

Лабораторная работа №1

Сетевые технологии

Иванов Сергей Владимирович, НПИбд-01-23

9 сентября 2025

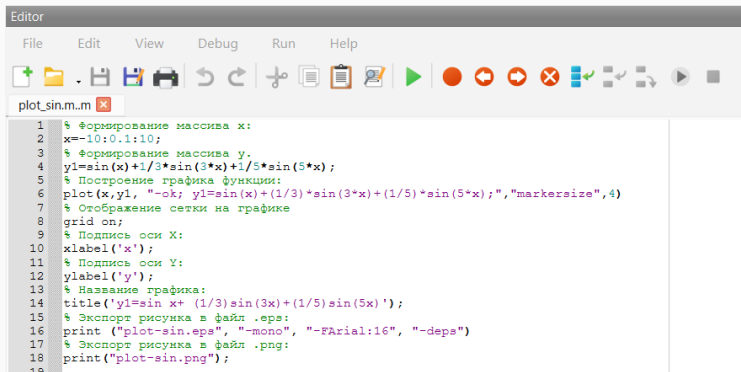
Российский университет дружбы народов, Москва, Россия

Изучение методов кодирования и модуляции сигналов с помощью высокоуровневого языка программирования Octave. Определение спектра и параметров сигнала. Демонстрация принципов модуляции сигнала на примере аналоговой амплитудной модуляции. Исследование свойства самосинхронизации сигнала.

Выполнение работы

Запустим Octave с оконным интерфейсом. Перейдем в окно редактора. Создадим новый сценарий. Сохраним его в рабочий каталог с именем. В окне редактора повторим листинг по построению графика функции (рис. 1).

Программа plot_sin.m

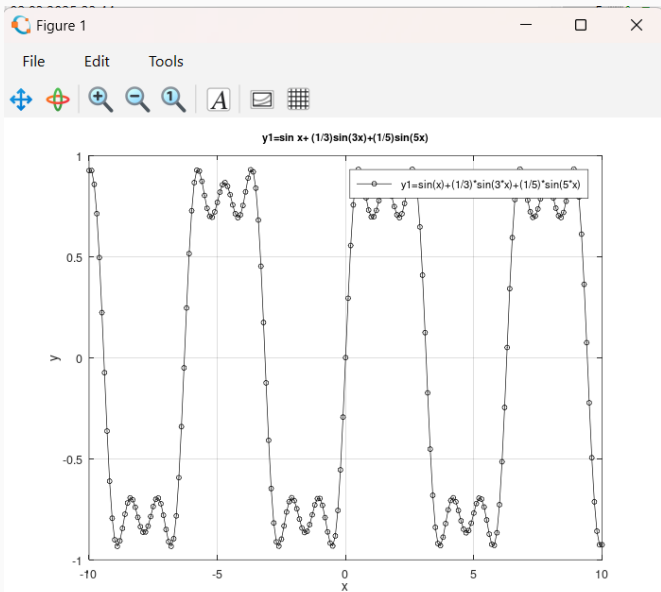


```
1 % формирование массива x:
2 x=-10:0.1:10;
3 % формирование массива y.
4 y1=sin(x)+1/3*sin(3*x)+1/5*sin(5*x);
5 % Построение графика функции:
6 plot(x,y1, "-ok; y1=sin(x)+(1/3)*sin(3*x)+(1/5)*sin(5*x);", "markersize", 4)
7 % Отображение сетки на графике
8 grid on;
9 % Подпись оси X:
10 xlabel('x');
11 % Подпись оси Y:
12 ylabel('y');
13 % Название графика:
14 title('y1=sin x+ (1/3)sin(3x)+(1/5)sin(5x)');
15 % Экспорт рисунка в файл .eps:
16 print ("plot-sin.eps", "-mono", "-FArial:16", "-deps")
17 % Экспорт рисунка в файл .png:
18 print("plot-sin.png");
19
```

