

## Лабораторная работа №1

### «Дискретизация аналоговых сигналов»

Целью курса «Радиофизическая лаборатория» является практические изучения разделов цифровой обработки сигналов, относящихся к вопросам дискретизации и спектрального анализа.

#### Задачи дисциплины

- изучение основ цифрового спектрального анализа с проведением компьютерного моделирования с помощью библиотек Numpy, Scipy, Matplotlib языка программирования Python;
- приобретение навыков анализа сигналов с помощью цифрового осциллографа;
- овладение алгоритмами расчета дискретного преобразования Фурье;
- получение необходимых базовых знаний для дальнейшего изучения цифрового спектрального анализа и цифровой фильтрации сигналов.

Курс «Радиофизическая лаборатория» является годовым.

#### Осенний семестр

- Выполняются две лабораторные работы.
- Недифференцированный зачет по итогам сдачи двух лабораторных работ.
- Необходимое и достаточное условие получения зачета в осеннем семестре: наличие положительных оценок по двум лабораторным работам.

#### Весенний семестр

- Выполняются две лабораторные работы.
- Дифференцированный зачет за курс по итогам сдачи четырех лабораторных работ (две из осеннего семестра, две из весеннего).
- Наличие положительных оценок по всем четырем работам является необходимым условием для зачета.

Занятия проходят в соответствии с расписанием на сайте института по четным или нечетным неделям.

На двух занятиях в семестре проходит сдача. Дату сдачи сообщает преподаватель на занятиях.

## Электронные ресурсы

Материалы размещаются в LMS и на сайте кафедры

**<http://kprf.mipt.ru/> ->**

**-> Учебные курсы ->**

**-> Радиофизическая лаборатория**

- Для гостевой авторизации можно вводить пару **student student5xx**
- Сайт доступен из локальной сети института и через интернет.
- При сохранении .ipynb файлов с сайта обратите внимание, что расширение должно быть **.ipynb** а не **.ipynb.txt**.

## Список рекомендуемой литературы

[1] Романюк Ю.А. Основы цифровой обработки сигналов. В 3-х ч. Ч.1. Свойства и преобразования дискретных сигналов. Изд. 2-Е, . М.: МФТИ, 2007. 332 с.

[2] Романюк Ю.А. Дискретное преобразование Фурье в цифровом спектральном анализе. Учебное пособие. М.: МФТИ, 2007. 120 с.

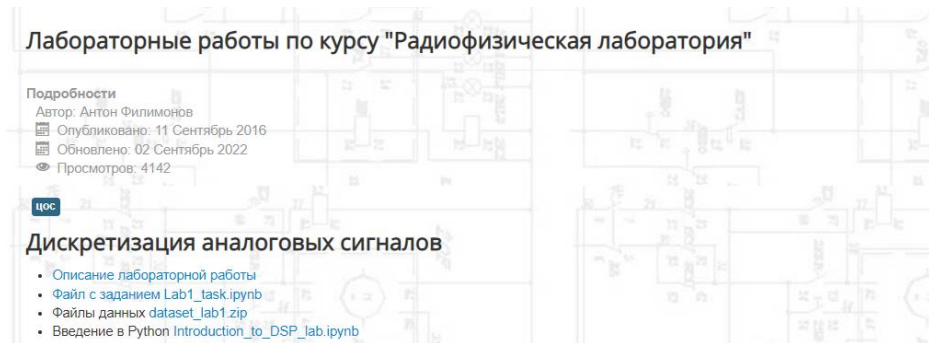
[3] Солонина А.И. Цифровая обработка сигналов в зеркале Matlab. СПб.: БХВ-Петербург, 2021. 560 с.

Учебные пособия [1], [2], [3] есть в библиотеке МФТИ.

## Лабораторная работа №1 «Дискретизация аналоговых сигналов»

- Занятие 1. Классификация сигналов: аналоговые, дискретные, цифровые.
- Занятие 2. Спектры импульсных и периодических сигналов.
- Занятие 3. Эффект наложения спектров при дискретизации сигналов.
- Занятие 4. Сдача.

