

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ  
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»

**Московский институт электроники и математики им. А.Н. Тихонова**  
Департамент электронной инженерии

**ОТЧЕТ**  
**ПО ДОМАШНЕЙ РАБОТЕ №2**  
по дисциплине «Основы построения инфокоммуникационных систем и сетей»

Выполнил:  
Цуркан Андрей Александрович,  
группа БИТ202

Москва 2022 г.

# Оглавление

Задание на выполнение домашней работы	3
Структурная схема функционирования реализуемой системы передачи	4
Листинг программы (Python)	5
Результаты имитационного моделирования	25
Выводы	34

## **Задание на выполнение домашней работы**

Построить модель, имитирующую функционирование системы передачи с частотным и временным разделением каналов (две трехканальные системы формируют групповые сигналы на основе временного разделения, затем эти два групповых сигнала поступают на вход двухканальной системы с частотным разделением каналов). На выходе приемного тракта выделить все 6 исходных первичных сигналов.

## Структурная схема функционирования реализуемой системы передачи

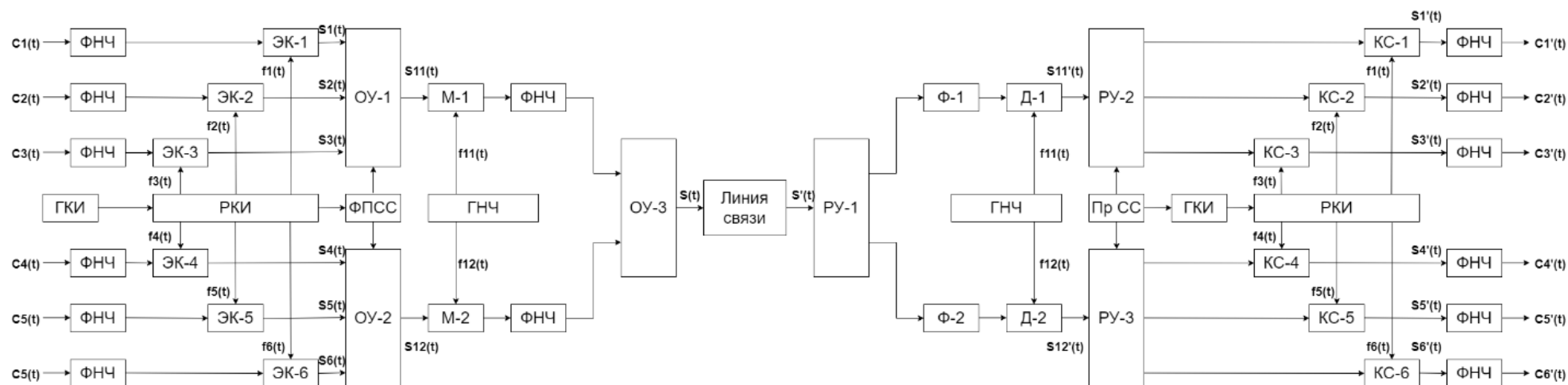


Рис.1. Структурная схема функционирования реализуемой системы передачи

ЭК – каналные электронные ключи;

КС – каналные селекторы;

ОУ – объединяющее устройство;

РУ – развязывающее устройство;

ГКИ – генератор канальных импульсов;

ФПСС – формирователь и передатчик синхросигнала;

РКИ – распределитель канальных импульсов;

ПрСС – приемник синхросигнала

## Листинг программы (Python)

```
import numpy as np
from math import *
import matplotlib.pyplot as plt

def f2w(f):
    return 2.0 * pi * f

def wb_pulse(t, Tc, fn, fv):
    freq = (fv + fn) * 0.5
    dt = 1.0 / (fv-fn)
    return exp(-(0.5*Tc-t)**2/dt**2*0.5)*sin(2.0*pi*freq*t)

def filter(time, signal, fl, fh):
    n = len(signal)
    freq = np.fft.fftfreq(n, time[1]-time[0])
    spectr = np.fft.fft(signal)
    for i in range(n):
        if not fl <= abs(freq[i]) <= fh:
            spectr[i] *= 0+0j
    return np.fft.ifft(spectr)

nch = 3
npp = 50

#-----Частоты первичных гармонических сигналов в каналах 1,
#2, 3 [кГц] обеих систем-----#
fc = np.array([1.0, 2.0, 3.0, 4.0, 5.0, 6.0])
#-----#

#-----Частоты сигналов-переносчиков в каналах 1, 2, 3
# [кГц] обеих систем-----#
fn = np.array([15.0, 20.5])
#-----#

#----int(input('Число временных отсчетов (должно быть не менее
#{0: .0f}) '.format(8*T*npp)))-----#
T = float(input('Временной интервал, мс '))
```

```

n = int(8*T)*npp*nch
#-----#
#-----Массивы первичных сигналов, поступающих на вход
каналов 1, 2, 3 обеих систем-----#
#-----1 система-----#
sig1 = [0] * n
sig2 = [0] * n
sig3 = [0] * n
#-----#
#-----2 система-----#
sig4 = [0] * n
sig5 = [0] * n
sig6 = [0] * n
#-----#
#-----#
#-----#
#-----Массивы модулированных первичных, канальных и группового
сигналов в тракте передачи-----#
#-----1 система-----#
mch_1_1 = [0] * n
mch_1_2 = [0] * n
mch_1_3 = [0] * n
#-----#
#-----2 система-----#
mch_2_1 = [0] * n
mch_2_2 = [0] * n
mch_2_3 = [0] * n
#-----#
#-----#
#-----#
#-----Массив моментов времени для отсчетов
сигналов-----#
time = [0] * n
#-----#
#-----#
#-----Шаг дискретизации по
времени-----#
h = T / (n-1)

```

```

#-----#
-----#

#-----Формирование первичных сигналов в
каналах систем-----#
for i in range(n):
    time[i] = i*h
    # -----1 система-----#
    sig1[i] = 1.0*cos(f2w(fc[0]-
0.2)*time[i])+1.5*cos(f2w(fc[0])*time[i])+2.0*cos(f2w(fc[0]+0.2
)*time[i])
    sig2[i] = 2.0*cos(f2w(fc[1]-
0.2)*time[i])+2.5*cos(f2w(fc[1])*time[i])+3.0*cos(f2w(fc[1]+0.2
)*time[i])
    sig3[i] = 1.5*cos(f2w(fc[2]-
0.2)*time[i])+2.0*cos(f2w(fc[2])*time[i])+2.5*cos(f2w(fc[2]+0.2
)*time[i])
    # -----#

    # -----2 система-----#
    sig4[i] = 1.2*cos(f2w(fc[3]-
0.2)*time[i])+1.7*cos(f2w(fc[3])*time[i])+2.2*cos(f2w(fc[3]+0.2
)*time[i])
    sig5[i] = 1.7*cos(f2w(fc[4]-
0.2)*time[i])+2.2*cos(f2w(fc[4])*time[i])+2.7*cos(f2w(fc[4]+0.2
)*time[i])
    sig6[i] = 2.5*cos(f2w(fc[5]-
0.2)*time[i])+3.0*cos(f2w(fc[5])*time[i])+3.5*cos(f2w(fc[5]+0.2
)*time[i])
    # -----#

#-----#
-----#

#-----ПППИ-----#
-----#

pcrp = [0] * n
#-----#
-----#

#-----Формирование канальных
сигналов-----#
for i in range(n//nch//npp):
    for j in range(nch):
        for k in range(npp):
            indx = i * nch * npp + j * npp + k

```

```

#-----Формирование ПППИ-----#
pcrp[indx] = 1.0 if k < 0.75*npp else 0
#-----#

#-----Канальные электронные ключи-----#
if j == 0:
    mch_1_1[indx] = pcrp[indx] * sig1[indx]
    mch_2_1[indx] = pcrp[indx] * sig4[indx]
elif j == 1:
    mch_1_2[indx] = pcrp[indx] * sig2[indx]
    mch_2_2[indx] = pcrp[indx] * sig5[indx]
elif j == 2:
    mch_1_3[indx] = pcrp[indx] * sig3[indx]
    mch_2_3[indx] = pcrp[indx] * sig6[indx]
#-----#

#-----#

#-----Формирование группового сигнала (объединение
канальных сигналов)-----#
tgrp_123 = np.array(np.array(mch_1_1) + np.array(mch_1_2) +
np.array(mch_1_3))
tgrp_456 = np.array(np.array(mch_2_1) + np.array(mch_2_2) +
np.array(mch_2_3))
#-----#

#-----Расчет спектра сигналов-----#

sp_freq = np.fft.fftfreq(n, T/n)

#-----1 система-----#
sp_sig1 = np.fft.fft(sig1)
sp_sig2 = np.fft.fft(sig2)
sp_sig3 = np.fft.fft(sig3)
#-----#

#-----2 система-----#
sp_sig4 = np.fft.fft(sig4)
sp_sig5 = np.fft.fft(sig5)
sp_sig6 = np.fft.fft(sig6)
#-----#

sp_pcrp = np.fft.fft(pcrp)

```



```

#-----1 система-----#
sp_mch_1_1 = np.fft.fft(mch_1_1)
sp_mch_1_2 = np.fft.fft(mch_1_2)
sp_mch_1_3 = np.fft.fft(mch_1_3)
#-----#

#-----2 система-----#
sp_mch_2_1 = np.fft.fft(mch_2_1)
sp_mch_2_2 = np.fft.fft(mch_2_2)
sp_mch_2_3 = np.fft.fft(mch_2_3)
#-----#

#-----Групповые сигналы-----#
sp_tgrp123 = np.fft.fft(tgrp_123)
sp_tgrp456 = np.fft.fft(tgrp_456)
#-----#
#-----#
#-----#

#-----Спектры исходных первичных сигналов на
входе СП ЧРК-----#
#-----1 система-----#
sp_sig1 = np.hypot(sp_sig1.real, sp_sig1.imag)/n*2
sp_sig2 = np.hypot(sp_sig2.real, sp_sig2.imag)/n*2
sp_sig3 = np.hypot(sp_sig3.real, sp_sig3.imag)/n*2
#-----#

#-----2 система-----#
sp_sig4 = np.hypot(sp_sig4.real, sp_sig4.imag)/n*2
sp_sig5 = np.hypot(sp_sig5.real, sp_sig5.imag)/n*2
sp_sig6 = np.hypot(sp_sig6.real, sp_sig6.imag)/n*2
#-----#
#-----#
#-----#

#-----Спектр ПППИ-----#
#-----#
sp_pcrp = np.hypot(sp_pcrp.real, sp_pcrp.imag)/n*2
#-----#
#-----#

#-----Спектры модулированных сигналов в
каналах 1, 2, 3-----#
#-----1 система-----#
sp_mch_1_1 = np.hypot(sp_mch_1_1.real, sp_mch_1_1.imag)/n*2

