باسمه تعالی

دانشگاه صنعتی شریف

دانشکده مهندسی برق

درس سیستم های مخابراتی

گزارش تمرین شمارهٔ یک

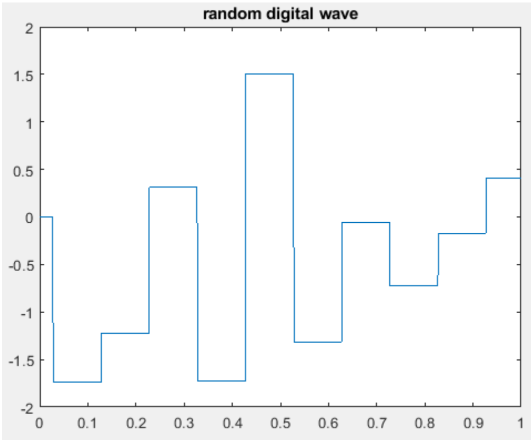
علی محرابیان96102331

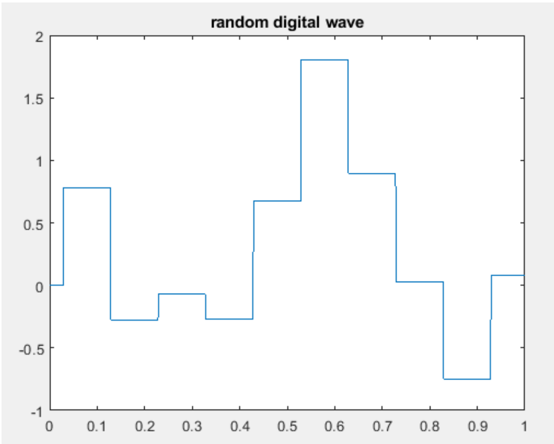
استاد: دکتر بهروزی

پاییز 1398

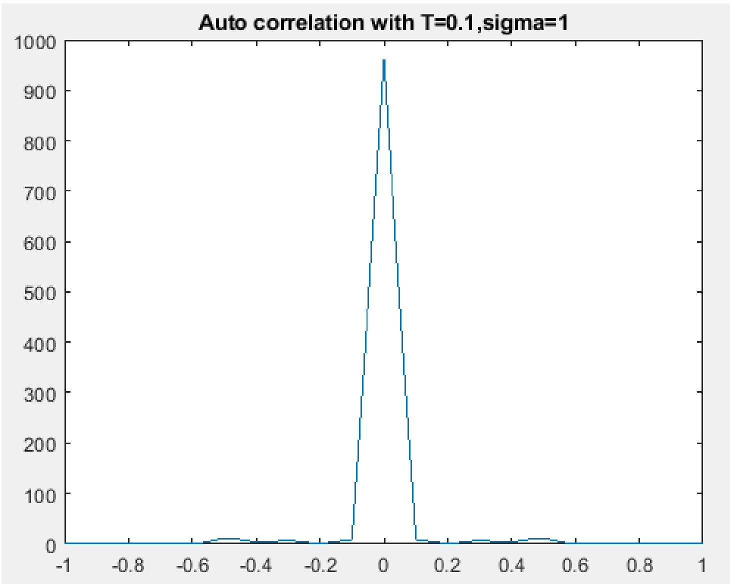
سوال 1:

تابع random_pulse را مطابق خواسته های سوال نوشتیم.دو خروجی از آن به صورت زیر است.

