title('数字双相码功率谱密度图','fontsize',12);
xlabel('f(kHz)','fontsize',12);
ylabel('P2(f)','fontsize',12);
grid on
```

数字双相码的仿真结果如图 3-8 所示。



图 3-8 数字双相码的 MATLAB 仿真结果

2. 数字双相码的 Simulink 仿真

48

用 Simulink 实现对数字双相码的仿真,仿真模型如图 3-9 所示,模型中各模块的主要参数设置见表 3-2。



模 块	参数名称	取 值			
	Probability of a zero	0.5			
Bernoulli Binary Generator	Initial seed	61			
	Sample time	1(s)			
Smitch	Criteria for passing first in put	u2>=Threshold			
Switch	Threshold	0.5			
	Period	1(s)			
Pulse Generator	Pulse Width	50($\%$ of period)			
	Phase delay	0			
	Period	1(s)			
Pulse Generator1	Pulse Width	50(% of period)			
	Phase delay	0.5			

表 3-2 数字双相码的 Simulink 仿真参数

Simulink 仿真结果如图 3-10 所示,示波器输出从上到下依次为原始信号波形、数字 双相码波形。



图 3-10 数字双相码的 Simulink 仿真结果

3.2.2 三阶高密度双极性码

三阶高密度双极性码(HDB₃码),其编码规则是:当信息代码中连"0"个数不大于 3 时,"1"码用正负脉冲交替表示;当信息代码中连"0"个数大于 3 时,将每 4 个连"0"串的 第 4 个"0"编码为与前一非"0"码同极性的正脉冲或负脉冲,该脉冲为破坏码或 V 码,为

------ MATLAB/System View通信原理实验与系统仿真(第2版)

保证加 V 码后输出仍无直流分量,则需要:①相邻 V 码的极性必须相反,为此当相邻 V 码间有偶数个"1"时,将后面的连"0"串中第 1 个"0"编码为 B 符号,B 符号的极性与前一 非"0"码的极性相反,而 B 符号后面的 V 码与 B 符号的极性相同;②V 码后面的非"0"符 号的极性再交替反转。HDB₃ 码是 CCITT 推荐作为 PCM 语音系统四次群线路接口码 型,在光缆传输系统中采用。图 3-11 是原码与 HDB₃ 码的波形图。



HDB₃ 码虽然编码很复杂,但解码规则很简单:若3连"0"前后非零脉冲同极性,则 将最后一个非零元素译为零,如+1000+1 就应该译成"10000";若2连"0"前后非零脉 冲极性相同,则两零前后都译为零,如-100-1 就应该译为0000。再将所有的-1变换 成+1后,就可以得到原消息代码。

```
HDB。的 MATLAB 仿真
源程序如下:
x=[101100000011100000010]; % 输入原码
                                    % 输出 v 初始化
y = x;
                                    8 计数器初始化
num = 0;
for k = 1:length(x)
  if x(k) == 1
                                   8 "1"计数器
     num = num + 1;
       if num/2 == fix(num/2)
                                   % 奇数个"1"时输出-1,进行极性交替
           y(k) = 1;
       else
           y(k) = -1;
       end
   end
end
% HDB₃ 编码
                                   % 连零计数器初始化
num = 0;
yh = y;
                                   8 输出初始化
                                   %极性标志初始化为0
sign = 0;
V = zeros(1, length(y));
                                   % 记录 Ⅴ脉冲位置的变量
B = zeros(1, length(y));
                                   % 记录 B 脉冲位置的变量
```

