

Matlab与信号处理

周治国

2018. 10

M A T L A B G U I P R O G R A M M I N G

MATLAB GUI

程序设计

王广 邢林芳 编著
Wang Guang Xing Linfang

资深作者编著，图书质量更有保证

资深MATLAB工程师执笔，系统总结GUI设计经验

提供配套源码，便于读者动手实践

理论联系实际，提供源代码下载，方便读者学习使用

内含丰富实例，利于读者二次开发

提供大量GUI编程设计实例，读者可以据此二次开发



源代码

清华大学出版社

第8章

数字滤波器设计-III

第11章 GUI实现滤波器设计

随着信息时代和数字世界的到来，数字信号处理已成为当今一门极其重要的学科和技术领域。数字信号处理在通信、语音、图像、自动控制、雷达、军事、航空航天、医疗和家用电器等众多领域得到了广泛的应用。在数字信号处理中，数字滤波器占有极其重要的地位。

现在数字滤波器可以用软件和设计专用的数字处理硬件两种方式来实现，用软件来实现数字滤波器优点是随着滤波器参数的改变，很容易改变滤波器的性能。根据数字滤波器单脉冲响应的时域特性可将数字滤波器分为两种，即IIR（**infinite impulse response**）无限长脉冲响应数字滤波器和FIR（**finite impulse response**）有限长脉冲响应数字滤波器。从功能上分类，可分为低通、高通、带通、带阻滤波器。

学习目标:

- (1) 了解IIR数字滤波器基本原理;
- (2) 了解FIR数字滤波器基本原理;
- (3) 学会基于GUI的数字滤波器设计。

11.1.2 IIR滤波器设计编程实现

【例 11-1】 设计 IIR 的 Butterworth 低通滤波器，其 $F_s=22050\text{Hz}$ ， $F_{y1}=3400\text{Hz}$ 。

程序如下：

```
Fs1 = 5000Hz, Rp = 2dB, Rs = 20dB
Fs = 22050; Fp1 = 3400; Fs1 = 5000; Rp = 3; Rs = 20;           % 设计指标
wpl = 2 * Fp1 / Fs; wsl = 2 * Fs1 / Fs;                       % 求归一化频率
% 确定 butterworth 的最小阶数 N 和频率参数 Wn
[n, Wn] = buttord(wpl, wsl, Rp, Rs);
[B, A] = butter(N, Wn);                                       % 确定传递函数的分子、分母系数
[h, f] = freqz(b, a, Nn, Fs_value);                          % 生成频率响应参数
plot(f, 20 * log(abs(h)))                                     % 画幅频响应图
plot(f, angle(h));                                          % 画相频响应图

% [N, Wn] = buttord(Wp, Ws, Rp, Rs) 确定 butterworth 的 N 和 Wn
% [N, Wn] = cheblord ( (Wp, Ws, Rp, Rs) 确定 Chebyshev 滤波器的 N 和 Wn
% [N, Wn] = cheb2ord (Wp, Ws, Rp, Rs) 确定 Chebyshev2 滤波器的 N 和 Wn
% [N, Wn] = ellipord (Wp, Ws, Rp, Rs) 确定椭圆(Ellipse)滤波器的 N 和 Wn
% [B, A] = butter(N, Wn, 'type') 设计 'type' 型巴特沃斯(Butterworth)滤波器 filter.
% [B, A] = cheby1 (N, R, Wn, 'type') 设计 'type' 型切比雪夫 I 滤波器 filter.
% [B, A] = cheby2 (N, R, Wn, 'type') 设计 'type' 型切比雪夫 II 滤波器 filter.
% [B, A] = ellip(N, Rp, Rs, Wn, 'type') 设计 'type' 型椭圆 filter.
```

程序效果图如图11-1所示。

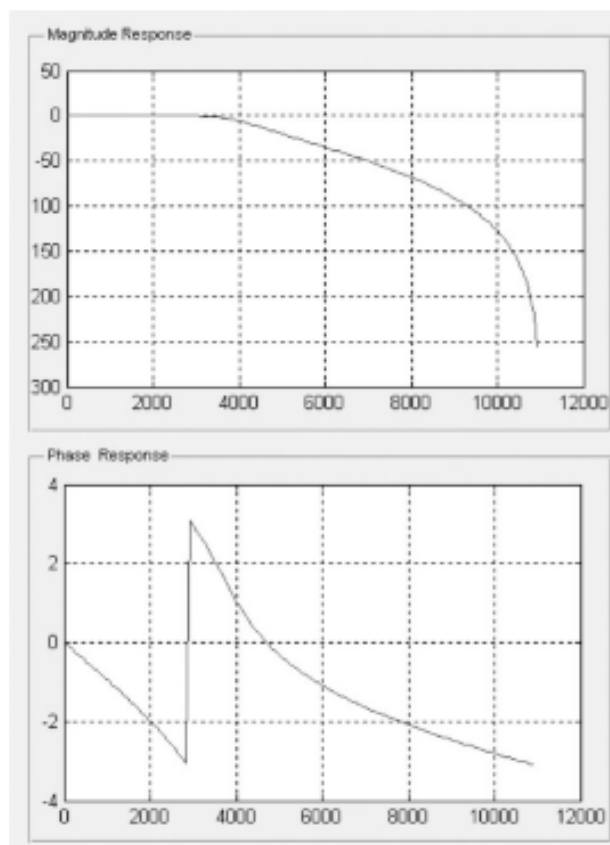


图11-1 Butterworth低通滤波器

11.2.2 FIR滤波器设计编程实现

【例11-2】 设计FIR的Blackman窗的低通滤波器，其中 $F_s=22050\text{Hz}$ ， $F_{p1}=3400\text{Hz}$ ， $F_{s1}=5000\text{Hz}$ ， $R_p=2\text{dB}$ ， $R_s=20\text{dB}$ 。