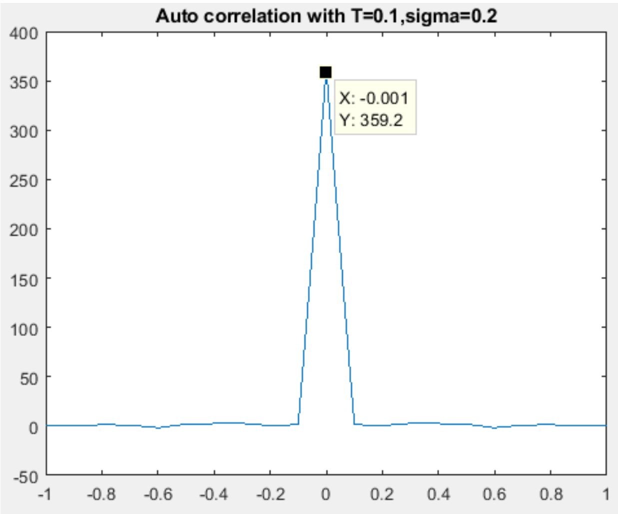


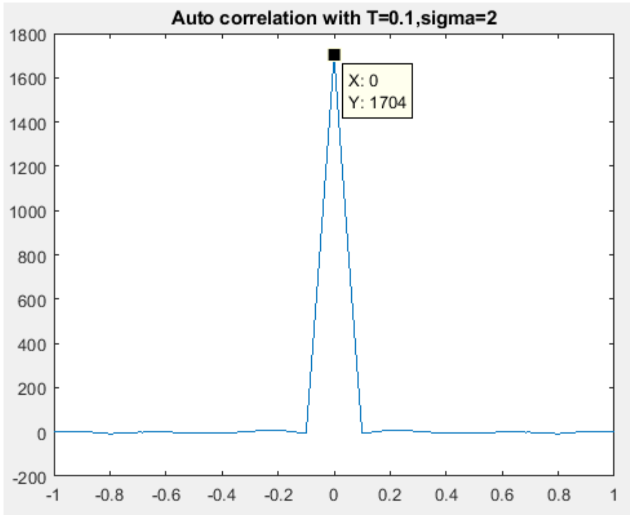


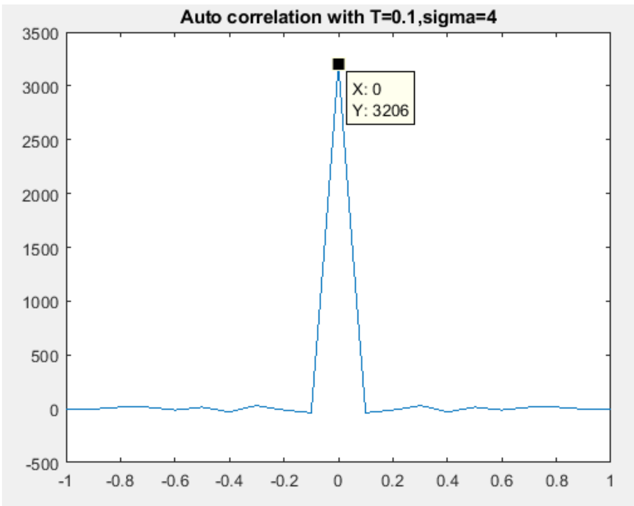
حال به کمک دستور xcorr،خودهمبستگی را به دست می آوریم.2000 نمونه تولید کرده و در نهایت میانگین آنها را می گیریم.



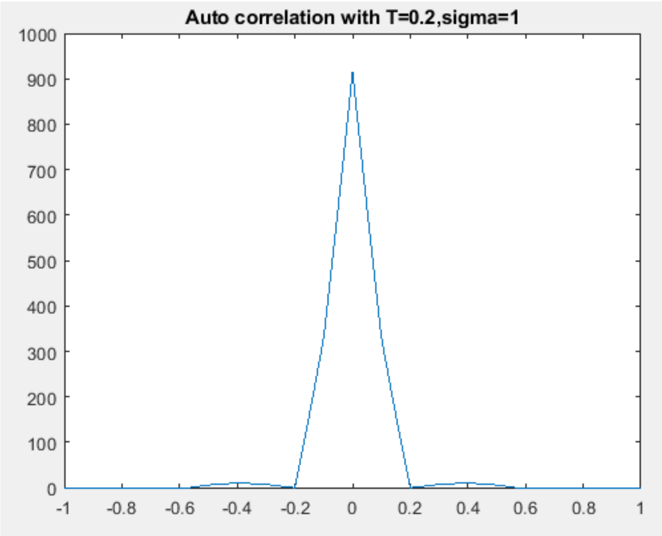
ابتدا تغییرات واریانس را بر تابع خودهمبستگی اعمال می کنیم.

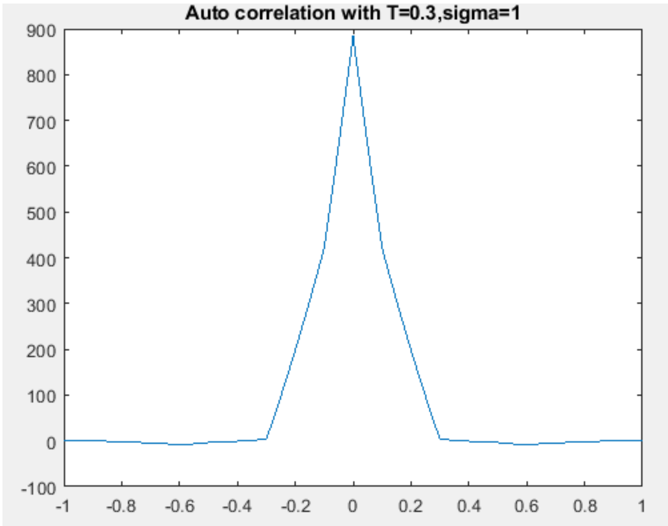






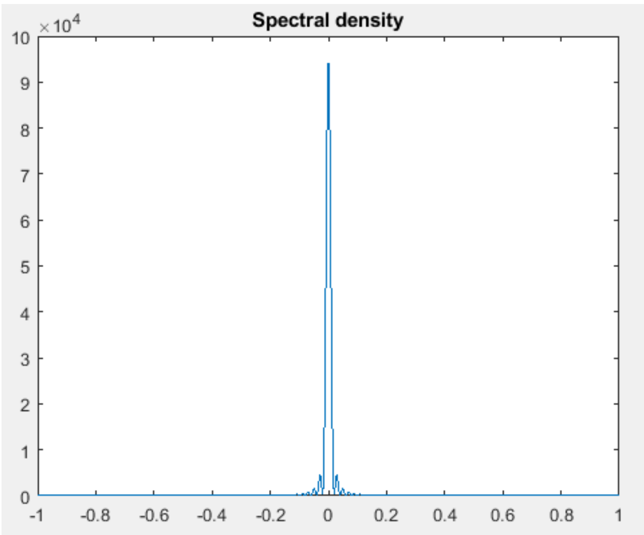
همان طور که مشاهده می شود با افزایش واریانس،دامنه تابع خودهمبستگی افزایش پیدا می کند ولی انحراف معیار خود سیگنال ثابت می ماند.در تمامی این تغییرات،T را ثابت فرض کردیم.حال در قسمت بعد،واریانس را ثابت فرض کرده و T را تغییر می دهیم.



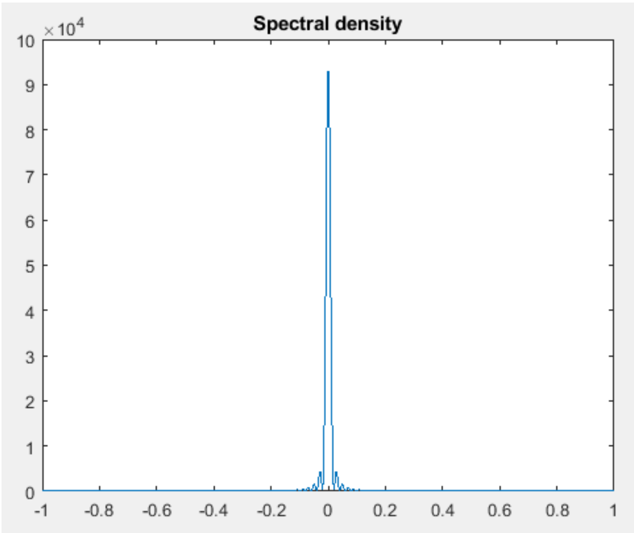


مشاهده می شود که با افزایش T،مقدار دامنه سیگنال مقدار کمی کاهش پیدا کرده و انحراف معیار سیگنال افزایش می یابد.