```

```

sp_mch_1_2 = np.hypot(sp_mch_1_2.real, sp_mch_1_2.imag)/n*2
sp_mch_1_3 = np.hypot(sp_mch_1_3.real, sp_mch_1_3.imag)/n*2
#-----#

#-----2 система-----#
sp_mch_2_1 = np.hypot(sp_mch_2_1.real, sp_mch_2_1.imag)/n*2
sp_mch_2_2 = np.hypot(sp_mch_2_2.real, sp_mch_2_2.imag)/n*2
sp_mch_2_3 = np.hypot(sp_mch_2_3.real, sp_mch_2_3.imag)/n*2
#-----#
#-----#
#-----#

#-----Спектр группового сигнала
на выходе-----#
sp_tgrp123 = np.hypot(sp_tgrp123.real, sp_tgrp123.imag)/n*2
sp_tgrp456 = np.hypot(sp_tgrp456.real, sp_tgrp456.imag)/n*2
#-----#
#-----#

#Построение графиков сигналов и их спектров в передающем тракте
fig, (ax0, ax1) = plt.subplots(nrows=2)
ax0.set_title('Распределения первичных сигналов система 1')
ax0.plot(time, sig1, 'tab:blue', lw=1, label='$Ch_1$')
ax0.plot(time, sig2, 'tab:orange', lw=1, label='$Ch_2$')
ax0.plot(time, sig3, 'tab:green', lw=1, label='$Ch_3$')

ax0.set_xlim(T/2-1, T/2+1)
ax0.set_xlabel("$t$, мс", fontsize=10)
ax0.set_ylabel("$V_{1,2,3}$, В", fontsize=10)
ax0.legend(loc='best')
ax1.set_title('Спектры первичных сигналов система 1')
ax1.plot(sp_freq[0:n//2], sp_sig1[0:n//2], 'tab:blue', lw=1,
label='$Ch_1$')
ax1.plot(sp_freq[0:n//2], sp_sig2[0:n//2], 'tab:orange', lw=1,
label='$Ch_2$')
ax1.plot(sp_freq[0:n//2], sp_sig3[0:n//2], 'tab:green', lw=1,
label='$Ch_3$')
ax1.set_xlim(0, 5)
ax1.set_xlabel("$f$, кГц", fontsize=10)
ax1.set_ylabel("$V_{1,2,3}$, В", fontsize=10)
ax1.legend(loc='best')
fig.tight_layout()

fig, (ax0, ax1) = plt.subplots(nrows=2)
ax0.set_title('ПППИ')

```

```

ax0.plot(time, pcrp, 'tab:blue', lw=1)

ax0.set_xlim(T/2-1, T/2+1)
ax0.set_xlabel("$t$, мс", fontsize=10)
ax0.set_ylabel("$V_{1,2,3}$, В", fontsize=10)
ax1.set_title('Спектр ПППИ')
ax1.plot(sp_freq[0:n//2], sp_pcrp[0:n//2], 'tab:blue', lw=1)

ax1.set_xlabel("$f$, кГц", fontsize=10)
ax1.set_ylabel("$V_{1,2,3}$, В", fontsize=10)
fig.tight_layout()

fig, (ax0, ax1) = plt.subplots(nrows=2)
ax0.set_title('Распределения модулированных сигналов система 1')
ax0.plot(time, mch_1_1, 'tab:blue', lw=1, label='$Ch_1$')
ax0.plot(time, mch_1_2, 'tab:orange', lw=1, label='$Ch_2$')
ax0.plot(time, mch_1_3, 'tab:green', lw=1, label='$Ch_3$')

ax0.set_xlim(T/2-1, T/2+1)
ax0.set_xlabel("$t$, мс", fontsize=10)
ax0.set_ylabel("$V_{1,2,3}$, В", fontsize=10)
ax0.legend(loc='best')
ax1.set_title('Спектры модулированных сигналов система 1')
ax1.plot(sp_freq[0:n//2], sp_mch_1_1[0:n//2], 'tab:blue', lw=1,
label='$Ch_1$')
ax1.plot(sp_freq[0:n//2], sp_mch_1_2[0:n//2], 'tab:orange',
lw=1, label='$Ch_2$')
ax1.plot(sp_freq[0:n//2], sp_mch_1_3[0:n//2], 'tab:green',
lw=1, label='$Ch_3$')
ax1.set_xlim(0, 25)
ax1.set_xlabel("$f$, кГц", fontsize=10)
ax1.set_ylabel("$V_{1,2,3}$, В", fontsize=10)
ax1.legend(loc='best')
fig.tight_layout()

fig, (ax0, ax1) = plt.subplots(nrows=2)
ax0.set_title('Групповой сигнал система 1')
ax0.plot(time, tgrp_123, 'tab:blue', lw=1)

ax0.set_xlim(T/2-1, T/2+1)
ax0.set_xlabel("$t$, мс", fontsize=10)
ax0.set_ylabel("$V_{1,2,3}$, В", fontsize=10)
ax1.set_title('Спектр группового сигнала система 1')
ax1.plot(sp_freq[0:n//2], sp_tgrp123[0:n//2], 'tab:blue', lw=1)

```

```

ax1.set_xlim(0, 25)
ax1.set_xlabel("$f$, кГц", fontsize=10)
ax1.set_ylabel("$V_{1,2,3}$, В", fontsize=10)
fig.tight_layout()

#-----#

fig, (ax0, ax1) = plt.subplots(nrows=2)
ax0.set_title('Распределения первичных сигналов система 2')
ax0.plot(time, sig4, 'tab:blue', lw=1, label='$Ch_1$')
ax0.plot(time, sig5, 'tab:orange', lw=1, label='$Ch_2$')
ax0.plot(time, sig6, 'tab:green', lw=1, label='$Ch_3$')

ax0.set_xlim(T/2-1, T/2+1)
ax0.set_xlabel("$t$, мс", fontsize=10)
ax0.set_ylabel("$V_{1,2,3}$, В", fontsize=10)
ax0.legend(loc='best')
ax1.set_title('Спектры первичных сигналов система 2')
ax1.plot(sp_freq[0:n//2], sp_sig4[0:n//2], 'tab:blue', lw=1,
label='$Ch_1$')
ax1.plot(sp_freq[0:n//2], sp_sig5[0:n//2], 'tab:orange', lw=1,
label='$Ch_2$')
ax1.plot(sp_freq[0:n//2], sp_sig6[0:n//2], 'tab:green', lw=1,
label='$Ch_3$')
ax1.set_xlim(0, 5)
ax1.set_xlabel("$f$, кГц", fontsize=10)
ax1.set_ylabel("$V_{1,2,3}$, В", fontsize=10)
ax1.legend(loc='best')
fig.tight_layout()

fig, (ax0, ax1) = plt.subplots(nrows=2)
ax0.set_title('Распределения модулированных сигналов система 2')
ax0.plot(time, mch_2_1, 'tab:blue', lw=1, label='$Ch_1$')
ax0.plot(time, mch_2_2, 'tab:orange', lw=1, label='$Ch_2$')
ax0.plot(time, mch_2_3, 'tab:green', lw=1, label='$Ch_3$')

ax0.set_xlim(T/2-1, T/2+1)
ax0.set_xlabel("$t$, мс", fontsize=10)
ax0.set_ylabel("$V_{1,2,3}$, В", fontsize=10)
ax0.legend(loc='best')
ax1.set_title('Спектры модулированных сигналов система 2')
ax1.plot(sp_freq[0:n//2], sp_mch_2_1[0:n//2], 'tab:blue', lw=1,
label='$Ch_1$')

```

```

ax1.plot(sp_freq[0:n//2], sp_mch_2_2[0:n//2], 'tab:orange',
lw=1, label='$Ch_2$')
ax1.plot(sp_freq[0:n//2], sp_mch_2_3[0:n//2], 'tab:green',
lw=1, label='$Ch_3$')
ax1.set_xlim(0, 25)
ax1.set_xlabel("$f$, кГц", fontsize=10)
ax1.set_ylabel("$V_{1,2,3}$, В", fontsize=10)
ax1.legend(loc='best')
fig.tight_layout()

fig, (ax0, ax1) = plt.subplots(nrows=2)
ax0.set_title('Групповой сигнал система 2')
ax0.plot(time, tgrp_456, 'tab:blue', lw=1)

ax0.set_xlim(T/2-1, T/2+1)
ax0.set_xlabel("$t$, мс", fontsize=10)
ax0.set_ylabel("$V_{1,2,3}$, В", fontsize=10)
ax1.set_title('Спектр группового сигнала система 2')
ax1.plot(sp_freq[0:n//2], sp_tgrp456[0:n//2], 'tab:blue', lw=1)
ax1.set_xlim(0, 25)
ax1.set_xlabel("$f$, кГц", fontsize=10)
ax1.set_ylabel("$V_{1,2,3}$, В", fontsize=10)
fig.tight_layout()
#-----
#-----#

'-----Из двух систем в
третью-----'

#-----Массивы модулированных первичных, канальных и
группового сигналов в тракте передачи-----#
mch1 = [0] * n
mch2 = [0] * n
#-----
#-----#

#-----Коэффициент модуляции-----#
m = 0.5
#-----
#-----#

#-----Модуляция первичных
сигналов-----#
for i in range(n):

```

```

    mch1[i] = (1.0 + m * tgrp_123[i])*cos(f2w(fn[0])*time[i])
    mch2[i] = (1.0 + m * tgrp_456[i])*cos(f2w(fn[1])*time[i])
#-----#
#-----Фильтрация модулированных сигналов
(выделение ВБП)-----#
tch1 = filter(time, mch1, fn[0]+0.3, fn[0]+3.4).real
tch2 = filter(time, mch2, fn[1]+0.3, fn[1]+3.4).real
#-----#
#-----Формирование группового сигнала
из двух систем-----#
tgrp_123456 = np.array(tch1.real + tch2.real).real
#-----#
#-----Расчет спектра
сигалов-----#
sp_mch1 = np.fft.fft(mch1)
sp_mch2 = np.fft.fft(mch2)

sp_tch1 = np.fft.fft(tch1)
sp_tch2 = np.fft.fft(tch2)

sp_tgrp_123456 = np.fft.fft(tgrp_123456)
#-----#
#-----Спектры модулированных
сигналов в каналах 1, 2-----#
sp_mch1 = np.hypot(sp_mch1.real, sp_mch1.imag)/n*2
sp_mch2 = np.hypot(sp_mch2.real, sp_mch2.imag)/n*2
#-----#
#-----Спектры канальных сигналов
до объединения-----#
sp_tch1 = np.hypot(sp_tch1.real, sp_tch1.imag)/n*2
sp_tch2 = np.hypot(sp_tch2.real, sp_tch2.imag)/n*2
#-----#
#-----Спектр группового сигнала на выходе тракта

