

Лабораторная работа №1

Тема «Методы кодирования и модуляция
сигналов»

по дисциплине «Сетевые технологии»

Выполнил: Щербак Маргарита Романовна

Студент группы: НПИбд-02-21

«07» сентября 2023г.

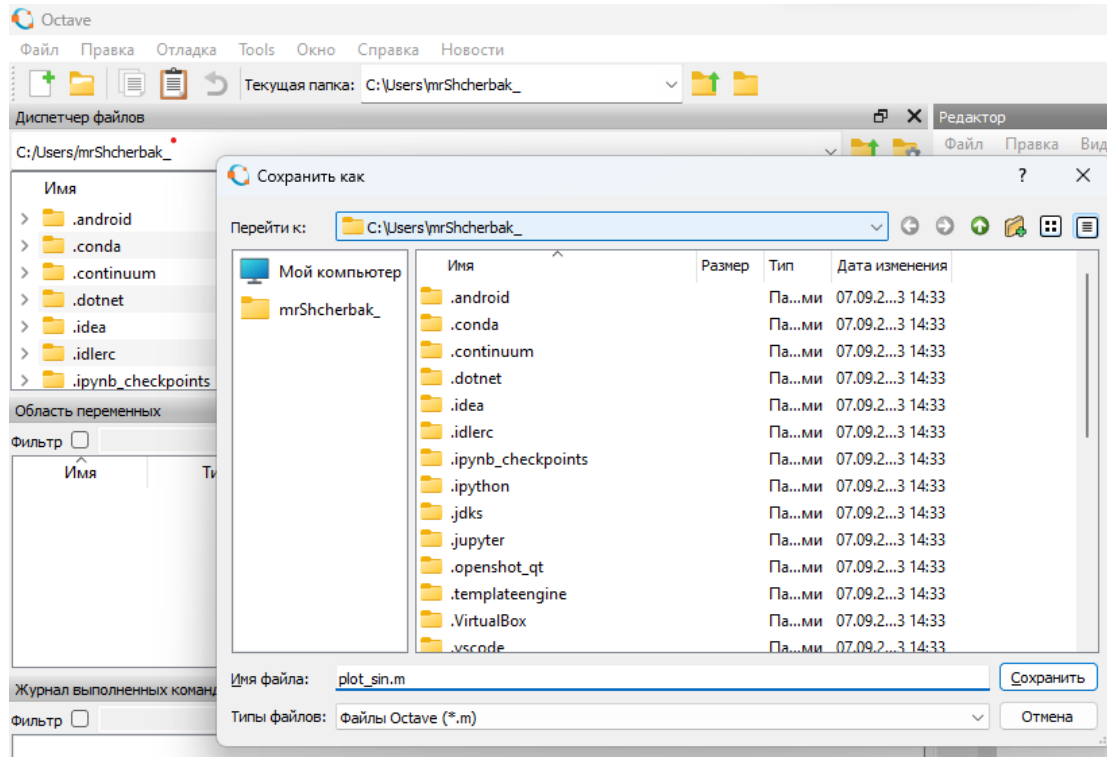
Цели работы:

Изучить методы кодирования и модуляции сигналов с помощью высокоуровневого языка программирования Octave. Определить спектр и параметры сигнала. Продемонстрировать принципы модуляции сигнала на примере аналоговой амплитудной модуляции. Исследовать свойства самосинхронизации сигнала.

Выполнение работы

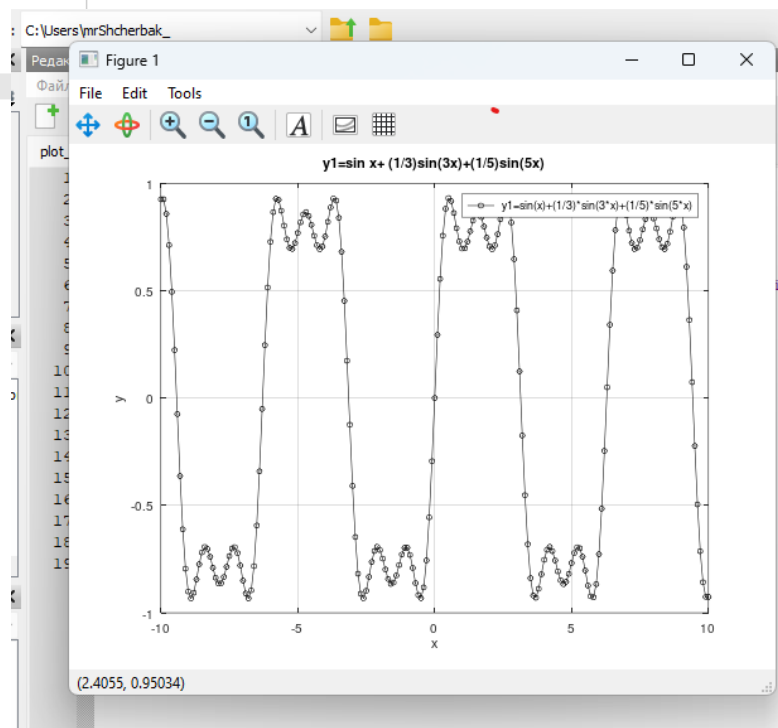
Построение графиков в Octave

Перешла в окно редактора и создала новый сценарий с именем plot_sin.m, затем сохранила его в свой рабочий каталог

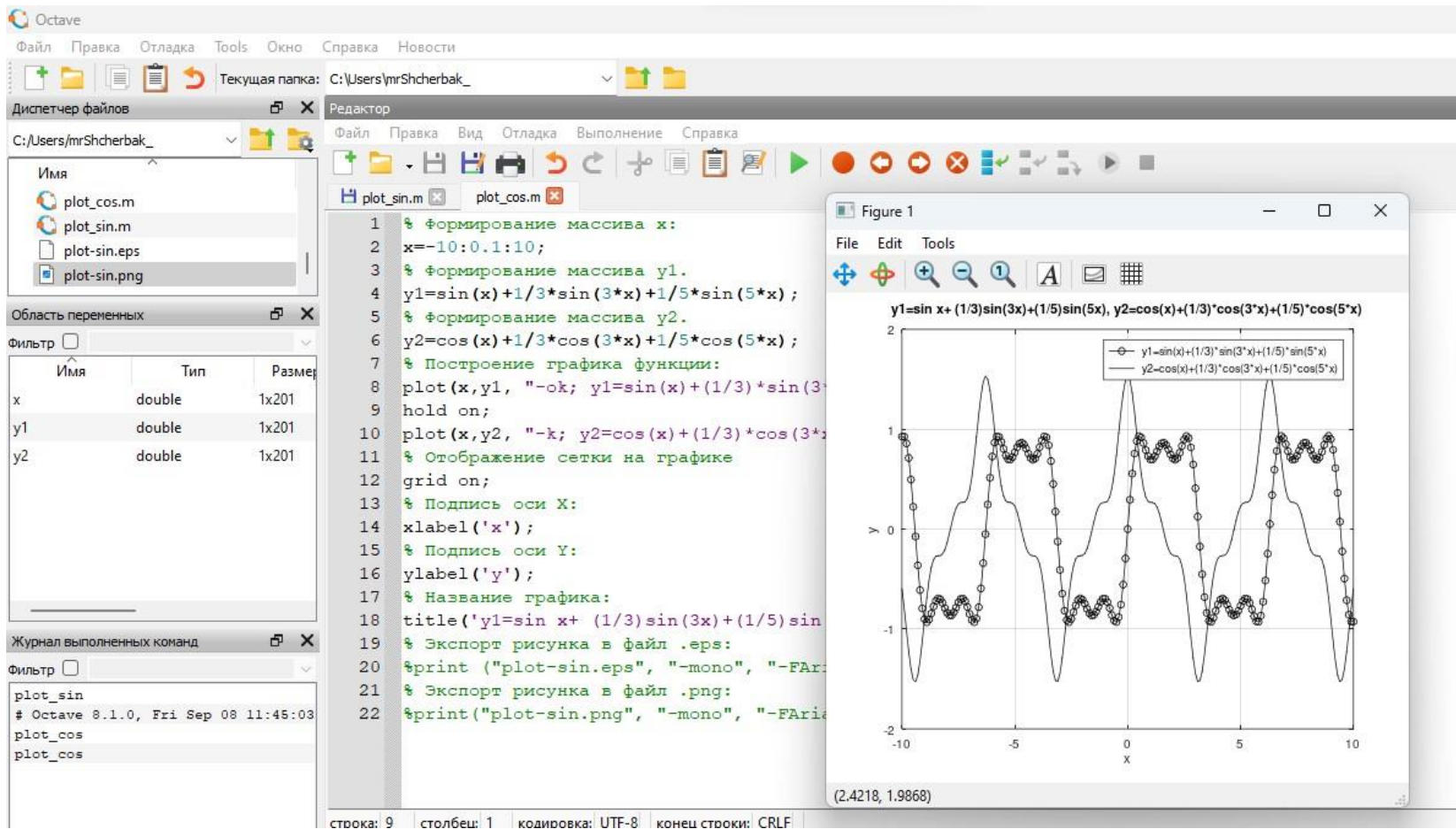


```
Справка  Новости
C:\Users\mrShcherbak_
Редактор
Файл  Правка  Вид  Отладка  Выполнение  Справка
plot_sin.m
1 % Формирование массива x:
2 x=-10:0.1:10;
3 % Формирование массива y.
4 y1=sin(x)+1/3*sin(3*x)+1/5*sin(5*x);
5 % Построение графика функции:
6 plot(x,y1, "-ok; y1=sin(x)+(1/3)*sin(3*x)+(1/5)*sin(5*x);", "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
```