حال به سراغ رسم تابع چگالی طیف می رویم.از تابع خودهمبستگی به دست آمده تبدیل فوریه گرفته و اندازه آن را در حوزه فرکانس رسم می کنیم.



قضیه wiener-kinchin بیان می‌کند که با تبدیل فوریه گرفتن از تابع خودهمبستگی،به تابع چگالی طیف می رسیم.راه دیگر،این است که برای هر نمونه،توان دوم اندازه تبدیل فوریه را محاسبه می‌کنیم.با توجه به این‌که باید expected value هر ستون را محاسبه کنیم ولی با توجه این‌که توزیع را نداریم،باید مقدار آن را تخمین بزنیم.از تخمین sample mean استفاده می‌کنیم.

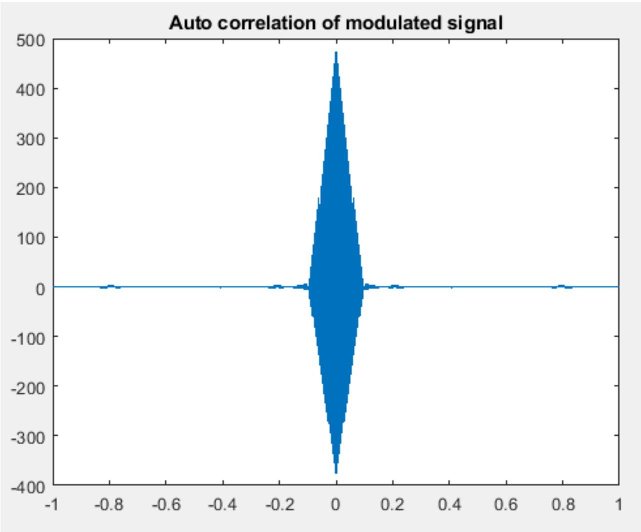


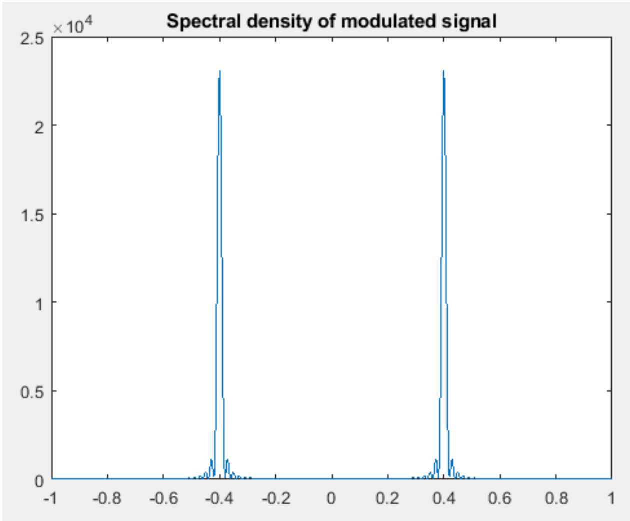
خطای به دست آمده در این حالت برابر با مقدار زیر است که حاکی از برقرار بودن قضیه دارد.

```
MSE =

   2.1152e-22
```

در قسمت آخر پس از یکسان کردن فرکانس نمونه برداری و ضرب سیگنال سینوسی،تابع خودهمبستگی و چگالی طیف را رسم می کنیم.

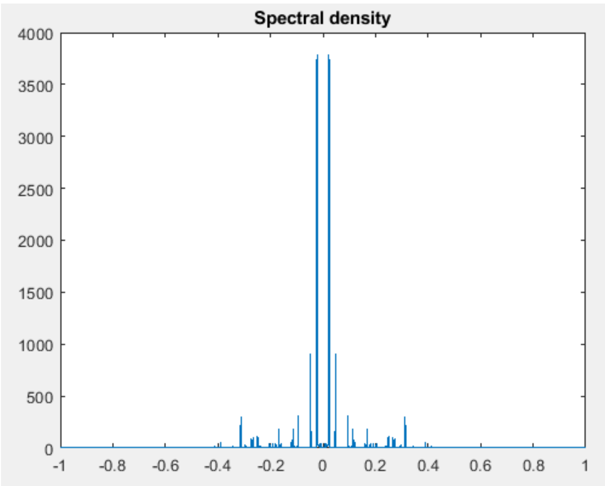




همان طور که مشاهده می شود،دامنه ما مقداری تضعیف شده و محتوای فرکانسی نیز جابه جا شده است.عمل مدولاسیون برای انتقال سیگنال در پهنای باند مشخص و محدودی که داریم ،استفاده می شود که این کار با ضرب سیگنال سینوسی در سیگنال اصلی حاصل می شود.

سوال 2:

فایل را لود کرده و تابع چگالی طیف آن را رسم می کنیم.



مشاهده می شود که سیگنال بیشتر دارای محتوا در فرکانس های پایین است.

