

به نام خدا



دانشگاه صنعتی شریف

دانشکده مهندسی برق

سیستم‌های مخابراتی - دکتر بهروزی - گروه ۳ و ۲

نیم‌سال اول ۹۸-۹۹

پروژه نهایی - مدولاسیون دیجیتال

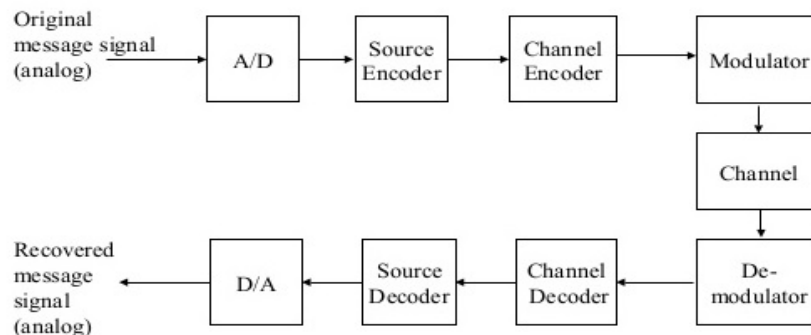
تاریخ تحویل: ۱۰ بهمن ۹۸

نکات قابل توجه و مواردی که باید رعایت شود:

- پروژه نهایی باید توسط گروه دو نفره مشخص شده انجام شود.
- تحویل پروژه در سامانه CW انجام می‌شود. تاریخ تحویل به هیچ وجه تمدید نخواهد شد. همچنین تحویل پروژه بعد از تاریخ ذکر شده و با تأخیر قابل قبول نیست. تحویل حضوری پروژه نیز متعاقباً هماهنگ خواهد شد.
- فایل تحویلی باید به فرمت zip یا rar. و حاوی موارد زیر باشد:
 - یک فایل m. شامل کدهای تمام سوالات
 - فایل‌های تمام توابعی که نوشته‌اید و در کد خود استفاده کرده‌اید
 - فایل گزارش به فرمت pdf. شامل پاسخ به تمام سوالات، نمودارها، نتایج خواسته شده، اثبات‌ها و محاسبات دستی (در صورت لزوم)
- نام فایلی که آپلود می‌کنید به صورت `Project_Name1_StudentID1_Name2_StudentID2` باشد.
- نمودارها باید عنوان مناسب داشته باشند.
- سعی کنید گزارشتان مختصر و کامل باشد، از توضیحات اضافی اجتناب کنید.
- کد خود را کامنت گذاری کنید و بخش‌های مختلف آن را توسط %% جدا کنید.
- از کپی کردن کد یا گزارش دیگران و یا قراردادن آن‌ها در اختیار دیگران بپرهیزید؛ در صورت مشاهده برخورد جدی خواهد شد.
- ۴۵٪ نمره مربوط به اجرای بدون خطای کد و عملکرد درست آن است. ۴۵٪ نمره به پاسخ به سوالات و نتایج درست گزارش تعلق دارد. خوانا بودن کد و تمیز بودن گزارش نیز هر کدام ۵٪ نمره را تشکیل می‌دهند.
- جهت رفع ابهام و اشکالات خود می‌توانید از طریق آدرس ایمیل `matlab.comsys@gmail.com` آن‌ها را مطرح کنید.

معرفی پروژه

در این پروژه هدف انتقال تصویر توسط سیستم مدولاسیون دیجیتال است. همانطور که از مطالب درس می‌دانید بلوک دیاگرام یک سیستم مدولاسیون دیجیتال به صورت شکل ۱ است:



شکل ۱: بلوک دیاگرام کلی مدولاسیون دیجیتال

به همراه فایل صورت پروژه تصاویری در پوشه pictures قرار دارند. از آن جایی که این تصاویر خودشان دیجیتال هستند، بلوک‌های A/D و D/A در این پروژه کاربرد ندارند. همچنین بلوک‌های Channel Encoder و Channel Decoder که برای اضافه کردن زوائد^۱ به سیگنال جهت تشخیص و تصحیح خطا هستند نیز در این پروژه بررسی و پیاده‌سازی نمی‌شوند. در فاز اول شما باید بلوک‌های مختلف را پیاده‌سازی کنید و در فاز دوم به آزمایش و تحلیل سیستم پیاده‌سازی شده پرداخته می‌شود. تصاویر این پروژه grayscale هستند بنابراین در ماتریس‌هایی به ابعاد تصویر یعنی 512×512 ذخیره می‌شوند. هر درایه این ماتریس‌ها از جنس uint8 و بنابراین عددی بین ۰ و ۲۵۵ است. برای کار با تصاویر از دستورات imread، imshow، imresize و ... استفاده کنید. پیش از هر کاری اندازه تصاویر را به 64×64 کاهش دهید تا در ارسال و دریافت آن‌ها به دلیل حجم بالا مشکلی ایجاد نشود.