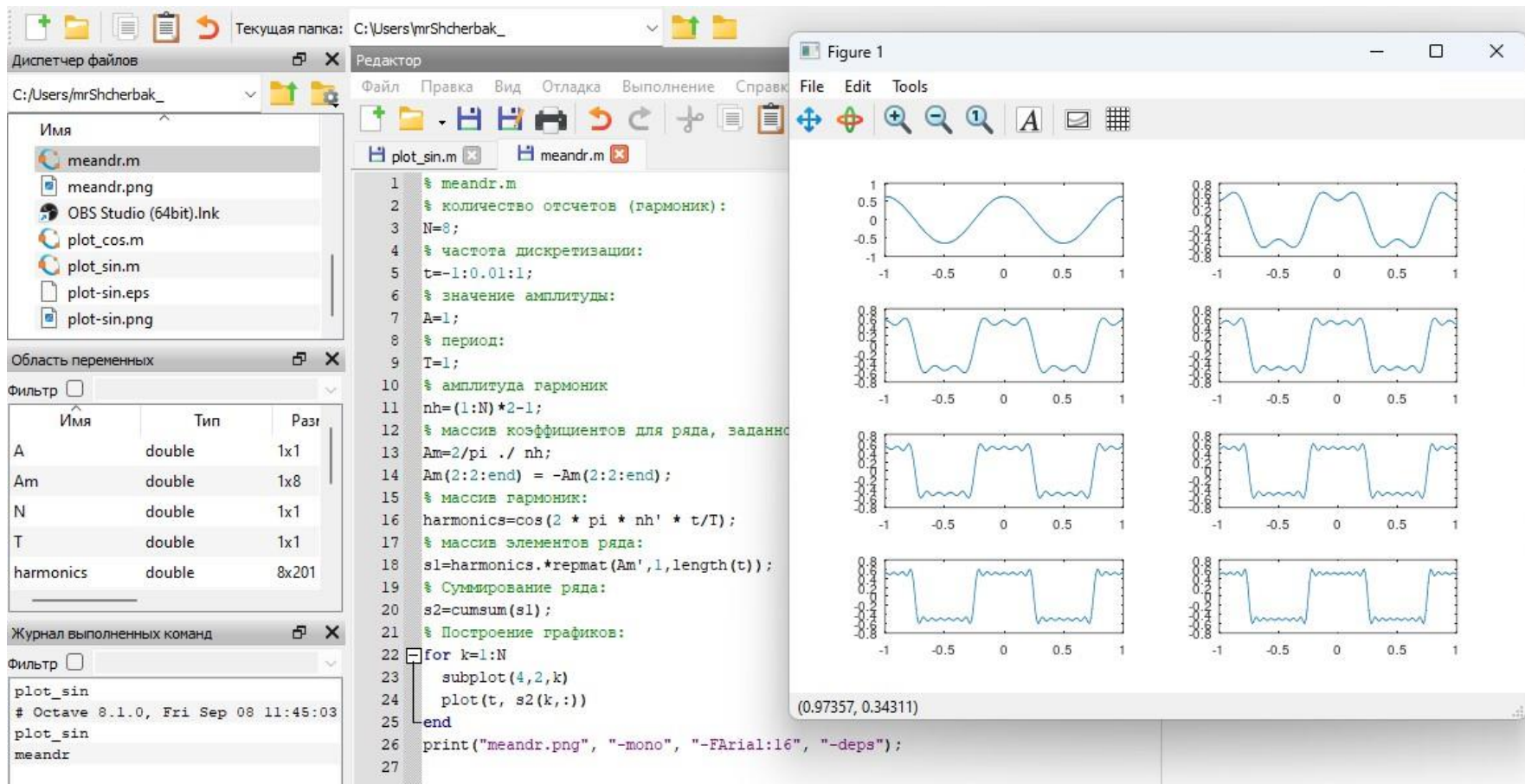
График функции $y = \sin x + 1/3 \sin 3x + 1/5 \sin 5x$ на интервале $[-10; 10]$



Создала сценарий под названием plot_cos.m и изменила его так, чтобы на одном графике располагались отличающиеся по типу линий графики функций $y_1 = \sin x + 1/3 \sin 3x + 1/5 \sin 5x$, $y_2 = \cos x + 1/3 \cos 3x + 1/5 \cos 5x$

