大连理工大学实验报告

学院（系）：电子信息与电气工程学部 专业：电子信息工程(英语强化) 班级：电英 1801

姓　　名：童博涵 学号：201883032 组：

实验时间： 实验室： 实验台：

指导教师签字： 成绩：

实验二 语音信号的调制解调

一、实验题目和结果

1. 信号的调制与解调

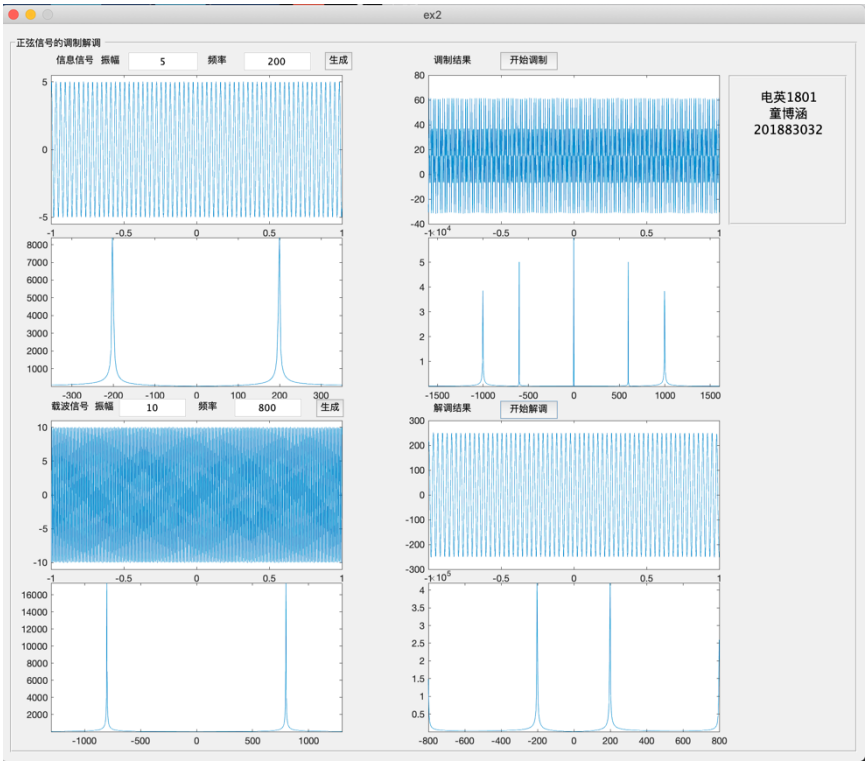

图 1. 调制与解调结果

2. 实验代码

```
function varargout = ex2(varargin)
% EX2 MATLAB code for ex2.fig
%      EX2, by itself, creates a new EX2 or raises the existing
%      singleton*.
%
```

```
%      H = EX2 returns the handle to a new EX2 or the handle to
%      the existing singleton*.
%
%      EX2('CALLBACK',hObject,eventData,handles,...) calls the local
%      function named CALLBACK in EX2.M with the given input arguments.
%
%      EX2('Property','Value',...) creates a new EX2 or raises the
%      existing singleton*.  Starting from the left, property value pairs are
%      applied to the GUI before ex2_OpeningFcn gets called.  An
%      unrecognized property name or invalid value makes property application
%      stop.  All inputs are passed to ex2_OpeningFcn via varargin.
%
%      *See GUI Options on GUIDE's Tools menu.  Choose "GUI allows only one
%      instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES

% Edit the above text to modify the response to help ex2

% Last Modified by GUIDE v2.5 27-May-2020 15:54:47

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn', @ex2_OpeningFcn, ...
                   'gui_OutputFcn',  @ex2_OutputFcn, ...
                   'gui_LayoutFcn',  [] , ...
                   'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT


% --- Executes just before ex2 is made visible.
function ex2_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
```

```
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to ex2 (see VARARGIN)

% Choose default command line output for ex2
handles.output = hObject;
handles.w = 0;
handles.w1 = 0;
handles.y = 0;
handles.y1 = 0;
handles.ym = 0;
% Update handles structure
guidata(hObject, handles);

% UIWAIT makes ex2 wait for user response (see UIRESUME)
% uiwait(handles.figure1);


% --- Outputs from this function are returned to the command line.
function varargout = ex2_OutputFcn(hObject, eventdata, handles)
% varargout  cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;



function edit_sig1_Callback(hObject, eventdata, handles)
% hObject    handle to edit_sig1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit_sig1 as text
%        str2double(get(hObject,'String')) returns contents of edit_sig1 as a double


% --- Executes during object creation, after setting all properties.
function edit_sig1_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit_sig1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called
```

```
% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end



function edit_sig2_Callback(hObject, eventdata, handles)
% hObject    handle to edit_sig2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit_sig2 as text
%        str2double(get(hObject,'String')) returns contents of edit_sig2 as a double


% --- Executes during object creation, after setting all properties.
function edit_sig2_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit_sig2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end



function edit_freq1_Callback(hObject, eventdata, handles)
% hObject    handle to edit_freq1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit_freq1 as text
%        str2double(get(hObject,'String')) returns contents of edit_freq1 as a double


% --- Executes during object creation, after setting all properties.