Рис. 1: Программа plot_sin.m

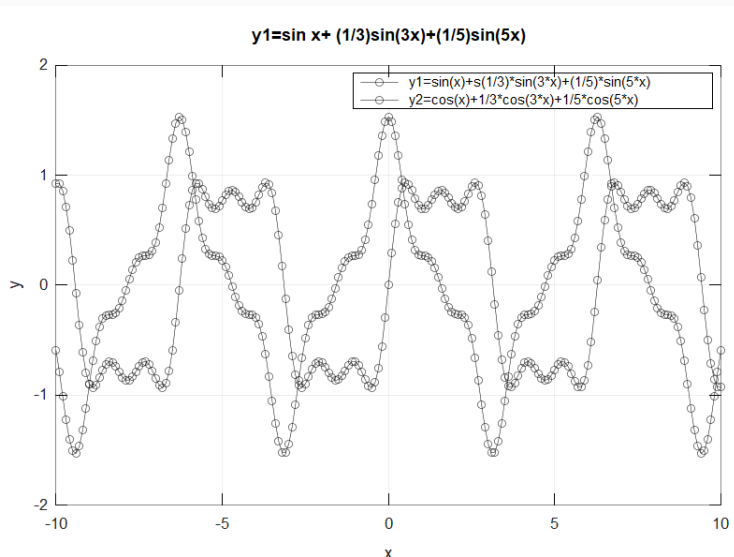
Запустим сценарий. В качестве результата выполнения кода открылось окно с построенным графиком (рис. 2) и в рабочем каталоге появились файлы с графиками в форматах `.eps`, `.png`.

Построенный график



Сохраним сценарий под другим названием и изменим его так, чтобы на одном графике располагались отличающиеся по типу линий графики функций (рис. 4)

График с cos



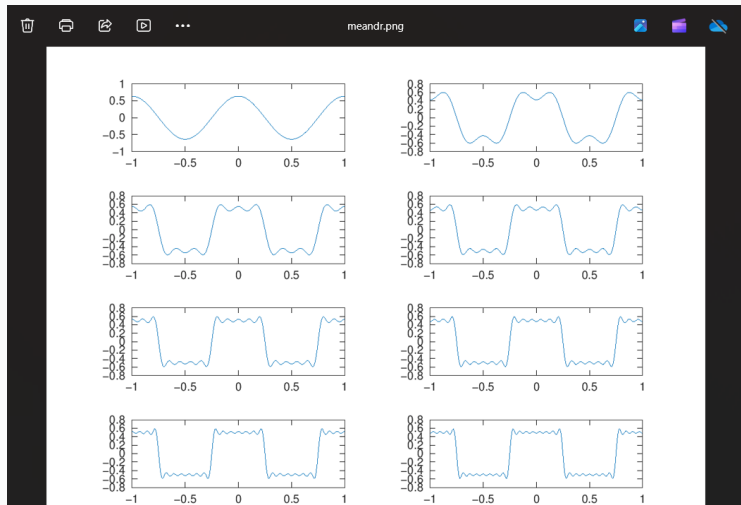
Разложение импульсного сигнала в частичный ряд Фурье

Создадим новый сценарий и сохраним его в рабочий каталог с именем meandr.m. В коде повторим листинг по построению графиков меандра. (рис. 5)

```
meandr.m ✕
1 % meandr.m
2 % количество отсчетов (гармоник):
3 N=8;
4 % частота дискретизации:
5 t=-1:0.01:1;
6 % значение амплитуды:
7 A=1;
8 % период:
9 T=1;
10 % амплитуда гармоник
11 nh=(1:N)*2-1;
12 % массив коэффициентов для ряда, заданного через cos:
13 Am=2/pi ./ nh;
14 Am(2:2:end) = -Am(2:2:end);
```

Рис. 5: Код meandr.m

Экспортируем полученный график в файл в формате .png. (рис. 6)



Меандр через синусы

Скорректируем код для реализации меандра через синусы. Получим графики. (рис. 7)

```
% количество отсчетов (гармоник):  
N=8;  
% частота дискретизации:  
t=-1:0.01:1;  
% значение амплитуды:  
A=1;  
% период:  
T=1;  
% амплитуда гармоник  
nh=(1:N)*2-1;  
% массив коэффициентов для ряда, зада  
Am=2/pi ./ nh;  
Am(2:2:end) = -Am(2:2:end);  
% массив гармоник:  
harmonics=sin(2 * pi * nh' * t/T);  
% массив элементов ряда:  
s1=harmonics.*repmat(Am',1,length(t))  
% Суммирование ряда:  
s2=cumsum(s1);  
% Построение графиков:  
for k=1:N  
    subplot(4,2,k)  
    plot(t, s2(k,:))  
end  
% Экспорт рисунка в файл .png:  
print("meandr-sin.png");
```