<http://kprf.mipt.ru/index.php/opisaniya-laboratorykh-rabot/51-laboratornye-raboty-po-tsifrovoj-obrabotke-signalov>



## Introduction\_to\_DSP\_lab.ipynb

Файл ipynb (для Jupyter Notebook) с введением в Python.

## Лр\_1\_Дискретизация\_сигналов.pdf

Описание лабораторной работы

## Lab1\_task.ipynb

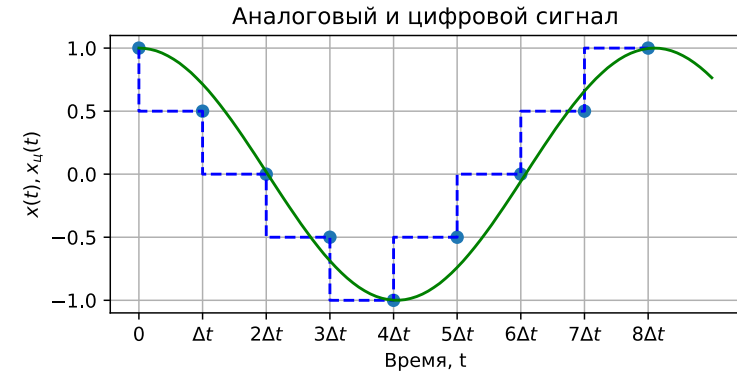
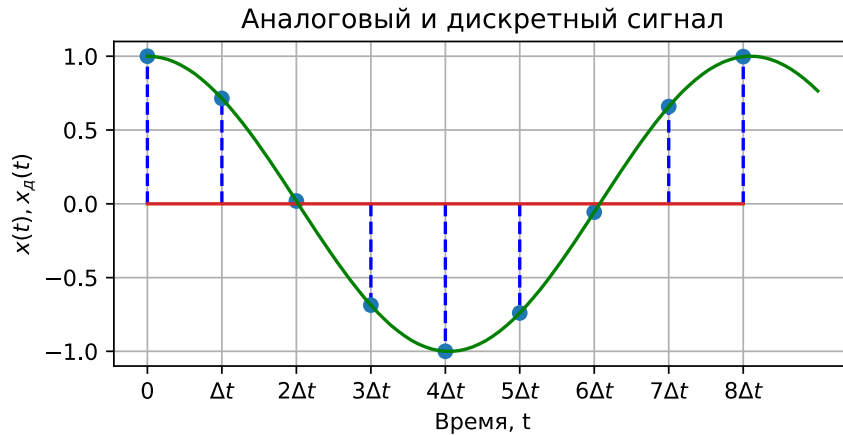
Задания лабораторной работы и шаблон отчета

## dataset\_lab1.zip

Файл с данными

## Классификация сигналов: аналоговые, дискретные, цифровые.

Под **сигналом** обычно понимают величину, отражающую состояние физической системы. Поэтому естественно рассматривать сигналы как функции, заданные в физических координатах. Примером могут служить одномерные сигналы, заданные как функции времени  $x(t)$ , двумерные сигналы заданные на плоскости  $I(x, y)$ . В качестве сигналов могут выступать различные величины. Пример одномерного сигнала — зависимость напряжения в сети от времени  $U(t) = A \cdot \cos(2\pi f_0 t)$ . Далее мы будем рассматривать в основном одномерные сигналы.



**Аналоговые** или непрерывные сигналы  $x(t)$  описываются непрерывными и кусочно-непрерывными функциями, причем как сама функция, так и ее аргумент могут принимать любые значения в пределах некоторого интервала.

**Дискретные сигналы**, могут быть описаны в виде счетного набора отсчетов (значений) в заданные моменты времени  $k\Delta t$ ,  $k \in \mathbb{Z}$ , где  $\Delta t$  — шаг дискретизации. Частота дискретизации  $f_d$  (размерность в Гц) — это величина, обратная шагу дискретизации  $f_d = 1 / \Delta t$ .