```
for k = 1: length(y)
  if y(k) == 0
                                      % 连"0"个数计数
      num = num + 1;
                                       8 如果4连"0"
      if num == 4
                                       8 计数器清零
        num = 0;
        yh(k) = 1 * yh(k - 4);
                                       8让4连"0"的最后一个"0"变为与前一个非零
                                       %符号相同极性的符号
        V(k) = yh(k);
                                       8 Ⅴ脉冲位置记录
                                       % 如果当前 Ⅴ符号与前一个 Ⅴ符号的极性相同
        if yh(k) == sign
          yh(k) = -1 * yh(k);
                                       %则让当前 V 符号极性反转
                                       % 添加 B 符号, 与 V 符号同极性
          yh(k-3) = yh(k);
          B(k-3) = yh(k);
                                       8 B脉冲位置记录
                                       % Ⅴ脉冲位置记录
          V(k) = yh(k);
          yh(k+1:length(y)) = -1 * yh(k+1:length(y));
                                       8 Ⅴ 后面的非零符号从 Ⅴ 开始再交替
       end
                                       % 记录前一个 Ⅴ符号的极性
      sign = yh(k);
    end
 else
     num = 0;
                                       8 当前输入为"1"则连零计数器清零
 end
end
                                       8 编码完成
≈ HDB, 解码
                                       % HDB, 码输入
input = yh;
                                       8 输出初始化
decode = input;
                                       8 极性标志初始化
sign = 0;
for k = 1:length(yh)
   if input(k) \sim = 0
                                       8 如果当前码与前一个非零码的极性相同
      if sign == yh(k)
                                      %则该码判为Ⅴ码并将*00Ⅴ清零
         decode(k-3:k) = [0 \ 0 \ 0 \ 0];
      end
                                       8 极性标志
      sign = input(k);
   end
end
                                       8 整流
decode = abs(decode);
subplot(3,1,1);stairs([0:length(x) - 1],x);axis([0 length(x) - 0.2 1.2]);
title('原码');
subplot(3,1,2); stairs([0:length(x) - 1], yh); axis([0 length(x) - 1.2 1.2]);
title('HDB3 编码');
subplot(3,1,3);stairs([0:length(x) - 1],decode);axis([0 length(x) - 0.2 1.2]);
title('HDB3 解码');
```

```
3
章
数
《字基带信号与发送滤波器的MATLAB/Simulink仿
真
```

第



3.3 发送滤波器仿真

在数字通信系统中,基带信号进入调制器前,波形是矩形脉冲,突变的上升沿和下降 沿包含高频成分较丰富,信号的频谱一般比较宽,通过带限信道时,单个符号的脉冲将延 伸到相邻符号的码元内,产生码间串扰。因此在信道带宽有限的条件下,要降低误码率, 需在信号传递前,通过发送滤波器(脉冲成形滤波器)对其进行脉冲成形处理,改善其频 谱特性,产生适合信道传输的波形。数字通信系统中常用的波形成形滤波器有升余弦脉 冲滤波器、平方根升余弦滤波器、高斯滤波器等。下面分别讨论这三种滤波器的特性及 仿真。

3.3.1 升余弦脉冲滤波器

余弦脉冲滤波器即系统函数具有余弦波的变化特点,如图 3-13 所示。余弦滚降系统的传输特性可用下式表示:

$$H(\omega) = \begin{cases} 1, & 0 \leq |\omega| \leq \frac{(1-\alpha)\pi}{T_s} \\ 1 + \cos\frac{T_s}{2\alpha} \left(\omega - \frac{(1-\alpha)\pi}{T_s} \right), & \frac{(1-\alpha)\pi}{T_s} \leq |\omega| \leq \frac{(1+\alpha)\pi}{T_s} \\ 0, & |\omega| \geq \frac{(1+\alpha)\pi}{T_s} \end{cases}$$
(3-1)

式中, α 为滚降系数; T_s 为码元间隔。由图 3-13 可以看出,滚降特性所形成的波形h(t)除采样点t=0处不为零外,其余采样点上均为零,并且"拖尾"现象随着 α 的增大而振荡幅度减小、衰减速度加快。



以下程序实现对升余弦滚降滤波器的仿真,升余弦滚降滤波器的频谱和时域波形仿 真结果分别如图 3-14 和图 3-15 所示。



图 3-15 升余弦滚降滤波器时域波形

53

第 3 章

数字基带信号与发送滤波器的MATLAB/Simulink仿真

------ MATLAB/System View 通信原理实验与系统仿真(第2版)

