



## گزارش 5 آزمایشگاه مخابرات دیجیتال

دانشگاه خلیج فارس

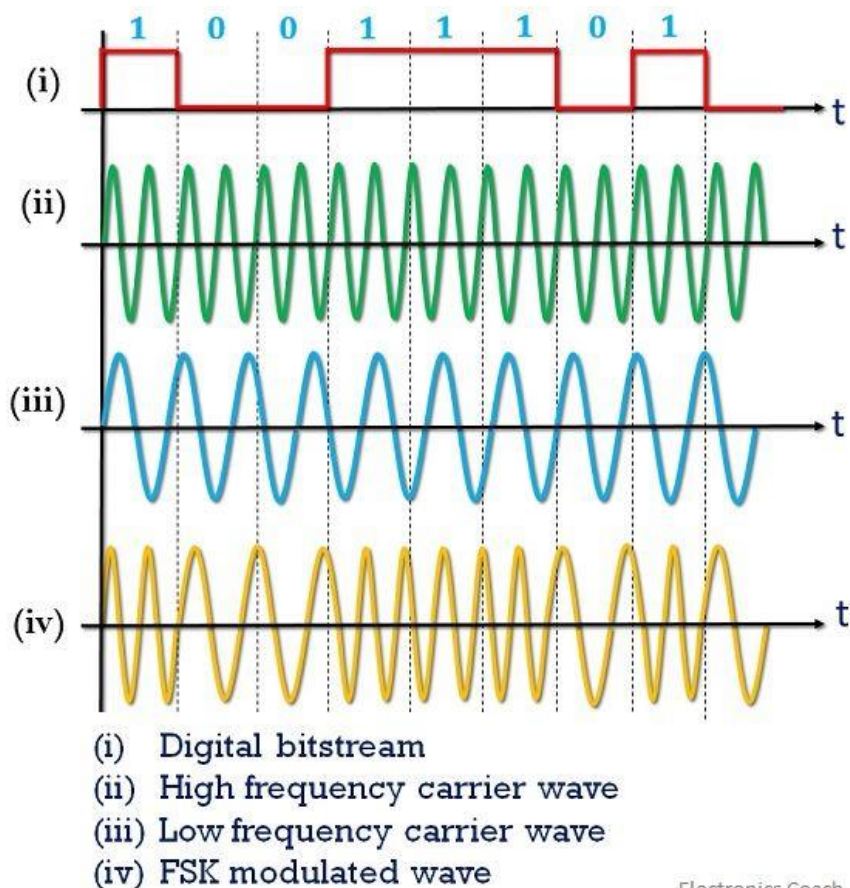
شماره دانشجویی: 990291614

نام: محمد صالح رایانی

هدف: شبیه سازی مدولاسیون فرکانس دیجیتال (FSK)

مدولاسیون FSK (Frequency Shift Keying) یک روش مدولاسیون دیجیتال در سیستم‌های فرستنده و گیرنده رادیویی است. در این روش، اطلاعات دیجیتال به صورت دنباله‌ای از بیت‌ها (صفر و یک) تمثیل می‌شوند و با استفاده از تغییر فرکانس در موج حامل رادیویی ارسال می‌شوند. در فرستنده FSK، سیگنال ورودی (بیت‌های دیجیتال) به سیگنال بی‌تی تبدیل می‌شود که برای هر بیت یک دوره زمانی مشخص دارد. در صورت وجود بیت یک، سیگنال بی‌تی برای آن دوره زمانی به فرکانس مدولاسیون یک می‌رسد؛ و در صورت وجود بیت صفر، سیگنال بی‌تی برای آن دوره زمانی به فرکانس مدولاسیون دیگری می‌رسد. به عبارت دیگر، فرکانس موج حامل در هر دوره زمانی برای نمایش بیت‌های دیجیتال تغییر می‌کند.

