**Исследование влияния помех различной формы в процессе передачи м-последовательности длиной 127 и дальнейшая обработка полученных сигналов с использованием нейронных сетей**

Для реализации эксперимента была собрана электронная схема, соответствующая структурной схеме на рисунке 1.

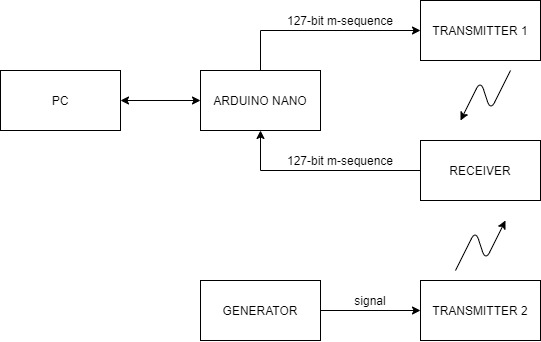


Рисунок 1 – Структурная схема экспериментальной установки

В качестве основного элемента схемы используется платформа Arduino Nano, к которой одновременно подключены передатчик 1 и приемник с несущими частотами 433 МГц. Параллельно в данной схеме присутствует независимый передатчик 2, подключенный к генератору сигнала, через него осуществляется передача помехи. Для записи принятой м-последовательности в текстовый файл Arduino Nano подключается к персональному компьютеру.

В коде программы Arduino Nano в массиве задана эталонная м-последовательность длиной 127. Программа последовательно считывает элементы массива и отправляет их в передатчик 1, затем считывает данные с приемника, записывает в массив и отправляет в последовательный порт компьютера. Далее с помощью вспомогательного скрипта на языке Python3 осуществляется чтение данных с последовательного порта и их запись в текстовый файл для последующей обработки. На генераторе выставляются различные параметры сигнала, который транслируется передатчиком 2. Таким образом, изменяя форму и амплитуду сигналов на генераторе, были получены текстовые файлы с принятыми м-последовательностями для различных условий. В последующем данные файлы будут использованы для обработки нейронными сетями.