function edit_freq1_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit_freq1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called
```

```
% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end



function edit_freq2_Callback(hObject, eventdata, handles)
% hObject    handle to edit_freq2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit_freq2 as text
%        str2double(get(hObject,'String')) returns contents of edit_freq2 as a double


% --- Executes during object creation, after setting all properties.
function edit_freq2_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit_freq2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end



% --- Executes on button press in start_modulate.
function start_modulate_Callback(hObject, eventdata, handles)
% hObject    handle to start_modulate (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% 开始调制
fs = 2000;
t = -1:1/fs:1;
A0 = max(handles.y1)*1.5;
A_mat = repmat(A0,1,length(handles.y));%直流信号
handles.ym = A_mat+handles.y.*handles.y1;%调制

plot(handles.sig_show3,t,handles.ym)
f_ym = fft(handles.ym);
```

```
N = length(t);
fm = (0:N-1)*fs/N-fs/2;

plot(handles.freq_show3,fm*2*pi,fftshift(abs(f_ym)))
axis(handles.freq_show3,[-2*handles.w1 2*handles.w1 -inf inf])
guidata(hObject, handles);


% --- Executes on button press in start_demodulate.
function start_demodulate_Callback(hObject, eventdata, handles)
% hObject    handle to start_demodulate (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% 开始解调
fs = 2000;
t = -1:1/fs:1;
p = handles.y1.*handles.ym;
FL = handles.w/(2*pi);%获得调制信号频率
FH = handles.w1/(2*pi);%获得载波信号频率
fL = FL+20;%设置通带截止频率
fH = fL+25;%设置阻带起始频率
Wp = fL/(fs/2);
Ws = fH/(fs/2);
Rp = 3;
Rs = 20;
[N,Wn] = buttord(Wp,Ws,Rp,Rs);
[B,A] = butter(N,Wn,'low');
m0 = filtfilt(B,A,p);
plot(handles.sig_show4,t,m0)
fm0 = fft(p);
len_N = length(t);
f = (0:len_N-1)*fs/len_N-fs/2;
plot(handles.freq_show4,2*pi*f,fftshift(abs(fm0)))
axis(handles.freq_show4,[-handles.w1 handles.w1 -inf inf])
guidata(hObject, handles);

if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end


% --- Executes on button press in gen_1.
function gen_1_Callback(hObject, eventdata, handles)
% hObject    handle to gen_1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
```

```
% handles    structure with handles and user data (see GUIDATA)
% 生成调制信号
fs = 2000;
t = -1:1/fs:1;
A1 = get(handles.edit_sig1,'String');
A1 = str2double(A1);
handles.w = get(handles.edit_freq1,'String');
handles.w = str2double(handles.w);
handles.y = A1*sin(t*handles.w);
fy = fft(handles.y);
N = length(t);
f = (0:N-1)*fs/N-fs/2;
plot(handles.sig_show1,t,handles.y)
axis(handles.sig_show1,[-inf inf -1.1*A1 1.1*A1])
plot(handles.freq_show1,f*2*pi,fftshift(abs(fy)))
axis(handles.freq_show1,[-handles.w-150 handles.w+150 -inf inf])
guidata(hObject, handles);

% --- Executes on button press in gen_2.
function gen_2_Callback(hObject, eventdata, handles)
% hObject    handle to gen_2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% 生成载波
fs = 2000;
t = -1:1/fs:1;
A2 = get(handles.edit_sig2,'String');
A2 = str2double(A2);
handles.w1 = get(handles.edit_freq2,'String');
handles.w1 = str2double(handles.w1);
handles.y1 = A2*sin(t*handles.w1);
fy = fft(handles.y1);
N = length(t);
f = (0:N-1)*fs/N-fs/2;
plot(handles.sig_show2,t,handles.y1)
axis(handles.sig_show2,[-inf inf -1.1*A2 1.1*A2])
plot(handles.freq_show2,f*2*pi,fftshift(abs(fy)))
axis(handles.freq_show2,[-handles.w1-500 handles.w1+500 -inf inf])
guidata(hObject, handles);
```