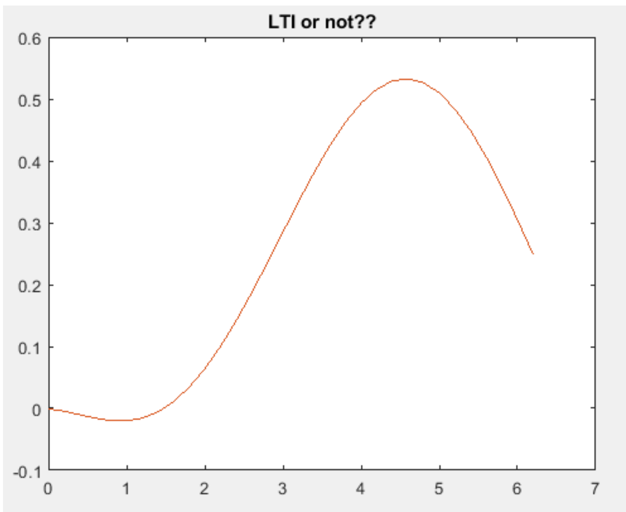
برای سنجیدن این نکته که آیا سیستم LTIهست یا نه،به ازای چند ورودی گوناگون،هرکدام از ویژگی هارا بررسی می کنیم.برای مثال ورودی های زیر را در نظر گرفته و خطی بودن سیستم را می سنجیم.

```
t=[0:0.2:2*pi];

X0=sin(t);
Y0=cos(t);

I1=Channel(2*X0-Y0);
O1=2*Channel(X0)-Channel(Y0);

plot(t,I1);
hold on
plot(t,O1);
title('LTI or not??')
```



همان طور که مشاهده شد،دو خروجی کاملا بر هم منطبق بودند.البته این مثال ادعایی بر اثبات خطی بودن سیستم نیست،ولی با مشاهده چند مورد می توان گفت که با تقریب خوبی سیستم خطی است.

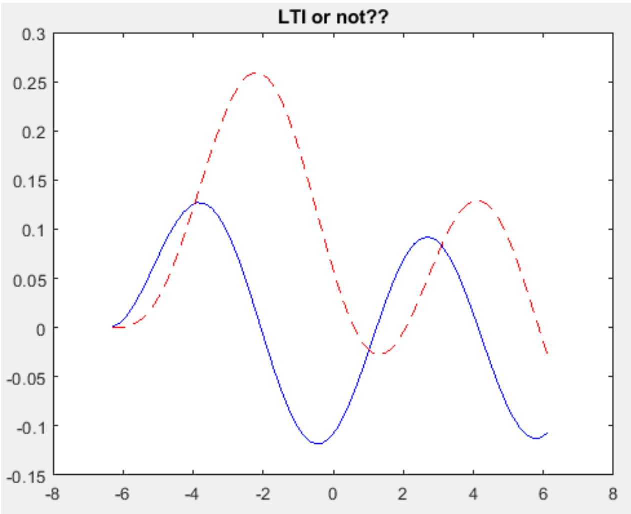
حال تغییر ناپذیر بودن با زمان را بررسی می کنیم.

```
%%
t=-2*pi:0.2:2*pi;

x1=cos(t);
y1=Channel(x1);

x2=cos(t-pi/2);
y2=Channel(x2);

plot(t,y1,'b');
hold on
plot(t,y2,'r--');
title('LTI or not??')
```



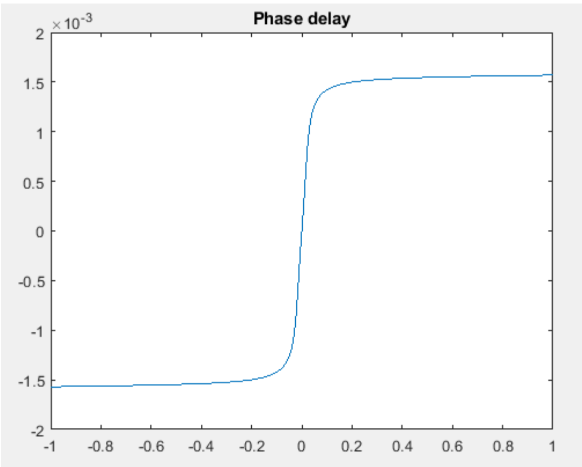
مشاهده می شود که سیگنال خط چین،به مقدار $\pi/2$ شیفت پیدا کرده ولی دامنه آن نیز دچار تغییراتی شده است. پس می توان گفت که سیستم به طور کلی تغییر ناپذیر با زمان نیست ولی با تقریب برای محاسبات قسمت های بعد آن را LTI می گیریم.در گزارش برای هر ویژگی یک مثال ذکر شده ولی در کد،مثال های دیگری را نیز بررسی کردیم.

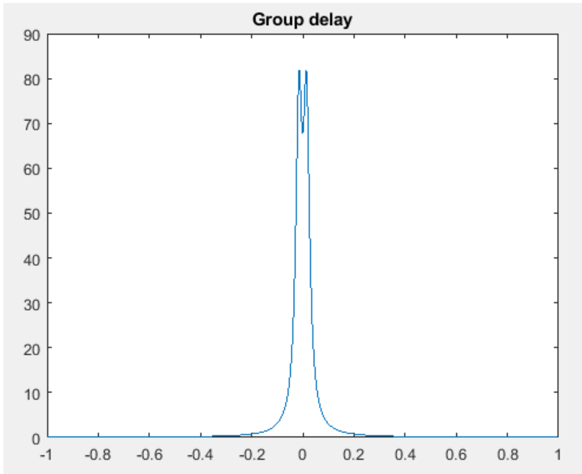
ابتدا پاسخ ضربه سیستم را تعیین می کنیم.یک ورودی ضربه به صورتی در مبدا برابر با یک و در بقیه نقاط برابر با صفر باشد،به سیستم می دهیم.حال به دنبال phase delay و group delay هستیم.