```

```

передачи и входе в тракт приема-----#
sp_tgrp_123456 = np.hypot(sp_tgrp_123456.real,
sp_tgrp_123456.imag)/n*2
#-----#
-----#

#-----Построение графиков сигналов и их спектров
в передающем тракте-----#
fig, (ax0, ax1) = plt.subplots(nrows=2)
ax0.set_title('Распределения первичных сигналов, сформированных
из двух систем')
ax0.plot(time, tgrp_123, 'tab:blue', lw=1, label='$Ch_1$')
ax0.plot(time, tgrp_456, 'tab:orange', lw=1, label='$Ch_2$')

ax0.set_xlim(T/2-1, T/2+1)
ax0.set_xlabel("$t$, мс", fontsize=10)
ax0.set_ylabel("$V_{1,2,3}$, В", fontsize=10)
ax0.legend(loc='best')
ax1.set_title('Спектры первичных сигналов, сформированных из
двух систем')
ax1.plot(sp_freq[0:n//2], sp_tgrp123[0:n//2], 'tab:blue', lw=1,
label='$Ch_1$')
ax1.plot(sp_freq[0:n//2], sp_tgrp456[0:n//2], 'tab:orange',
lw=1, label='$Ch_2$')
ax1.set_xlim(0, 5)
ax1.set_xlabel("$f$, кГц", fontsize=10)
ax1.set_ylabel("$V_{1,2,3}$, В", fontsize=10)
ax1.legend(loc='best')
fig.tight_layout()

fig, (ax0, ax1) = plt.subplots(nrows=2)
ax0.set_title('Распределения модулированных сигналов,
сформированных из двух систем')
ax0.plot(time, mch1, 'tab:blue', lw=1, label='$Ch_1$')
ax0.plot(time, mch2, 'tab:orange', lw=1, label='$Ch_2$')

ax0.set_xlim(T/2-1, T/2+1)
ax0.set_xlabel("$t$, мс", fontsize=10)
ax0.set_ylabel("$V_{1,2,3}$, В", fontsize=10)
ax0.legend(loc='best')
ax1.set_title('Спектры модулированных сигналов, сформированных
из двух систем')
ax1.plot(sp_freq[0:n//2], sp_mch1[0:n//2], 'tab:blue', lw=1,
label='$Ch_1$')
ax1.plot(sp_freq[0:n//2], sp_mch2[0:n//2], 'tab:orange', lw=1,

```

```

label='$Ch_2$')
ax1.set_xlim(10, 25)
ax1.set_xlabel("$f$, кГц", fontsize=10)
ax1.set_ylabel("$V_{1,2,3}$, В", fontsize=10)
ax1.legend(loc='best')
fig.tight_layout()

fig, (ax0, ax1) = plt.subplots(nrows=2)
ax0.set_title('Распределения канальных сигналов, сформированных
из двух систем')
ax0.plot(time, tch1, 'tab:blue', lw=1, label='$Ch_1$')
ax0.plot(time, tch2, 'tab:orange', lw=1, label='$Ch_2$')

ax0.set_xlim(T/2-1, T/2+1)
ax0.set_xlabel("$t$, мс", fontsize=10)
ax0.set_ylabel("$V_{1,2,3}$, В", fontsize=10)
ax0.legend(loc='best')
ax1.set_title('Спектры канальных сигналов, сформированных из
двух систем')
ax1.plot(sp_freq[0:n//2], sp_tch1[0:n//2], 'tab:blue', lw=1,
label='$Ch_1$')
ax1.plot(sp_freq[0:n//2], sp_tch2[0:n//2], 'tab:orange', lw=1,
label='$Ch_2$')
ax1.set_xlim(10, 25)
ax1.set_xlabel("$f$, кГц", fontsize=10)
ax1.set_ylabel("$V_{1,2,3}$, В", fontsize=10)
ax1.legend(loc='best')
fig.tight_layout()

fig, (ax0, ax1) = plt.subplots(nrows=2)
ax0.set_title('Финальный сигнал')
ax0.plot(time, tgrp_123456, 'tab:blue', lw=1)

ax0.set_xlim(T/2-1, T/2+1)
ax0.set_xlabel("$t$, мс", fontsize=10)
ax0.set_ylabel("$V_{1,2,3}$, В", fontsize=10)
ax1.set_title('Спектр финального сигнала')
ax1.plot(sp_freq[0:n//2], sp_tgrp_123456[0:n//2], 'tab:blue',
lw=1)
ax1.set_xlim(10, 25)
ax1.set_xlabel("$f$, кГц", fontsize=10)
ax1.set_ylabel("$V_{1,2,3}$, В", fontsize=10)
fig.tight_layout()
#-----#
-----#

```



```

'-----На приемке-----'
'Частотный фильтр выделяем 2 системы'
#-----Фильтрация группового сигнала с выделением
канальных сигналов-----#
rch1 = filter(time, tgrp_123456, fn[0]+0.3, fn[0]+3.4).real
rch2 = filter(time, tgrp_123456, fn[1]+0.3, fn[1]+3.4).real
#-----#

#-----Демодуляция канальных
сигналов-----#
for i in range(n):
    mch1[i] = (1.0 + m * cos(f2w(fn[0])*time[i]))*rch1[i]
    mch2[i] = (1.0 + m * cos(f2w(fn[1])*time[i]))*rch2[i]
#-----#

#-----Канальные ФНЧ для выделения первичных
сигналов-----#
rsig1 = filter(time, mch1, 0.3, 3.4).real
rsig2 = filter(time, mch2, 0.3, 3.4).real
#-----#

#-----Спектры канальных сигналов после
разделения-----#
sp_rch1 = np.fft.fft(rch1)
sp_rch2 = np.fft.fft(rch2)

sp_rch1 = np.hypot(sp_rch1.real, sp_rch1.imag)/n*2
sp_rch2 = np.hypot(sp_rch2.real, sp_rch2.imag)/n*2
#-----#

#-----Спектры демодулированных сигналов в
каналах 1, 2-----#
sp_mch1 = np.fft.fft(mch1)
sp_mch2 = np.fft.fft(mch2)

sp_mch1 = np.hypot(sp_mch1.real, sp_mch1.imag)/n*2
sp_mch2 = np.hypot(sp_mch2.real, sp_mch2.imag)/n*2
#-----#

```

```

#-----Спектры первичных сигналов, выделенных из
канальных на выходе СП ЧРК-----#
sp_rsig1 = np.fft.fft(rsig1)
sp_rsig2 = np.fft.fft(rsig2)

sp_rsig1 = np.hypot(sp_rsig1.real, sp_rsig1.imag)/n*2
sp_rsig2 = np.hypot(sp_rsig2.real, sp_rsig2.imag)/n*2
#-----#
#-----#
'Частотный фильтр выделяем 2 системы КОНЕЦ'

#-----Построение графиков сигналов и их спектров на
приемном участке МСП ЧРК-----#
fig, (ax0, ax1) = plt.subplots(nrows=2)
ax0.set_title('Распределения канальных сигналов в тракте
приема')
ax0.plot(time, rch1, 'tab:blue', lw=1, label='$Ch_1$')
ax0.plot(time, rch2, 'tab:orange', lw=1, label='$Ch_2$')

ax0.set_xlim(T/2-1, T/2+1)
ax0.set_xlabel("$t$, мс", fontsize=10)
ax0.set_ylabel("$V_{1,2,3}$, В", fontsize=10)
ax0.legend(loc='best')
ax1.set_title('Спектры канальных сигналов в тракте приема')
ax1.plot(sp_freq[0:n//2], sp_rch1[0:n//2], 'tab:blue', lw=1,
label='$Ch_1$')
ax1.plot(sp_freq[0:n//2], sp_rch2[0:n//2], 'tab:orange', lw=1,
label='$Ch_2$')
ax1.set_xlim(10, 25)
ax1.set_xlabel("$f$, кГц", fontsize=10)
ax1.set_ylabel("$V_{1,2,3}$, В", fontsize=10)
ax1.legend(loc='best')
fig.tight_layout()

fig, (ax0, ax1) = plt.subplots(nrows=2)
ax0.set_title('Распределения демодулированных сигналов')
ax0.plot(time, mch1, 'tab:blue', lw=1, label='$Ch_1$')
ax0.plot(time, mch2, 'tab:orange', lw=1, label='$Ch_2$')
ax0.set_xlim(T/2-1, T/2+1)
ax0.set_xlabel("$t$, мс", fontsize=10)
ax0.set_ylabel("$V_{1,2,3}$, В", fontsize=10)
ax0.legend(loc='best')

ax1.set_title('Спектры демодулированных сигналов')

```

```

ax1.plot(sp_freq[0:n//2], sp_mch1[0:n//2], 'tab:blue', lw=1,
label='$Ch_1$')
ax1.plot(sp_freq[0:n//2], sp_mch2[0:n//2], 'tab:orange', lw=1,
label='$Ch_2$')
ax1.set_xlim(0, 5)
ax1.set_xlabel("$f$, кгц", fontsize=10)
ax1.set_ylabel("$V_{1,2,3}$, В", fontsize=10)
ax1.legend(loc='best')
fig.tight_layout()

fig, (ax0, ax1) = plt.subplots(nrows=2)
ax0.set_title('Распределения первичных сигналов')
ax0.plot(time, rsig1, 'tab:blue', lw=1, label='$Ch_1$')
ax0.plot(time, rsig2, 'tab:orange', lw=1, label='$Ch_2$')

ax0.set_xlim(T/2-1, T/2+1)
ax0.set_xlabel("$t$, мс", fontsize=10)
ax0.set_ylabel("$V_{1,2,3}$, В", fontsize=10)
ax0.legend(loc='best')
ax1.set_title('Спектры первичных сигналов')
ax1.plot(sp_freq[0:n//2], sp_rsig1[0:n//2], 'tab:blue', lw=1,
label='$Ch_1$')
ax1.plot(sp_freq[0:n//2], sp_rsig2[0:n//2], 'tab:orange', lw=1,
label='$Ch_2$')
ax1.set_xlim(0, 5)
ax1.set_xlabel("$f$, кгц", fontsize=10)
ax1.set_ylabel("$V_{1,2,3}$, В", fontsize=10)
ax1.legend(loc='best')
fig.tight_layout()
#-----#

#-----Разделение группового сигнала на
канальные сигналы-----#
for i in range(n//nch//npp):
    for j in range(nch):
        for k in range(npp):
            indx = i*nch*npp+j*npp+k
            # -----1 система-----#
            mch_1_1[indx] = 0
            mch_1_2[indx] = 0
            mch_1_3[indx] = 0
            # -----#