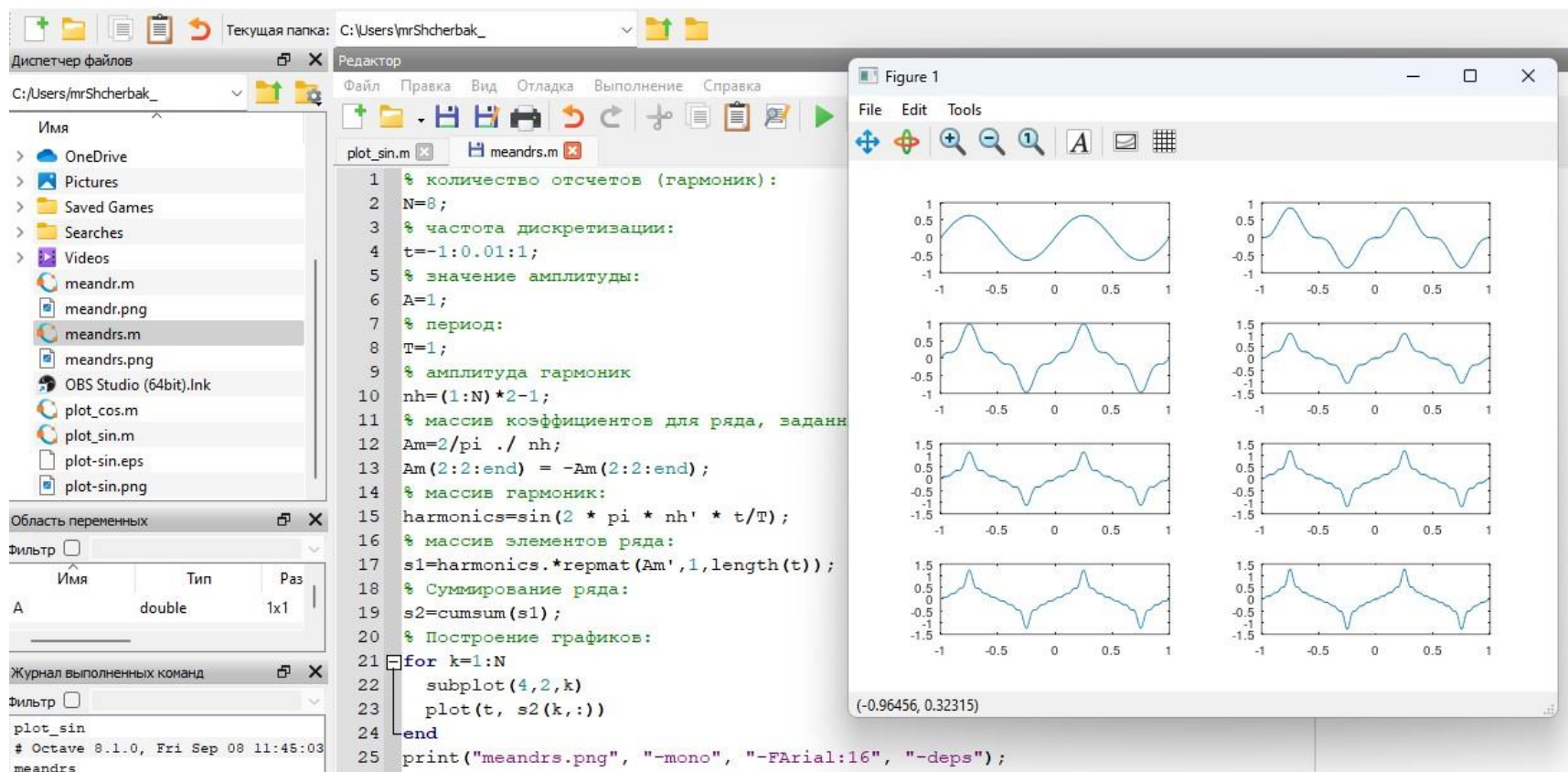


Разложение импульсного сигнала в частичный ряд Фурье



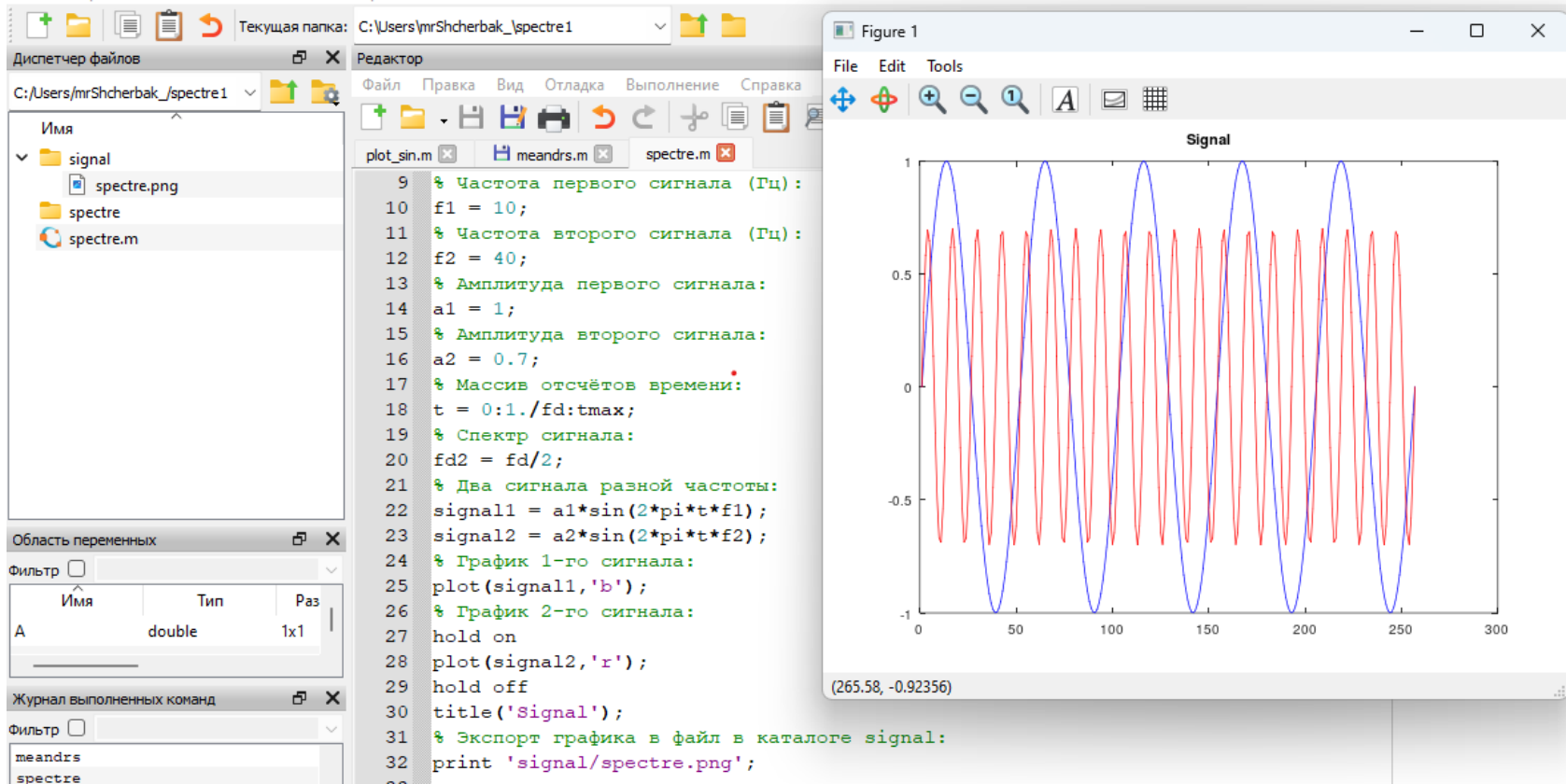
Листинг m-файла и графики меандра, реализованные с различным количеством гармоник

