

به نام خدا



دانشگاه تهران

پردیس دانشکده‌های فنی

دانشکده مهندسی برق و کامپیوتر

گزارش تمرین کامپیوتری 2

اصول سیستم‌های مخابراتی

دکتر صباغیان

امیر مرتضی رضائی

810101429

نیمسال اول 1403-1404

فهرست:

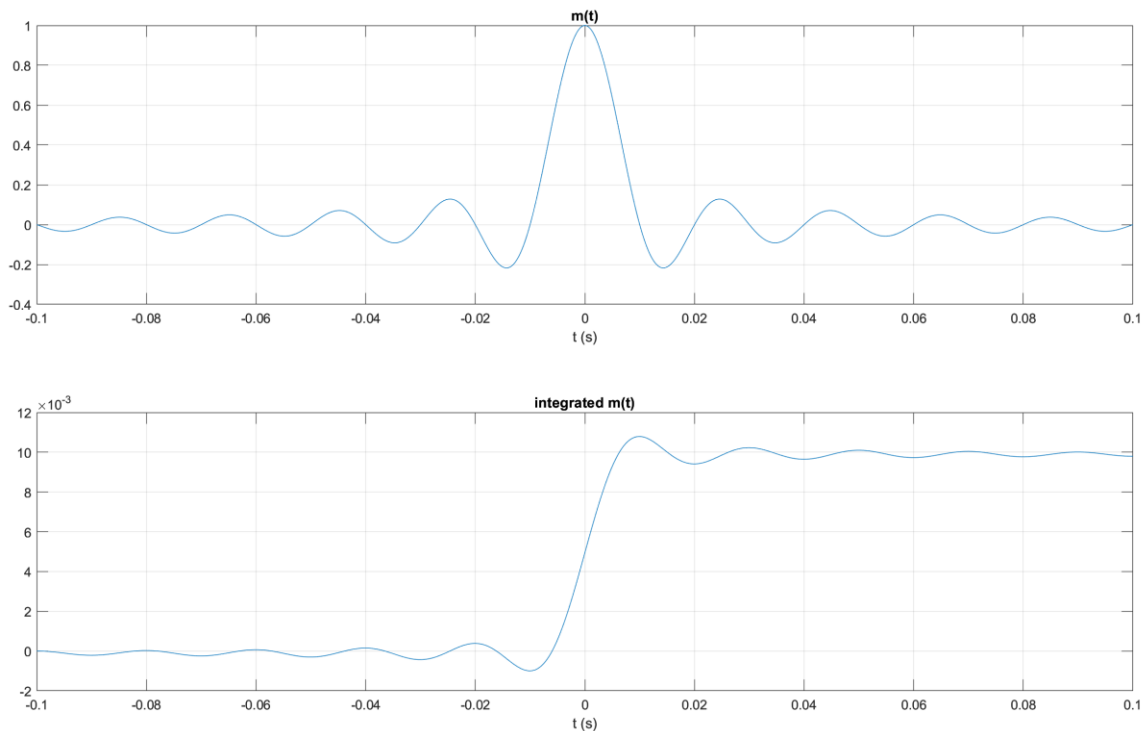
تمرین 1: 3.

تمرین 2: 5.

تمرین ۱:

قسمت الف: مشاهده سیگنال در حوزه زمان و فرکانس

سیگنال پیام و انتگرال آن مطابق صورت سوال پیاده‌سازی و رسم شدند. برای ترم انتگرال گیر نیز از تابع cumsum استفاده شد که با ضرب حاصل آن در هر بازه زمانی (T_s) گویی انتگرال سیگنال را محاسبه کرده‌ایم.