3. 使用 simulink 设计幅度调制和相干解调系统

3.1 主界面设计

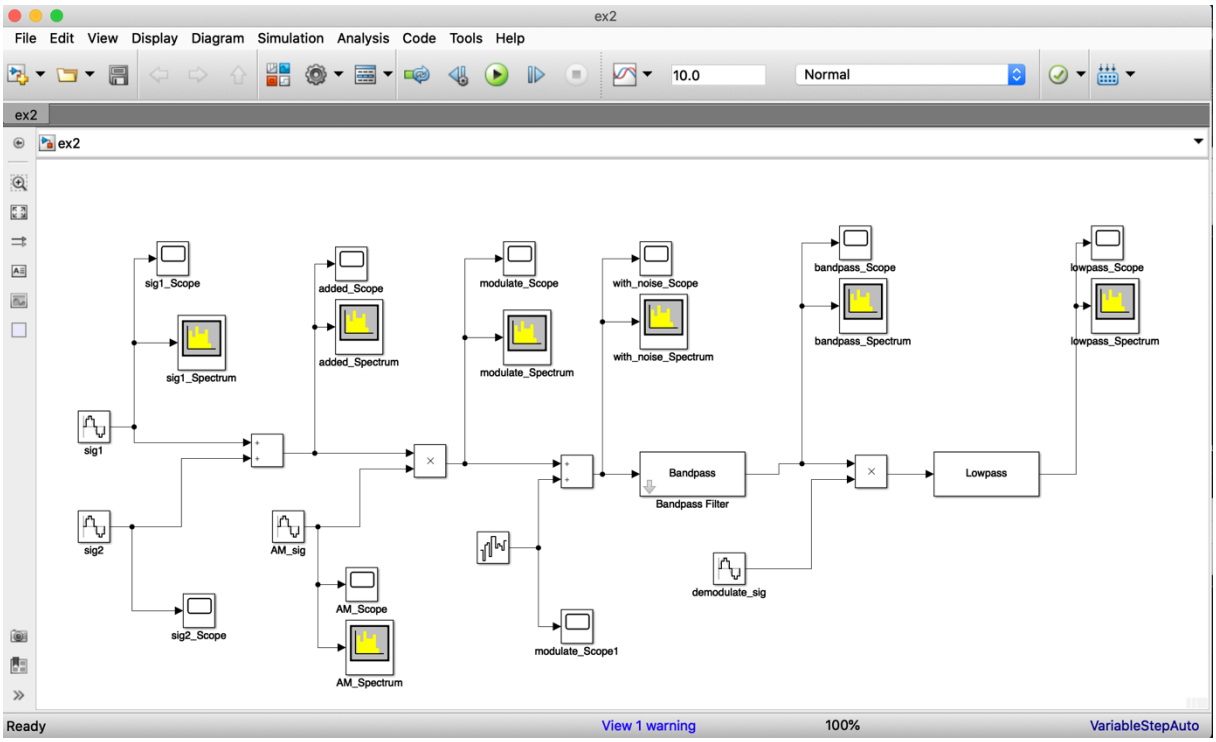

图 2. 系统主界面

3.2 输入和调制信号设置

如图 3 所示，输入信号为频率为 80rad/s，振幅为 2 的正弦信号，与幅度为 1 的直流信号相加后得到调制信号。图 4 为载波信号，频率为 500rad/s，振幅为 100。系统的抽样频率为 2000Hz。

