Санкт-Петербургский Государственный Электротехнический Университет "ЛЭТИ"

кафедра физики

Задание №1 по дисциплине

"Физические основы информационных технологий"

Название: Искривление луча в оптическом канале

|  |  |
| --- | --- |
| Фамилия И.О.: | Павлов Д.Р. |
| группа: | 9303 |
| Преподаватель: | Альтмарк А.М. |
| Итоговый балл: |  |
|  |  |

Крайний срок сдачи: 23.10.21

.

Санкт-Петербург 2021

Условие задания

Найти длину траектории светового луча *S* в прямолинейном оптоволоконном канале, Рис.1. Функцию распределения показателя преломления n1(y) по поперечной координате Y, начальный угол ввода луча α в волновод, длину канала *L*, диаметр канала D можно взять в таблице 1. Ввод луча осуществляется из центральной части канала с координатой *y*=0. Параметры *L* и D даны в безразмерных координатах.

Необходимо построить график траектории луча, а также записать ответ *S* в безразмерных единицах в текстовый файл LR1\result.txt. Помимо текстового файла result.txt в папке LR1 должен находиться Word-файл с отчетом, а также файл с кодом (Python, Mathcad, Mathematica). Для лучшего понимания отчетности смотрите папку “Пример организации яндекс-папки студентов”.

Пример содержания файла result.txt:

4.53258

n2

Y

Z

2DDD

n1

α

n2

*L*

Рисунок.1

**Вариант 19**

**Основные теоретические положения.**

**Закон преломления**.

При прохождении луча света через границу различных сред направление распространения изменяется. Это изменение зависит от соотношения показателей преломления данных сред:

,

где – угол падения, – угол преломления, и – показатели преломления в первой и второй среде соответственно.

**Закон полного отражения**.

При отражении луча света от граничного слоя (например, воздуха и стекла) угол падения равен точно углу отражения .

**Волоконно-оптическая связь**.

Способ передачи информации, использующий в качестве носителя информационного сигнала электромагнитное излучение оптического (ближнего инфракрасного) диапазона, а в качестве направляющих систем — волоконно-оптические кабели. В основе волоконно-оптической связи лежит явление полного внутреннего отражения электромагнитных волн на границе раздела диэлектриков с разными показателями преломления. Оптическое волокно состоит из двух элементов — сердцевины, являющейся непосредственным световодом, и оболочки. Показатель преломления сердцевины несколько больше показателя преломления оболочки, благодаря чему луч света, испытывая многократные переотражения на границе сердцевина-оболочка, распространяется в сердцевине, не покидая её.

**Выполнение работы**

Разобьём ось Y, вдоль которой меняется показатель преломления, и для каждого разбиения посчитаем угол, на который отклоняется луч, проходя через среду. Для каждого разбиения вычисляется функция распределения показателя преломления n1(y), затем по закону преломления света вычисляется синус угла преломления. Геометрическая длина пути вычисляется согласно определению синуса и теореме Пифагора.

Таким образом были получены длина траектории светового луча и следующий график (см. рисунок 3):