程序命令如下：

```
Fs = 22050; Fp1 = 3400; Fs1 = 5000; Rp = 3; Rs = 20; n = 75; % 设计指标
wpl = 2 * Fp1 / Fs; ws1 = 2 * Fs1 / Fs; % 求归一化频率
% 确定的最小阶数 N 和频率参数 Wn
[n, Wn] = buttord(wpl, ws1, Rp, Rs);
[b, a] = butter(N, Wn); % 确定传递函数的分子、分母系数
w = blackman(n + 1); % 确定窗口值
% w = boxcar(n + 1);
% w = bartlett(n + 1);
% w = triang(n + 1);
% w = hanning(n + 1);
% w = hamming(n + 1);
b = fir1(n, wn, w); % 确定传递函数的分母系数
[h, f] = freqz(b, 1); % 生成频率响应参数
plot(f, 20 * log(abs(h))) % 画幅频响应图
plot(f, angle(h)); % 画相频响应图
```

运行程序效果图如图11-2所示。

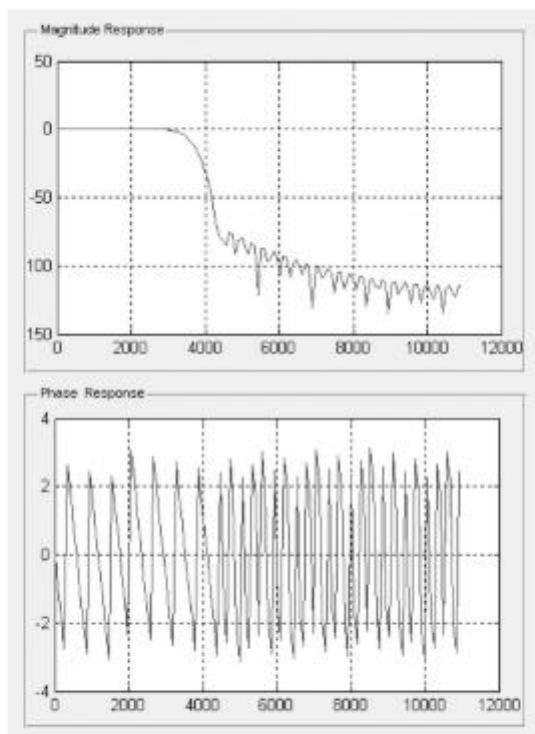


图11-2 Blackman窗低通滤波器

11.3 基于GUI的数字滤波器设计与实现

“滤波器设计”可以实现如下功能：

- (1) 能够实现人机交互；
- (2) 在下拉Digital Filter菜单里可以选择IIR或FIR滤波器设计；
- (3) 当选择IIR滤波器时，能够选择巴特沃斯（Butterworth）、切比雪夫 I（Chebyshev I）滤波器、切比雪夫 II（Chebyshev II）滤波器、椭圆（Ellipse）滤波器；
- (4) 当选择FIR滤波器时，能够选择Boxcar、Bartlett、Blackman、Hanning、Hamming、Kaiser窗口设计滤波器；
- (5) 在下拉菜单Filter Type中选择滤波器类型，能够选择Lowpass、Highpass、Bandpass、Bandstop四种类型；
- (6) 在下拉菜单Display Type中选择图形显示类型，能够选择Linear、Logarithmic两种类型；
- (7) 在设计滤波器阶数时，可选择自定义阶数和利用最小阶数设计滤波器，并显示最小阶数；
- (8) 在参数输入中，可输入抽样频率 F_s 、滤波器通带临界频率（ F_{p1} 、

Fp2)、滤波器阻带临界频率 (Fst1、Fst2)、通带内的最大衰减Rp、通带内的最小衰减Rs;

(9) 设计的滤波器的相频响应和幅频响应显示在界面中。

MATLAB中的属性控制非常多, 要设置哪些对象的属性, 哪些对象属性可以不设置, 都需具体问题具体分析。接下来再通过控件布置编辑器来设置控件的对齐方式及分布等, 以完善界面功能。按要求设计好的“滤波器设计”主面板共包括4个区域:

(1) 图形区: 用于显示各模块的仿真曲线;

(2) 参数设区: 由一个静态文本框和一个编辑框以及类型选择按钮组成, 实时地进行系统参数的设定和滤波器原型的选定;

(3) 对象模型区: 由下拉菜单选定数字滤波器类型, 当用户的输入参数发生变化时, 可通过单击响应的Run按钮, 实现设计结果的实时刷新与显示;