Скорректировала код для реализации меандра через синусы

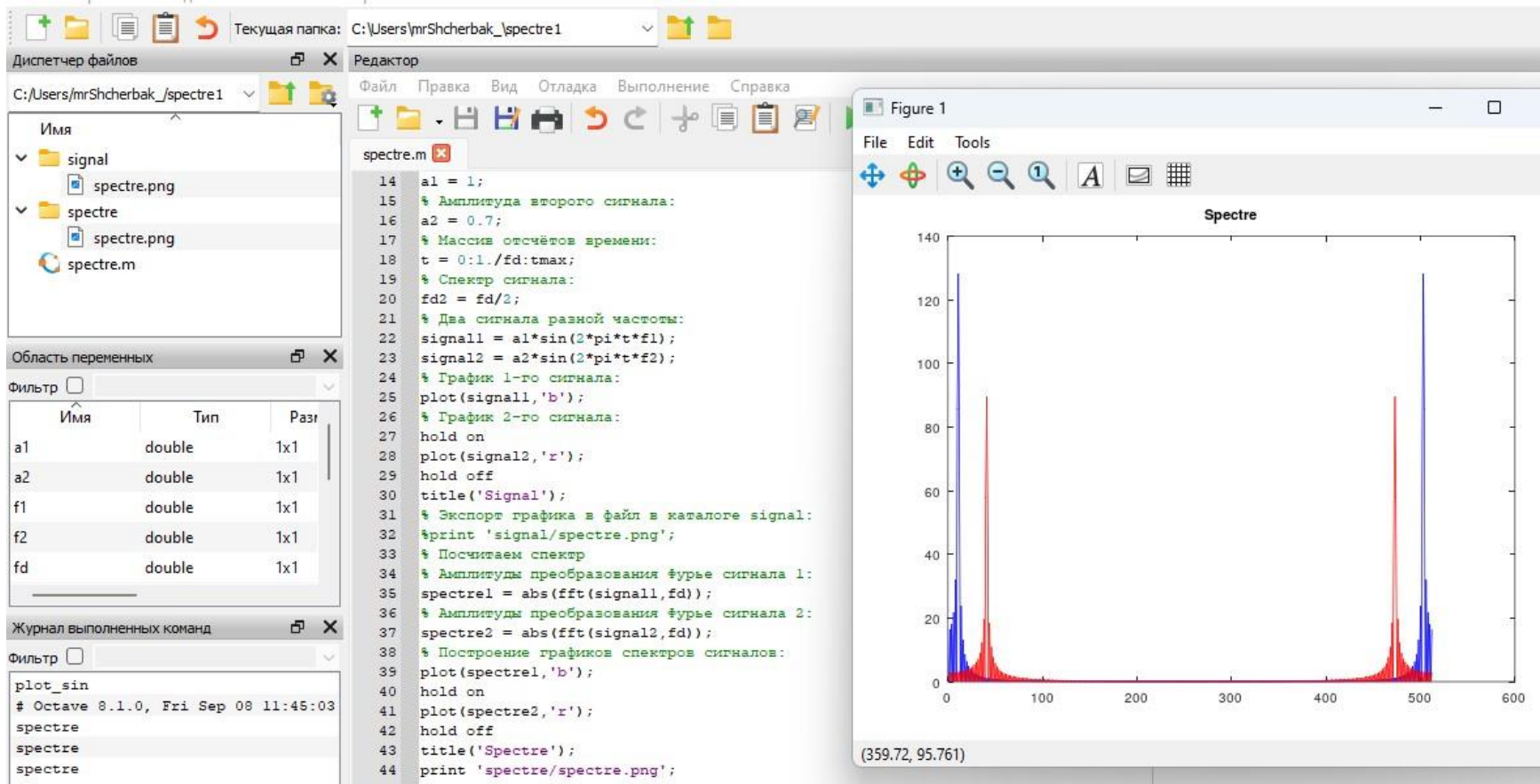


Листинг m-файла через синусы, графики меандра и файл meandrs в формате .png

Определение спектра и параметров сигнала

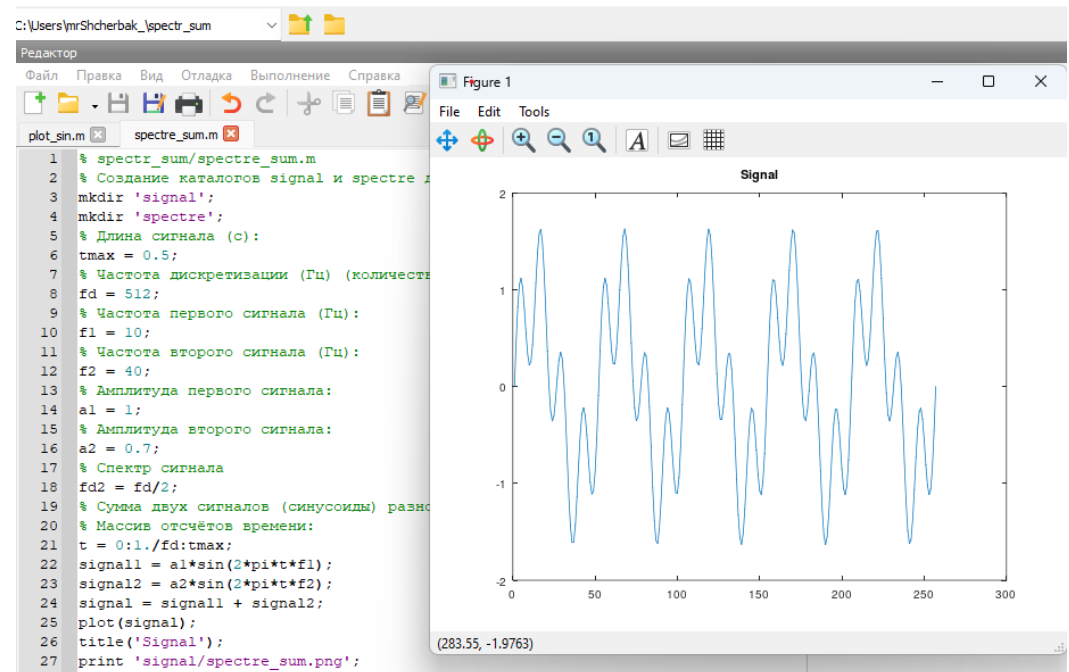
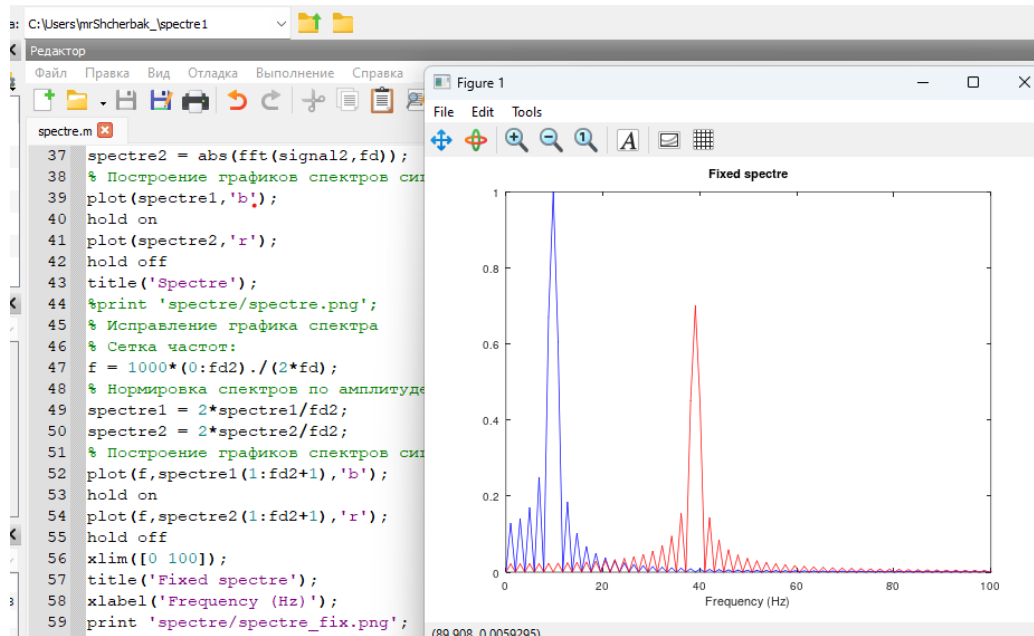


