

Группа М32051

К работе допущен \_\_\_\_\_

Студент Пантелеев, Артамонов

Работа выполнена \_\_\_\_\_

Преподаватель Ефремова Е.А.

Отчет принят \_\_\_\_\_

## **Рабочий протокол и отчет по задаче от лектора №1**

## Задание

Сформировать импульс, близкий к прямоугольному, из спектра плоских гармонических волн с несущей длиной волны  $\lambda_0 = 1.5$  мкм и длительностью  $\Delta t \approx 10$  нс. Какая спектральная ширина пакета? Промоделировать его прохождение через среду с заданной дисперсией фазовой скорости  $v_{\text{фаз}} = \sqrt{c^2 + b^2 \lambda^2}$  для  $b = \{1, 10, 100\}$ . Определить характерное время расплывания пакета.

Закодировать сообщение (короткое слово) такими импульсами и промоделировать передачу такого сообщения до времен порядка характерного времени расплывания пакета.

## Решение

Предоставленный код генерирует сигнал меандра для передачи сообщения в виде двоичного кода. Он использует функцию «meander(binary\_digits, frequency)», которая принимает двоичный код сообщения и количество точек на графике для каждого бита кода («frequency»).

Алгоритм работы функции «meander» следующий:

- 1) Преобразование двоичного кода в список цифр.
- 2) Расчет длины сигнала, которая равна длине списка цифр, умноженной на переменную «frequency».
- 3) Для каждого бита кода определяется диапазон времени, в течение которого сигнал будет иметь соответствующее значение.
- 4) Значения списка заполняются 1 или 0 в зависимости от значения текущего бита, и результатом является список значений сигнала меандра.

Для отображения полученных значений на графике используется библиотека «Matplotlib». В результате выполнения данного кода будет построен график сигнала меандра, содержащий значения, соответствующие переданному двоичному коду слова «physics».

Также отмечается, что значение переменной «frequency» может быть изменено при необходимости для увеличения или уменьшения числа точек на графике. Это позволяет контролировать детализацию и гладкость графика сигнала меандра.

## Исходный код

```
import matplotlib.pyplot as plt

def to_binary_digits(s):
    """
    Переводит строку в двоичный код (список нулей и единиц)

    :param s:
    :return:
    """
    return [int(digit) for i in s for digit in f"{ord(i):b}"]

def meander(binary_digits, frequency):
    """
    Функция преобразует двоичный код в список значений функции меандра.

    :param binary_digits: двоичный код
    :param frequency: количество точек на графике для каждого бита кода (начало и конец)
    :return: список значений функции меандра
    """
    return [int(digit) for digit in binary_digits for _ in range(frequency)]

word = "physics"
word_binary_digits = to_binary_digits(word)

print(f"Word: {word}")
print(f"Binary: {word_binary_digits}")

meander_values = meander(
    binary_digits=word_binary_digits,
    frequency=2
)

x_labels = list(range(len(meander_values)))
plt.plot(x_labels, meander_values)
plt.xlabel("Time")
plt.ylabel("Value")
plt.title(f"Word: {word}")
plt.show()
```

Исходный код опубликован на GitHub и может быть скачан и запущен:  
<https://github.com/SmartOven/Python/blob/master/Physics/LecTask1/main.py>