(4) 数据显示区: 对应于图形显示结果, 实时显示滤波器阶次和分子分母多项式系数。

11.3.1 “滤波器设计” 界面设计

在MATLAB命令行窗口中输入guide，打开GUIDE界面，然后拖入所要的图形控件，按需要修改外观和空间属性，直至满足要求，如图11-3所示。单击GUIDE界面上方的Run按钮，会生成一个FIG文件，一个M文件，其中FIG文件就是界面的图形，M文件是界面的回调函数，在M文件里每个控件的回调函数都已经自动生成，控件要做的工作就是在文件框架下定义某些特殊要求的状态并补充完整回调函数，单击控件时激活回调程序完成一定的功能。

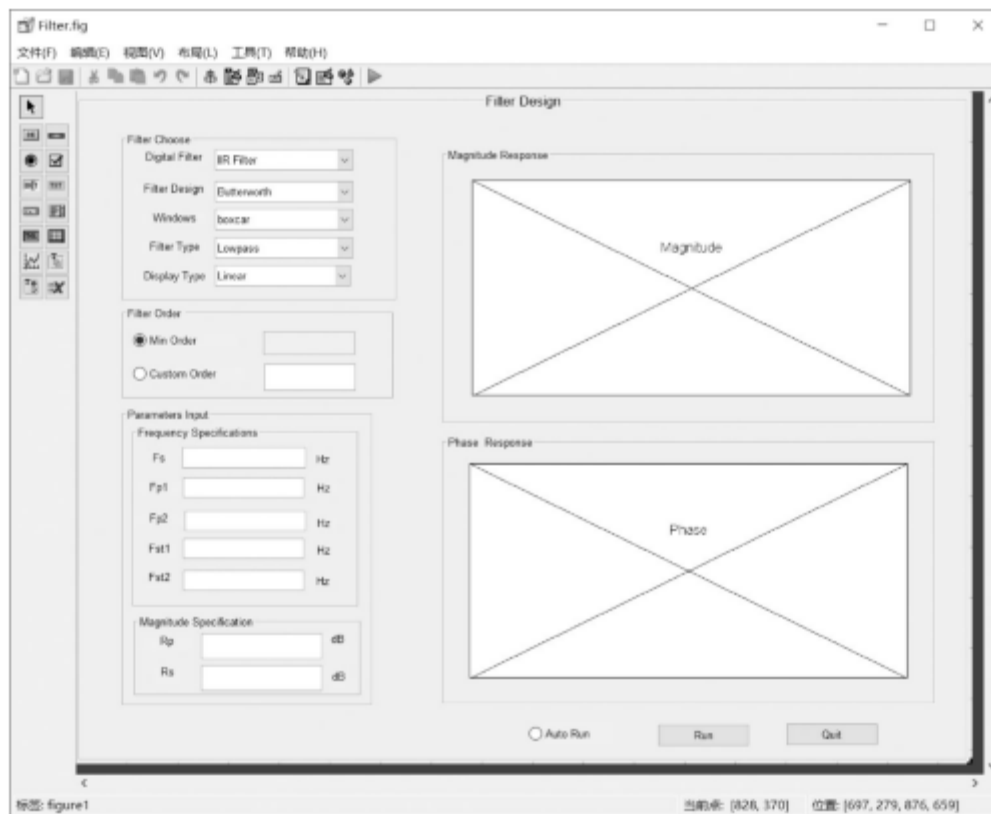


图11-3 GUI界面设计

11.3.2 “滤波器设计”回调函数

1. Digital Filter下拉菜单

在下拉Digital Filter菜单里可以选择IIR或FIR滤波器设计。选择IIR或者FIR屏蔽相应的选项，若选择IIR选项时，则FIR的Windows的选择窗口不能使用，程序指令如下：

```
function DigitalFilter_Callback(hObject, eventdata, handles)
% 读取此时选择的滤波器选择, "IIR、FIR"
DigitalFilter_value = get(handles.DigitalFilter, 'Value');
    % 当选择了 IIR 时, 使窗口选项屏蔽
    if(DigitalFilter_value == 1)
        set(handles.FilterDesign, 'enable', 'on');
        set(handles.Windows, 'enable', 'off')
    % 当选择了 FIR 时, 使 IIR 的滤波器 (Butterworth、Chebyshev I 等) 选项屏蔽
    else
        set(handles.FilterDesign, 'enable', 'off');
        set(handles.Windows, 'enable', 'on')
    end
```

相应的程序效果图如图11-4和图11-5所示。



图11-4 IIR滤波器选择



图11-5 FIR滤波器选择

2. Filter Type下拉菜单

在下拉菜单**Filter Type**中选择滤波器类型。选择低通或者高通滤波器时，隐藏相应的带通和带阻的第二个临界频率，使其频率参数不能输入，程序指令如下：

```

function FilterType_Callback(hObject, eventdata, handles)
% 读取此时设计的滤波器的类型, "Lowpass、Highpass、Bandpass、Bandstop"
FilterType_value = get(handles.FilterType, 'Value');
% 当选择 Lowpass 或者 Highpass, 屏蔽相应的临界频率的显示
if((FilterType_value == 1) || (FilterType_value == 2))
    set(handles.Fp2, 'visible', 'off'); set(handles.Fs2, 'visible', 'off');
    set(handles.text17, 'visible', 'off'); set(handles.text19, 'visible', 'off');
    set(handles.text6, 'visible', 'off'); set(handles.text10, 'visible', 'off');
else
% 当选择 Bandpass 或者 Bandstop, 使相应的临界频率的显示, 使参数能够输入
    if((FilterType_value == 3) || (FilterType_value == 4))
set(handles.Fp2, 'visible', 'on'); set(handles.Fs2, 'visible', 'on');
        set(handles.text17, 'visible', 'on'); set(handles.text19, 'visible', 'on');
set(handles.text6, 'visible', 'on'); set(handles.text10, 'visible', 'on');
    end
end
end