```
8升余弦滚降系统
clear all;
close all;
Ts = 1;
N sample = 17;
dt = Ts/N sample;
df = 1.0/(20.0 \times Ts);
t = -10 * Ts:dt:10 * Ts;
f = -2/Ts:df:2/Ts;
alpha = [0, 0.5, 1];
for n = 1:length(alpha)
     for k = 1:length(f)
          if abs(f(k)) > 0.5 \times (1 + alpha(n))/Ts
               Xf(n,k) = 0;
          elseif abs(f(k)) < 0.5 * (1 - alpha(n))/Ts
              Xf(n,k) = Ts;
          else
               Xf(n,k) = 0.5 * Ts * (1 + cos(pi * Ts/(alpha(n) + eps) * (abs(f(k)) - 0.5 * (1 - cos(pi + Ts/(alpha(n) + eps)))))
alpha(n))/Ts)));
          end
     end
     xt(n,:) = sinc(t/Ts). * (cos(alpha(n) * pi * t/Ts))./(1-4 * alpha(n)^2 * t.^2/Ts^2 +
eps);
end
figure(1)
plot(f, Xf(1, :), 'b', f, Xf(2, :), 'r', f, Xf(3, :), 'k');
axis([-1101.2]);xlabel('f/Ts');ylabel('升余弦滚降频谱');
legend('alpha = 0', 'alpha = 0.5', 'alpha = 1');
figure(2)
plot(t, xt(1, :), 'b', t, xt(2, :), 'r', t, xt(3, :), 'k');
legend('alpha = 0', 'alpha = 0.5', 'alpha = 1');
axis([-1010-0.51.1]);xlabel('t');ylabel('升余弦滚降波形);
```

下面来仿真升余弦滤波器对信号的影响。采用 MATLAB 工具箱中专门用于升余 弦 FIR 滤波器的指令[NUM, DEN]=RCOSINE(Fd,Fs,TYPE_FLAG,R),可以返回 一个具有升余弦过渡带的低通线性相位 FIR 滤波器,截止频率为 Fd,滚降系数为 R,采 样频率为 Fs,TYPE_FLAG 规定设计的是规范的升余弦滚降滤波器(normal)还是平方 根升余弦滤波器(sqrt),用整型参数 Delay 设定延时。

```
% 设置参量,采用4倍采样速率,滚降系数为0.5
Fd=1;Fs=4;Delay=2;R=0.5;
% ------建立升余弦滚降滤波器 ------ %
[yf,tf]=rcosine(Fd,Fs,'fir/normal',R,Delay);
% 画图得到升余弦滚降滤波器波形
% b1=ones(1,length(t2)); % 滤波器输入矩形脉冲
figure(1);
```