از روابط روبرو کمک می گیریم.

$$\tau_g(\omega) = -\frac{d\phi(\omega)}{d\omega}$$

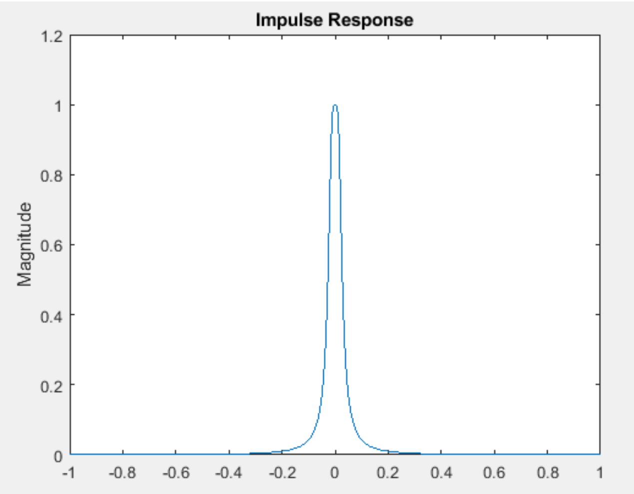
$$\tau_\phi(\omega) = -\frac{\phi(\omega)}{\omega} .$$

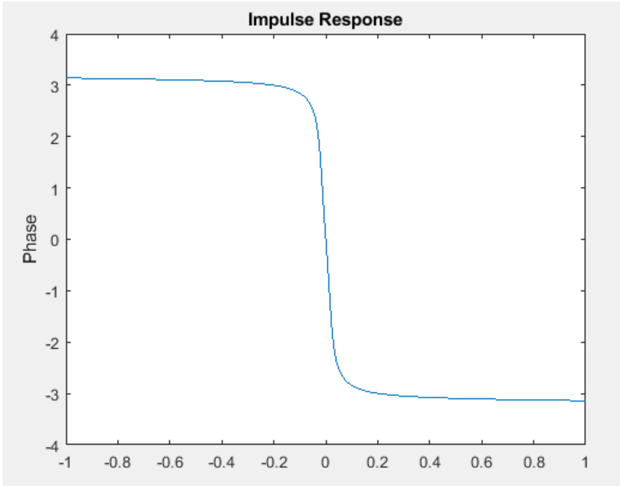




می توان حدس زد که محتوا های فرکانس های پایین شیفت پیدا می کند ولی برای فرکانس های بالا این اتفاق نمی افتد که این امر موجب اعوجاج می شود.

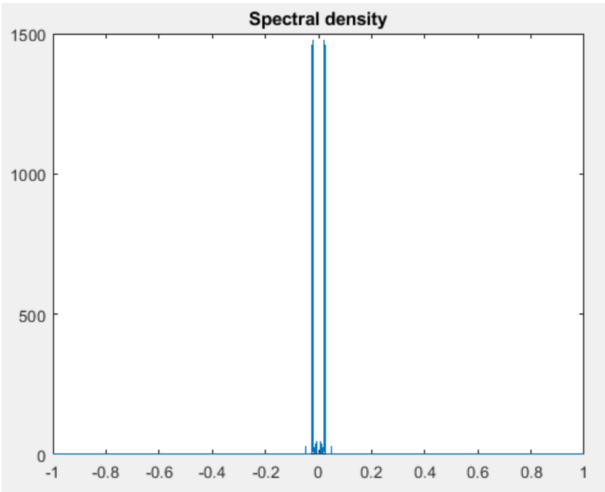
محتوای فرکانسی پاسخ ضربه را رسم می کنیم.





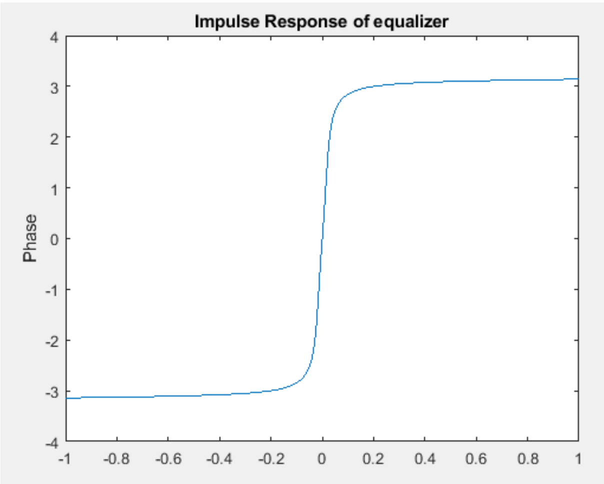
مشاهده می شود که فرکانس های بالا تضعیف می شوند که خود باعث اعوجاج می شود.

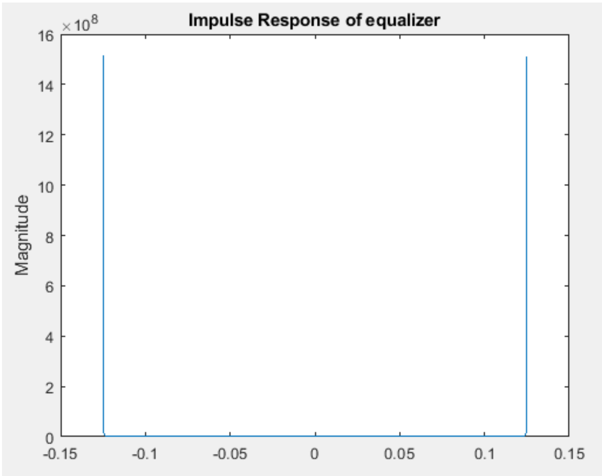
طیف چگالی سیگنال صوتی پس از عبور از کانال،به صورت زیر است.

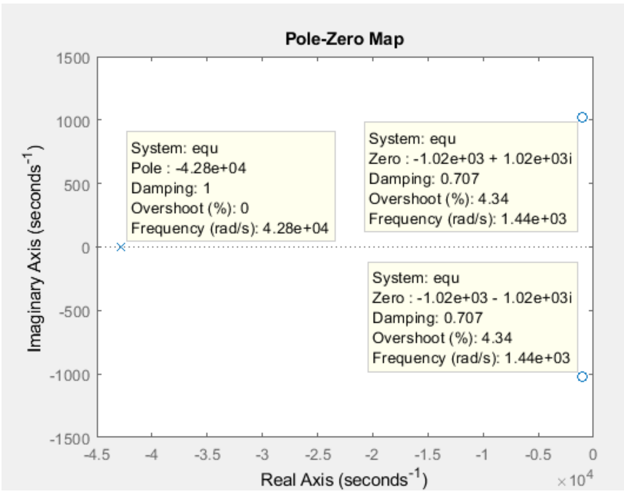


همان طور که درست حدس زده شد،فرکانس های بالا تضعیف شده و دیگر اثری از آن ها نمی بینیم.

