

# Отчёт

Корпускулярно-волновой дуализм

4 Апреля 2024

## 1 Явление интерференции. Работа с интерферометром

### 1.1 Интерференция

**Интерференция света** — (интерференция электромагнитных волн) — перераспределение интенсивности света в результате наложения (*суперпозиции*) нескольких световых волн. Это явление обычно характеризуется чередующимися в пространстве максимумами и минимумами интенсивности света. Конкретный вид такого распределения интенсивности света в пространстве или на экране, куда падает свет, называется *интерференционной картиной*.

### 1.2 Принцип работы интерферометра Майкельсона

Луч лазера, попадая на светоделительную пластинку, разделяется на два луча (половина отразилась и еще половина преломилась), один из них попадает на зеркало, установленное под углом в  $90^\circ$  к падающему лучу, отражается от него и, приломляясь, попадает в коллиматор. Вторая часть луча, которая приломилась через полуволновую пластинку, отражается от зеркала на пьезоэлементе (оно будет колебаться) и, попадая обратно на полуволновую пластинку, отражается от неё и попадает в коллиматор (рис. 1). На выходе коллиматора будет видна *интерференционная картина* (рис. 2).

### 1.3 Принцип настройки интерферометра

1. Вращением нижнего колеса держателя светоделительной пластинки нужно повернуть его на угол  $45^\circ$  к лазеру по вертикали.
2. Вращением нижнего колеса держателя светоделительной пластинки нужно повернуть его на угол  $90^\circ$  к лазеру по горизонтали.
3. Вращением нижнего колеса держателя обыкновенного зеркала нужно совместить входящий и выходящий луч к зеркалу по вертикали.
4. Вращением верхнего колеса держателя обыкновенного зеркала нужно совместить входящий и выходящий луч к зеркалу по горизонтали.

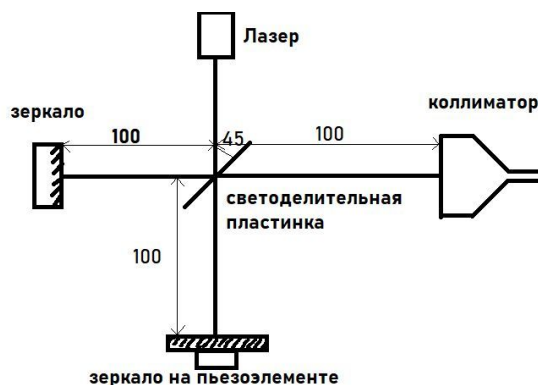


Рис. 1: Схема интерферометра Майкельсона

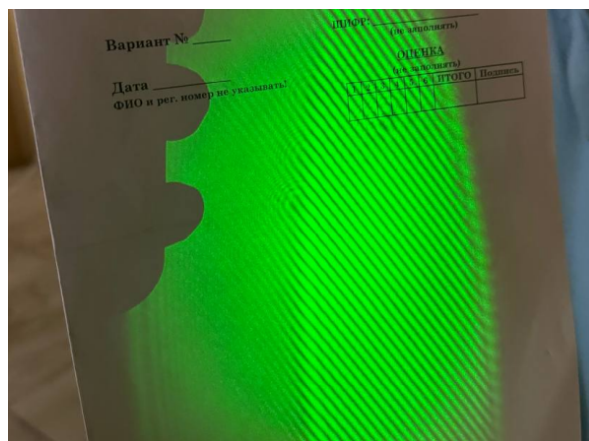


Рис. 2: Интерференционная картина

5. Вращением нижнего колеса держателя зеркала на пьезоэлементе нужно совместить входящий и выходящий луч к зеркалу по вертикали.
6. Вращением верхнего колеса держателя зеркала на пьезоэлементе нужно совместить входящий и выходящий луч к зеркалу по горизонтали.
7. Если лучи, идущие в коллиматор, не совпали, повторить все сначала.

#### 1.4 Принцип работы с готовым устройством

Для работы с высокоточным измерителем перемещений необходимо соблюдать технику безопасности: надеть специальные защитные очки, снять абсолютно все вещи с отражающими элементами. Также в ходе использования не трогать линзы или иные оптические элементы; в случае загрязнения протереть спиртовой салфеткой.

Для запуска интерферометра необходимо включить лазер, вставить оптоволоконный провод в коллиматор и в фотоприемник. Далее следует подключить провод фотоприемника к осциллографу, сохранить результат измерений и перенести его на компьютер, где скопировать данные в консоль программы **Python**. Программа выведет результат в виде характеристик премещения.

## 2 Погрешности

### 2.1 Основные формулы нахождения погрешности

Для нахождения средней квадратичной погрешности используем формулу:

$$S_{a_i} = \sqrt{\frac{\sum_i \Delta a_i^2}{n-1}} \quad (1)$$

где  $n$  - это количество измерений,  $a_i$  это  $i$ -е измерение данной величины, а  $\Delta a_i = a_i - \frac{\sum_i a_i}{n}$ .