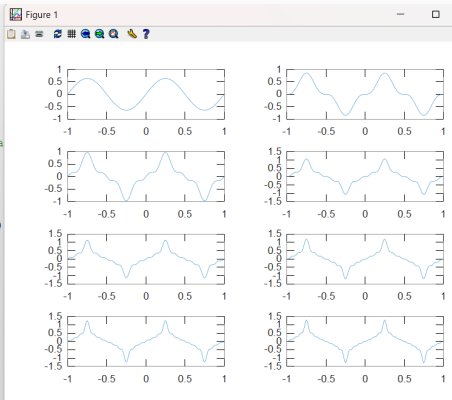


Рис. 7: Меандр через синусы

График сигналов разной частоты

Создадим каталог `spectre1` и в нём новый сценарий с именем, `spectre.m`. В коде повторим листинг по построению сигналов разной частоты. (рис. 8)

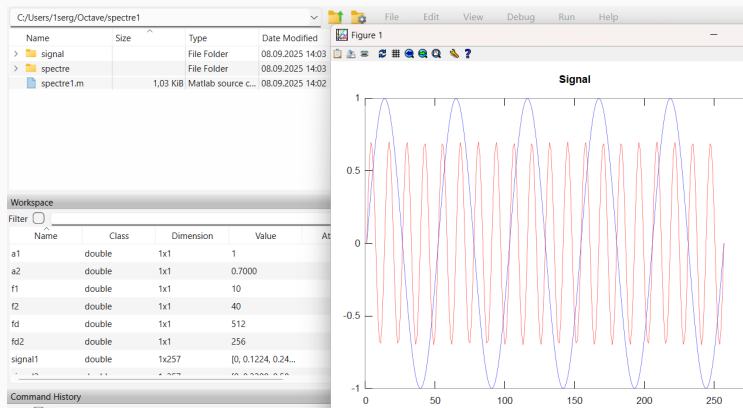
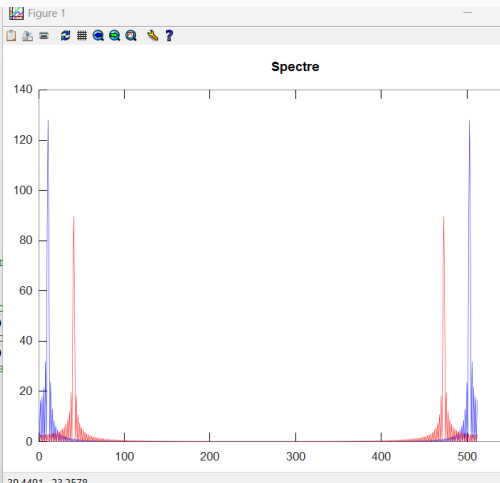


Рис. 8: График сигналов разной частоты

График спектра сигнала

Добавим код для нахождения спектров сигналов с помощью быстрого преобразования Фурье и получим график. (рис. 9)

```
% Амплитуда второго сигнала:
a2 = 0.7;
% Массив отсчётов времени:
t = 0:1./fd:tmax;
% Спектр сигнала:
fd2 = fd/2;
% Два сигнала разной частоты:
signal1 = a1*sin(2*pi*t*f1);
signal2 = a2*sin(2*pi*t*f2);
% График 1-го сигнала:
plot(signal1,'b');
% График 2-го сигнала:
hold on
plot(signal2,'r');
hold off
title('Signal');
% Экспорт графика в файл в кат
print 'signal/spectre.png';
% Посчитаем спектр
% Амплитуды преобразования Фур
spectre1 = abs(fft(signal1,fd)
% Амплитуды преобразования Фур
spectre2 = abs(fft(signal2,fd)
% Построение графиков спектров
plot(spectre1,'b');
hold on
plot(spectre2,'r');
hold off
title('Spectre');
print 'spectre/spectre.png';
```



Скорректированный график спектра

Скорректируем график спектра: отбросим дублирующие отрицательные частоты, а также примем в расчёт то, что на каждом шаге вычисления быстрого преобразования Фурье происходит суммирование амплитуд сигналов. (рис. 10)