برای جبران اعوجاج کانال،با توجه به این که فرکانس های پایین ثابت ماندند و فرکانس های بالا تضعیف شدند، باید سیستمی را طراحی کنیم که فرکانس های بالا را تقویت کند.به کمک دستور iddata،سیستم را با یک صفر و دو قطب تخمین می زنیم.equalizer که اعوجاج خروجی را درست می کند،برابر با عکس پاسخ سیستم اصلی است.نمودار صفر وقطب و پاسخ فرکانسی equalizer به صورت زیر است.







با توجه به این که قطبی در خارج از دایره واحد داریم،سیستم علّی نبوده و پیاده سازی آن در عمل امکان پذیر نیست.

سوال 3:

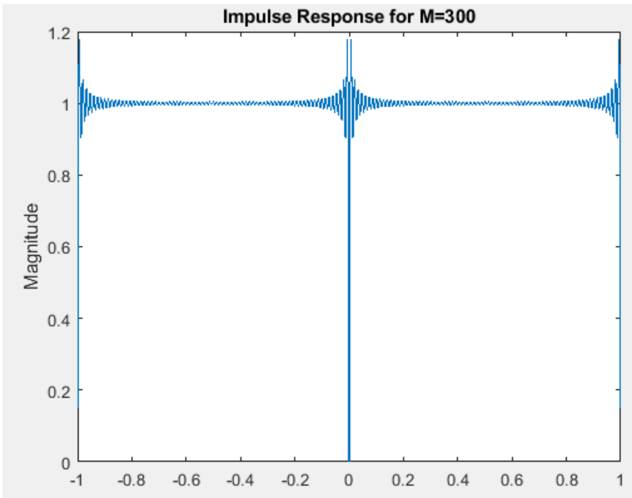
ابتدا پاسخ ضربه تبدیل هیلبرت را به دست می آوریم.

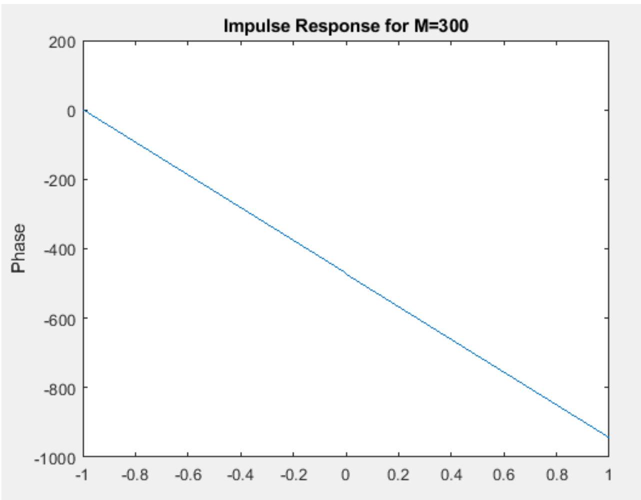
$$f\{u(t)\} = \frac{1}{j2\pi f} + \frac{1}{2}\delta(t) \quad \rightarrow \quad f\{sgn(t)\} = \frac{1}{j\pi f} \quad \rightarrow \quad f\left\{\frac{1}{j\pi t}\right\} = sgn(-f) \quad \rightarrow$$

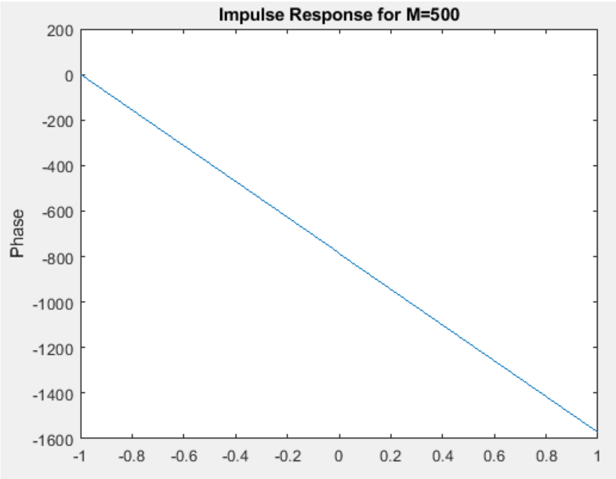
$$f\left\{\frac{1}{\pi t}\right\} = -jsgn(f)$$

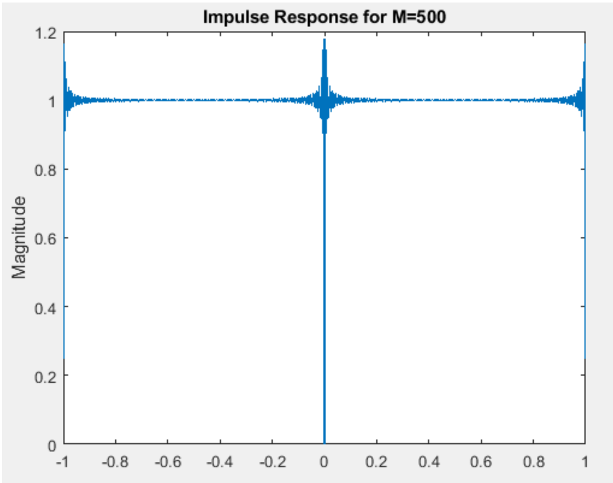
عبارت به دست آمده در حوزه زمان مطلقاً انتگرال پذیر نمی باشد،پس این فیلتر پایدار نیست.