فاز اول

بلوک‌های Source Encoder و Source Decoder

نقش این بلوک تبدیل دنباله سمبل‌ها به دنباله باینری است (توجه کنید در این پروژه هر پیکسل تصویر یک سمبل است). این کار با هدف فشرده‌سازی و کاهش بیت بر سمبل انجام می‌شود. یک روش انجام کدگذاری منبع استفاده از الگوریتم کدگذاری هافمن^۲ است. کدگذاری هافمن روشی برای فشرده‌سازی بی‌اتلاف اطلاعات است. این روش به استفاده از جدول کد طول متغیر برای کد کردن هر کدام از سمبل‌های مبدأ (مانند کاراکترهای یک متن) برمی‌گردد. جدول کد طول متغیر از روشی بخصوص مبنی بر احتمال وقوع هر کدام از سمبل‌های مبدأ بدست می‌آید. در کدگذاری هافمن، روشی به نام کدهای پیشوندی^۳ استفاده می‌شود. یعنی در این روش رشته‌ای که نشان دهنده یک سمبل خاص است هیچ‌گاه پیشوند رشته دیگر که نمایانگر سمبل دیگری است، نمی‌باشد و سمبل‌های پرکاربردتر با رشته‌های بیتی کوتاه‌تری نسبت به آن‌هایی که کاربردشان کمتر است، نشان داده می‌شوند. الگوریتم هافمن به صورت زیر است:

^۱redundancy

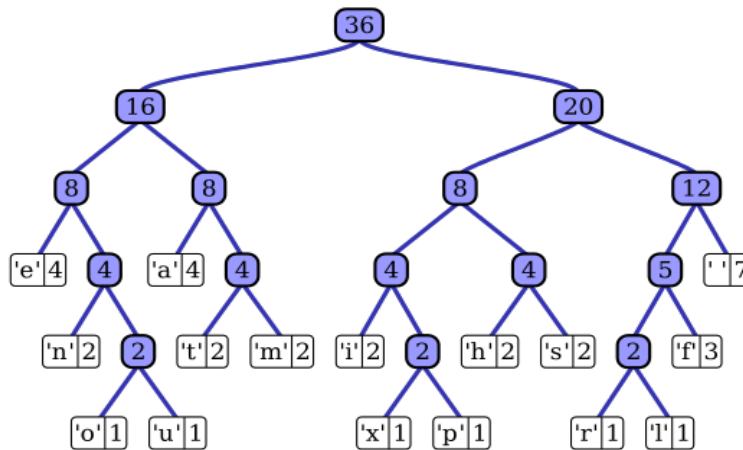
^۲Huffman coding

^۳prefix code

HUF(A, f)
 If $|A| = 1$ then return a single vertex.
 Let w and y be the symbols with the lowest frequencies.
 Let $A' = A \setminus \{w, y\} + \{z\}$.
 Let $f'(x) = f(x)$ for all $x \in A' \setminus \{z\}$, and let $f'(z) = f(w) + f(y)$.
 $T' = \text{HUF}(A', f')$.
 Create T from T' by adding w and y as children of z .
return T

شکل ۲: الگوریتم کدگذاری هافمن

A مجموعه سمبل‌ها و $f(x)$ برای هر $x \in A$ فراوانی سمبل x در داده‌ها (در اینجا دیتاست تصاویر) است. برای روشن‌تر شدن موضوع به مثال زیر که کدگذاری جمله «this is an example of huffman tree» را نشان می‌دهد توجه کنید:



شکل ۳: مثال الگوریتم کدگذاری هافمن

پس از تشکیل درخت از ریشه به سمت برگ‌ها حرکت می‌کنیم تا کد هر برگ را به دست آوریم. برای این کار به صورت اختیاری یک از جهت‌ها را صفر و دیگری را یک می‌گیریم. و مسیری که از ریشه تا برگ طی می‌شود کد آن سمبل را مشخص می‌کند. مثلاً کدهای معادل درخت بالا مطابق جدول ۱ است. برای مطالعه بیشتر در مورد کدگذاری هافمن می‌توانید در اینترنت جستجو کنید.

جدول ۱: مثال الگوریتم کدگذاری هافمن

کد	تکرار	سمبل
۱۱۱	۷	space
۰۱۰	۴	a
۰۰۰	۴	e
۱۱۰۱	۳	f
۱۰۱۰	۲	h
۱۰۰۰	۲	i
۰۱۱۱	۲	m
۰۰۱۰	۲	n
۱۰۱۱	۲	s
۰۱۱۰	۲	t
۱۱۰۰۱	۱	l
۰۰۱۱۰	۱	o
۱۰۰۱۱	۱	p
۱۱۰۰۰	۱	r
۰۰۱۱۱	۱	u
۱۰۰۱۰	۱	x

با توجه به روش معرفی شده بلوک‌های Source Encoder و Source Decoder را بدون به کارگیری توابع آماده متلب پیاده‌سازی کنید. سپس با کدگذاری و کدگشایی چند تصویر صحت آن‌ها را آزمایش کرده و نشان دهید (مثال‌ها را در گزارش خود بیاورید).