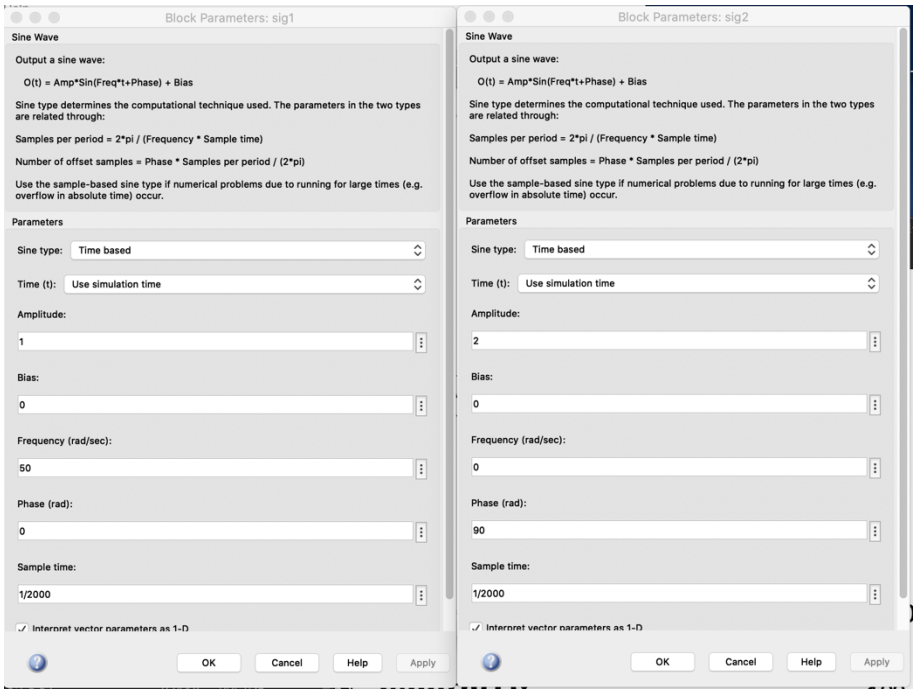

图 3.输入信号参数

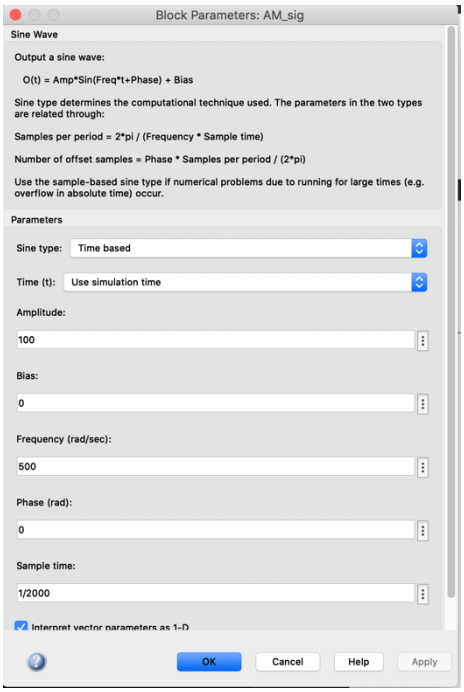

图 4. 调制信号

3.3 带通滤波器设计

图 5 和图 6 分别给出了起滤除噪声作用的带通滤波器的参数和幅频响应曲线。

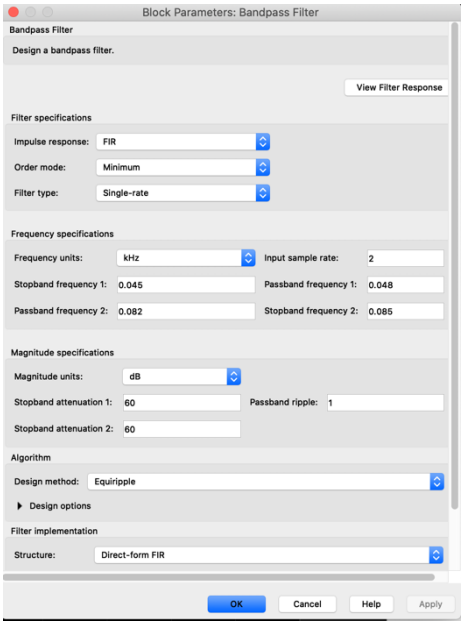

图 5. 带通滤波器参数

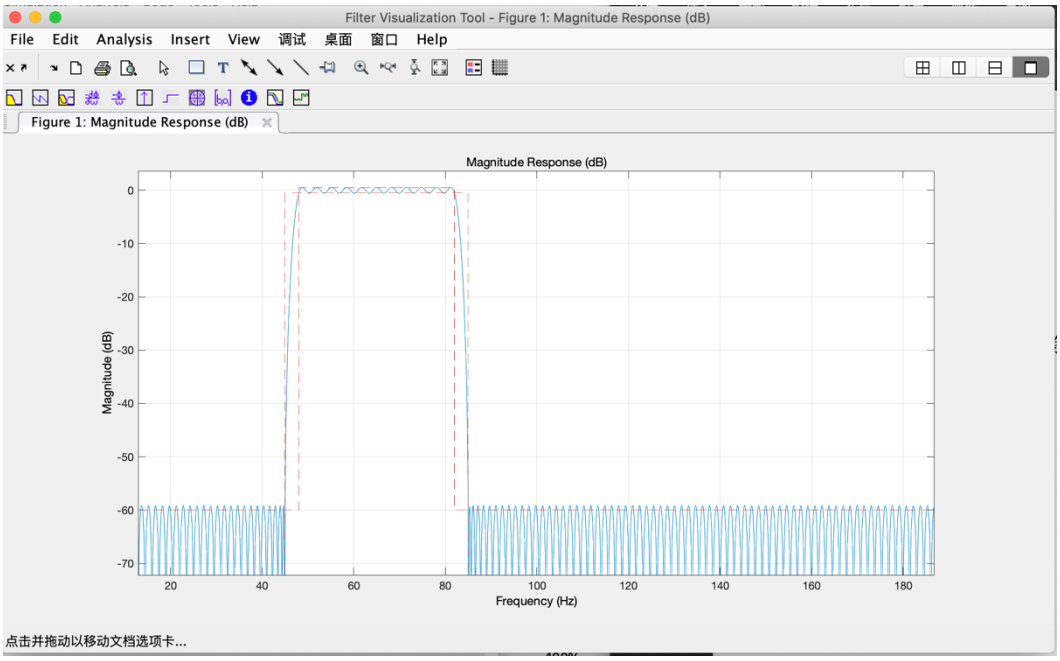