**Цифровые сигналы**, помимо того, что они являются дискретными, могут принимать лишь конечное число значений, соответствующих уровням квантования. Процесс преобразования аналогового сигнала в цифровой состоит из дискретизации и квантования, которые осуществляются аналого-цифровым преобразователем (АЦП). Обычно число уровней квантования  $2^m$ , где  $m$  — разрядность АЦП.

Для дискретных сигналов будем использовать следующие описания.

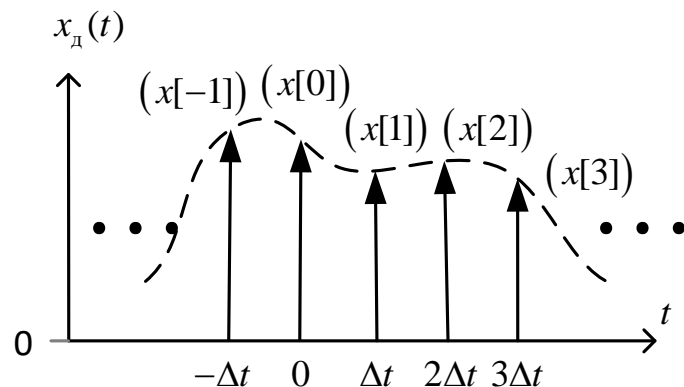
### 1) Функция дискретного времени $k$ .

Это описание дискретного сигнала в виде последовательности отсчетов  $x[k]$  в заданные моменты времени  $k\Delta t$ ,  $k \in \mathbb{Z}$ , где  $\Delta t$  — шаг дискретизации. Далее мы будем использовать квадратные скобки для обозначения функций дискретного аргумента.

### 2) Функция непрерывного времени $t$ (континуальная запись).

$$x_d(t) = \sum_{k=-\infty}^{\infty} x[k] \delta(t - k\Delta t)$$

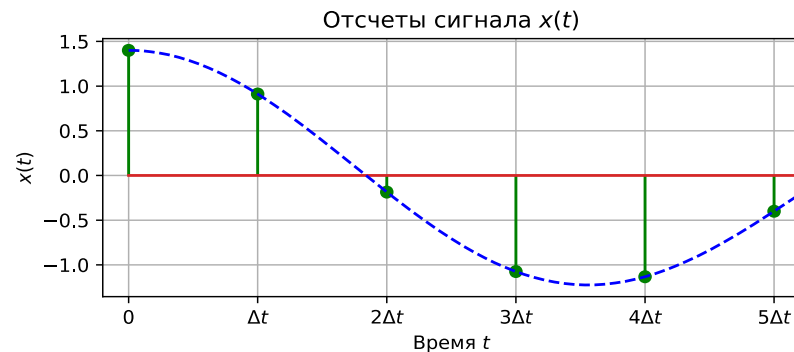
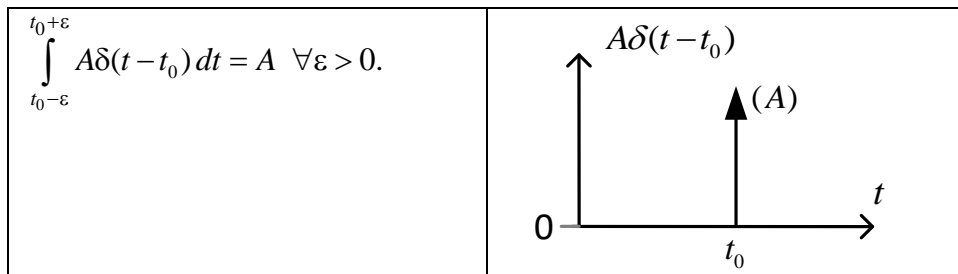
В этой записи дискретный сигнал представляет собой последовательность дельта-функций с площадями  $x[k]$ .