Скорректированный график спектра

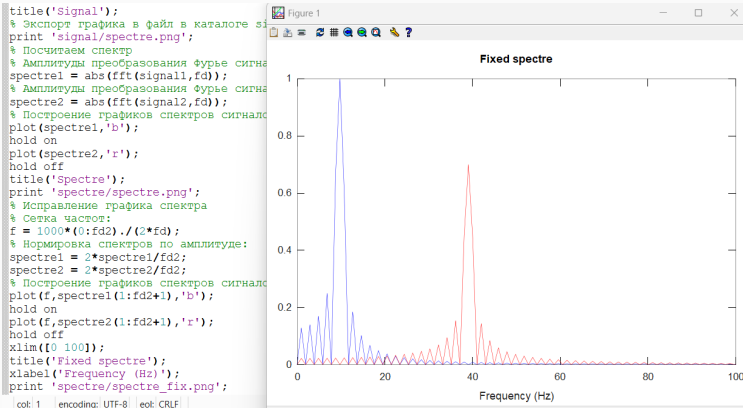


Рис. 10: Скорректированный график спектра

Найдем спектр суммы рассмотренных сигналов, создав каталог `spectr_sum` и файл в нём `spectre_sum.m`. В результате получился аналогичный предыдущему результат, т.е. спектр суммы сигналов равен сумме спектров сигналов, что вытекает из свойств преобразования Фурье. (рис. 11)

Графики суммарного сигнала

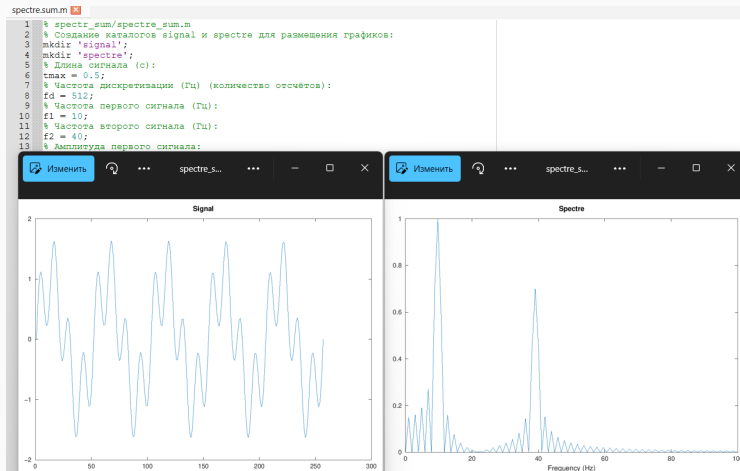


Рис. 11: Графики суммарного сигнала

Амплитудная модуляция

Создадим каталог modulation и в нём новый сценарий с именем am.m. Добавим в файле am.m код из листинга. (рис. 12)

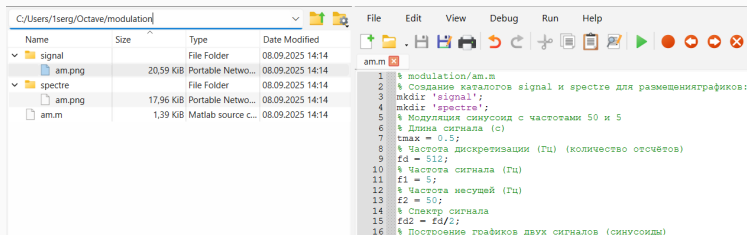


Рис. 12: Каталог modulation

Графики амплитудной модуляции

В результате получаем, что спектр произведения представляет собой свёртку спектров (рис. 13)

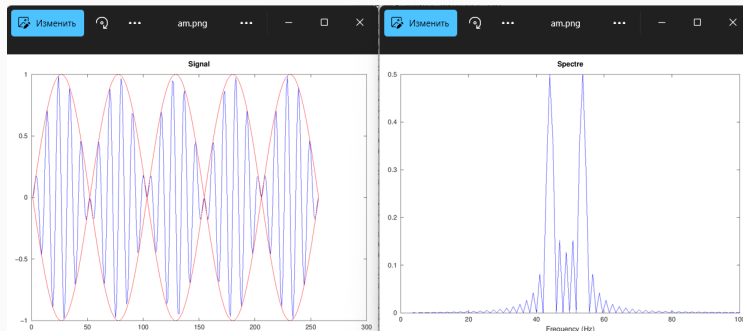











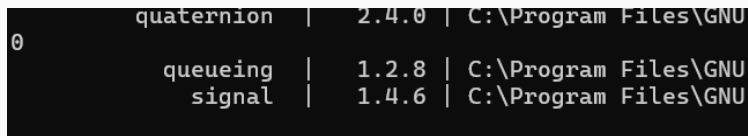
Рис. 13: Графики амплитудной модуляции

Кодирование сигнала. Исследование свойства самосинхронизации сигнала

Создадим каталог coding и в нём файлы main.m, maptowave.m, unipolar.m, ami.m, bipolarnrz.m, bipolarrrz.m, manchester.m, diffmanc.m, calcspectre.m (рис. 14)