```

相应的程序效果图如图11-6和图11-7所示。

Parameters Input

Frequency Specifications

Fs Hz

Fp1 Hz

Fp2 Hz

Fst1 Hz

Fst2 Hz

图11-6 带通和带阻滤波器频率参数输入

Parameters Input

Frequency Specifications

Fs Hz

Fp1 Hz

Fst1 Hz

图11-7 低通和高通滤波器频率参数输入

3. Min Order和Custom Order按钮

在设计滤波器阶数时，可选择自定义阶数和利用最小阶数设计滤波器，并显示最小阶数，这里涉及的控件有Min Order和Custom Order，其回调函数如下：

1) 按钮Min Order

选择使用最小阶数设置时，屏蔽自定义阶数的输入。

```
function MinOrder_Callback(hObject, eventdata, handles)
MinOrder_value = get(handles.MinOrder, 'Value');
    if(MinOrder_value == 1)
        set(handles.Order, 'visible', 'off');
    else
        set(handles.Order, 'visible', 'on');
    end
```

相应的程序效果图如图11-8所示。

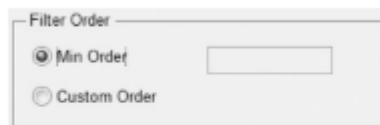


图11-8 使用最小阶数设计Filter

2) 按钮Custom Order

选择自定义阶数时，显示阶数输入框。

```
function CustomOrderButton_Callback(hObject, eventdata, handles)
    CustomOrderButton_value = get(handles.CustomOrderButton, 'Value');
    if(CustomOrderButton_value == 0)
        set(handles.Order, 'visible', 'off');
    else
        set(handles.Order, 'visible', 'on');
    end
end
```

相应的程序效果图如图11-9所示。

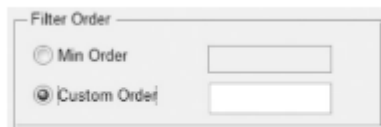


图11-9 使用自定义阶数设计Filter

3) 其中涉及Run控件中的程序如下:

显示最小阶数:

```
set(handles.MinOrderDisplay, 'string', num2str(n))
```

当选择了自定义阶数时, 读取自定义阶数:

```
if(MinOrder_value == 0)
    n = str2double(get(handles.Order, 'String'))
end
```

4. Fp1、Fp2、Fst1、Fst2文本编辑框

参数输入时，在输入抽样频率 F_s 的前提下，判断滤波器通带临界频率（Fp1、Fp2）、滤波器阻带临界频率（Fst1、Fst2）的归一化频率 w_{p1} 、 w_{p2} 、 w_{s1} 、 w_{s2} 是否在 $[0, 1]$ ，如不正确则显示错误对话框。Fp2、Fst1、Fst2的回调函数同理，程序如下：

```
function Fp1_Callback(hObject, eventdata, handles)
% 检查输入的通带临界频率 Fp1 的归一化频率是否在[0,1]之间
Fs_value = str2double(get(handles.Fs, 'String'));
Fp1_value = str2double(get(handles.Fp1, 'String'));
wp1 = 2 * Fp1_value / Fs_value;
% 如果不在[0,1]之间,显示输入错误对话框
if(wp1 >= 1)
    errordlg('wp1 = 2 * Fp1/Fs, 归一化频率不在【0,1】之间, 请输入正确的参数', '错误信息')
end
```

相应的程序效果图如图11-10所示。



图11-10 输入参数错误信息框显示

5. Run按钮

根据输入的参数显示设计的滤波器的幅频特性和相频特性图，其程序如下：

```
function Run_Callback(hObject, eventdata, handles)
% 点击 Run 立即运行 AutoChoose.m 文件, 实现滤波器设计程序
AutoChoose(handles)
```

6. Auto Run按钮

当选择了Auto Run按钮时，能够根据所选的IIR的滤波器（Butterworth、Chebyshev I 等）、FIR的窗口选项、图形显示类型（Linear、Logarithmic）立即显示滤波器的幅频特性和相频特性图，其中IIR的滤波器（Butterworth、Chebyshev I 等）的回调函数程序如下：

```
function FilterDesign_Callback(hObject, eventdata, handles)
AutoRun_value = get(handles.AutoRun, 'Value');
% 当选择了 Filter(Butterworth、Chebyshev I 等) 中一种 Filter 运行 AutoChoose.m 文件
if(AutoRun_value == 1)
    AutoChoose(handles)
end
```

7. Quit按钮

退出滤波器设计窗口，其程序如下：

```
function Quit_Callback(hObject, eventdata, handles)
%单击 Quit 按钮退出
Close
```