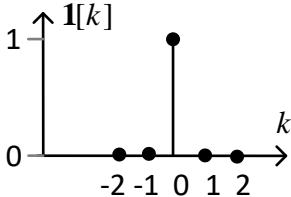
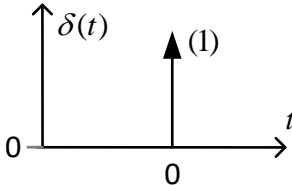
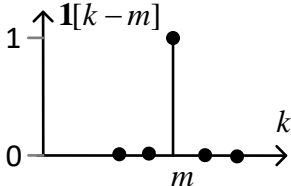
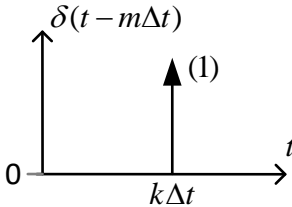


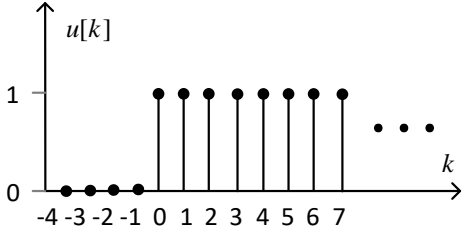
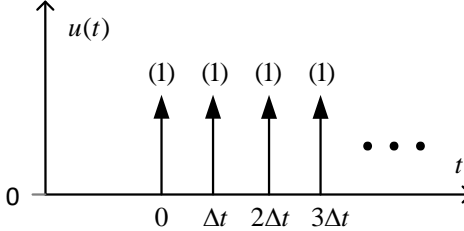
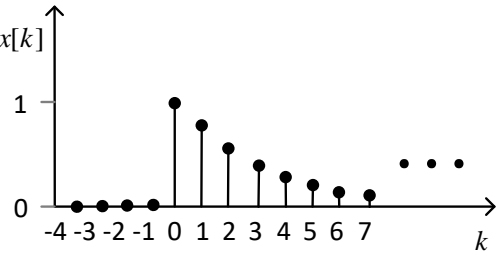
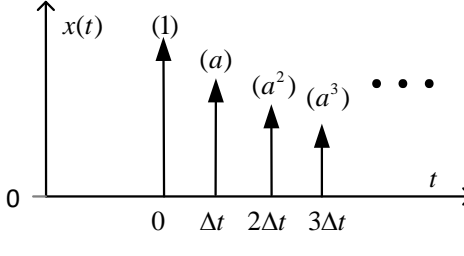
Для дельта-функции справедливо соотношение

$$\int_{t_0-\varepsilon}^{t_0+\varepsilon} \delta(t-t_0) dt = 1 \quad \forall \varepsilon > 0.$$

Рассмотрим сигнал  $y(t) = A\delta(t-t_0)$ ,  $A \in \mathbb{C}$ . Это дельта-функция в точке  $t_0$  оси времени. Площадь под графиком

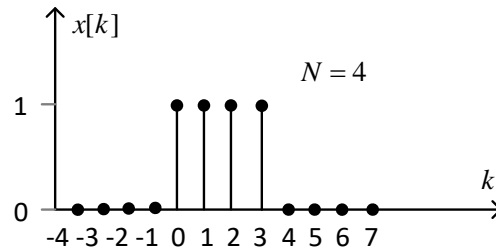


Дискретный сигнал	Описание в виде функции дискретного времени	Описание в виде функции непрерывного времени
Единичный импульс в точке 0	$\mathbf{1}[k] = \begin{cases} 1, & \text{при } k = 0, \\ 0, & \text{при } k \neq 0. \end{cases}$ <p><math>\mathbf{1}[k]</math> — единичный импульс</p> 	$\delta(t)$ <p><math>\delta(t)</math> — дельта-функция Дирака</p> 
Единичный импульс в точке m	$\mathbf{1}[k - m] = \begin{cases} 1, & \text{при } k = m, \\ 0, & \text{при } k \neq m. \end{cases}$ 	$\delta(t - m\Delta t)$ 