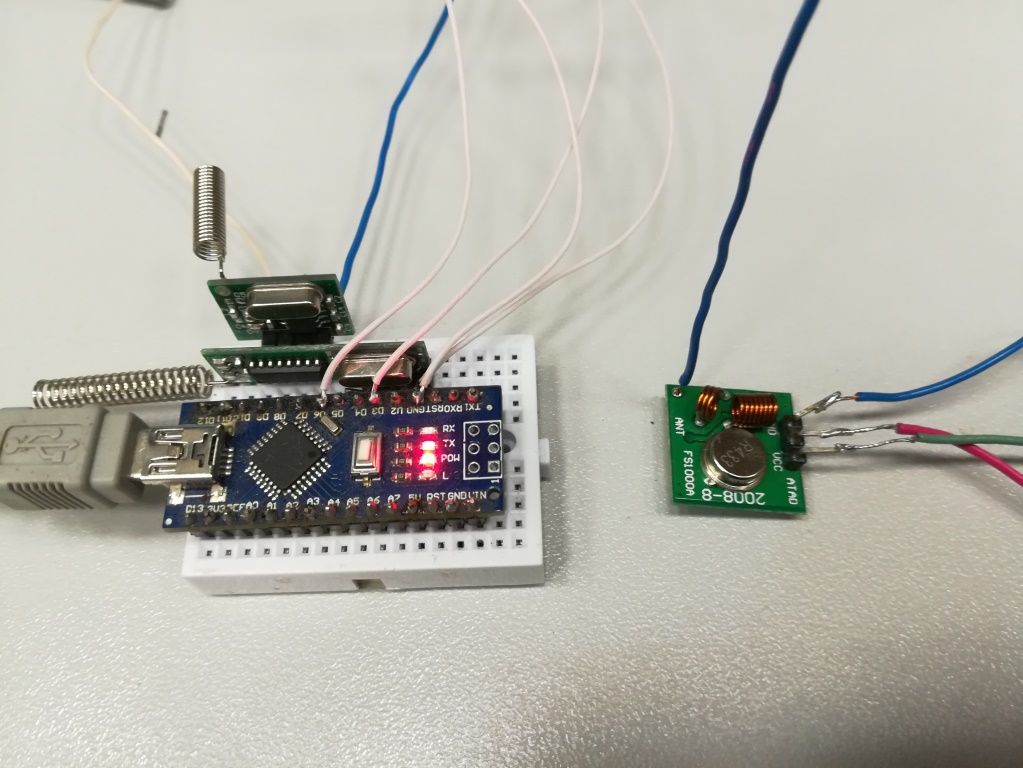


Рисунок 2 – Электронная схема в сборе

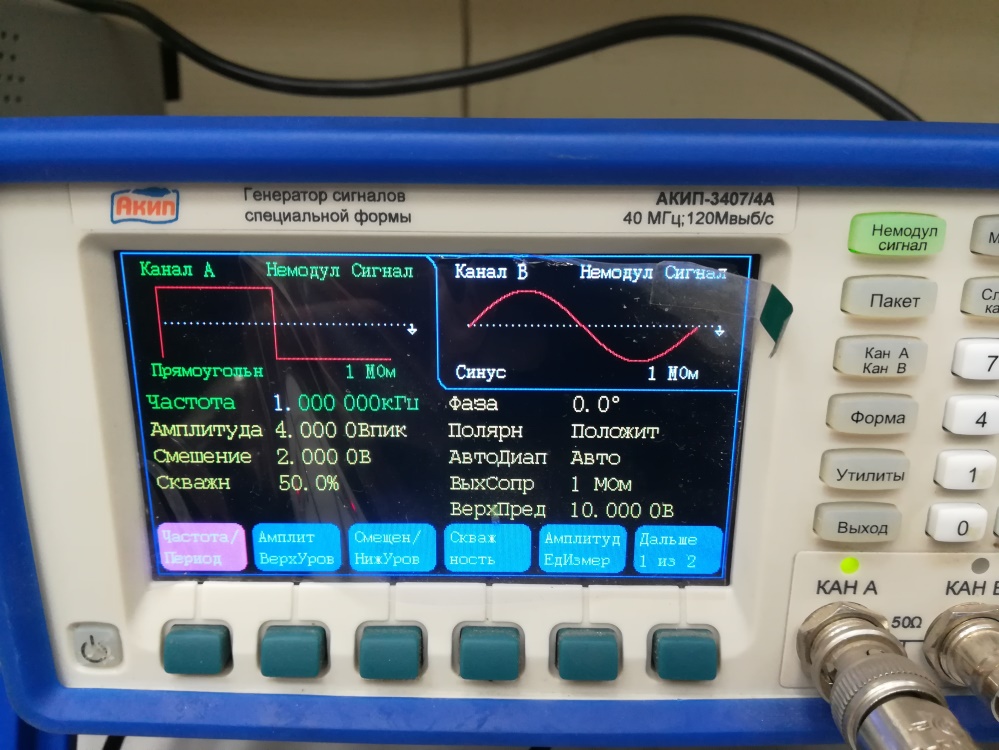


Рисунок 3 – Параметры генератора для импульсов 1 КГц

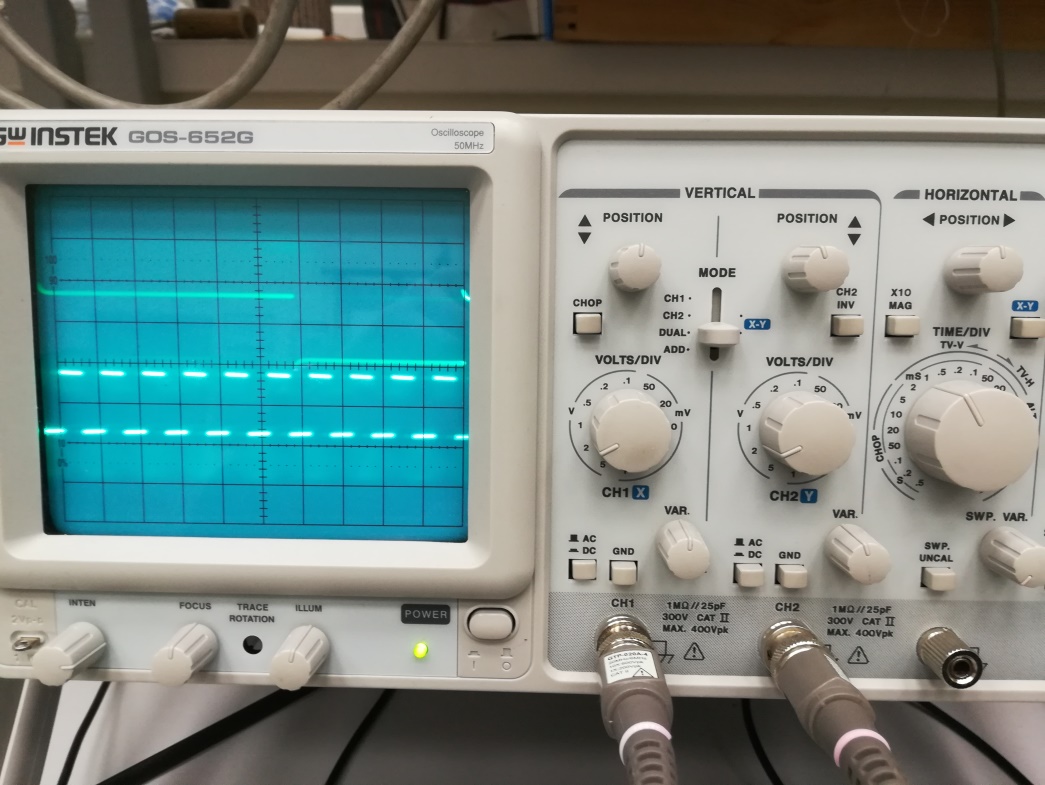


Рисунок 4 – Сигналы м-последовательности и генератора

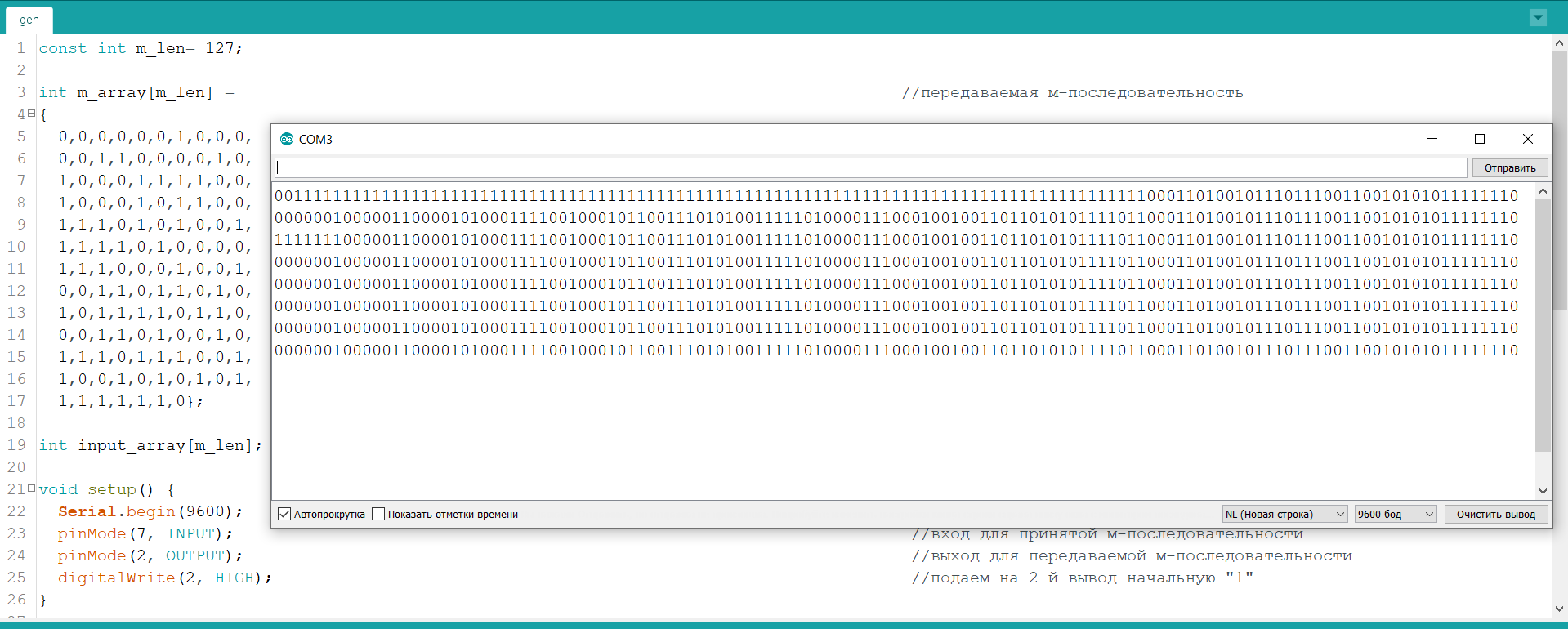


Рисунок 5 – М-последовательность без помехи

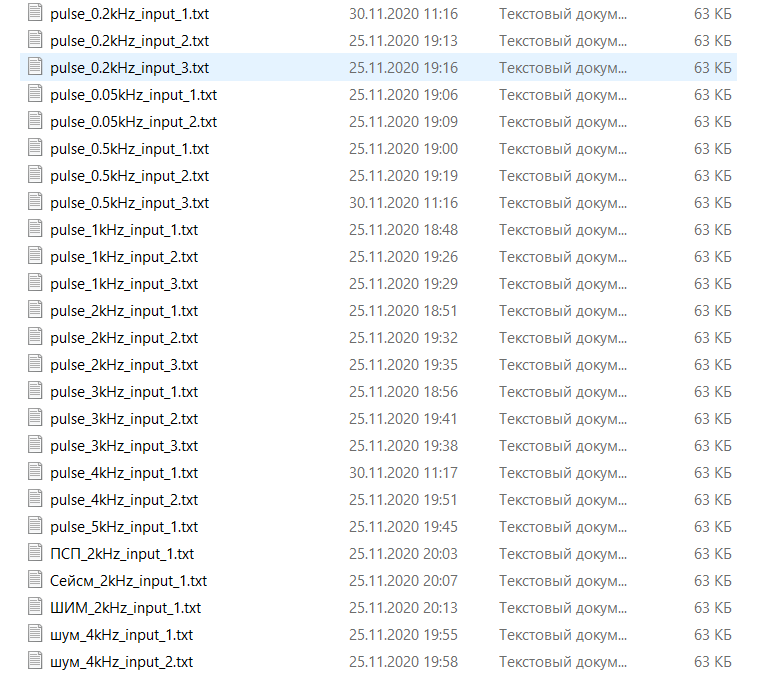


Рисунок 6 – Текстовые файлы для различных условий

Код программы Arduino Nano:

const int m\_len= 127; //длина м-последовательности

int m\_array[m\_len] = //эталонная м-последовательность

{

0,0,0,0,0,0,1,0,0,0,0,0,1,1,0,0,0,0,1,0,

1,0,0,0,1,1,1,1,0,0,1,0,0,0,1,0,1,1,0,0,

1,1,1,0,1,0,1,0,0,1,1,1,1,1,0,1,0,0,0,0,

1,1,1,0,0,0,1,0,0,1,0,0,1,1,0,1,1,0,1,0,

1,0,1,1,1,1,0,1,1,0,0,0,1,1,0,1,0,0,1,0,

1,1,1,0,1,1,1,0,0,1,1,0,0,1,0,1,0,1,0,1,

1,1,1,1,1,1,0};

int input\_array[m\_len]; //входящая м-последовательность

void setup() { //параметры инициализациии

Serial.begin(9600); //скорость обмена

pinMode(2, INPUT); //вход данных с приемника

pinMode(4, OUTPUT); //вывод питания приемника