Листинг m-файла и два синусоидальных сигнала разной частоты

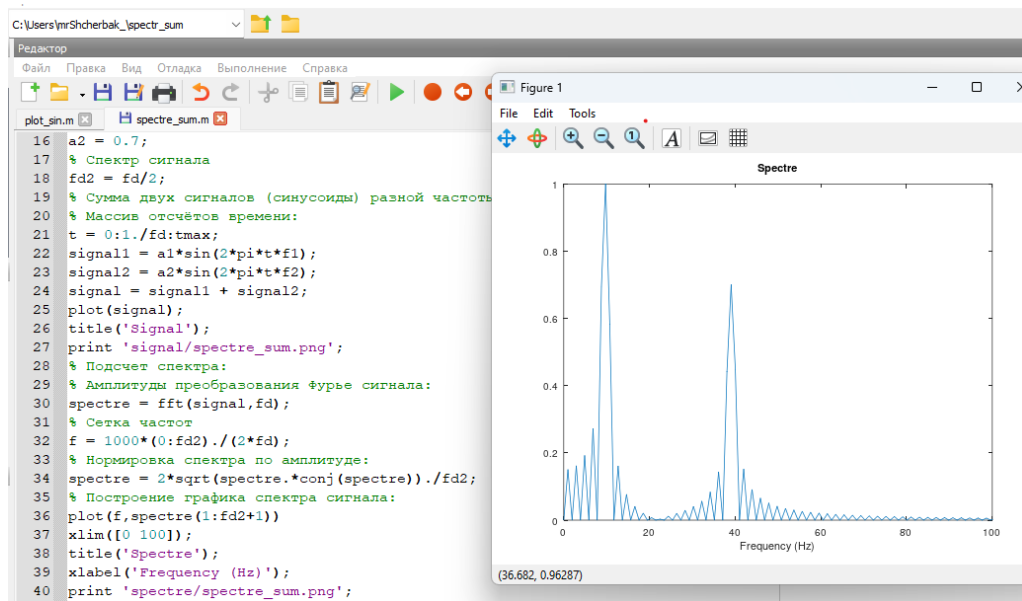


Листинг m-файла и график спектров синусоидальных сигналов

Листинг m-файла и исправленный график спектров синусоидальных сигналов

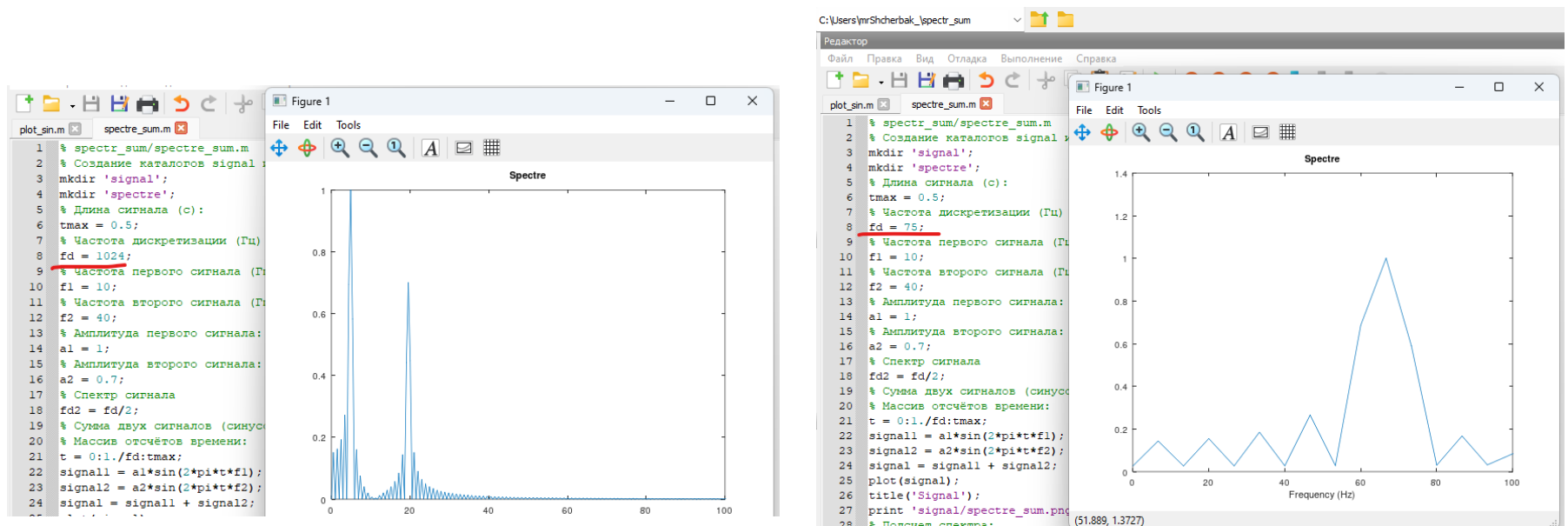


Листинг файла `spectre_sum.m` и
суммарный сигнал



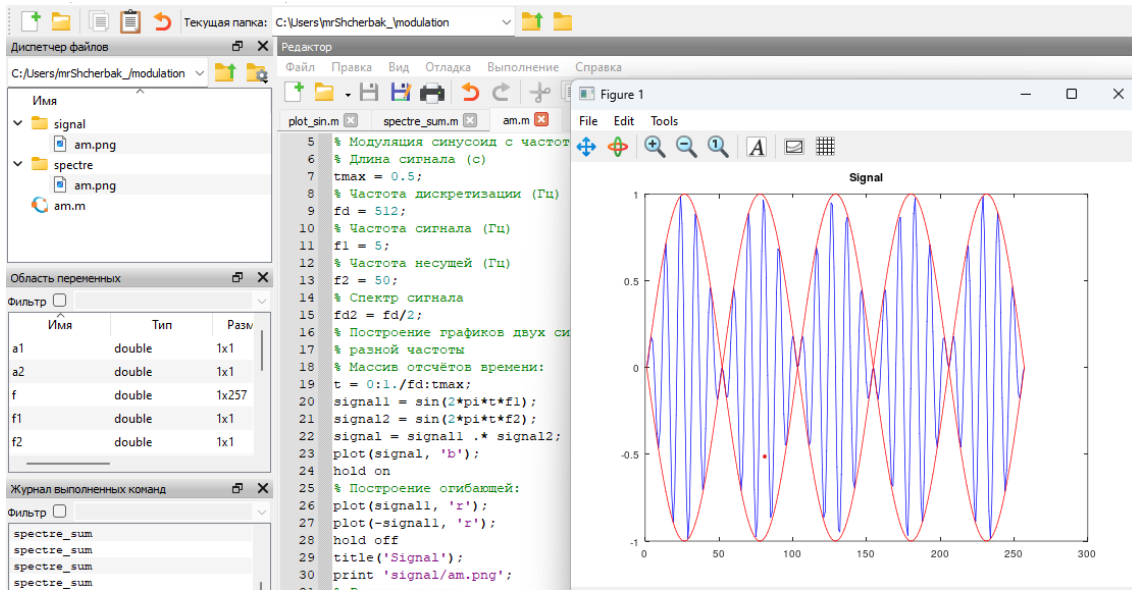
Листинг файла `spectre_sum.m` и спектр суммарного сигнала

Выполнила задание с другой частотой дискретизации – взяла значения 75 и 1024 Гц

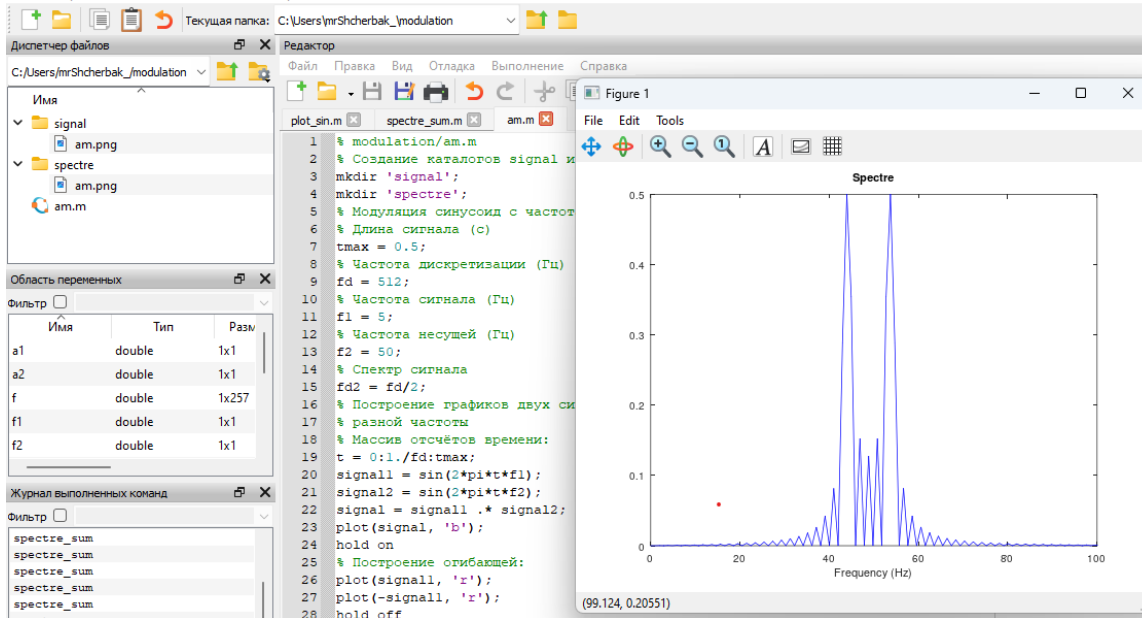


Амплитудная модуляция

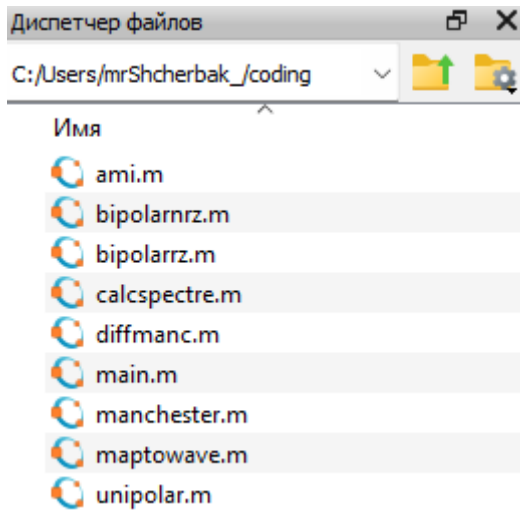
Листинг файла am.m.
Сигнал и огибающая при
амплитудной модуляции