11.3.3 AutoChoose.m程序的编写

AutoChoose.m程序的编写如下：

11.3.4 运行和结果显示

在图形界面下，按“运行”按钮进入界面后，选择要设计的滤波器选项，编辑框中输入要求设计的数字滤波器的性能指标，选择Run命令按钮，出现模拟低通原型滤波器幅频响应和相频响应曲线最小阶次。当选择了Auto Run按钮，在“滤波器类型选择”旁的下拉菜单框中选择其他类型的滤波器会立刻出现此类型的滤波器的图形。

【例11-3】 IIR的Butterworth低通滤波器，其中 $F_s=1000\text{Hz}$ ， $F_{p1}=100\text{Hz}$ ， $F_{s1}=300\text{Hz}$ ， $R_p=3\text{dB}$ ， $R_s=20\text{dB}$ ， $n=4$ ，运行结果如图11-11和图11-12所示。

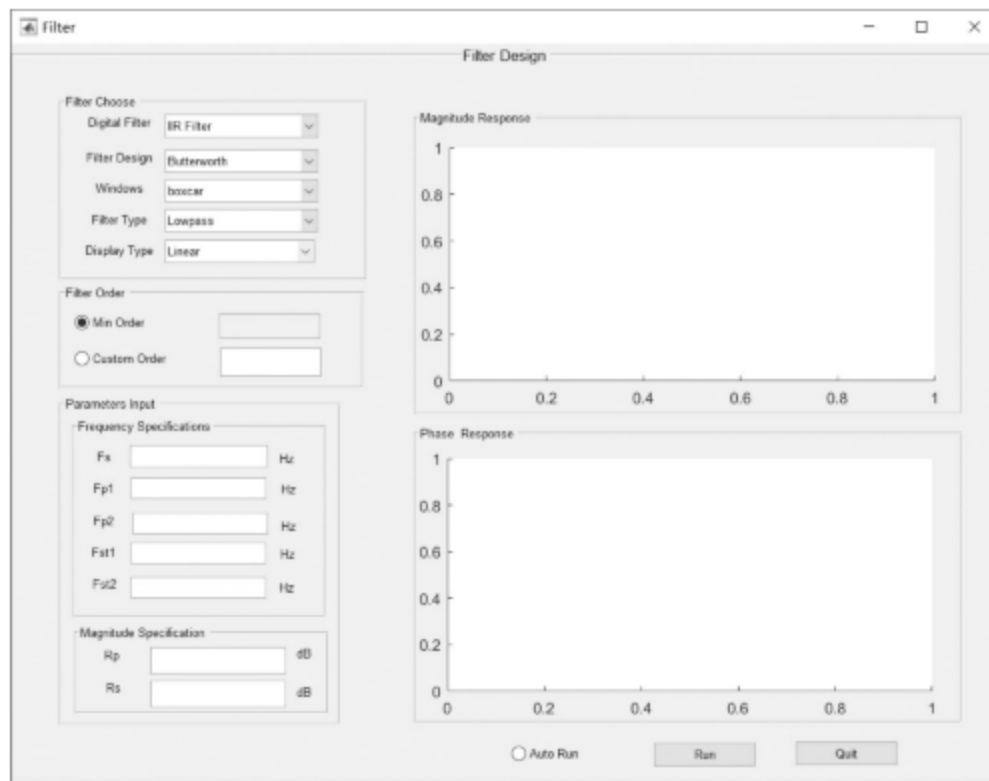


图11-11 运行显示界面

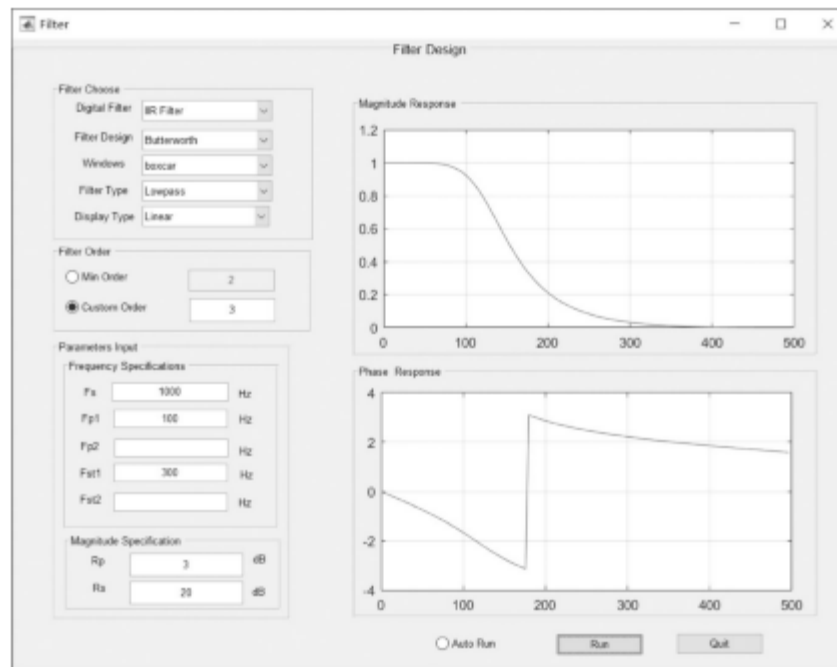


图11-12 Butterworth低通滤波器

【例 11-4】 设计 IIR 低通滤波器，其中 $F_s=1000\text{Hz}$ ， $F_{p1}=100\text{Hz}$ ， $F_{s1}=300\text{Hz}$ ， $R_p=3\text{dB}$ ， $U_s=20\text{dB}$ ， $n=4$ ，利用四种不同滤波器进行设计。