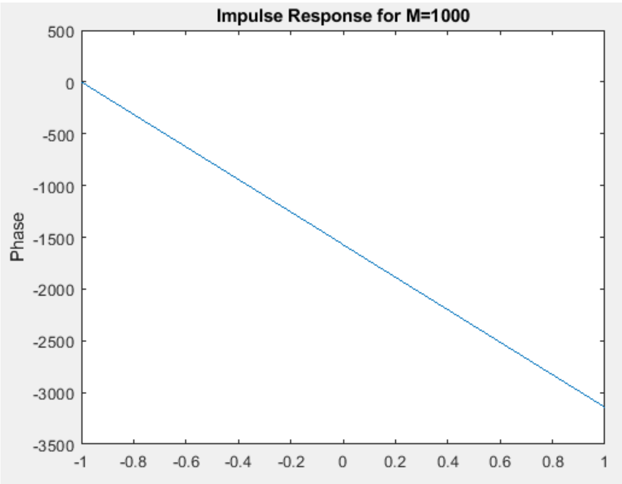
طبق خواسته سوال،پنجره خواسته شده را می سازیم.پاسخ فرکانسی برای M ها مختلف به صورت زیر است.

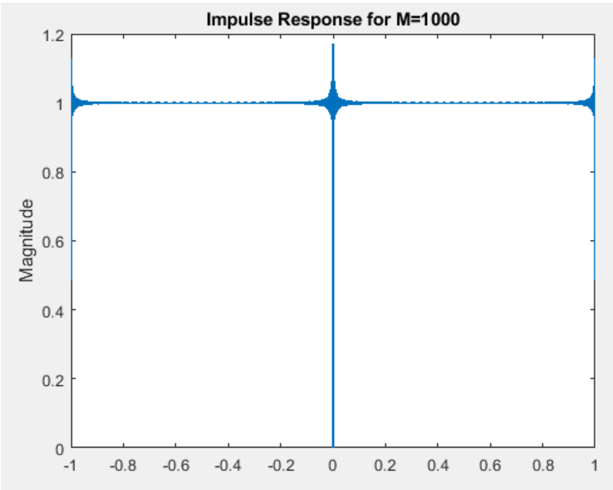






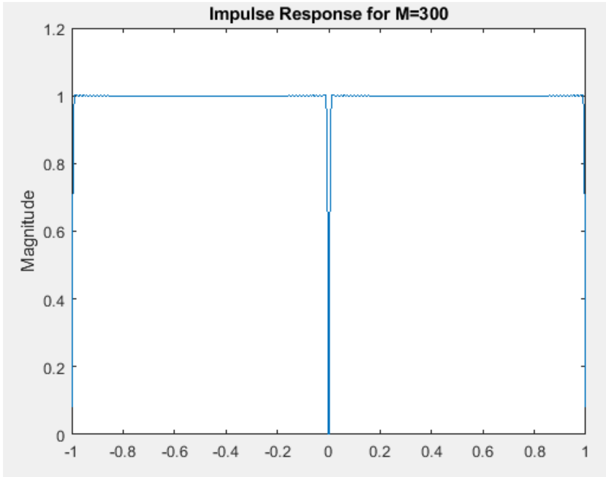


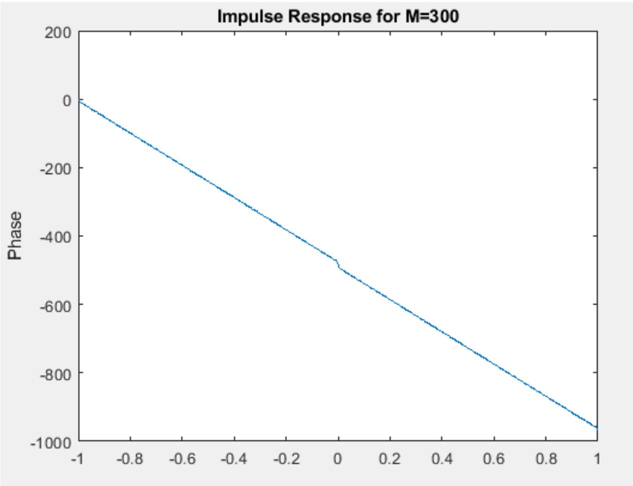


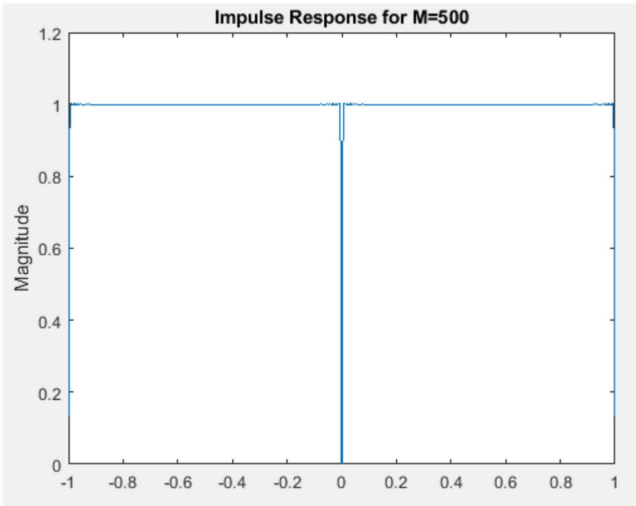


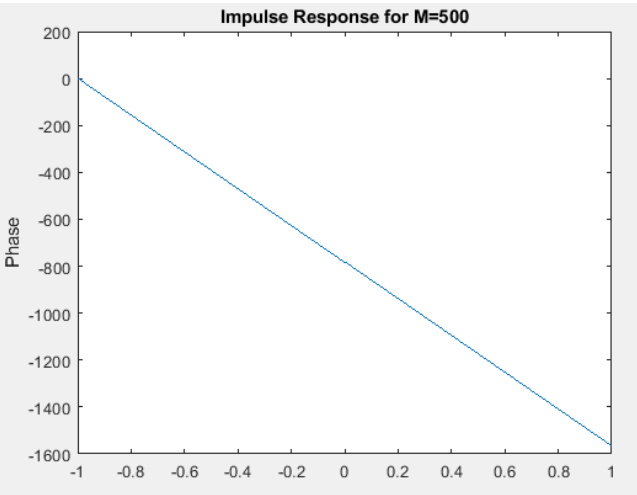
مشاهده می شود که هرچه M بیشتر باشد،سریع تر به مقدار ثابت رسیده و نوسانات کمتری دارد.برای فاز نیز می بینیم که شیب در هرمرحله افزایش می یابد.