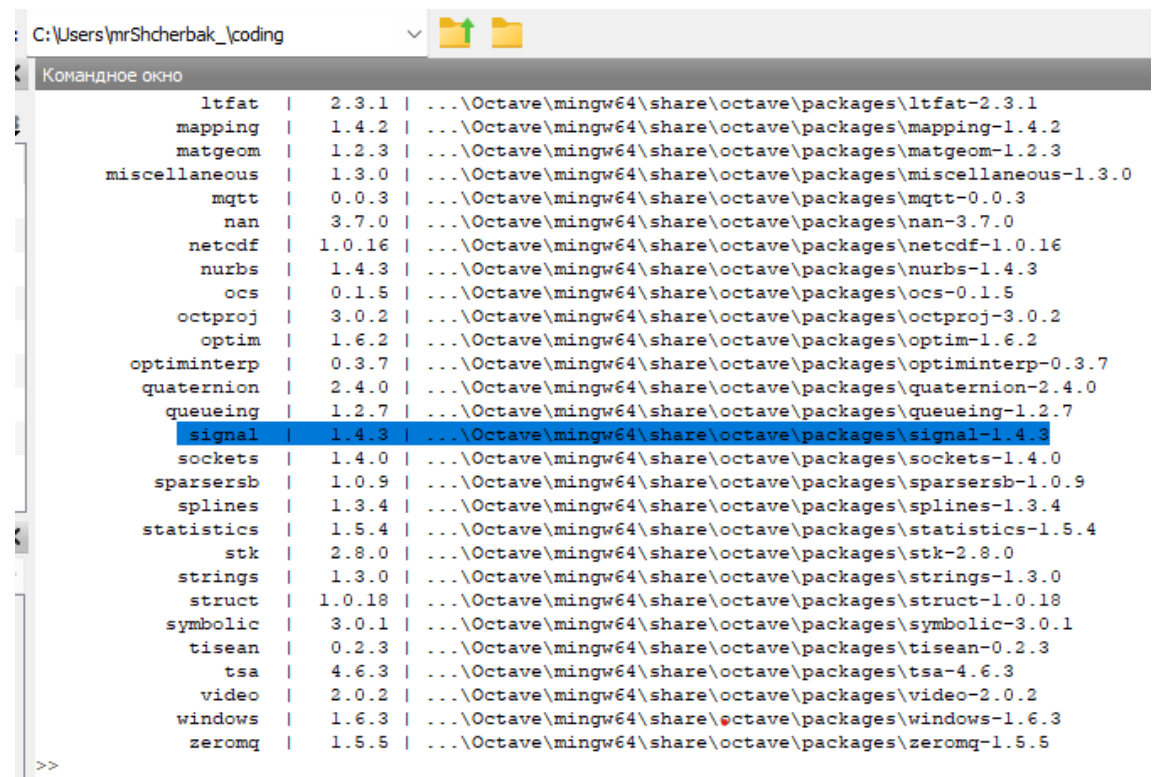
Листинг файла am.m и
спектр сигнала при
амплитудной модуляции



Кодирование сигнала. Исследование свойства самосинхронизации сигнала



Установлен пакет расширений signal



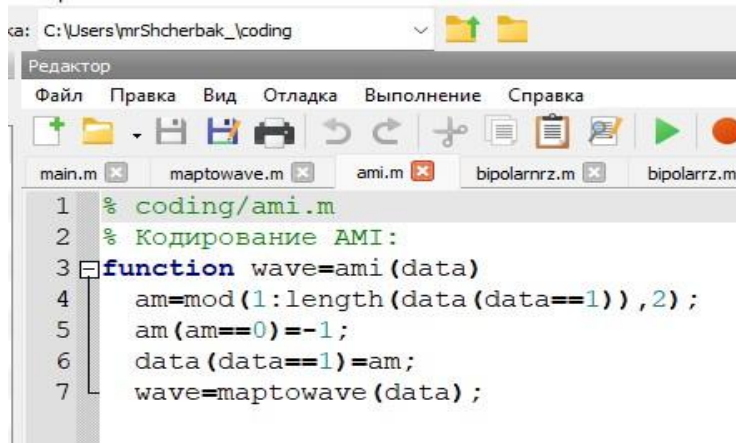
```
1 % coding/main.m
2 % Подключение пакета signal:
3 pkg load signal;
4 % Входная кодовая последовательность:
5 data=[0 1 0 0 1 1 0 0 0 1 1 0];
6 % Входная кодовая последовательность для проверки свойства самосинхронизации:
7 data_sync=[0 0 0 0 0 0 0 1 1 1 1 1];
8 % Входная кодовая последовательность для построения спектра сигнала:
9 data_spectre=[0 1 0 1 0 1 0 1 0 1 0 1];
10 % Создание каталогов signal, sync и spectre для размещения графиков:
11 mkdir 'signal';
12 mkdir 'sync';
13 mkdir 'spectre';
14 axis("auto");
15 % Униполярное кодирование
16 wave=unipolar(data);
17 plot(wave);
18 ylim([-1 6]);
19 title('Unipolar');
20 print 'signal/unipolar.png';
21 % Кодирование ami
22 wave=ami(data);
23 plot(wave)
```

В файле main.m
подключила пакет
signal

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

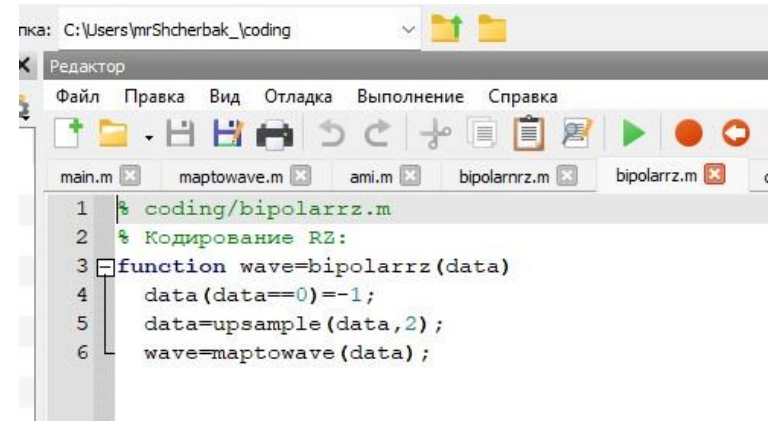
```
1 % coding/unipolar.m
2 % Униполярное кодирование:
3 function wave=unipolar(data)
4     wave=maptowave(data);
```


В файлах unipolar.m, ami.m, bipolarnrz.m, bipolarrrz.m, manchester.m, diffmanc.m прописала соответствующие функции преобразования кодовой последовательности data с вызовом функции maptowave для построения соответствующего графика



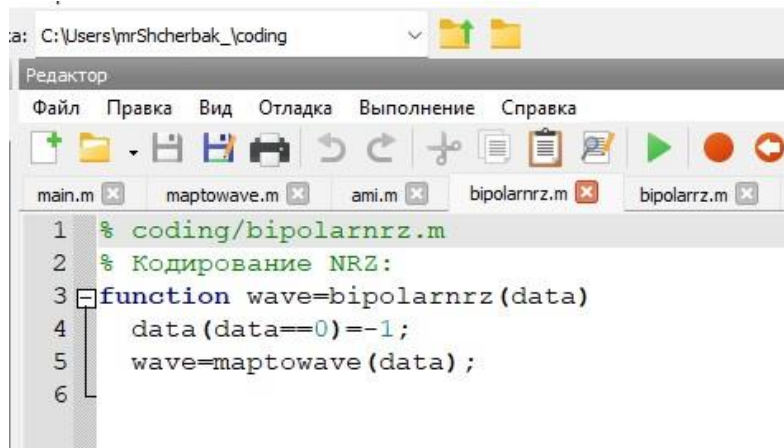
Скриншот редактора MATLAB, показывающий код файла ami.m. В заголовке окна указан путь к папке coding. В меню и панели инструментов видны стандартные опции. Таблицей вкладок открыты файлы main.m, maptowave.m, ami.m, bipolarnrz.m и bipolarrrz.m. Код в файле ami.m:

```
1 % coding/ami.m
2 % Кодирование AMI:
3 function wave=ami(data)
4     am=mod(1:length(data(data==1)),2);
5     am(am==0)=-1;
6     data(data==1)=am;
7     wave=maptowave(data);
```