#### 2.1.1 Период и количество интерференционных полос

Для нахождения погрешности периода колебаний зеркала и количества интерференционных полос умножим среднюю квадратическую погрешность на коэффициент Стьюдента с доверительной вероятностью 0,95.

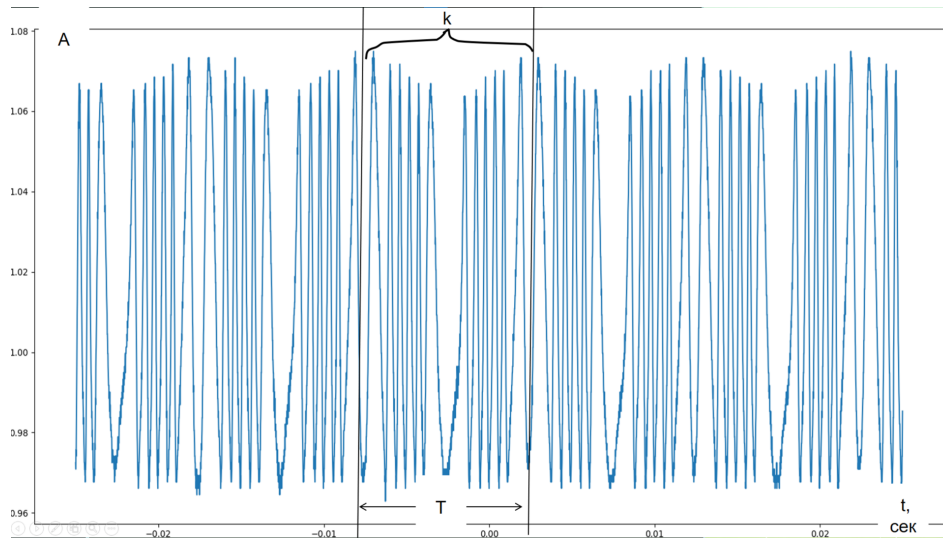


Рис. 3: Изначальные входные данные сигнала с осциллографа

### 2.1.2 Частота колебаний зеркала

Для нахождения погрешности нужно умножить саму величину на относительную погрешность:

$$\Delta\nu = \frac{1}{T} * \frac{\Delta T}{T} = \frac{\Delta T}{T^2} \quad (2)$$

Где  $T$  и  $\Delta T$  - это период колебаний и погрешность периода колебаний соответственно (см. пункт 2.1.1)

### 2.1.3 Амплитуда колебаний зеркала

Амплитуда колебаний  $A = \frac{k\lambda}{4}$ , где  $k$  - это количество прошедших интерференционных полос за полный цикл колебания зеркала, а  $\lambda$  - это длина волны зеленого лазера, равная 532 нанометров. Из этого можно посчитать погрешность амплитуды колебаний, где  $\Delta k$  - это погрешность  $k$ .

$$\Delta A = \frac{\Delta k * \lambda}{4} \quad (3)$$

### 2.1.4 Средняя скорость зеркала

Средняя скорость тела считается по формуле:  $v = \frac{S}{t}$ , где  $S$  и  $t$  - это общий путь и общее время соответственно, где  $S = \frac{\lambda k}{2}$  и  $t = T$ , а средняя скорость равна  $v = \frac{\lambda k}{2T}$ . Поэтому можем найти погрешность средней скорости в силу правила нахождения погрешности:

$$\Delta v = \frac{k\Delta\lambda}{2T} + \frac{k\lambda\Delta T}{2T^2} \quad (4)$$

### 2.1.5 Длина волны

Погрешность длины волны составляет 0,1 нанометров (Длина волны  $\lambda = 532$  нанометра).

## 3 Принцип работы программы

На вход даются данные с осциллографа (рис. 3): время и амплитуда. По этим данным программа выдает частоту колебаний зеркала, амплитуду перемещения зеркала и среднюю скорость перемещения зеркала.

Код был написан при помощи 3-х библиотек: numpy, matplotlib.pyplot и scipy.fftpack. При помощи последней сигнал с осциллографа фильтруется с использованием функций: fftfreq, ifft и fft (рис. 4). Сигнал считывается с консоли.