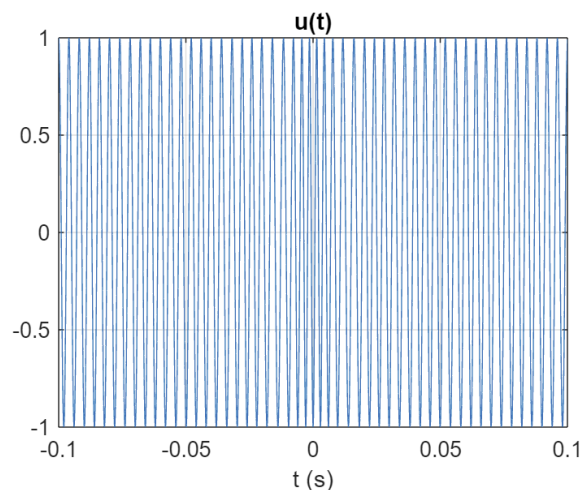


قسمت ب: مدولاسیون FM

مدولاتور FM، به صورت فانکشنی با همین نام پیاده‌سازی شد. مدولاسیون طبق رابطه زیر در این تابع محاسبه می‌گردد:

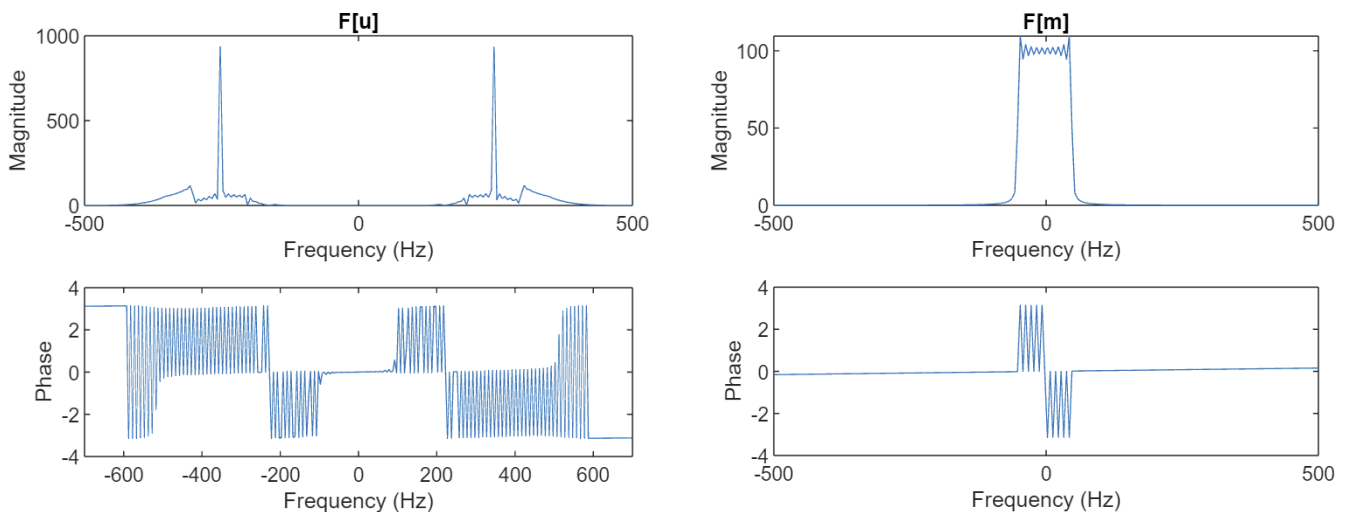
$$u(t) = \cos(2\pi f_c t + 2\pi k_f \int_{-\infty}^t m(\tau) d\tau)$$

سیگنال مدوله شده به صورت زیر می‌باشد:



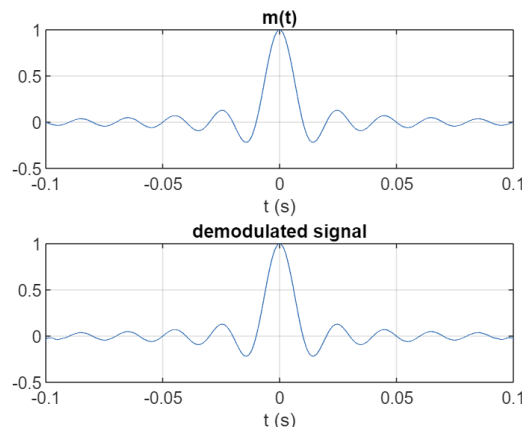
قسمت ج: محاسبه تبدیل فوریه توابع

تبدیل فوریه سیگنال‌های $m(t)$ و $u(t)$ با استفاده از توابع `fft` و `fftshift` محاسبه و رسم گردید. تابع `fftshift` برای جابجایی مرکز تبدیل فوریه به کار می‌رود. در تبدیل فوریه، نتایج به طور معمول به‌طور متوالی از منفی به مثبت در فرکانس‌ها مرتب می‌شوند، به طوری که فرکانس‌های منفی در ابتدا و فرکانس‌های مثبت در انتها قرار می‌گیرند. با استفاده از `fftshift`، این آرایه‌ها به گونه‌ای جابجا می‌شوند که فرکانس‌های صفر (فرکانس پایه) به مرکز بردار تبدیل فوریه منتقل شوند. در واقع `fftshift` برای انتقال صفر فرکانس به وسط آرایه‌ای است که تبدیل فوریه آن محاسبه شده است. نتیجه نهایی در بازه‌ی -500 تا 500 هرتز به صورت زیر می‌باشد:



قسمت د: دمدولاسیون سیگنال و بازیابی سیگنال پیام

در ابتدا با استفاده از تبدیل هیلبرت¹ و تابع `angle`، فاز سیگنال دمدوله شده استخراج شده و با استفاده از تابع `unwrap`، اثر تاخوردگی فاز به میزان 2π برطرف شد. سپس فاز سیگنال حامل ($2\pi f_c$) از فاز بدست آمده کسر شد. در ادامه مشتق نتیجه محاسبه و مقدار آن بر $2\pi f_\Delta$ تقسیم شد تا سیگنال پیام بدست آید. در نهایت نیز سیگنال پیام اصلی و سیگنال دمدوله شده به صورت زیر رسم شدند.



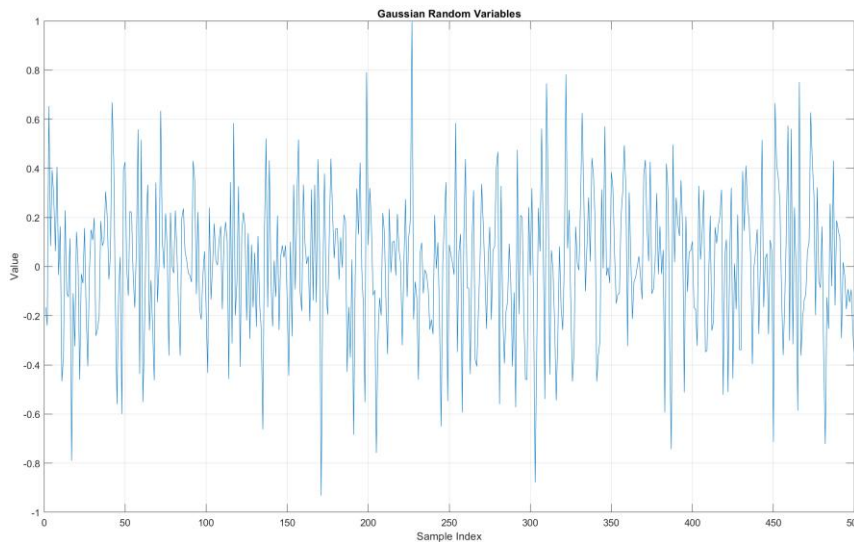
همان‌طور که مشاهده می‌گردد، سیگنال دمدوله شده به میزان بسیار خوبی مشابه سیگنال پیام اصلی می‌باشد.