图 6.带通滤波器幅频响应曲线

3.4 低通滤波器设计

图 5 和图 6 分别给出了起滤除噪声作用的低通滤波器的参数和幅频响应曲线。

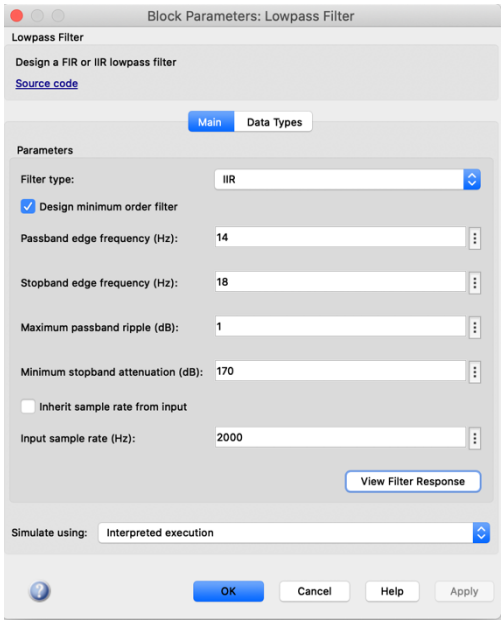

图 7.低通滤波器参数

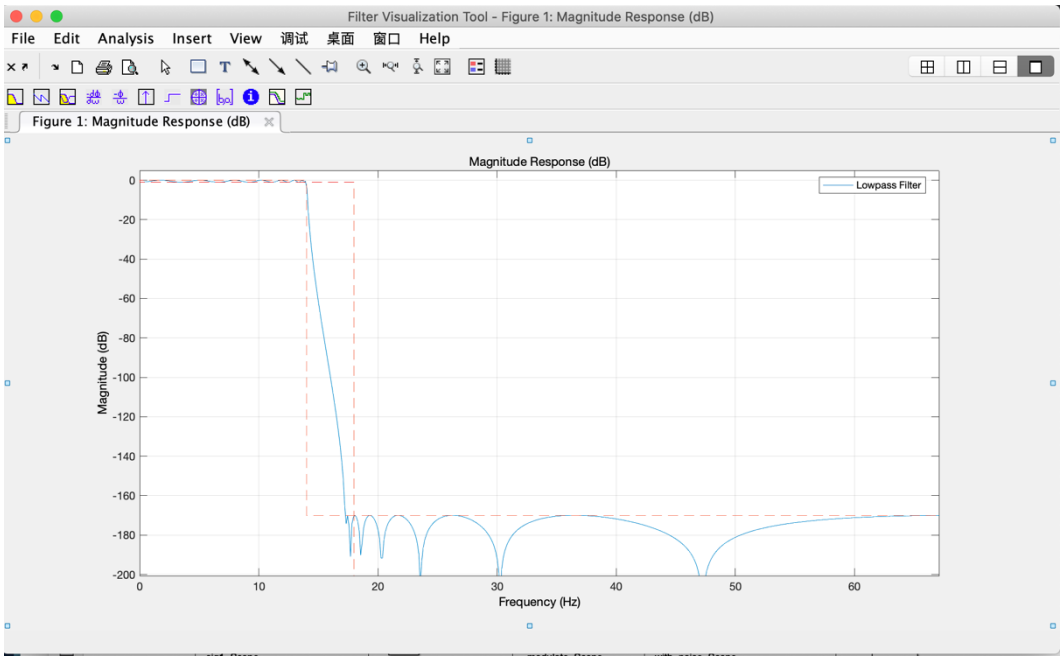

图 8.低通滤波器幅频响应

3.5 仿真结果

图 9、图 10、图 11、图 12 分别为调制信号、调制后加入噪声的信号、通过带通滤波器后信号和解调后信号的幅频图。图 13 为调制解调前后信号时域图像的对比。