```

```

# -----2 система-----#
mch_2_1[indx] = 0
mch_2_2[indx] = 0
mch_2_3[indx] = 0
# -----#

#-----Канальные электронные ключи-----#
if j == 0:
    mch_1_1[indx] = tgrp_123[indx]
    mch_2_1[indx] = tgrp_456[indx]
elif j == 1:
    mch_1_2[indx] = tgrp_123[indx]
    mch_2_2[indx] = tgrp_456[indx]
elif j == 2:
    mch_1_3[indx] = tgrp_123[indx]
    mch_2_3[indx] = tgrp_456[indx]
# -----#

#-----#

#-----Канальные ФНЧ для выделения первичных
сигналов-----#
# -----1 система-----#
final_sig_1 = filter(time, mch_1_1, 0.3, 3.4).real
final_sig_2 = filter(time, mch_1_2, 0.3, 3.4).real
final_sig_3 = filter(time, mch_1_3, 0.3, 3.4).real
# -----#

# -----2 система-----#
final_sig_4 = filter(time, mch_2_1, 0.3, 4).real
final_sig_5 = filter(time, mch_2_2, 0.3, 4).real
final_sig_6 = filter(time, mch_2_3, 0.3, 4).real
# -----#

#-----#

#-----Спектры канальных сигналов после
разделения-----#
# -----1 система-----#
sp_mch_1_1 = np.fft.fft(mch_1_1)
sp_mch_1_2 = np.fft.fft(mch_1_2)
sp_mch_1_3 = np.fft.fft(mch_1_3)

sp_mch_1_1 = np.hypot(sp_mch_1_1.real, sp_mch_1_1.imag)/n*2
sp_mch_1_2 = np.hypot(sp_mch_1_2.real, sp_mch_1_2.imag)/n*2

```

```

sp_mch_1_3 = np.hypot(sp_mch_1_3.real, sp_mch_1_3.imag)/n*2
# -----#

# -----2 система-----#
sp_mch_2_1 = np.fft.fft(mch_2_1)
sp_mch_2_2 = np.fft.fft(mch_2_2)
sp_mch_2_3 = np.fft.fft(mch_2_3)

sp_mch_2_1 = np.hypot(sp_mch_2_1.real, sp_mch_2_1.imag)/n*2
sp_mch_2_2 = np.hypot(sp_mch_2_2.real, sp_mch_2_2.imag)/n*2
sp_mch_2_3 = np.hypot(sp_mch_2_3.real, sp_mch_2_3.imag)/n*2
# -----#
#-----#
-----#

#-----Спектры первичных сигналов, выделенных из
канальных на выходе СП ЧРК-----#
# -----1 система-----#
sp_final_sig_1 = np.fft.fft(final_sig_1)
sp_final_sig_2 = np.fft.fft(final_sig_2)
sp_final_sig_3 = np.fft.fft(final_sig_3)

sp_final_sig_1 = np.hypot(sp_final_sig_1.real,
sp_final_sig_1.imag)/n*2
sp_final_sig_2 = np.hypot(sp_final_sig_2.real,
sp_final_sig_2.imag)/n*2
sp_final_sig_3 = np.hypot(sp_final_sig_3.real,
sp_final_sig_3.imag)/n*2
# -----#

# -----2 система-----#
sp_final_sig_4 = np.fft.fft(final_sig_4)
sp_final_sig_5 = np.fft.fft(final_sig_5)
sp_final_sig_6 = np.fft.fft(final_sig_6)

sp_final_sig_4 = np.hypot(sp_final_sig_4.real,
sp_final_sig_4.imag)/n*2
sp_final_sig_5 = np.hypot(sp_final_sig_5.real,
sp_final_sig_5.imag)/n*2
sp_final_sig_6 = np.hypot(sp_final_sig_6.real,
sp_final_sig_6.imag)/n*2
# -----#
#-----#
-----#

```

```

#-----Построение графиков сигналов и их спектров на
приемном участке МСП ЧРК-----#
fig, (ax0, ax1) = plt.subplots(nrows=2)
ax0.set_title('Распределения канальных сигналов в тракте приема
система 1')
ax0.plot(time, mch_1_1, 'tab:blue', lw=1, label='$Ch_1$')
ax0.plot(time, mch_1_2, 'tab:orange', lw=1, label='$Ch_2$')
ax0.plot(time, mch_1_3, 'tab:green', lw=1, label='$Ch_3$')

ax0.set_xlim(T/2-1, T/2+1)
ax0.set_xlabel("$t$, мс", fontsize=10)
ax0.set_ylabel("$V_{1,2,3}$, В", fontsize=10)
ax0.legend(loc='best')
ax1.set_title('Спектры канальных сигналов в тракте приема
система 1')
ax1.plot(sp_freq[0:n//2], sp_mch_1_1[0:n//2], 'tab:blue', lw=1,
label='$Ch_1$')
ax1.plot(sp_freq[0:n//2], sp_mch_1_2[0:n//2], 'tab:orange',
lw=1, label='$Ch_2$')
ax1.plot(sp_freq[0:n//2], sp_mch_1_3[0:n//2], 'tab:green',
lw=1, label='$Ch_3$')
ax1.set_xlim(0, 25)
ax1.set_xlabel("$f$, кГц", fontsize=10)
ax1.set_ylabel("$V_{1,2,3}$, В", fontsize=10)
ax1.legend(loc='best')
fig.tight_layout()

fig, (ax0, ax1) = plt.subplots(nrows=2)
ax0.set_title('Распределения первичных сигналов система 1')
ax0.plot(time, final_sig_1, 'tab:blue', lw=1, label='$Ch_1$')
ax0.plot(time, final_sig_2, 'tab:orange', lw=1, label='$Ch_2$')
ax0.plot(time, final_sig_3, 'tab:green', lw=1, label='$Ch_3$')

ax0.set_xlim(T/2-1, T/2+1)
ax0.set_xlabel("$t$, мс", fontsize=10)
ax0.set_ylabel("$V_{1,2,3}$, В", fontsize=10)
ax0.legend(loc='best')
ax1.set_title('Спектры первичных сигналов система 1')
ax1.plot(sp_freq[0:n//2], sp_final_sig_1[0:n//2], 'tab:blue',
lw=1, label='$Ch_1$')
ax1.plot(sp_freq[0:n//2], sp_final_sig_2[0:n//2], 'tab:orange',
lw=1, label='$Ch_2$')
ax1.plot(sp_freq[0:n//2], sp_final_sig_3[0:n//2], 'tab:green',
lw=1, label='$Ch_3$')
ax1.set_xlim(0, 5)

```

```

ax1.set_xlabel("$f$, кГц", fontsize=10)
ax1.set_ylabel("$V_{1,2,3}$, В", fontsize=10)
ax1.legend(loc='best')
fig.tight_layout()

fig, (ax0, ax1) = plt.subplots(nrows=2)
ax0.set_title('Распределения канальных сигналов в тракте приема
система 2')
ax0.plot(time, mch_2_1, 'tab:blue', lw=1, label='$Ch_1$')
ax0.plot(time, mch_2_2, 'tab:orange', lw=1, label='$Ch_2$')
ax0.plot(time, mch_2_3, 'tab:green', lw=1, label='$Ch_3$')

ax0.set_xlim(T/2-1, T/2+1)
ax0.set_xlabel("$t$, мс", fontsize=10)
ax0.set_ylabel("$V_{1,2,3}$, В", fontsize=10)
ax0.legend(loc='best')
ax1.set_title('Спектры канальных сигналов в тракте приема
система 2')
ax1.plot(sp_freq[0:n//2], sp_mch_2_1[0:n//2], 'tab:blue', lw=1,
label='$Ch_1$')
ax1.plot(sp_freq[0:n//2], sp_mch_2_2[0:n//2], 'tab:orange',
lw=1, label='$Ch_2$')
ax1.plot(sp_freq[0:n//2], sp_mch_2_3[0:n//2], 'tab:green',
lw=1, label='$Ch_3$')
ax1.set_xlim(0, 25)
ax1.set_xlabel("$f$, кГц", fontsize=10)
ax1.set_ylabel("$V_{1,2,3}$, В", fontsize=10)
ax1.legend(loc='best')
fig.tight_layout()

fig, (ax0, ax1) = plt.subplots(nrows=2)
ax0.set_title('Распределения первичных сигналов система 2')
ax0.plot(time, final_sig_4, 'tab:blue', lw=1, label='$Ch_1$')
ax0.plot(time, final_sig_5, 'tab:orange', lw=1, label='$Ch_2$')
ax0.plot(time, final_sig_6, 'tab:green', lw=1, label='$Ch_3$')
ax0.set_xlim(T/2-1, T/2+1)
ax0.set_xlabel("$t$, мс", fontsize=10)
ax0.set_ylabel("$V_{1,2,3}$, В", fontsize=10)
ax0.legend(loc='best')
ax1.set_title('Спектры первичных сигналов система 2')
ax1.plot(sp_freq[0:n//2], sp_final_sig_4[0:n//2], 'tab:blue',
lw=1, label='$Ch_1$')
ax1.plot(sp_freq[0:n//2], sp_final_sig_5[0:n//2], 'tab:orange',
lw=1, label='$Ch_2$')
ax1.plot(sp_freq[0:n//2], sp_final_sig_6[0:n//2], 'tab:green',

```

```
lw=1, label='$Ch_3$')
ax1.set_xlim(0, 5)
ax1.set_xlabel("$f$, κΓϣ", fontsize=10)
ax1.set_ylabel("$V_{1,2,3}$, B", fontsize=10)
ax1.legend(loc='best')
fig.tight_layout()
plt.show()
#-----
-----#
```



## Результаты имитационного моделирования

```
C:\Users\Хот9К\PycharmProjects\Laboratho
Временной интервал, мс 10
```

Рис.2. Входные параметры

В тракте передачи до объединения сигналов двух систем происходит несколько процессов, а именно формирование шести сигналов различной частоты, их поступление на входы, модулирование с помощью ПППИ (рисунок 4), преобразование в каналные сигналы, группировка сигналов (рисунки 3–12). В итоге получаем финальный сигнал, которые передается по линии передачи. (рисунок 13) Далее происходит временная модуляция каждого из двух сигналов, фильтрация сигналов от НГЧ до ВГЧ, объединение каналных сигналов в один и построение их спектра. В тракте приема процессы будут происходить в обратном порядке, за исключением того, что демодулированные сигналы пройдут еще через ФНЧ. (рисунок 14–20).