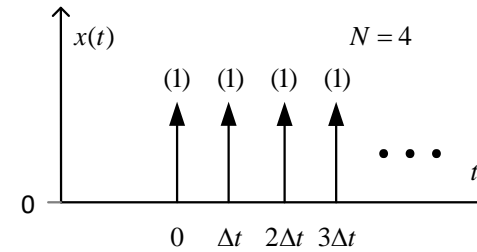
<p>Дискретная функция включения</p>	$u[k] = \begin{cases} 1, & \text{при } k \geq 0, \\ 0, & \text{при } k < 0. \end{cases}$ 	$\sum_{m=0}^{\infty} \delta(t - m\Delta t)$ 
<p>Дискретная экспонента</p>	$x[k] = \begin{cases} a^k, & \text{при } k \geq 0, \\ 0, & \text{при } k < 0. \end{cases}$  <p style="text-align: center;">случай <math>0 &lt; a &lt; 1</math></p>	$\sum_{m=0}^{\infty} a^m \delta(t - m\Delta t)$  <p style="text-align: center;">случай <math>0 &lt; a &lt; 1</math></p>

Последовательность из  $N$  единичных импульсов

$$x[k] = \sum_{m=0}^{N-1} \mathbf{1}[k - m]$$



$$x(t) = \sum_{m=0}^{N-1} \delta(t - m\Delta t)$$





## Практическая часть

В качестве средства моделирования мы будем использовать библиотеки языка Python 3 Numpy, SciPy, Matplotlib.

**Anaconda** — дистрибутив языков программирования Python и R, включающий набор популярных свободных библиотек, таких как NumPy, SciPy и Matplotlib.

Сайт проекта <https://www.anaconda.com/>

**Spyder** — свободная и кроссплатформенная интерактивная IDE для научных расчетов на языке Python, обеспечивающая простоту использования функциональных возможностей и легковесность программной части.

**Jupyter Notebook** — интерактивная оболочка для различных языков программирования (в том числе Python), позволяющая объединить код, текст, формулы, результаты вычислений и графики в один документ и распространять его для других пользователей с сохранением возможности повторения вычислений. Расширение файлов **.ipynb** (IPython Notebook).

**Google Colaboratory** — онлайн-сервис, позволяющий работать с блокнотами **.ipynb** и производить эффективные вычисления с помощью виртуальных машин в облаке.

Сайт проекта <https://colab.research.google.com/>

**NumPy** — это библиотека с открытым исходным кодом, которая поддерживает работу с многомерными массивами и позволяет выполнять высокоуровневые математические операции над ними.

Соглашение об импорте:

```
import numpy as np
```

NumPy API Reference <https://numpy.org/doc/stable/reference/index.html>

Примеры одномерных массивов

```
np.array([1,2,3,4])
```

```
np.arange(start=1, stop=5, step=1)
```

```
np.zeros(4)
```

```
np.ones(4)
```

```
np.random.rand(4)
```

```
np.linspace(start=1, stop=25, num=4)
```

```
k=np.arange(10)
```

```
x=np.sin(2*np.pi*0.2*k)
```

```
y=np.cos(2*np.pi*0.2*k)
```

```
z=np.exp(-1j*2*np.pi*0.2*k)
```

**SciPy**— библиотека для языка программирования Python с открытым исходным кодом, предназначенная для выполнения научных и инженерных расчётов.

SciPy Reference <https://docs.scipy.org/doc/scipy-1.5.2/reference/>

- Fourier Transforms (scipy.fft) Дискретное преобразование Фурье
- Signal Processing (scipy.signal) Обработка сигналов
- Linear Algebra (scipy.linalg) Линейная алгебра
- Optimization (scipy.optimize) Численные методы оптимизации
- Interpolation (scipy.interpolate) Интерполяция
- Multidimensional image processing (scipy.ndimage) Цифровая обработка изображений

**Matplotlib** — библиотека на языке программирования Python для визуализации данных двумерной и трехмерной графикой.

<https://matplotlib.org/>

```
import matplotlib.pyplot as plt
plt.figure(figsize=[7, 4], dpi=100)
x=np.array([1,2,3,4])
k=np.arange(x.size)
plt.stem(k,x)
plt.xlabel("k")
plt.ylabel("x(k)")
plt.xticks(ticks=np.arange(x.size))
plt.title("Сигнал x(k)")
plt.grid()
plt.show()
```