C:/Users/1serg/Octave/coding			
Name	Size	Type	Date Modified
 ami.m	0 байты	Matlab source c...	08.09.2025 14:19
 bipolarnrz.m	0 байты	Matlab source c...	08.09.2025 14:19
 bipolarrrz.m	0 байты	Matlab source c...	08.09.2025 14:19
 calcspectre.m	0 байты	Matlab source c...	08.09.2025 14:20
 diffmanc.m	0 байты	Matlab source c...	08.09.2025 14:20
 main.m	0 байты	Matlab source c...	08.09.2025 14:18
 manchester.m	0 байты	Matlab source c...	08.09.2025 14:19
 maptowave.m	0 байты	Matlab source c...	08.09.2025 14:18
 unipolar.m	0 байты	Matlab source c...	08.09.2025 14:18

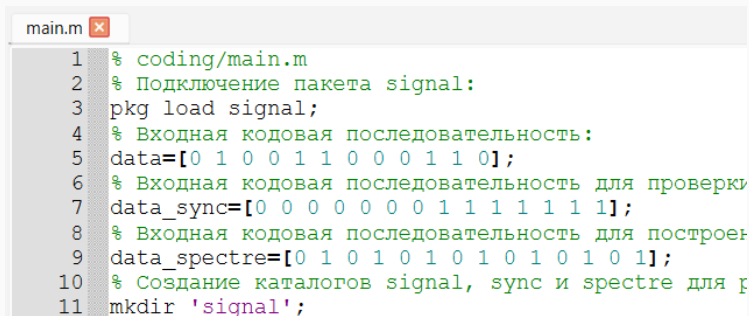
В окне интерпретатора команд проверим, установлен ли пакет расширений signal. Видим, что он установлен. (рис. 15)



```
0      quaternion | 2.4.0 | C:\Program Files\GNU
      queueing   | 1.2.8 | C:\Program Files\GNU
      signal     | 1.4.6 | C:\Program Files\GNU
```

Рис. 15: Проверка пакета

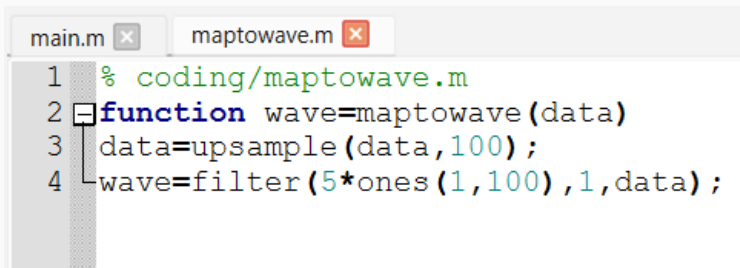
В файле main.m подключим пакет signal и скопируем программу из листинга. (рис. 16)



```
1 % coding/main.m
2 % Подключение пакета signal:
3 pkg load signal;
4 % Входная кодовая последовательность:
5 data=[0 1 0 0 1 1 0 0 0 1 1 0];
6 % Входная кодовая последовательность для проверки
7 data_sync=[0 0 0 0 0 0 0 1 1 1 1 1 1];
8 % Входная кодовая последовательность для построения спектра
9 data_spectre=[0 1 0 1 0 1 0 1 0 1 0 1 0 1];
10 % Создание каталогов signal, sync и spectre для хранения данных
11 mkdir 'signal';
```

Рис. 16: Код main.m

В файле maptowave.m пропишем функцию, которая по входному битовому потоку строит график сигнала. (рис. 17)

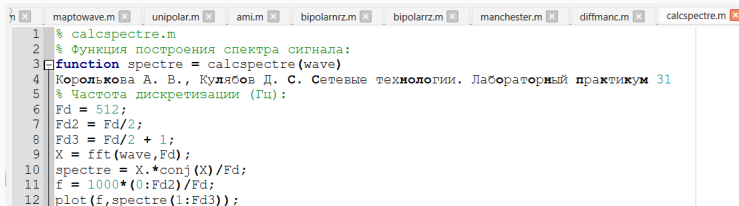


```
main.m x maptowave.m x
1 % coding/maptowave.m
2 function wave=maptowave(data)
3     data=upsample(data,100);
4     wave=filter(5*ones(1,100),1,data);
```