设计运行结果分别如图 11-13、图 11-14、图 11-15 和图 11-16 所示。

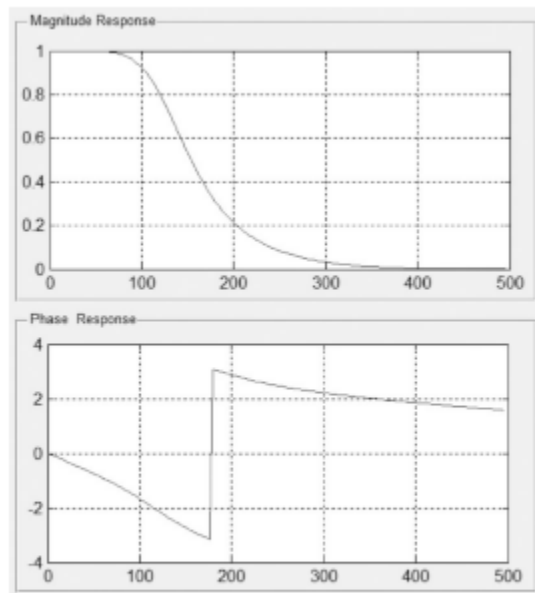


图 11-13 Butterworth 低通滤波器

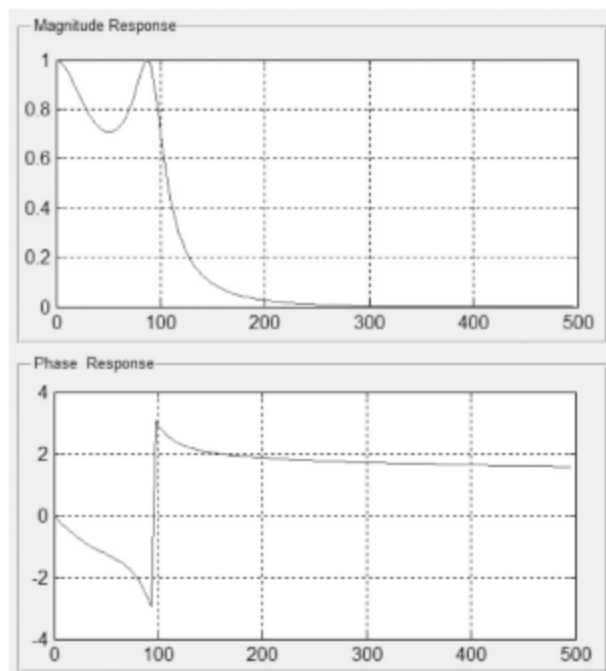


图11-14 Chebyshev I 低通滤波器

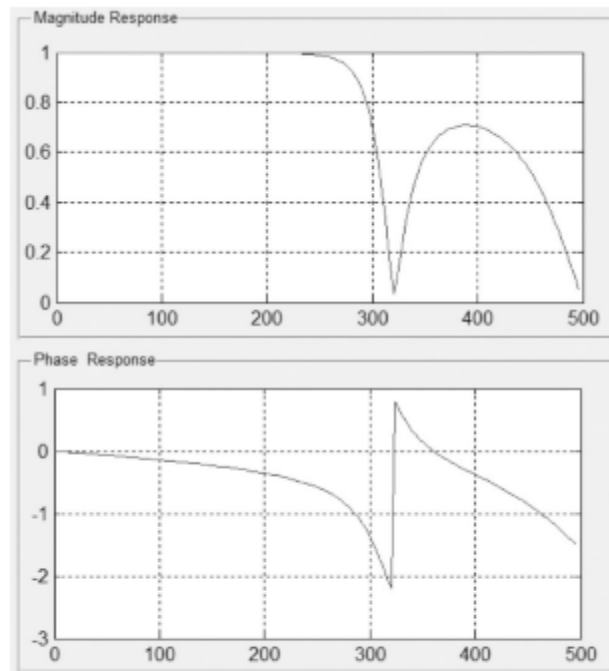


图11-15 Chebyshev II 低通滤波器

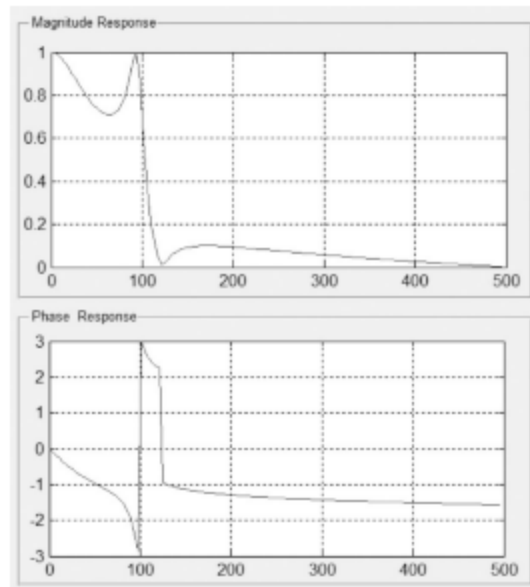


图11-16 Ellipse低通滤波器

【例 11-5】 设计 FIR 带通滤波器，其 $F_s=4000\text{Hz}$ ， $F_{p1}=900\text{Hz}$ ， $F_{p2}=1300\text{Hz}$ ， $F_{s1}=600\text{Hz}$ ， $F_{s2}=1500\text{Hz}$ ， $R_p=1\text{dB}$ ， $U_s=40\text{dB}$ ， $n=20$ ， $R_p=3\text{dB}$ ， $U_s=20\text{dB}$ ， $n=4$ ，利用四种不同滤波器进行设计。