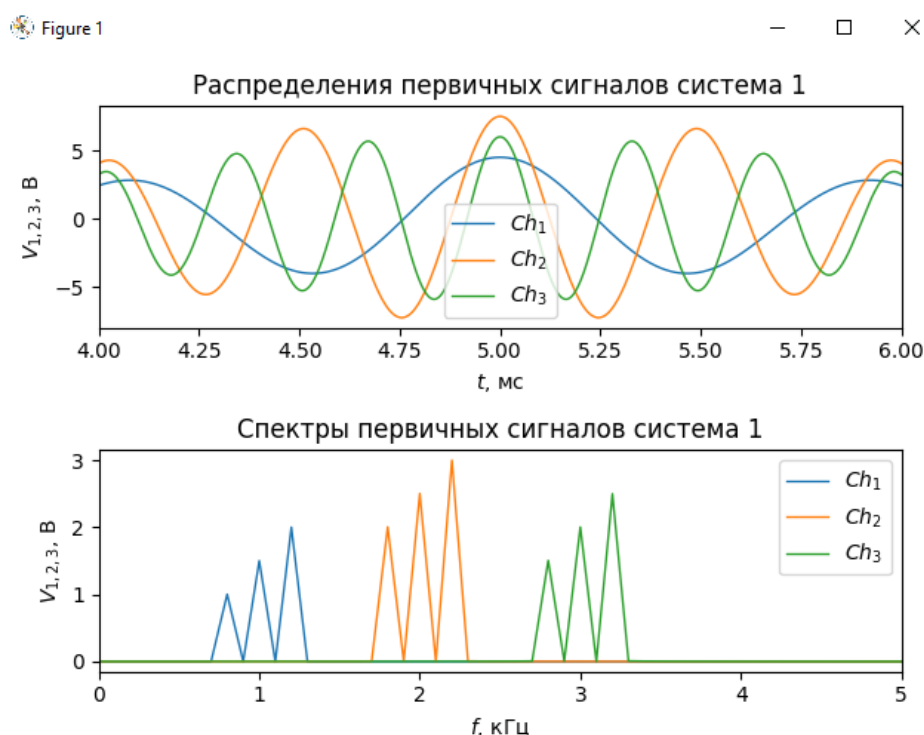


Рис.3. Распределение первичных сигналов первой системы и их спектры

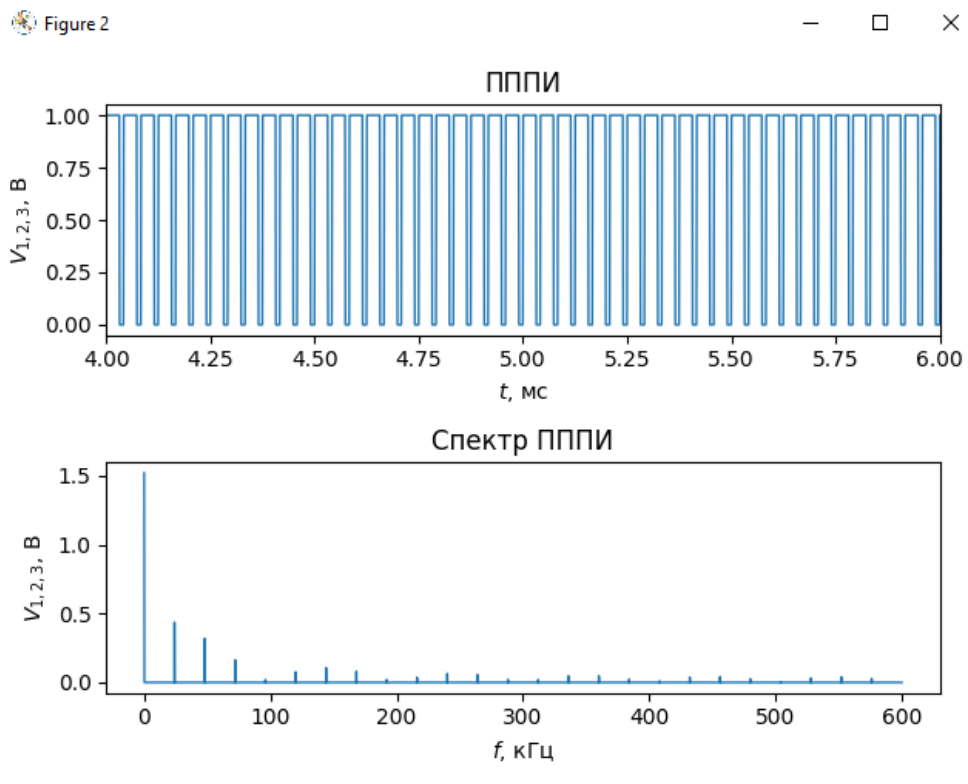


Рис.4. Периодическая последовательность прямоугольных импульсов и ее спектр

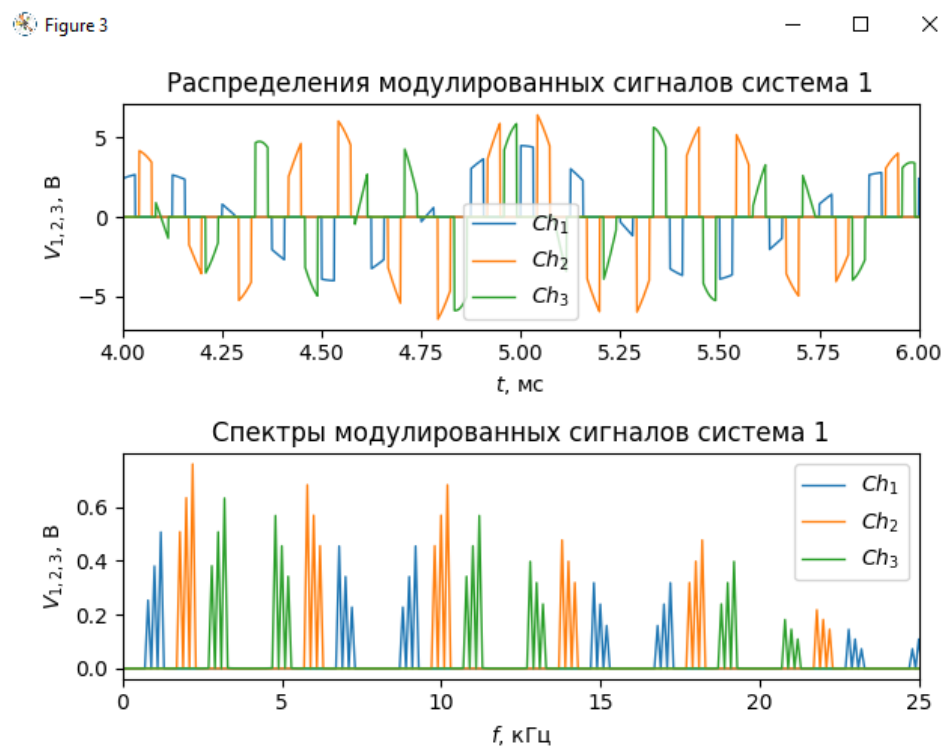


Рис.5. Распределение модулированных сигналов в первой системе

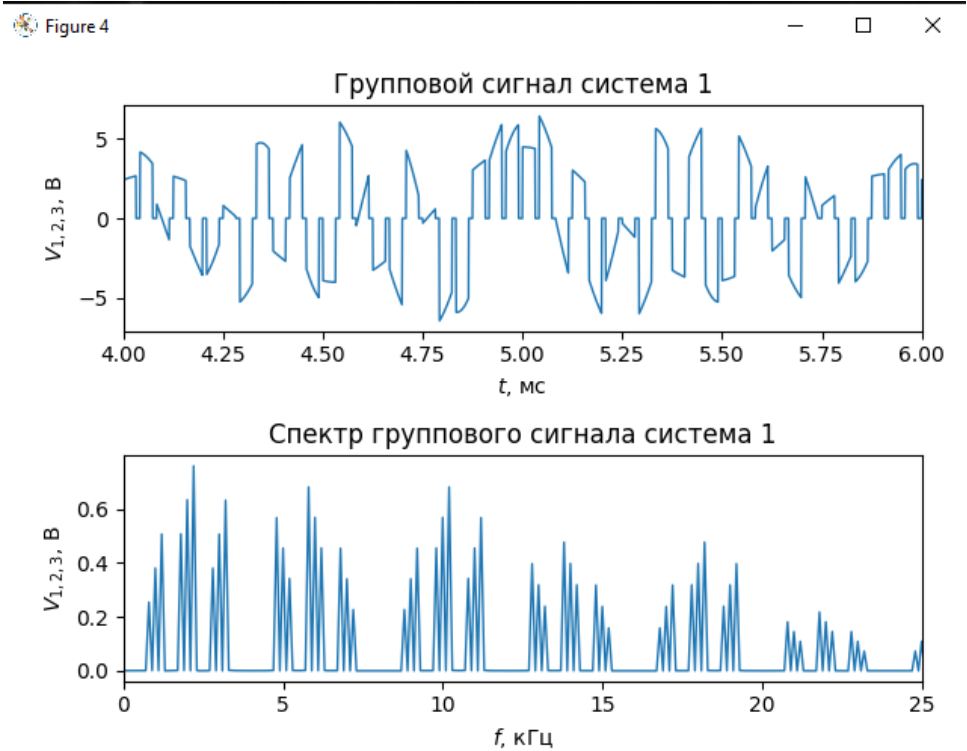


Рис.6. Групповой сигнал первой системы и его спектр

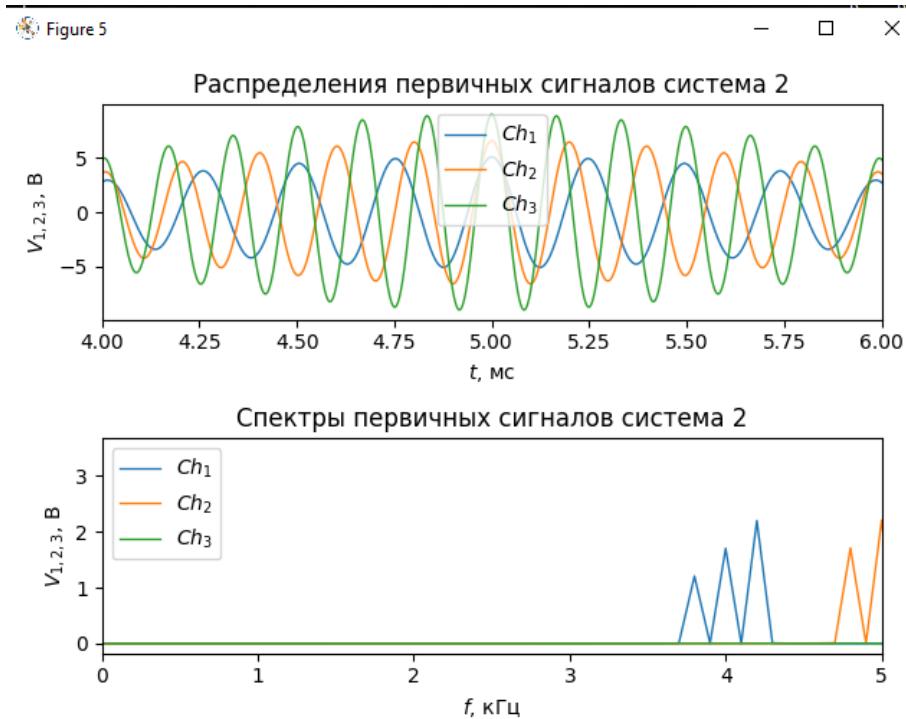


Рис.7. Распределение первичных сигналов второй системы

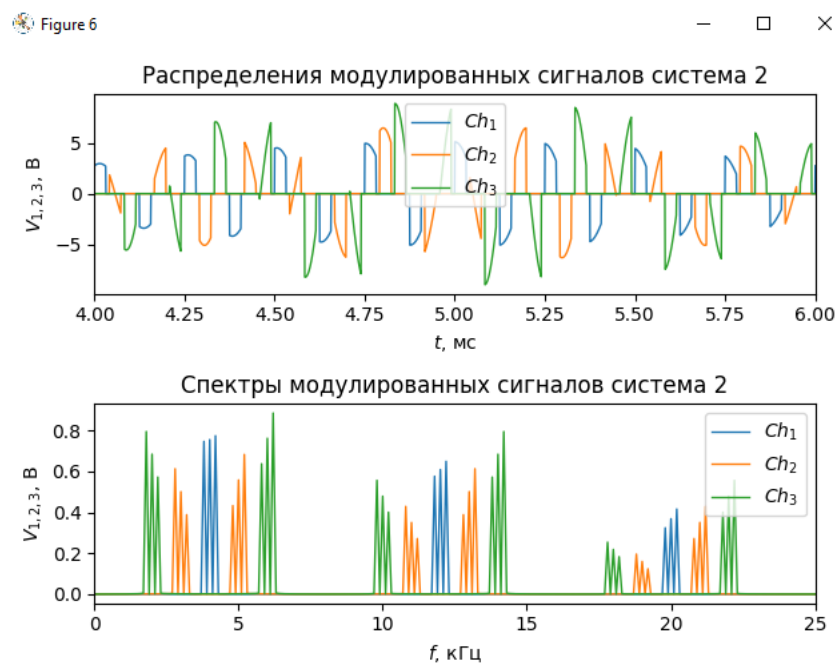