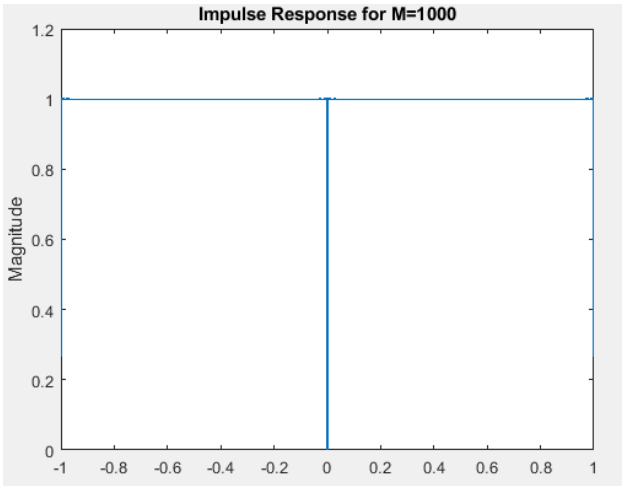
حال مراحل قبل را برای پنجره hamming تکرار می کنیم.

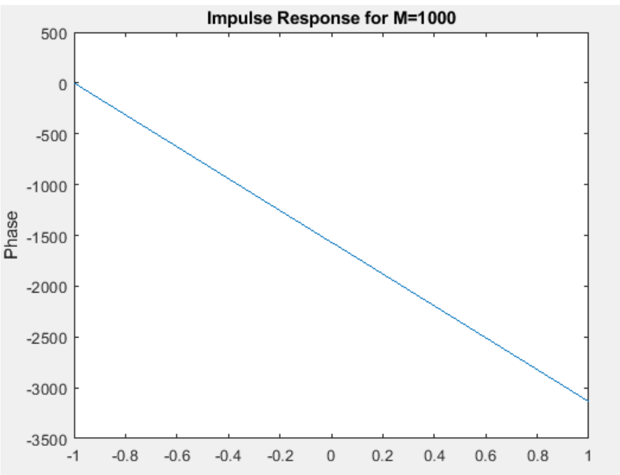












مشاهده می شود که هرچه M بیشتر باشد،سریع تر به مقدار ثابت رسیده و نوسانات کمتری دارد.برای فاز نیز می بینیم که شیب در هرمرحله افزایش می یابد.

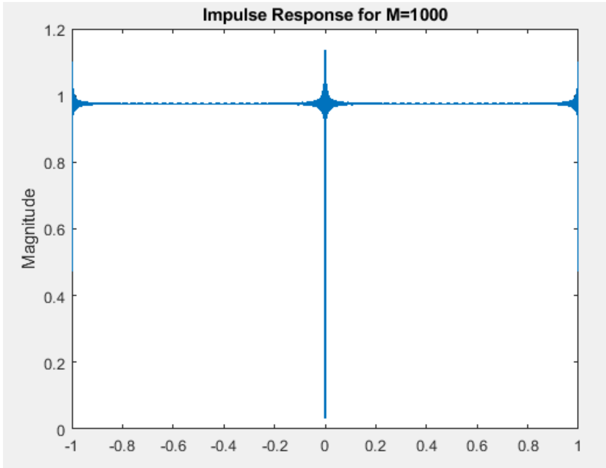
شباهت دو پنجره در این است که فاز یکسانی داشته و هردو پس از نوساناتی در دامنه به مقدار ثابتی می رسند. تفاوت این دو در نوسانات دامنه است.پنجره همینگ سریع تر به مقدار ثابت خود رسیده و نوسانات کمتری نسبت به پنجره نرمال دارد.

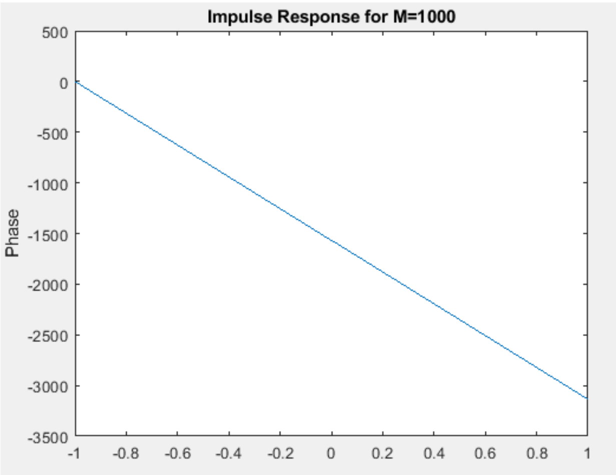
تعریف پنجره kaiser به صورت زیر است.

$$w[n] = w_0\left(\tfrac{L}{N}(n-N/2)\right) = \frac{I_0\left[\pi\alpha\sqrt{1-\left(\frac{2n}{N}-1\right)^2}\right]}{I_0[\pi\alpha]}, \quad 0 \le n \le N,$$

I انتگرال بسل اصلاح شده مرتبه صفر است و α پارامتر عبارت است.مقدار α را در محاسبات برابر با 0.1 در نظر می گیریم.

برای پنجره به طول M=1000 پاسخ به صورت زیر است.





همان طور که مشاهده می شود،رفتار فرکانسی آن تقریباً شبیه با دو پنجره قبلی است.کاربرد این پنجره در Audio coding پیشرفته است.