设计运行结果分别如图 11-17、图 11-18、图 11-19、图 11-20、图 11-21、图 11-22 所示。

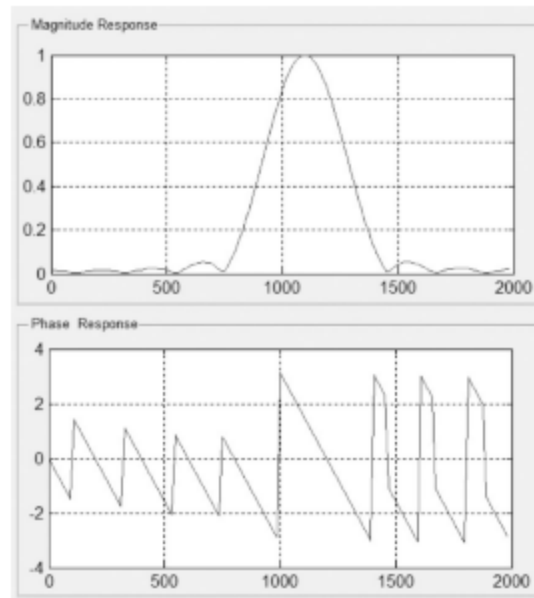


图11-17 Boxar窗带通滤波器

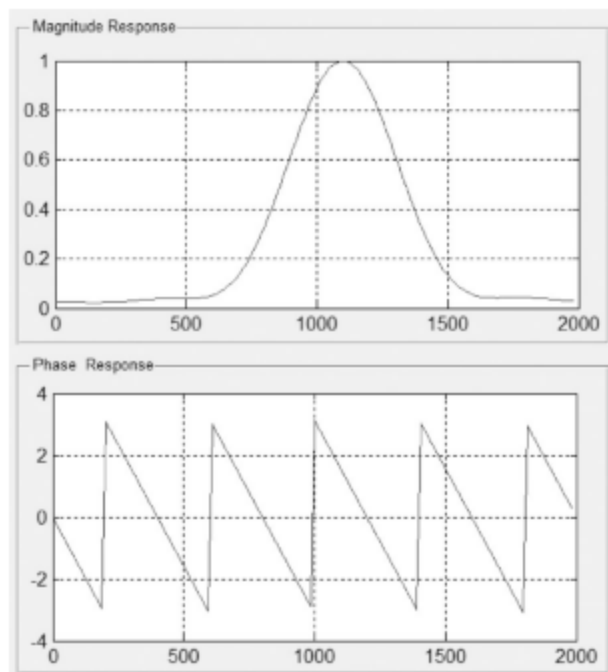


图11-18 Bartlett窗带通滤波器

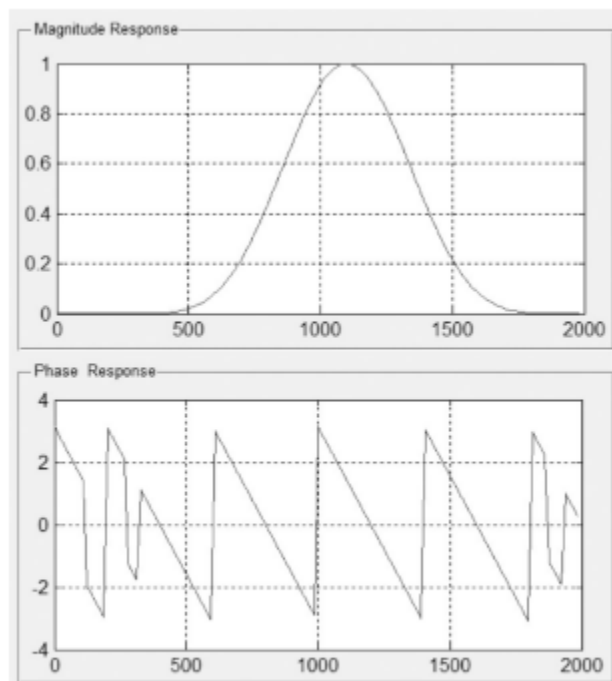


图11-19 Blackman窗带通滤波器

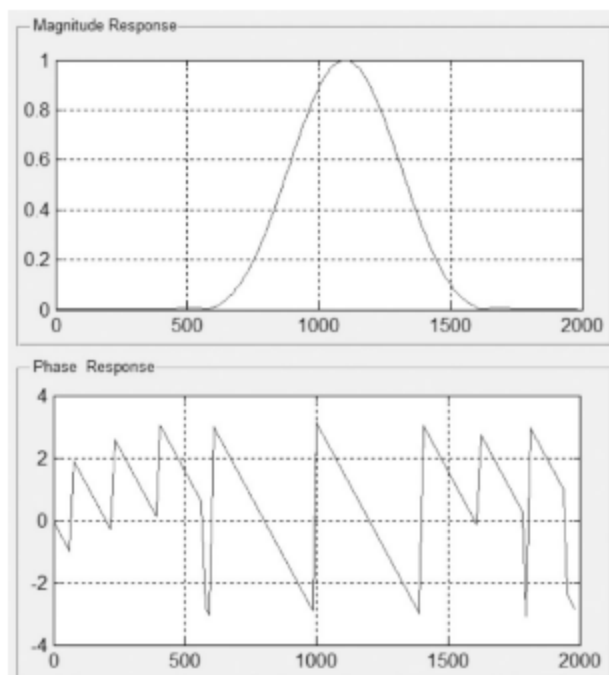


图11-20 Hanning窗带通滤波器

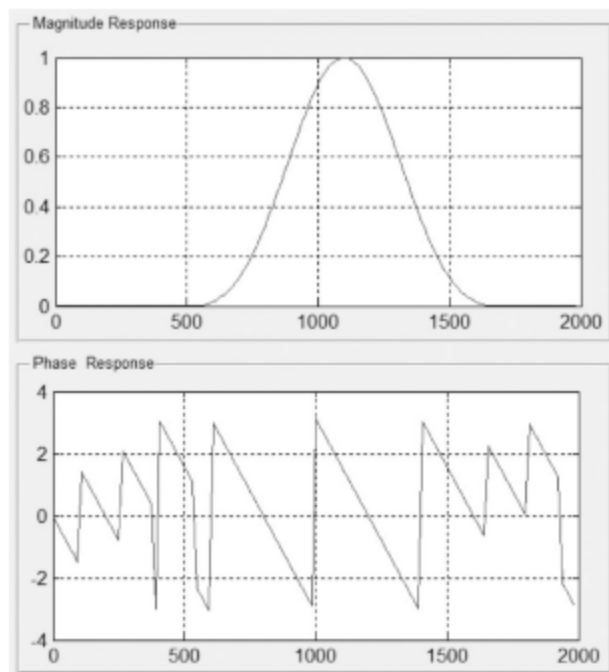


图11-21 Hamming窗带通滤波器

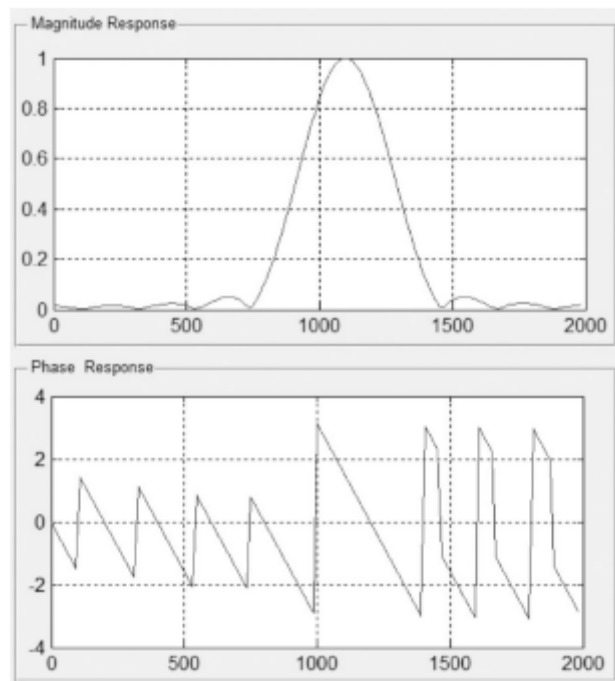


图11-22 Kaiser窗带通滤波器

本章小结

GUI是实现人机交互的中介，具有强大的功能，可以完成许多复杂的程序模块。使用它需要具有一定的知识储备和必要的经验技巧。需要了解函数句柄等必要基础知识，熟悉各控件对象的基本属性和方法操作，知晓不同控件的合适使用条件及其特有的功能，并会采用不同的使用手段来实现相同功能的设计。同时还需要详细掌握菜单和控件。菜单很简单，就是弄清楚菜单之间的关系和如何调用就可以。控件的使用主要是用好 `CreateFcn` 和 `Callback` 属性。`CreateFcn` 中的语句就是在程序运行时，就立即执行脚本。如果希望界面可控，那么最好用 `Callback` 属性。在相应控件下，添加相应的脚本就可以实现比较复杂计算绘图等功能。

在设计GUI的时候，要注意一定的原则和步骤，分析界面所要求实现的主要功能，明确设计任务，构思草图，设计界面和属性，编写对象的相应代码，实现控件的交互调用。

对于GUI在数字信号处理中的应用中，数字信号处理这门学科的知识是基础，要掌握数字信号处理的相关知识的原理后，并用代码来实现，才能很好地结合MATLAB进行GUI编程。

目录

□FDATool

□SPTool