Рис. 17: Файл maptowave.m

Прописывание функций

В файлах unipolar.m, ami.m, bipolarnrz.m, bipolarrrz.m, manchester.m, diffmanc.m пропишем соответствующие коды листингов. (рис. 18)



```
1 % calcspectre.m
2 % Функция построения спектра сигнала:
3 function spectre = calcspectre(wave)
4 Королькова А. В., Кулябов Д. С. Сетевые технологии. Лабораторный практикум 31
5 % Частота дискретизации (Гц):
6 Fd = 512;
7 Fd2 = Fd/2;
8 Fd3 = Fd/2 + 1;
9 X = fft(wave,Fd);
10 spectre = X.*conj(X)/Fd;
11 f = 1000*(0:Fd2)/Fd;
12 plot(f,spectre(1:Fd3));
```

Рис. 18: Прописывание функций

Запустим главный скрипт `main.m`. В каталоге `signal` получены файлы с графиками кодированного сигнала, в каталоге `sync` — файлы с графиками, иллюстрирующими свойства самосинхронизации, в каталоге `spectre` — файлы с графиками спектров сигналов. (рис. 19)

Полученные графики

▼	signal		File Folder	08.09.2025 14:27	main
	ami.png	12,71 KiB	Portable Netwo...	08.09.2025 14:29	
	unipolar.png	13,26 KiB	Portable Netwo...	08.09.2025 14:29	
	manchester....	13,45 KiB	Portable Netwo...	08.09.2025 14:29	
	diffmanc.png	13,96 KiB	Portable Netwo...	08.09.2025 14:29	
	bipolarrz.png	14,04 KiB	Portable Netwo...	08.09.2025 14:29	
	bipolarnrz.p...	14,51 KiB	Portable Netwo...	08.09.2025 14:29	
▼	spectre		File Folder	08.09.2025 14:27	1:
	ami.png	14,96 KiB	Portable Netwo...	08.09.2025 14:29	1:
	unipolar.png	16,37 KiB	Portable Netwo...	08.09.2025 14:29	1:
	manchester....	16,69 KiB	Portable Netwo...	08.09.2025 14:29	1:
	bipolarnrz.p...	17,82 KiB	Portable Netwo...	08.09.2025 14:29	1:
	bipolarrz.png	18,02 KiB	Portable Netwo...	08.09.2025 14:29	1:
	diffmanc.png	19,07 KiB	Portable Netwo...	08.09.2025 14:29	1:
▼	sync		File Folder	08.09.2025 14:27	2:
	ami.png	12,77 KiB	Portable Netwo...	08.09.2025 14:29	2:
	unipolar.png	13,23 KiB	Portable Netwo...	08.09.2025 14:29	2:
	manchester....	13,82 KiB	Portable Netwo...	08.09.2025 14:29	2:
	diffmanc.png	14,30 KiB	Portable Netwo...	08.09.2025 14:29	2:
	bipolarrz.png	14,35 KiB	Portable Netwo...	08.09.2025 14:29	2:
	bipolarnrz.p...	14,48 KiB	Portable Netwo...	08.09.2025 14:29	2:

Рис. 19: Полученные графики

Задание с другой частотой дискретизации.

1) Что будет, если взять частоту дискретизации меньше 80 Гц?

При снижении частоты дискретизации ниже 80 Гц высокочастотные составляющие сигнала могут быть потеряны, что приведёт к искажению результатов спектрального анализа.

Задание с другой частотой дискретизации

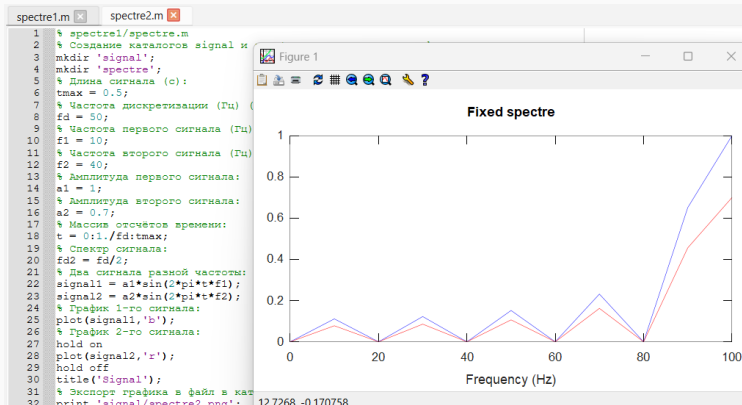


Рис. 20: Задание с другой частотой дискретизации

Вывод

В рамках лабораторной работы мы изучили методов кодирования и модуляции сигналов с помощью высокоуровневого языка программирования Octave. Определили спектра и параметров сигнала. Продемонстрировали принципы модуляции сигнала на примере аналоговой амплитудной модуляции. Исследовали свойства самосинхронизации сигнала.