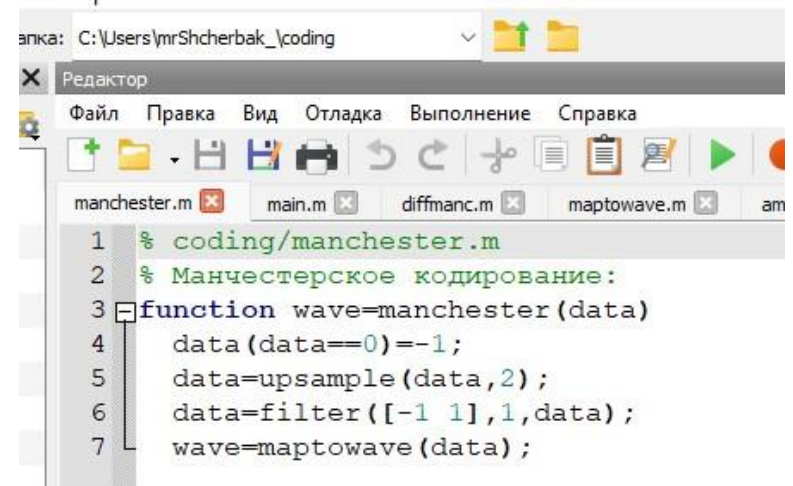
Скриншот редактора MATLAB, показывающий код файла bipolarrrz.m. В заголовке окна указан путь к папке coding. В меню и панели инструментов видны стандартные опции. Таблицей вкладок открыты файлы main.m, maptowave.m, ami.m, bipolarnrz.m и bipolarrrz.m. Код в файле bipolarrrz.m:

```
1 % coding/bipolarrrz.m
2 % Кодирование RZ:
3 function wave=bipolarrrz(data)
4     data(data==0)=-1;
5     data=upsample(data,2);
6     wave=maptowave(data);
```



Скриншот редактора MATLAB, показывающий код файла bipolarnrz.m. В заголовке окна указан путь к папке coding. В меню и панели инструментов видны стандартные опции. Таблицей вкладок открыты файлы main.m, maptowave.m, ami.m, bipolarnrz.m и bipolarrrz.m. Код в файле bipolarnrz.m:

```
1 % coding/bipolarnrz.m
2 % Кодирование NRZ:
3 function wave=bipolarnrz(data)
4     data(data==0)=-1;
5     wave=maptowave(data);
6
```



Скриншот редактора MATLAB, показывающий код файла manchester.m. В заголовке окна указан путь к папке coding. В меню и панели инструментов видны стандартные опции. Таблицей вкладок открыты файлы manchester.m, main.m, diffmanc.m, maptowave.m и ami.m. Код в файле manchester.m:

```
1 % coding/manchester.m
2 % Манчестерское кодирование:
3 function wave=manchester(data)
4     data(data==0)=-1;
5     data=upsample(data,2);
6     data=filter([-1 1],1,data);
7     wave=maptowave(data);
```


Папка: C:\Users\mrShcherbak_\coding

Редактор

Файл Правка Вид Отладка Выполнение Справка

manchester.m main.m diffmanc.m maptowave.m am

```
1 % coding/manchester.m
2 % Манчестерское кодирование:
3 function wave=manchester(data)
4     data(data==0)=-1;
5     data=upsample(data,2);
6     data=filter([-1 1],1,data);
7     wave=maptowave(data);
```

Папка: C:\Users\mrShcherbak_\coding

Редактор

Файл Правка Вид Отладка Выполнение Справка

main.m maptowave.m ami.m bipolarrrz.m bipolarrrz.m

```
1 % calcspectre.m
2 % функция построения спектра сигнала:
3 function spectre = calcspectre(wave)
4 % Частота дискретизации (Гц):
5     Fd = 512;
6     Fd2 = Fd/2;
7     Fd3 = Fd/2 + 1;
8     X = fft(wave,Fd);
9     spectre = X.*conj(X)/Fd;
10    f = 1000*(0:Fd2)/Fd;
11    plot(f,spectre(1:Fd3));
12    xlabel('Frequency (Hz)');
```