```
subplot(3,1,1);
plot(yf);
grid;
xlabel('Time');
ylabel('Amplitude');
title('升余弦滚降滤波器 h(t)');
% ------ 定义一个与二元序列对应的时间序列作为原始信号 ------ %
x = [zeros(1,10), ones(1,10), ones(1,10), zeros(1,10), zeros(1,10), zeros(1,10)];
y = filter(yf,tf,x)/Fs;
8 画出原始信号波形
subplot(3,1,2);
plot(x);
axis([0,61, -0.2,1.2]);
title('原始信号');
8 ------ 原始信号通过升余弦滚降滤波器后的输出
subplot(3,1,3);
plot(y);
axis([2,61, -0.2,1.2]);
title('滤波后输出')
grid;
```

由图 3-16 可见,原始信号通过该升余弦滚降滤波器可以使波形平滑,有效地改变 突变的上升沿和下降沿,从而消除波形中的高频成分,达到降低码间串扰的可能性,提 高频带利用率的效果。由于滤波器的影响,原始信号和滤波后信号之间存在一定的 延迟。



3.3.2 平方根升余弦滤波器

可以将波形成形滤波器放置在收发两端,即在发送端和接收端分别用一个平方根 升余弦滤波器(平方根升余弦函数),并且使两个滤波器满足匹配滤波原则,则既实现 升余弦滤波器的作用,也满足匹配滤波器的实现,从而提升接收端信噪比,更便于准确 接收信号。如果不考虑由于信道引起的码间串扰,两个平方根升余弦函数相乘(相当 于时域卷积)就得到升余弦形式的合成的系统传输函数。平方根升余弦滤波器可以表 示为

$$H_{\mathrm{T}}(\omega) = H_{\mathrm{R}}(\omega) = H(\omega) = \begin{cases} 1, & 0 \leqslant |\omega| \leqslant \frac{(1-\alpha)\pi}{T_{\mathrm{s}}} \\ \sqrt{1+\cos\frac{T_{\mathrm{s}}}{2\alpha}\left(\omega-\frac{(1-\alpha)\pi}{T_{\mathrm{s}}}\right)}, & \frac{(1-\alpha)\pi}{T_{\mathrm{s}}} \leqslant |\omega| \leqslant \frac{(1+\alpha)\pi}{T_{\mathrm{s}}} \\ 0, & |\omega| \geqslant \frac{(1+\alpha)\pi}{T_{\mathrm{s}}} \end{cases} \end{cases}$$

$$(3-2)$$

平方根升余弦冲激响应的表达式为

$$h(t) = \frac{\sqrt{T}}{\pi t \left(T^2 - 16t^2 T^2\right)} \left\{T^2 \sin\left[2\pi f_0 \left(1 - \alpha\right)t\right] + 4\alpha t T \cos\left[2\pi f_0 \left(1 + \alpha\right)t\right]\right\}$$
(3-3)

采用 MATLAB 工具箱中专门用于升余弦 FIR 滤波器的指令[NUM, DEN] = RCOSINE(Fd,Fs,TYPE_FLAG,R),可以返回一个具有平方根升余弦过渡带的低通线性相位 FIR 滤波器,截止频率为 Fd,滚降系数为 R,采样频率为 Fs,TYPE_FLAG 用来规定滤波器类型,整型参数 Delay 设定延时。

```
%设置参量,采用4倍采样速率,滚降系数为0.5
Fd = 1; Fs = 4; Delay = 2; R = 0.5;
[yf,tf] = rcosine(Fd,Fs,'sqrt',R,Delay);
8 画图得到升余弦滚降滤波器波形
% b1 = ones(1, length(t2));
                             % 滤波器输入矩形脉冲
figure(1);
subplot(3,1,1);
plot(yf);
grid;
xlabel('Time');
ylabel('Amplitude');
title('平方根升余弦滚降滤波器 h(t)');
x = [zeros(1,10), ones(1,10), ones(1,10), zeros(1,10), zeros(1,10), zeros(1,10)];
y1 = filter(yf, tf, x)/(Fs^{0.5});
% y2 = filter(yf,tf,y1)/(Fs^0.5);
```