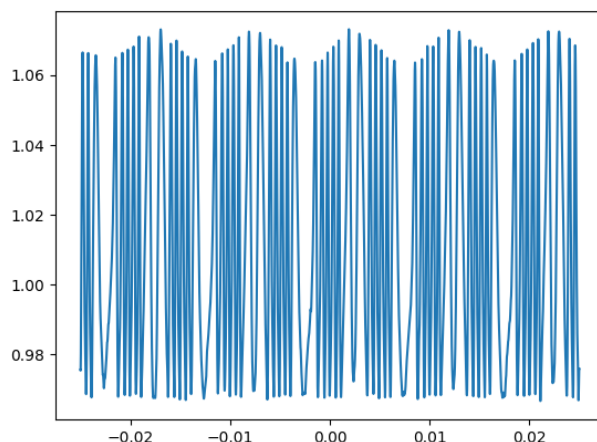


Рис. 4: Данные с осциллографа после быстрого преобразования Фурье

### 3.1

Заметим, что линии, которые разделяют график на "столбики", более пологие, чем линии внутри столбиков. Найдем среднее значение модуля снижения значения на оси  $y$  при помощи цикла.

```
1 def average_reduction_value(ampl_list):
2     reduction = []
3     for i in range(1, len(ampl_list) - 1):
4         reduction.append(np.abs(ampl_list[i] - ampl_list[i+1]))
5     return sum(reduction)/len(reduction)
```

### 3.2

Далее отметим точки, значения модуля снижения которых меньше среднего (линии более пологие), и которые, при прибавлении к ним среднего значения модуля снижения на оси  $y$ , становятся больше максимальной точки по оси  $y$  на графике. Также, между моментами времени этих точек должно пройти больше 0.0001 секунды.

```
1 sr_snij = average_reduction_value(ampl_list)
2 sr_zn = average_value(ampl_list)
3 vremena = []
4 launch = 0
5
6 for i in range(1, len(ampl_list) - 1):
7     if np.abs(ampl_list[i] - ampl_list[i+1]) < sr_snij and launch == 0:
8         launch = 1
9     if ampl_list[i] + sr_snij >= max(ampl_list) and launch == 1:
10        if len(vremena) == 0:
11            vremena.append([time_list[i], ampl_list[i]])
12        elif np.abs(vremena[-1][0] - time_list[i]) > 0.0001:
13            vremena.append([time_list[i], ampl_list[i]])
14    if np.abs(ampl_list[i] - ampl_list[i+1]) >= sr_snij and launch == 1:
15        launch = 0
```

### 3.3

Найдем между этими точками среднее расстояние по времени. Так как в одном "столбике" выделяется несколько точек, разнесем точки разных "столбиков" в разные списки, проверяя, больше

ли настоящее расстояние по времени между точками среднего расстояния, и, если больше, разность их в разные списки. В каждом списке находим максимальную по оси y точку (высшую точку в столбике). Между этими точками находим среднее расстояние по времени. Это и есть период колебания зеркала.

```

1 list_of_up = []
2 distance_in_time = []
3 for i in range(len(vremena) - 1):
4     tim, ampl = vremena[i]
5     distance_in_time.append(np.abs(vremena[i+1][0] - tim))
6 average_distance = sum(distance_in_time)/len(distance_in_time)
7 total_columns = []
8 interval_value = []
9 for i in range(len(distance_in_time)):
10     if distance_in_time[i] <= average_distance:
11         interval_value.append(i)
12     else:
13         interval_value.append(i)
14         total_columns.append(interval_value)
15         interval_value = []
16 for times in total_columns:
17     maximums = 0
18     total = []
19     for t in times:
20         if vremena[t][1] > maximums:
21             maximums = vremena[t][1]
22             total = vremena[t]
23     list_of_up.append(total)
24 periods = []
25 for i in range(len(list_of_up) - 1):
26     tim, ampl = list_of_up[i]
27     periods.append(np.abs(list_of_up[i+1][0] - tim))
28
29 period = sum(periods)/len(periods)

```

### 3.4

Теперь найдем количество полос, обозначающих прохождение интерференционных полос, найдя среднее значение на оси y и пройдя циклом по значениям на оси y. Если значение было больше или равно среднему и меньше или равно среднему значению плюс среднее значение снижения, и при этом находилось далее 0.0001 секунды от предыдущего подходящего значения, количество полос увеличивалось на один. Далее разделим количество полос на количество "столбиков", которое равно количеству секунд в графике разделить на период одного "столбика".

```

1 for i in range(len(time_list)):
2     a = len(list_of_lines) == 0
3     if range_of_values[0] <= ampl_list[i] <= range_of_values[1]:
4         if a:
5             list_of_lines.append(i)
6         else:
7             if np.abs(time_list[list_of_lines[-1]] - time_list[i]) > 0.0001:
8                 list_of_lines.append(i)
9
10 counts_of_lines = round(len(list_of_lines)/wave)

```

По формулам из пункта 2 найдем амплитуду, частоту и среднюю скорость зеркала, а также погрешность

## 4 Результаты обработки эталонных и экспериментальных данных

### 4.1 Эталонные

Частота колебания зеркала равна 93.604 Гц, программная погрешность может быть равна 12.701 Гц

Амплитуда перемещений зеркала равно 1729.0 нм, программная погрешность может быть равна 0.325 нм

Средняя скорость перемещения зеркала равна 323681.747 нм/с, программная погрешность может быть равна 43979.987 нм/с

Все данные были округлены с точностью до трёх знаков после запятой