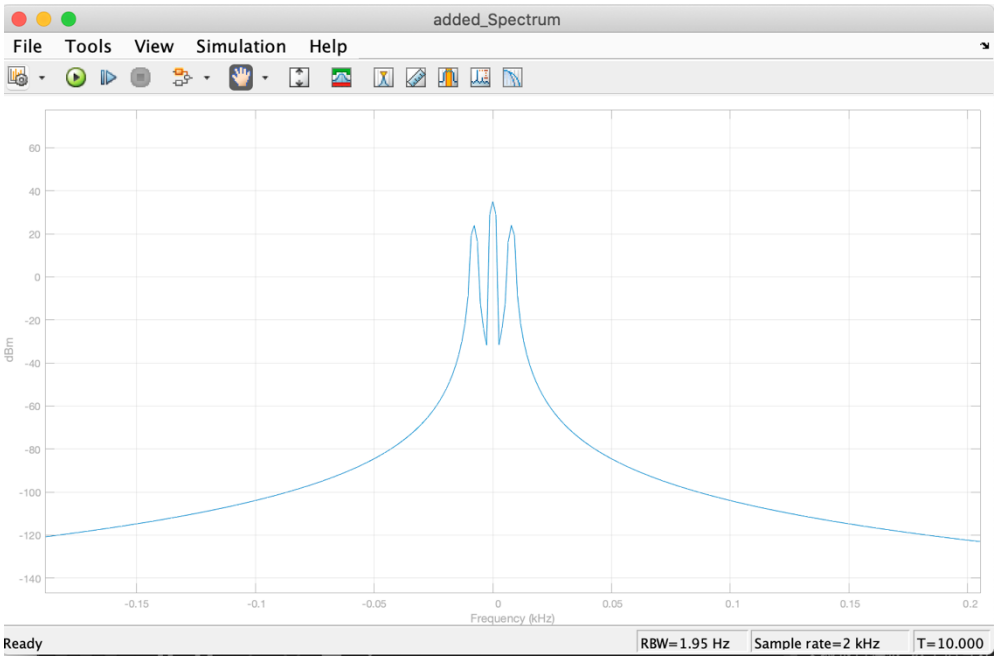

图 9.调制信号

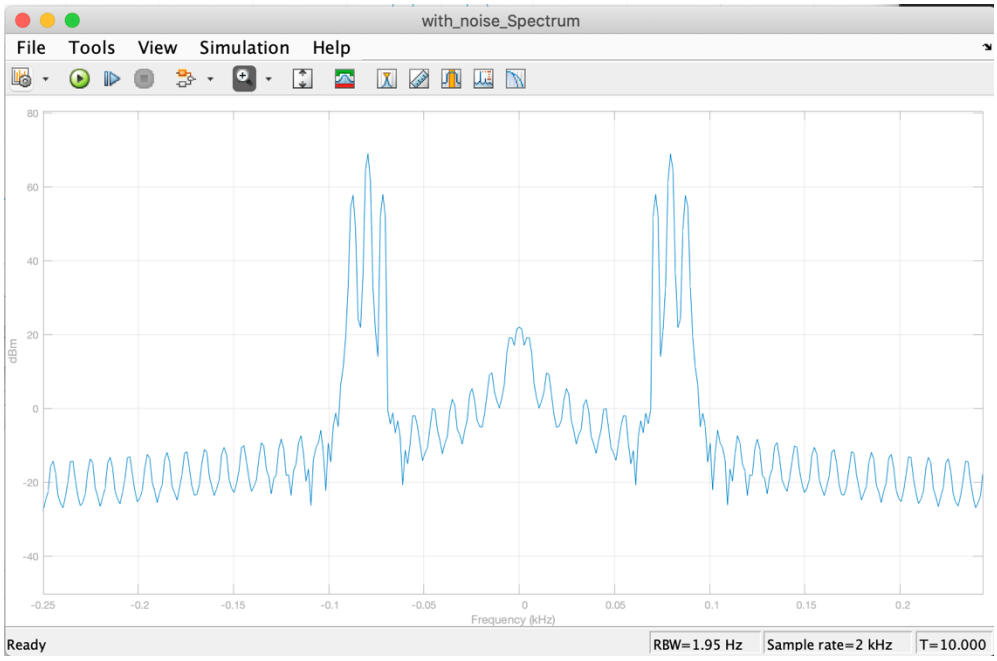

图 10.调制后加入噪声

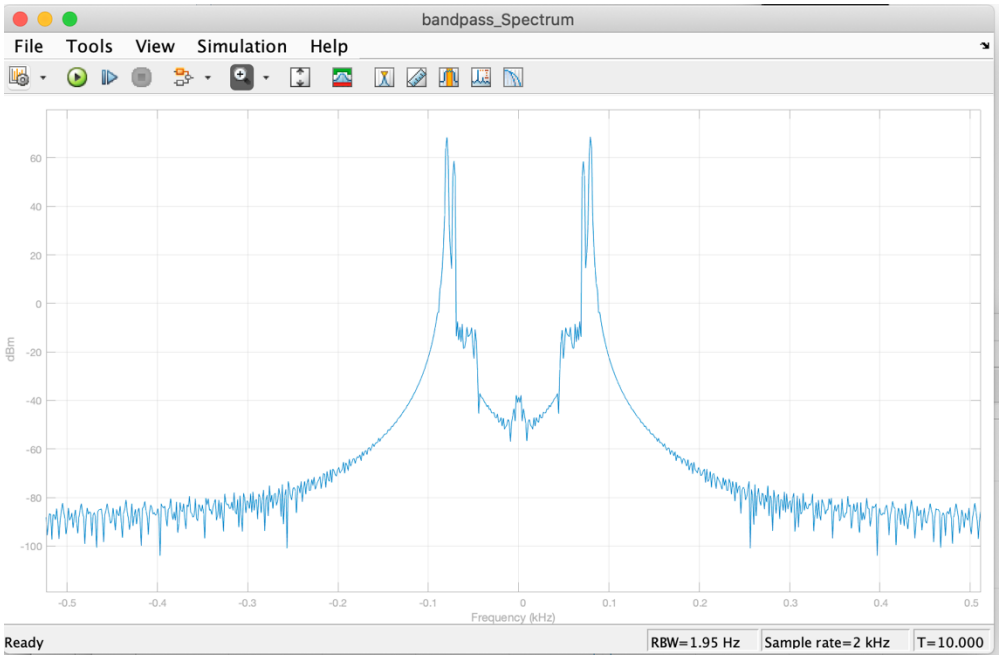

图 11.通过带通滤波器