```
第
3
章
数字基带信号与发送滤波器的MATLAB/Simulink仿真
```

```
% 画出原始信号波形
subplot(3,1,2);
plot(x);
axis([0,61, -0.2,1.2]);
title('原始信号');
% ------原始信号通过升余弦滚降滤波器后的输出 ------%
subplot(3,1,3);
plot(y1);
axis([2,61, -0.2,1.2]);
title('滤波后输出')
grid;
```

由图 3-17 可见,原始信号通过该平方根升余弦滚降滤波器后也可以使波形平滑,有效地改变突变的上升沿和下降沿,作用与升余弦滤波器类似。实际应用中,收、发两端的 平方根升余弦滚降滤波器可以按照匹配滤波器的原则进行设计。



3.3.3 高斯滤波器

在一些通信场合(如移动通信),对信号带外辐射功率的限制十分严格,比如要求衰减达到 70~80dB 以上,从而减小对邻道的干扰,这时可以采用高斯低通滤波器。高斯低 通滤波器的特点是:

(1) 带宽窄,具有良好的截止特性;

(2) 具有较低的过冲脉冲响应,以防止调制器的瞬间频偏过大;

(3)保持滤波器输出脉冲的面积不变,以便于进行相干解调。

高斯型滤波器的传输函数 H(f)为

----- MATLAB/System View 通信原理实验与系统仿真(第2版)

$$H(f) = \exp(-a^2 f^2)$$
 (3-4)

高斯型滤波器的冲激响应为

$$h(t) = \frac{\sqrt{\pi}}{a} \exp\left(-\frac{\pi^2}{a^2} t^2\right)$$
(3-5)

下面通过仿真来说明高斯脉冲成形滤波器对矩形脉冲输入的影响。假设 b(t)是高度为 1、宽度为 T_b 的矩形脉冲,则 b(t)通过高斯脉冲成形滤波器的输出波形 g(t)为

$$g(t) = h(t) * b(t) = \int_{t-\frac{T_{\rm b}}{2}}^{t+\frac{T_{\rm b}}{2}} \frac{\sqrt{\pi}}{a} \exp\left(-\frac{\pi^2}{a^2}\tau^2\right) d\tau$$
$$= \frac{1}{2} \left\{ \operatorname{erfc}\left[\frac{\pi}{a} \left(t - \frac{T_{\rm b}}{2}\right] - \operatorname{erfc}\left[\frac{\pi}{a} \left(t + \frac{T_{\rm b}}{2}\right)\right] \right\}$$
(3-6)

源代码如下:

t1 = -1.5:0.01:1.5;t2 = -1.5:0.01:1.5;b1 = [zeros(1,50), ones(1,200), zeros(1,51)]; %% ------产牛滤波器 1, a = 0.25 ------ %% $y1 = sqrt(pi)/0.25 * exp(-((pi * t1).^2)/0.25.^2);$ z1 = 0.5 * (erfc(pi/0.25 * (t1 - 1)) - erfc(pi/0.25 * (t1 + 1)));%% -----产生滤波器 2, a = 0.5 ----- %% $y_2 = sgrt(p_1)/0.5 * exp(-((p_1 * t_1).^2)/0.5.^2);$ z2 = 0.5 * (erfc(pi/0.5 * (t1 - 1)) - erfc(pi/0.5 * (t1 + 1)));%% -----产生滤波器 3, a = 1 ----- %% $y3 = sqrt(pi) * exp(-(pi * t1).^{2});$ z3 = 0.5 * (erfc(pi * (t1 - 1)) - erfc(pi * (t1 + 1))); %% -----产生滤波器 4, a = 2 ----- %% $y4 = sqrt(pi)/2 * exp(-((pi * t1).^2)/2.^2);$ z4 = 0.5 * (erfc(pi/2 * (t1 - 1)) - erfc(pi/2 * (t1 + 1)));subplot(3,1,1), plot(t2,b1); axis([-1.5 1.5 0 1.2]); xlabel('t/T'); ylabel('b(t)'); title('高斯脉冲成形滤波器的输入(矩形脉冲)'); subplot(3,1,2),plot(t1,y1,'r',t1,y2,'g',t1,y3,'b',t1,y4,'m'); legend('\alpha = 0.25', '\alpha = 0.5', '\alpha = 1', '\alpha = 2'); xlabel('t/T');ylabel('h(t)');title('高斯脉冲成形滤波器的冲激响应 h(t)'); subplot(3,1,3),plot(t1,z1,'r',t1,z2,'g',t1,z3,'b',t1,z4,'m'); xlabel('t/T');ylabel('g(t)'); legend('\alpha = 0.25, ', '\alpha = 0.5', '\alpha = 1', '\alpha = 2'); axis([-1.5 1.5 0 1.2]);title('高斯脉冲成形滤波器的输出');

由图 3-18 可以看出,矩形脉冲通过高斯脉冲成形滤波器后变成了高斯脉冲,有效地 改变了矩形波突变的上升沿和下降沿。