Рис.8. Распределение модулированных сигналов второй системы

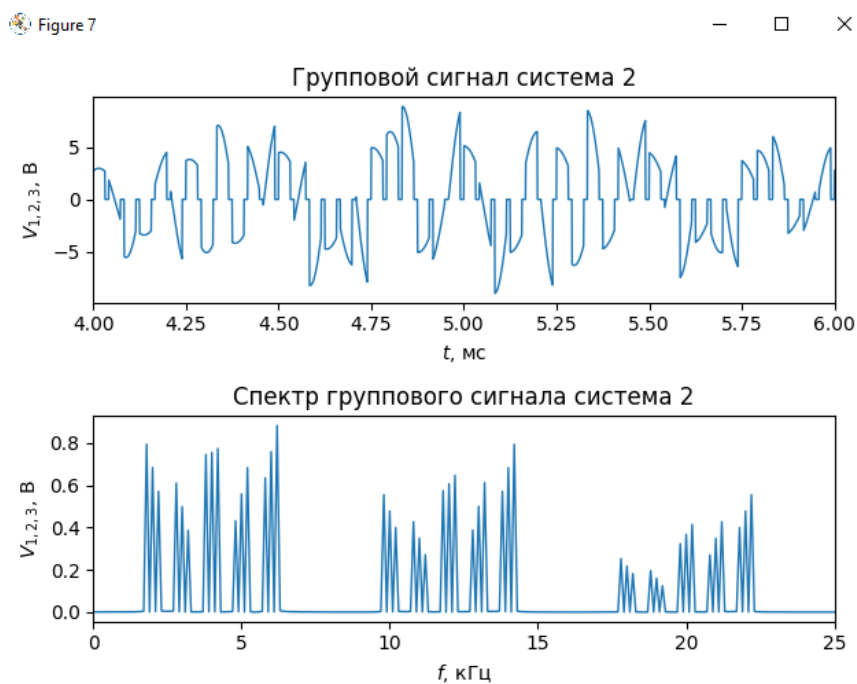


Рис.9. Групповой сигнал второй системы

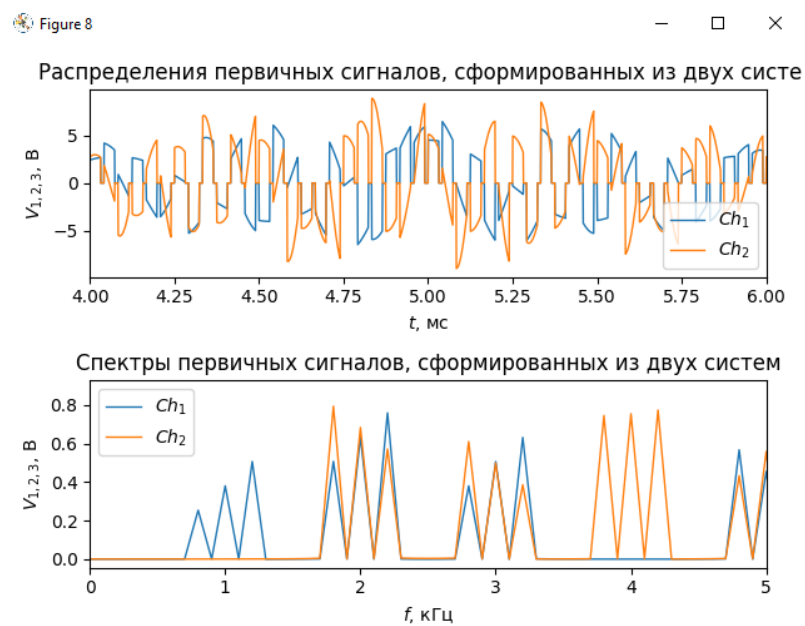


Рис.10. Распределение первичных сигналов, сформированных в первых двух системах

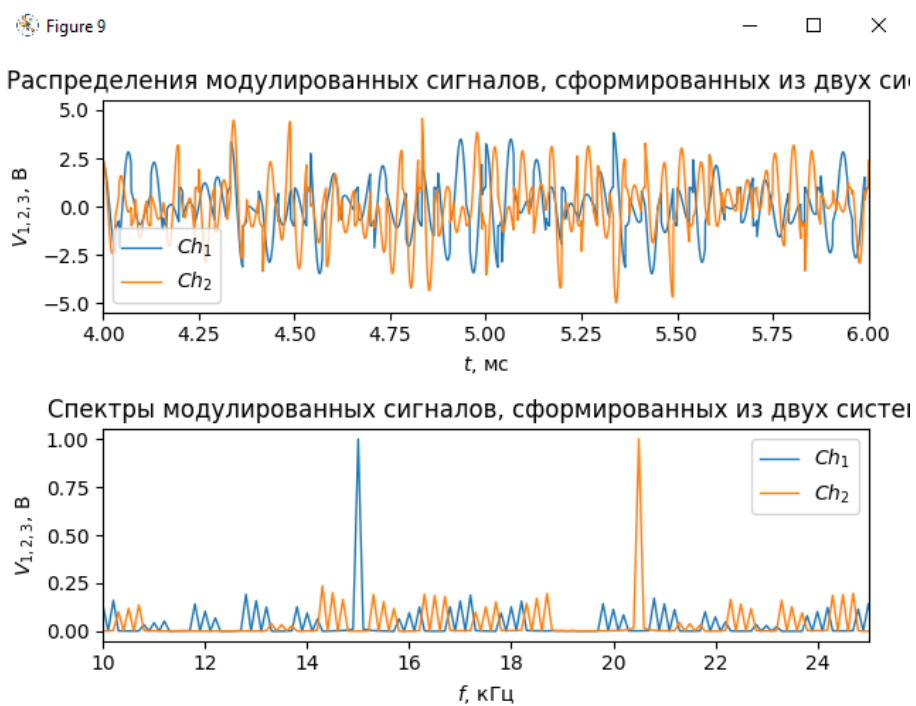


Рис.11. Модуляция групповых сигналов из двух первых систем

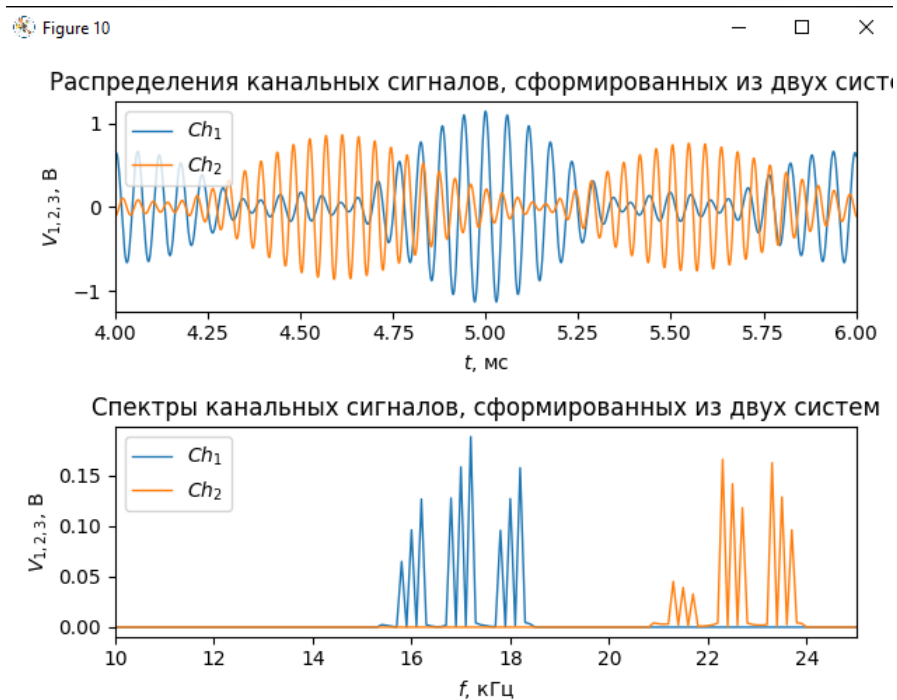


Рис.12. Распределения канальных сигналов из двух первых систем

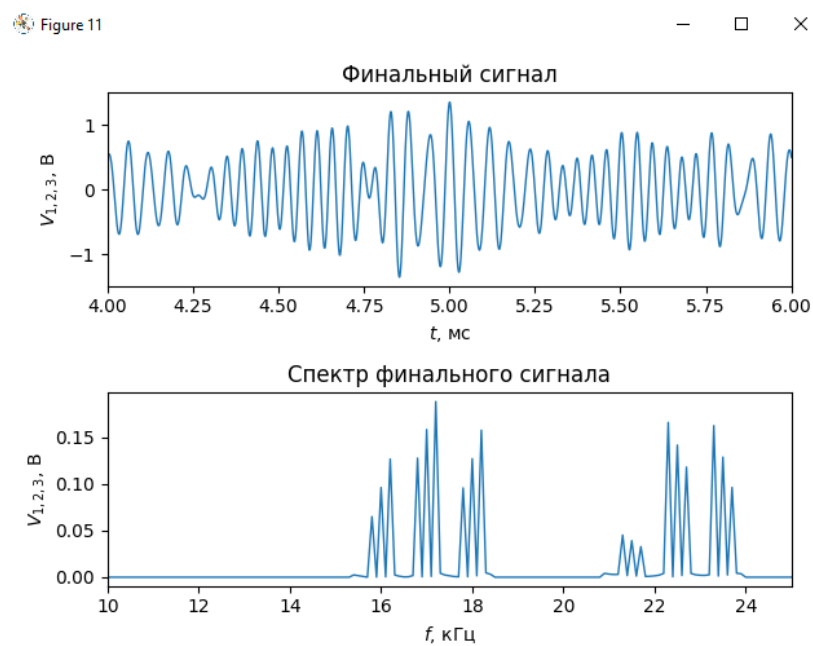