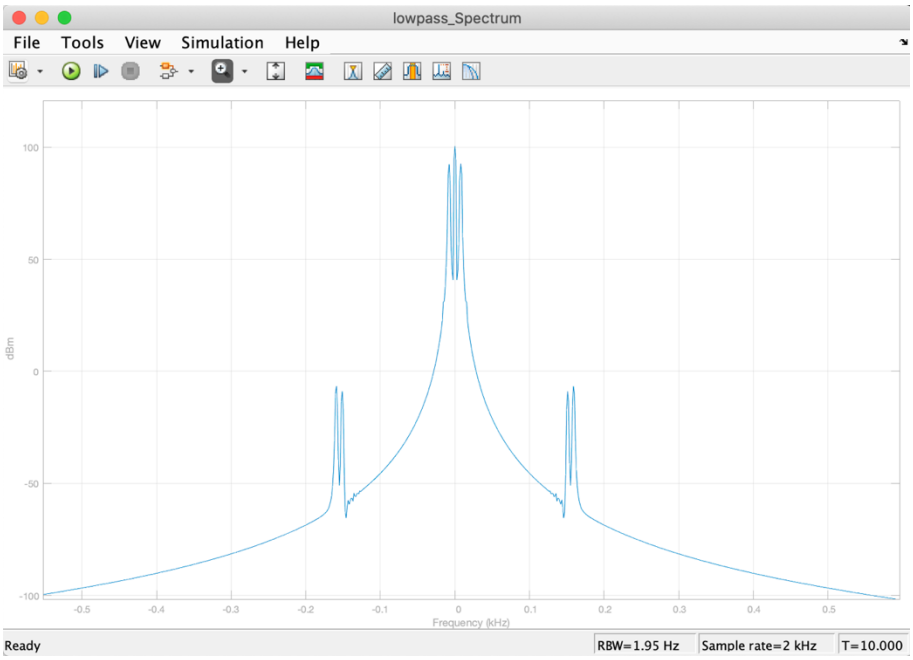

图 12.解调后信号

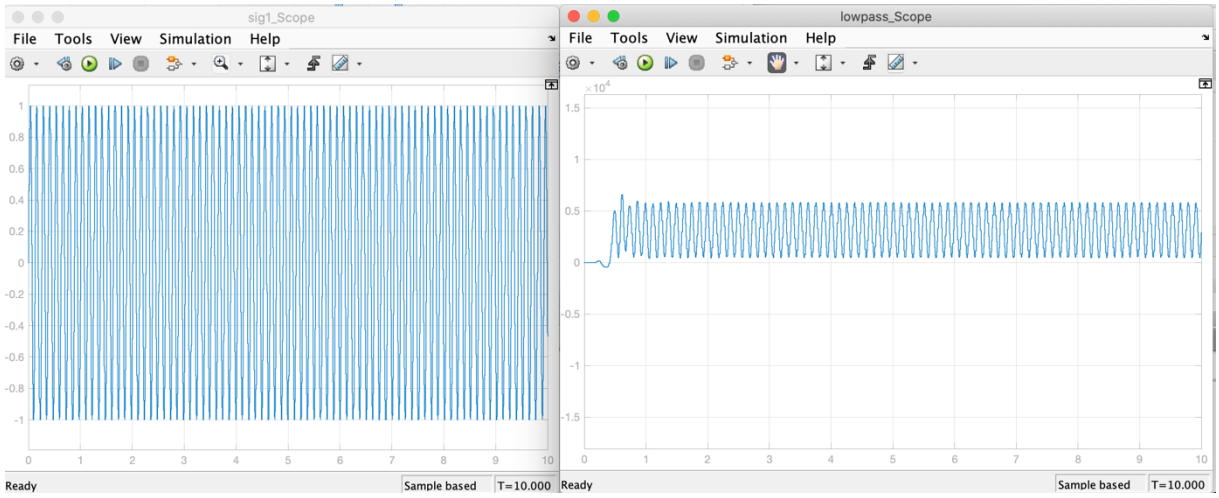

图 13.处理前后信号对比

二、实验总结

1. 在任务 1 中，设计低通滤波器时，低通滤波器的通带截止频率 fL 和通带起始频率 fH 应较为接近，使得滤波器的过渡带更小，否则无法滤除噪声、载波信号等不需要的信号。
2. 在任务 2 中，若需生成直流信号，可将正弦信号生成器的相位设为 90，振幅为信号高度，频率为 0。
3. 在设计滤除噪声的带通滤波器时，选择信号频率后，输入的参数为归一化后的频率，需要将频率转换成归一化后的频率作为参数输入。
4. 任务 2 中设计低通滤波器时，需要将滤波器的类型更改为 IIR 型，否则解调之后的信号持续时间较短。