digitalWrite(4, HIGH); //на вывод питания приемника 5В

pinMode(7, OUTPUT); //земля передатчика

digitalWrite(7, LOW); //низкий уровень на землю передат.

pinMode(8, OUTPUT); //вывод питания передатчика

digitalWrite(8, HIGH); //на вывод питания передатчика 5В

pinMode(9, OUTPUT); //выход данных на передатчик

digitalWrite(9, HIGH); //нач. значение на выводе данных

}

void my\_print() { //функция вывода в монитор порта

for (int j=0; j<m\_len; j++)

{

Serial.print(String(input\_array[j])); //выводим входящую м-посл.

}

Serial.print("\n"); //выводим символ переноса

}

void loop() { //основной цикл

for (int j=0; j<m\_len; j++)

{

if (m\_array[j] == 1){ //если элемент массива равен 1

digitalWrite(9, HIGH); //отправляем в передатчик "1"

}

if (m\_array[j] == 0){ //если элемент массива равен 0

digitalWrite(9, LOW); //отправляем в передатчик "0"

}

delay(1); //пауза 1 мс

if (digitalRead(2) == HIGH){ //если на входе высокий уровень

input\_array[j] = 1; //записываем в массив 1

}

if (digitalRead(2) == LOW){ //если на входе низкий уровень

input\_array[j] = 0; //записываем в массив 0

}

delay(1); //пауза 1 мс

}