بلوک‌های Modulator و Demodulator

- تابعی با عنوان Modulator بنویسید که در ورودی دنباله‌ای از سمبل‌ها (اعداد ۰ تا M-۱) دریافت کند و در خروجی شکل موج مدوله شده به روش M-ary PSK^۴ را تولید کند.

$$S_m = \begin{cases} \sqrt{\frac{2}{T_s}} \cos\left(2\pi f_c t + \frac{(2m+1)\pi}{M}\right) & 0 \leq t \leq T_s \\ 0 & otherwise \end{cases}$$

$$m = 0, \dots, M-1$$

- تابعی با عنوان Demodulator بنویسید که در ورودی سیگنال دریافت شده را بگیرد و با استفاده از همبستگی سنج (Correlator) و یا فیلتر منطبق (Matched Filter) شکل موج متناظر با هر پیام را به فضای برداری سیگنال تصویر کند. خروجی این تابع دنباله‌ای از بردارها است که بردار m-ام i-نشان دهنده‌ی تصویر امین-i پیام دریافتی در فضای برداری سیگنال است. (خروجی این تابع همان بردارهای موجود در Constellation سیگنال هستند).

- تابعی با عنوان Detector بنویسید که دنباله بردارهای حاصل از Demodulation را دریافت کند و با قاعده تصمیم‌گیری بهینه، در مورد هرکدام از پیام‌ها تصمیم بگیرد و مقدار تخمین زده شده برای سمبل متناظر را در خروجی تولید کند. خروجی این تابع در حالت ایده‌آل باید برابر ورودی تابع Modulator باشد. مجموع این دو تابع عمل دمدولاسیون را انجام می‌دهند و در واقع بلوک Demodulation در بلوک دیاگرام شکل ۱ را می‌سازند.

^۴Phase Shift Keying

فاز دوم

الف) بلوک‌های پیاده‌سازی شده را در کنار یکدیگر قرار دهید تا سیستم تکمیل شود. برای این کار باید خروجی Source Encoder که یک رشته بیتی است را به قطعه‌هایی با طول مشخص تقسیم کنید به طوری که عددی بین ۰ و $M-1$ شود و بتوانید آن را به مدولاتوری که پیاده‌سازی کرده‌اید بدهید. عکس همین عمل نیز بعد از دمدولاسیون انجام می‌شود.

ب) برای سادگی در تست کانال را ایده‌آل در نظر می‌گیریم، و فقط پهنای باند سیگنال عبوری را محدود می‌کنیم. برای این مسئله، تابعی با عنوان Channel بنویسید که در ورودی سیگنال ارسالی، فرکانس نمونه برداری، و فرکانس حامل را دریافت کرده و در خروجی سیگنال رسیده به گیرنده را تحویل دهد. (صرفاً قرار است یک فیلتر میان‌گذر بنویسید!)

در تمام قسمت‌های بعدی مشخصات زیر را در نظر بگیرید:

جدول ۲: پارامترهای شبیه‌سازی

مقدار	پارامتر
۱۰۰KHz	فرکانس نمونه‌برداری
۱۰ms	طول هر سمبل (T_s)
۱۰KHz	فرکانس حامل (f_c)
۱۰KHz	فرکانس مرکزی کانال
۵KHz	پهنای باند کانال
۴	M

پ) طیف سیگنال عبوری از کانال را رسم کنید و پهنای باند میانگین آن را گزارش کنید. چند نمونه را در گزارش بیاورید.

ت) با فرض اینکه نویز AWGN قبل از ورود سیگنال به کانال با آن جمع می‌شود. با ارسال و دریافت تصاویر توسط سیستم احتمال خطا را بر حسب واریانس نویز رسم کنید. رفتار حدی احتمال خطا را توجیه کنید.

ث) (امتیازی) با ارسال و دریافت تصاویر توسط سیستم SNR میانگین روی تمام تصاویر را قبل و بعد از دمدولاسیون گزارش کنید. همچنین چند نمونه از تصاویر ارسالی و دریافتی را در گزارش خود بیاورید.

ج) ۶ مقدار مختلف برای واریانس نویز انتخاب کنید (به نحوی که تا حد خوبی بازه‌های معنا دار را پوشش دهد). برای هر کدام از این واریانس‌ها، نمودار بردارهای خروجی تابع Demodulation را رسم کنید. (قرار است تقریبی از آنچه که در درس با عنوان Constellation یاد گرفته‌اید را ببینید). همچنین آستانه تصمیم‌گیری بهینه Detector را بر روی این نمودارها رسم کنید. تاثیر نویز را روی این نمودار توضیح دهید.

چ) (امتیازی) فرض کنید که گیرنده و فرستنده دقیقاً همزمان نیستند و در نتیجه در گیرنده اختلاف فاز ϕ نسبت به گیرنده وجود دارد. برای سه حالت $\phi = \frac{\pi}{۱۲}, \frac{\pi}{۶}, \frac{\pi}{۳}$ قسمت قبل را تکرار کنید.