Папка: C:\Users\mrShcherbak_\coding

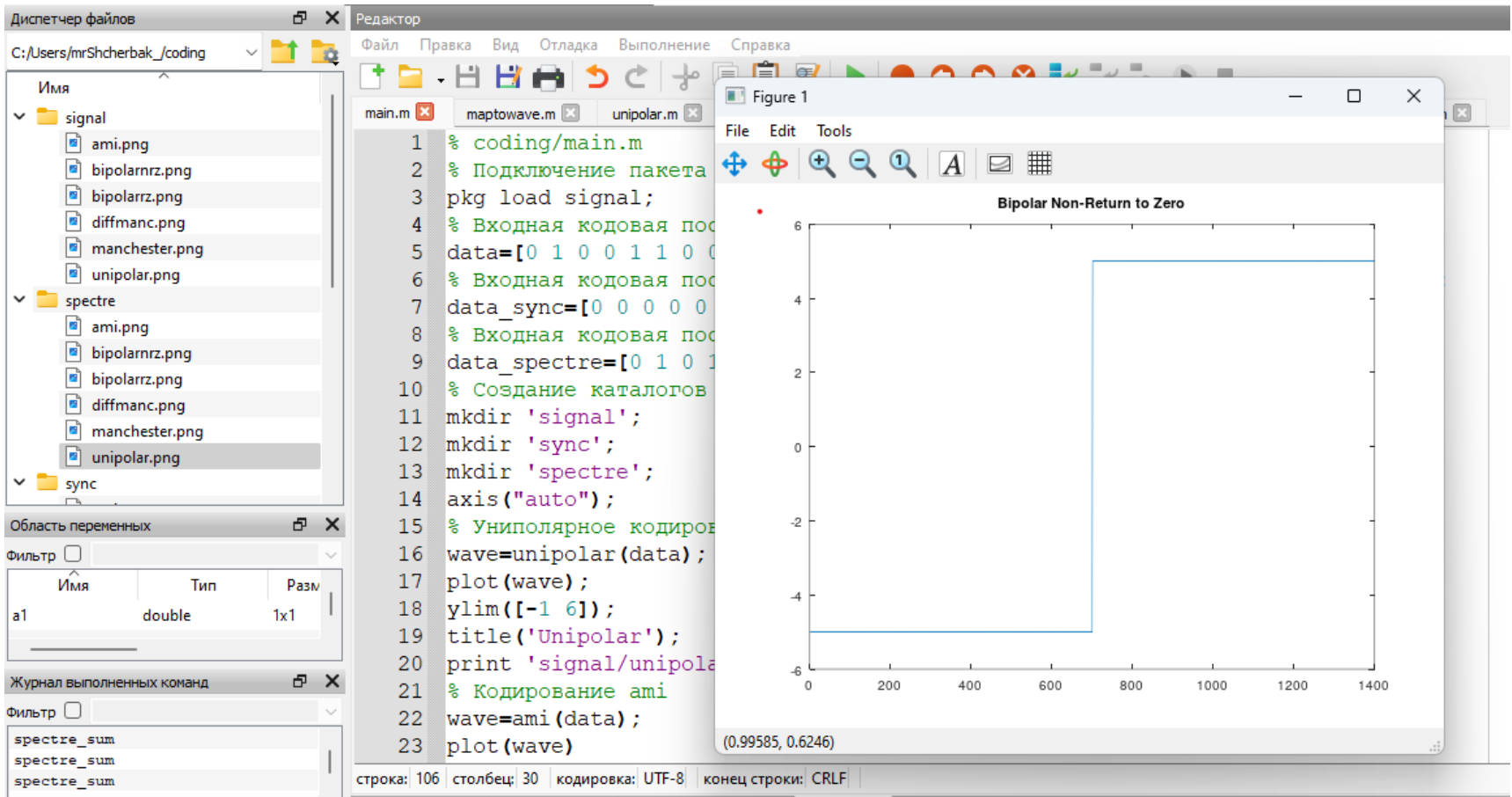
Редактор

Файл Правка Вид Отладка Выполнение Справка

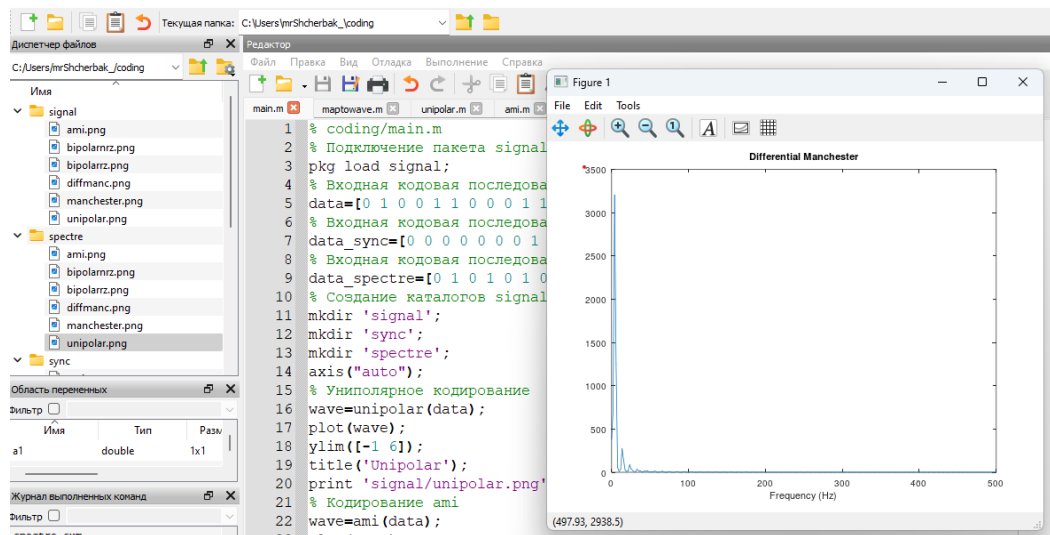
main.m diffmanc.m maptowave.m ami.m bipolarrrz.m bipolarrrz.m

```
1 % coding/diffmanc.m
2 % Дифференциальное манчестерское кодирование
3 function wave=diffmanc(data)
4     data=filter(1,[1 1],data);
5     data=mod(data,2);
6     wave=manchester(data);
```

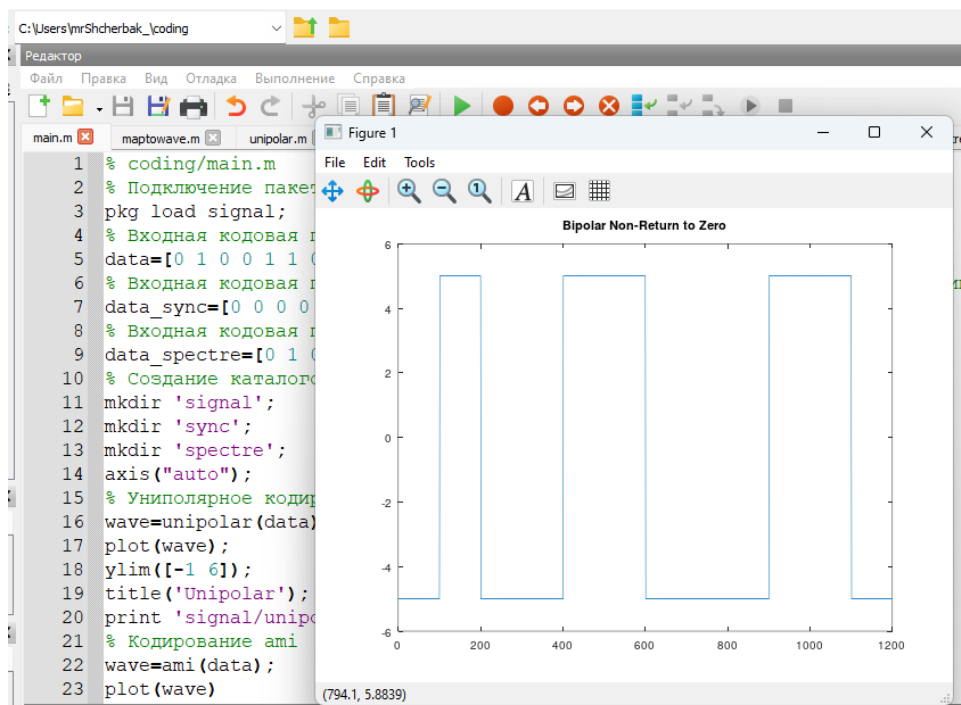
Запустила главный скрипт main.m. В каталоге signal получены файлы с графиками кодированного сигнала, в каталоге sync — файлы с графиками, иллюстрирующими свойства самосинхронизации, в каталоге spectre — файлы с графиками спектров сигналов.



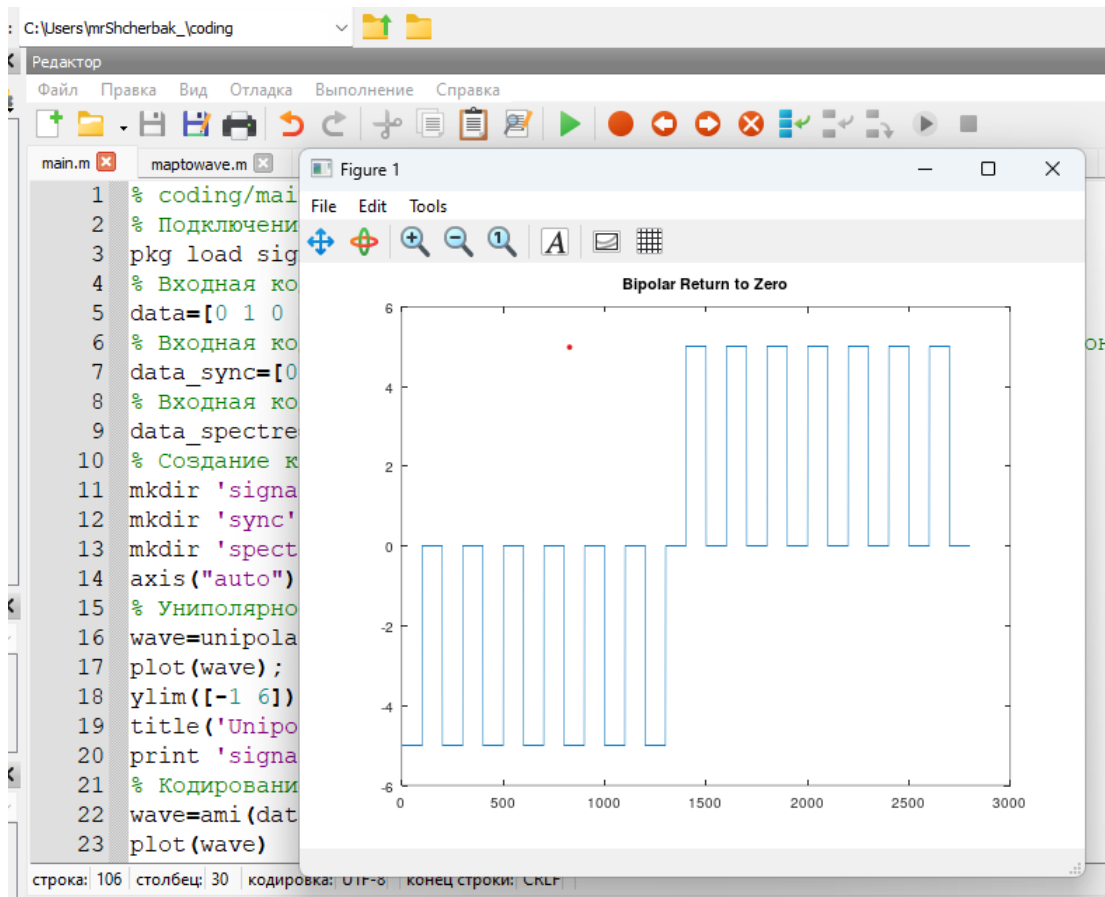
Кодирование NRZ: нет самосинхронизации



Дифференциальное
манчестерское кодирование:
спектр сигнала



Кодирование NRZ



Кодирование RZ: есть самосинхронизация

Вывод: таким образом, в ходе выполнения л/р №1, я изучила методы кодирования и модуляции сигналов с помощью высокоуровневого языка программирования Octave. Определила спектр и параметры сигнала. Продемонстрировала принципы модуляции сигнала на примере аналоговой амплитудной модуляции. Исследовала свойства самосинхронизации сигнала.