¹ <https://dsp.stackexchange.com/questions/73356/how-does-the-hilbert-transform-extract-the-in-phase-and-quadrature-phase-compone>

تمرین ۲:

قسمت الف: تولید مقادیر تصادفی

500 مقدار تصادفی با توزیع نرمال با میانگین 0 و واریانس 1 تولید کرده و آنها را در بازه‌ی $(-1,1)$ نرمالایز نموده و رسم کردیم.

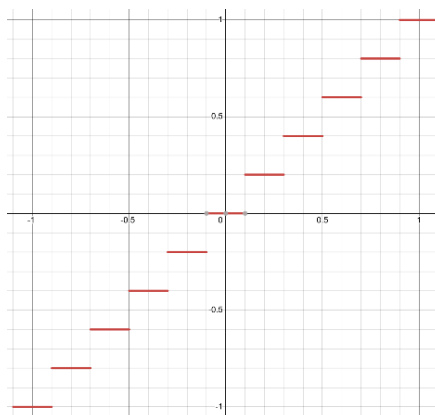


قسمت ب: کوانتیزاسیون و محاسبه SQNR

می‌دانیم هنگامی که q سطح کوانتیزاسیون داشته باشیم و سیگنال‌مان بین $(-1,1)$ نرمالایز شده باشد، فاصله سطوح کوانتیزاسیون برابر با $2/q$ خواهد بود.

برای اعمال کوانتیزاسیون، با کم کردن حداقل مقدار از تمام عناصر داده، داده‌ها را طوری جابجا می‌کنیم که حداقل مقدار برابر با صفر شود. سپس این مقدار جابجا شده را بر روی یک گام کوانتیزاسیون تقسیم می‌کنیم. این گام نشان‌دهنده اندازه هر بخش از دامنه است که ما می‌خواهیم داده‌ها را در آن تقسیم کنیم. در نهایت، با استفاده از تابع گرد کردن، مقادیر به نزدیک‌ترین عدد صحیح گرد می‌شوند. این عمل باعث می‌شود که هر مقدار به نزدیک‌ترین سطح گسسته^۲ تبدیل شود.

در ادامه مقادیر گسسته که قبلاً محاسبه شده‌اند (مقادیر گرد شده) دوباره با گام کوانتیزاسیون ضرب می‌شوند تا به دامنه‌ای که در آن کار کردیم برگردیم. و در نهایت، حداقل مقدار اولیه دوباره به مقادیر اضافه می‌شود تا داده‌ها به دامنه اصلی خود بازگردند.



^۲ quantization level

در ادامه، خطای کوانتیزاسیون برای هر یک از داده‌ها را به صورت اختلاف مقدار اصلی سیگنال و مقدار کوانتیزه شده آن محاسبه می‌کنیم. سپس، توان سیگنال و توان نویز کوانتیزاسیون را به صورت میانگین مربعات داده‌ها محاسبه نموده و با تقسیم آن‌ها بر یکدیگر و لگاریتم گرفتن از حاصل، SQNR را بر حسب dB بدست می‌آوریم:

$$\text{SQNR} = 30.3736 \text{ dB}$$

قسمت ج: نمایش مقادیر محاسبه شده

در این مرحله 5 مقدار اول دنباله تصادفی، مقادیر کوانتیزه و کلمات PCM نمایش داده شده‌اند. از آنجا که PCM دنباله باینری را نمایش می‌دهد و در کل q سطح کوانتیزاسیون داریم، بنابراین حداقل بیت‌های لازم برای نمایش آن به صورت زیر بدست می‌آید.

$$v = \log_2 q = \log_2 64 = 6$$

در نهایت نتیجه به صورت زیر خواهد بود:

Original Values:	Quantized Values:	PCM Codes in decimal	PCM Codes in binary:
-0.1656	-0.1503	25	0011001
-0.2395	-0.2441	22	0010110
0.6535	0.6622	51	0110011
0.0839	0.0684	32	0100000
0.3928	0.3809	42	0101010

قسمت د: رسم خطای کوانتیزاسیون

در این قسمت، خطای کوانتیزاسیون را که در قسمت ب بدست آورده بودیم، رسم می‌کنیم.