Рис.13. Объединенный сигнал из первых двух систем

Далее происходит обратное преобразование в тракте приема таким образом, чтобы попытаться получить исходные сигналы на выходе

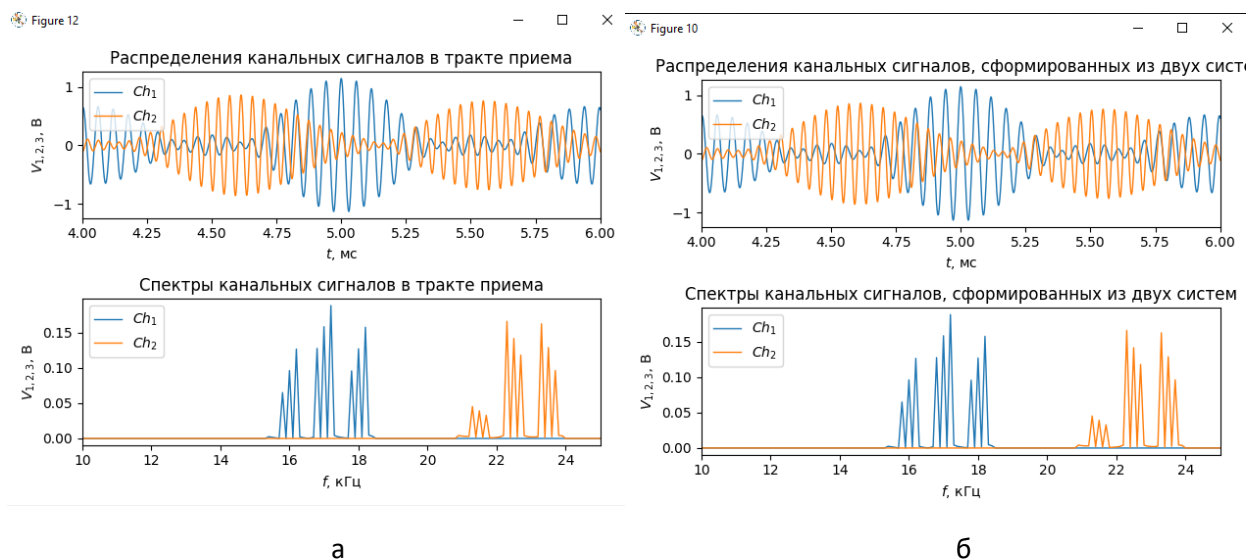


Рис.14. Сравнение распределения канальных сигналов в тракте приема(а) и передачи(б)

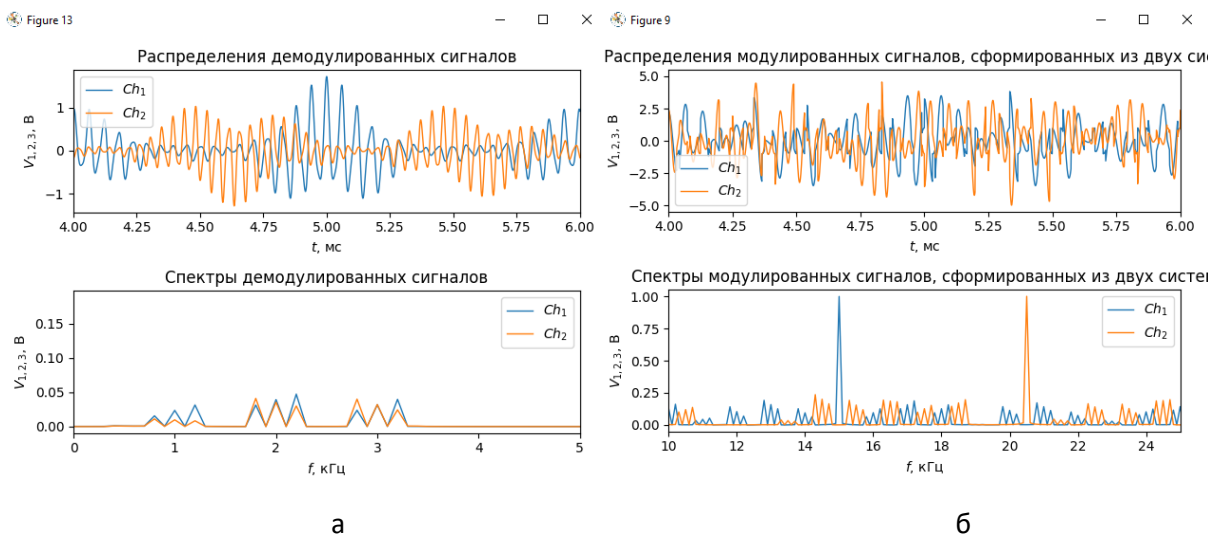


Рис.15. Сравнение распределения демодулированных сигналов в тракте приема(а) и передачи(б)

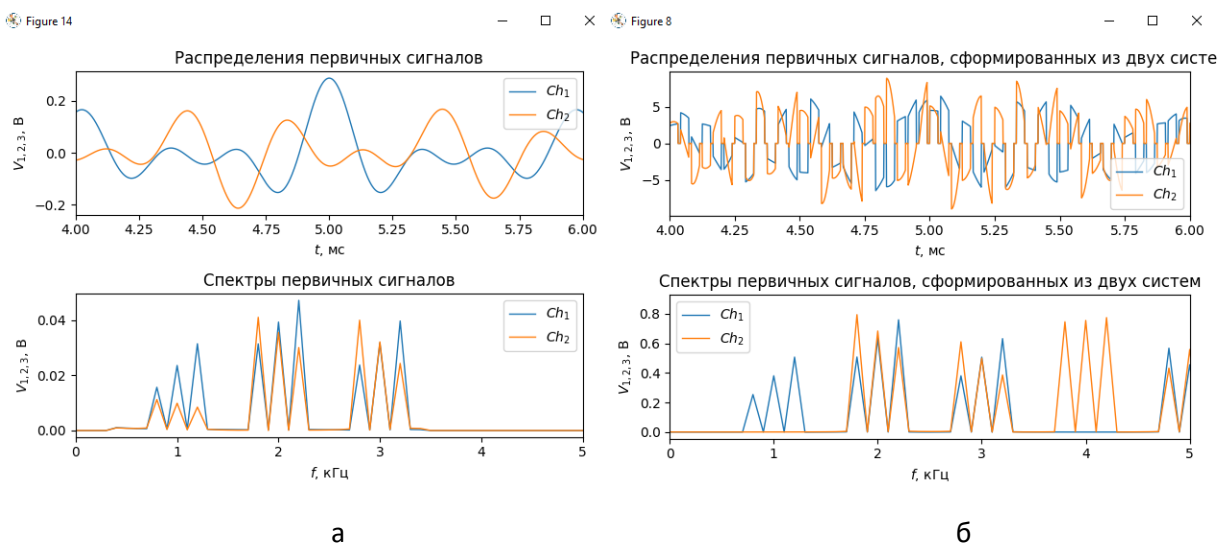


Рис.16. Сравнение распределения первичных сигналов двух систем в тракте приема(а) и передачи(б)