در گیرنده FSK، سیگنال دریافت شده توسط آنتن رادیویی تقویت و پس از تقسیم بندی به دوره‌های زمانی، آنالیز می‌شود. با تشخیص تغییر فرکانس موج در هر دوره زمانی، بیت‌های دیجیتال اصلی بازیابی می‌شوند. مدولاسیون FSK به دلیل قابلیت تحمل به نویز بالا و مقاومت در برابر اختلالات محیطی، برای انتقال داده‌های با سرعت بالا و در فواصل بزرگتر میان فرستنده و گیرنده استفاده می‌شود. همچنین، به دلیل ساختار ساده و عملکرد استاندارد، در بسیاری از سیستم‌های ارتباطات بی‌سیم استفاده می‌شود. در تصویر زیر شکل امواج رادیویی در مدولاسیون FSK را که از سایت Electronics coach برداشته شده است، می‌توانید مشاهده کنید.



کد متلب:

```
clc;clear;close all;

b = [1 0 0 1 1 1 0 1 0 1 1 0];
t = 0:0.001:1;
T = 0:0.001:length(b);
Signal1 = [];
Signal2 = [];
Fi = [2 10];
Fc = 5;
```

خط اول کد، پنجره فرمان، متغیرها و فیگرها را می‌بندد. متغیر  $b$  سیگنال بیت پیام است. متغیرهای  $t$  و  $T$  زمان دوره پالس حامل و کل سیگنال است.  $Signal1$  برای نگهداری سیگنال مدوله شده با فرکانس حامل درون  $Fi$  است و  $Signal2$  برای نگهداری سیگنال مدوله شده با فرکانس حامل  $Fc$  جمع شده با فرکانس های پیام در  $Fi$  است.

```
for i=1:length(b)
    s1= sin(2.*pi.*Fi(b(i)+1).*t)/(sum(abs(sin(2.*pi.*Fi(b(i)+1).*t)).^2));
    Signal1 = cat(2, Signal1, s1);

    s2= sin(2.*pi.*(Fc + Fi(b(i)+1)).*t)/(sum(abs(sin(2.*pi.*(Fc +
Fi(b(i)+1)).*t)).^2));
    Signal2 = cat(2, Signal2, s2);
end
```

حلقه for برای مدوله کردن هر بیت سیگنال و ذخیره سازی آن در  $Signal1$  و  $Signal2$  است. در تابع سینوس ها  $Fi(b(i)+1)$  به معنای این است که اولین یا دومین عنصر درون  $Fi$  انتخاب خواهد شد و این بستگی به مقدار نام  $b$  دارد.

```
figure

subplot(2,1,1)

stairs(0:length(b), [b b(end)], 'LineWidth', 1.5)
title('Signal bits')
ylim([-1.5 1.5])
xlim([0 length(b)])

subplot(2,1,2)
plot(T, Signal1(1:end-(length(b)-1)), 'LineWidth', 1.5)
title('Full modulated signal')
xlim([0 length(b)])
ylim([-0.005 0.005])

figure
subplot(2,1,1)
stairs(0:length(b), [b b(end)], 'LineWidth', 1.5)
title('Signal bits')
ylim([-1.5 1.5])
xlim([0 length(b)])
```

```
subplot(2,1,2)
plot(T, Signal2(1:end-(length(b)-1)), 'LineWidth',1.5)
title('Full modulated signal')
xlim([0 length(b)])
ylim([-0.005 0.005])
```

در کد بالا دو عدد فیگور باز می‌شود برای حالت های مختلف فرکانس حامل. دو زیرمجموعه می‌سازیم، یک نمودار سیگنال پیام با تابع **stairs** نشان می‌دهیم، ورودی اول این تابع محل روی محور افقی را مشخص می‌کند و ورودی دوم مقدار آن نقاط را مشخص می‌کند. تابع *plot* نمودار مقادیر ورودی دوم خود را بر حسب ورودی اول نشان می‌دهد. علت قراردادن  $(1:\text{end}-(\text{length}(b)-1))$  بعد سیگنال ها اصلاح کردن مشکل عدم تطابق اندازه بردار ها است.