my\_print(); //вызываем функцию сообщения

delay(10); //пауза 10 мс

}

Код вспомогательного скрипта на языке Python3 для чтения последовательного порта и записи данных в файл:

import serial # импорт библиотеки

import time # импорт библиотеки

idcom = [] # пустой массив

m\_file = 'pulse\_0.2kHz\_input\_1.txt' # имя файла

i = 0 # начальный индекс

ser = serial.Serial(

port = 'COM3',\

baudrate=9600,\

parity=serial.PARITY\_NONE,\

stopbits=serial.STOPBITS\_ONE,\

bytesize=serial.EIGHTBITS,\

timeout=None)

print("connected to: " + ser.portstr + "\n")

files1 = open(m\_file, 'w') # открываем файл в режиме перезаписи

while (i != 500):

m = '' # объявляем строковую переменную

for line in ser.readline():

idcom.append(line)

for j in range(len(idcom)):

if idcom[j] == 48: #если принят 0

m = m + '0' #добавить в строку 0

if idcom[j] == 49: #если принят 1

m = m + '1' #добавить в строку 1

print(m + "\n")

files1.write(m +'\n') # запись в файл

del(m) # удаляем строковую переменную

idcom.clear() # очищаем массив

i += 1

files1.close() # закрываем файл

print('запись закончена') # сообщение в консоль