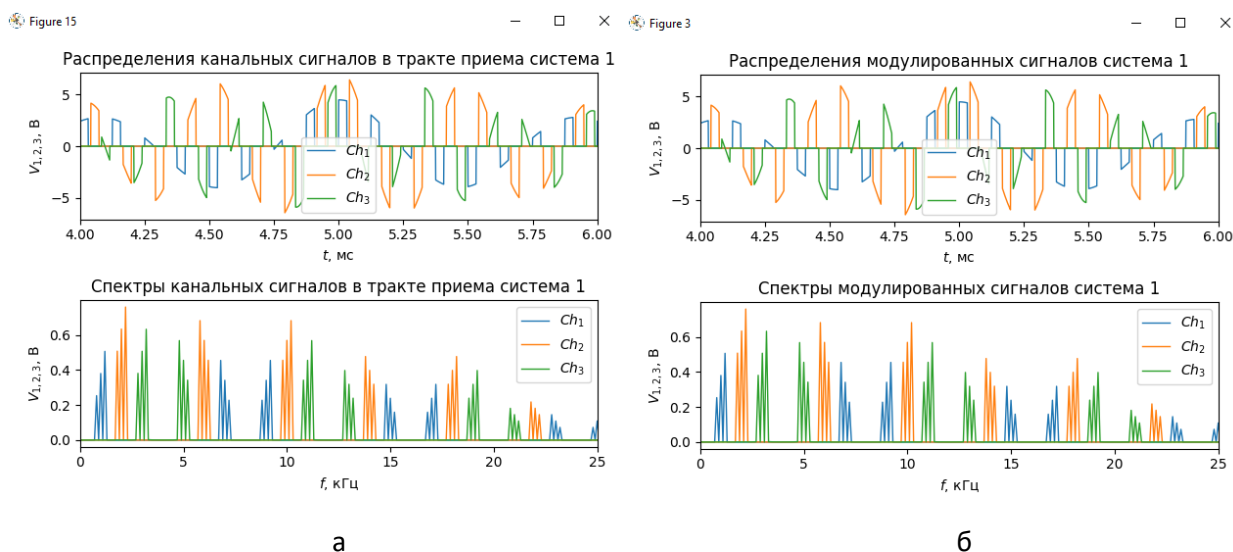


Рис.17. Сравнение распределения модулированных канальных сигналов первой системы в тракте приема(а) и передачи(б)

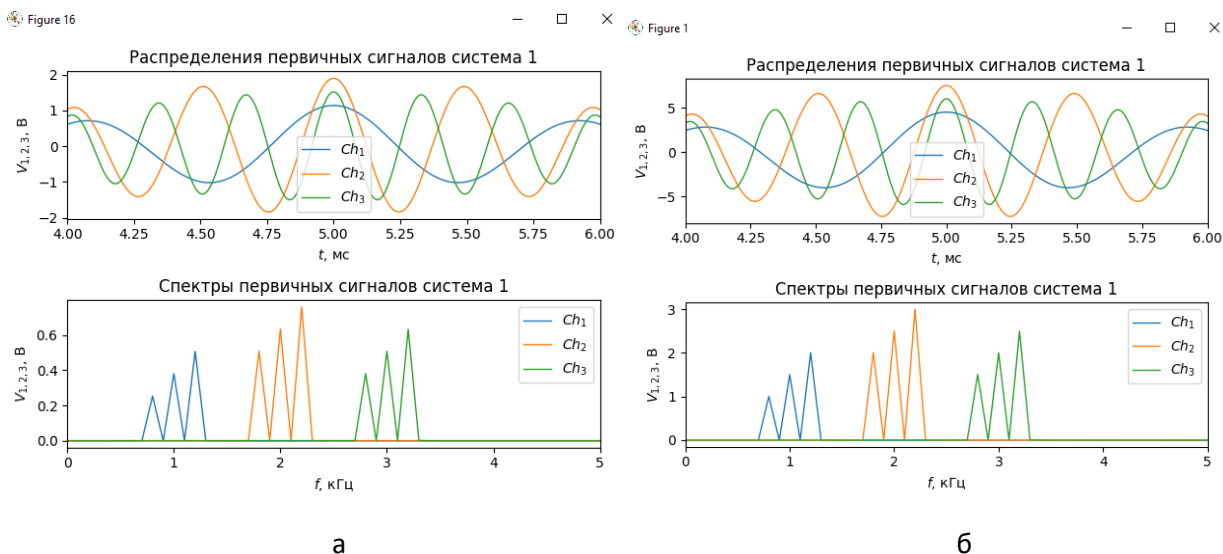


Рис.18. Сравнение распределения первичных сигналов первой системы в тракте приема(а) и передачи(б)

Первичные сигналы на входе передатчика и на выходе приемника абсолютно идентичны, но выходные сигналы заметно ослаблены по амплитуде. Это обусловлено математической стороной процессов модуляции и демодуляции и не несет никакой опасности.



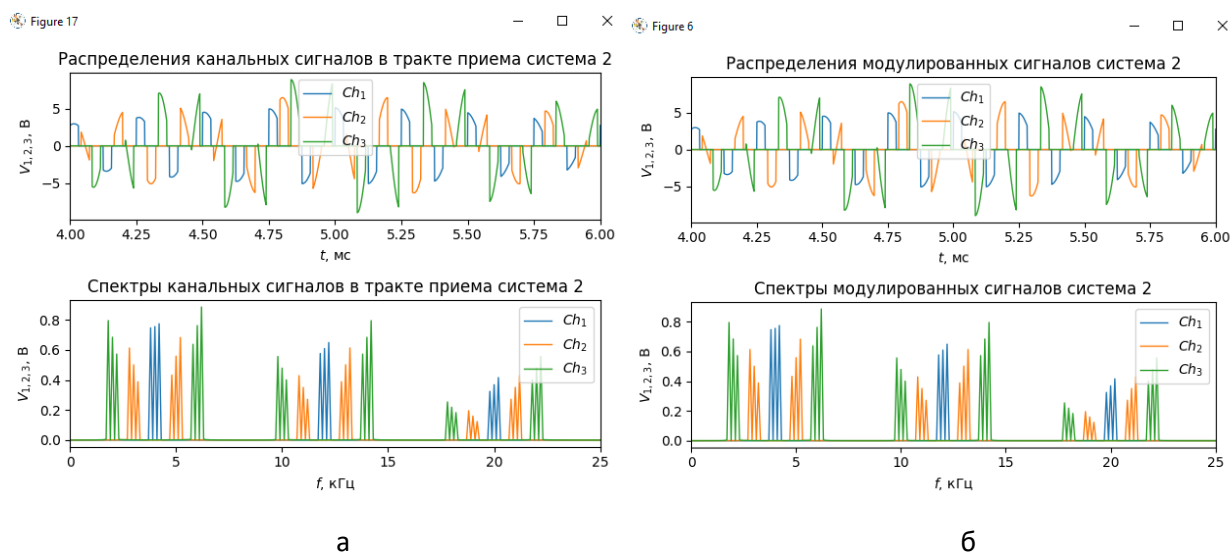


Рис.19. Сравнение распределения канальных сигналов второй системы в тракте приема(а) и передачи(б)

Модуляция в данном случае выполняется по методу АИМ-1. Сразу можно заметить, что в спектре присутствуют исходные сигналы, с уменьшенной амплитудой, а более детальное сравнение показывает, что форма осталась идентичной.

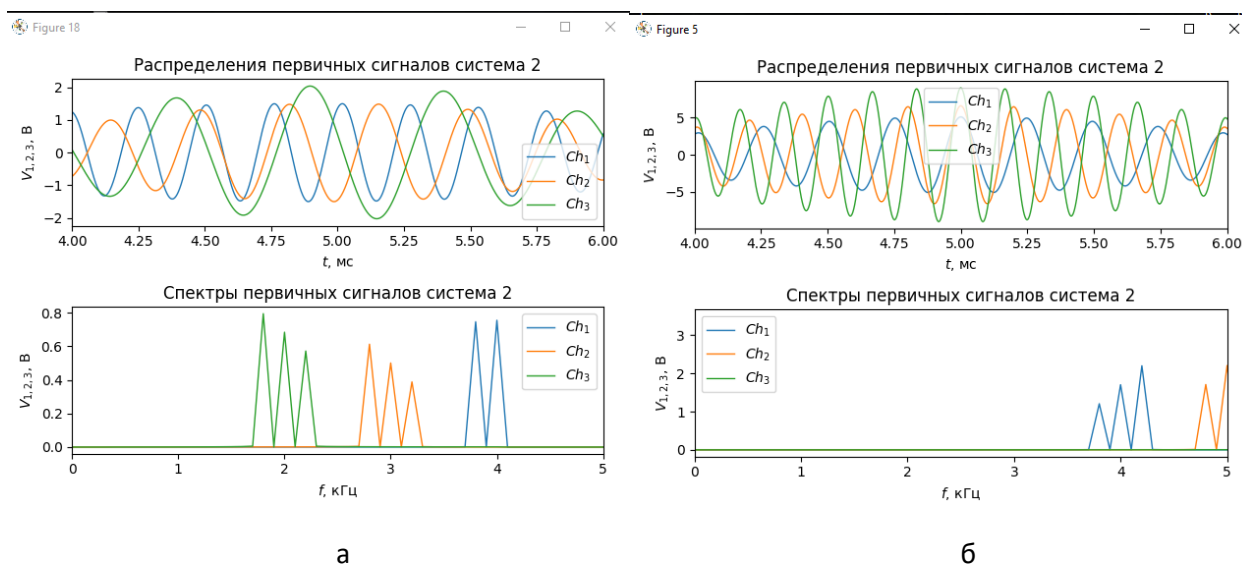


Рис.20. Сравнение распределения первичных сигналов второй системы в тракте приема(а) и передачи(б)

По итогу, у обоих сигналов, после прохождения системы, уменьшилась амплитуда как сигналов, так и пиков на спектре данных сигналов. Уменьшилась приблизительно в 5 раз.

## Выводы

Таким образом, в данной домашней работе, была с нуля реализована модель, имитирующую функционирование системы передачи с частотным и временным разделением каналов. Были окончательно усвоен способ мышления, при написании подобных моделей и работы с библиотеками `matplotlib`, `math`, `numpy`. Попробовал реализовать функциональную схему. Касательно общего хода работы, в целом получилось все, однако первичные сигналы второй системы не совпали на входе и выходе как по пикам спектров, так и по самим сигналам. Предположу, что это произошло из-за того, что теорема Котельникова для данной системы была не выполнена.

По результатам работы, отчет и листинг программы были выгружены на GitHub:

[https://github.com/AndreyTss/Fundamentals\\_Of\\_Building\\_Infocommunication\\_Systems\\_And\\_Networks/tree/master/HW](https://github.com/AndreyTss/Fundamentals_Of_Building_Infocommunication_Systems_And_Networks/tree/master/HW)