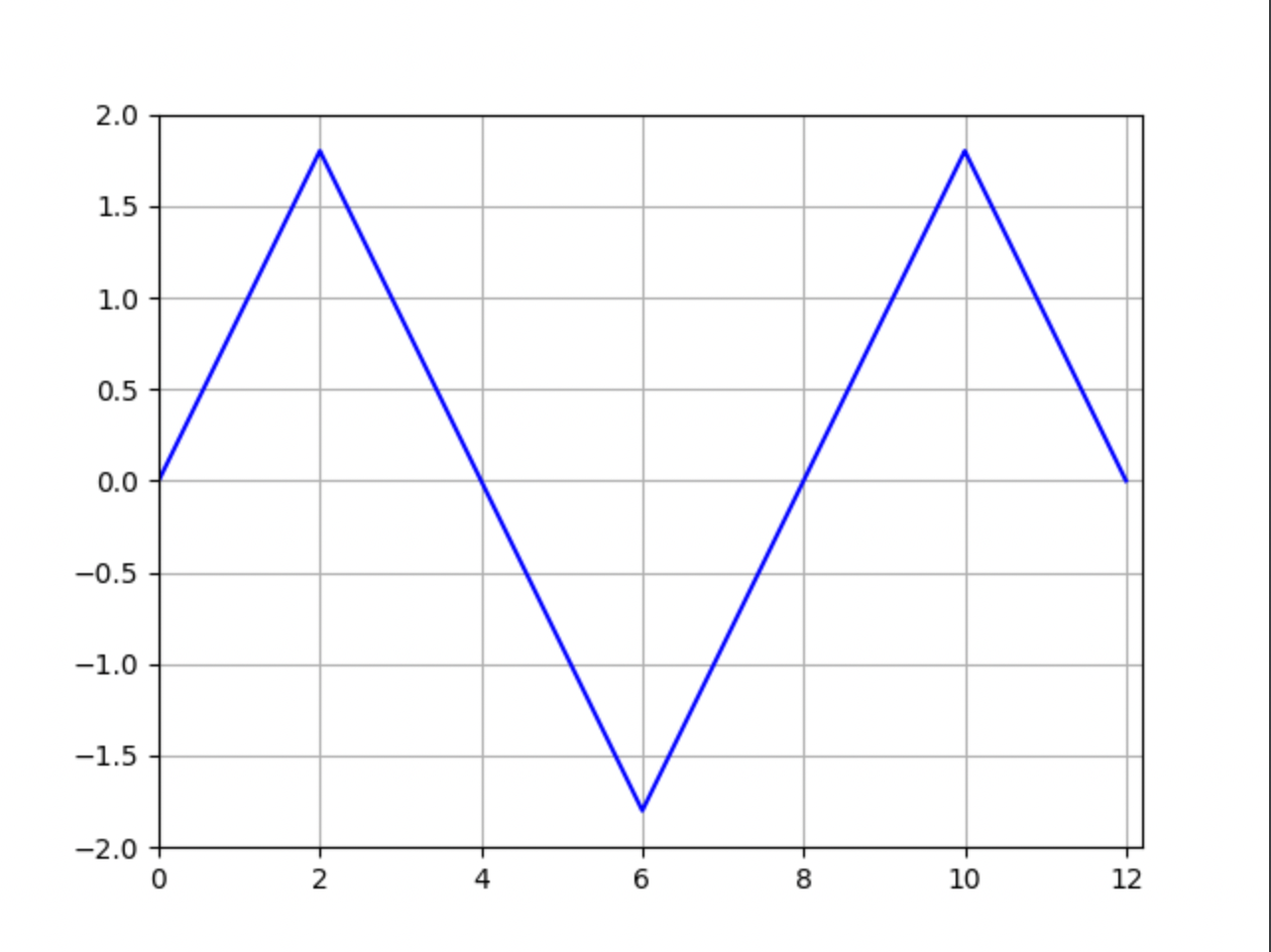


Рисунок 2 – График траектории оптического луча

**ПРОГРАММА MAIN.PY**

import math

import matplotlib.pyplot as plt

D = 1.8

L = 12

alpha\_0 = math.radians(90 - 42)

dy = 0.0001

n2 = 1

def n(y):

return 1.4 + 0.3 \* math.cos(math.radians(0.8 \* pow(y, 4)))

def calc\_sin\_beta(sin\_alpha, n1, n2):

return sin\_alpha \* n1 / n2

def calculate\_trajectory(dy):

z, y = 0.0, 0.0

points = [(z, y)]

sin\_alpha = math.sin(alpha\_0)

n1 = n(y)

res = 0.0

while z < L:

n2 = n(y)

sin\_beta = calc\_sin\_beta(sin\_alpha, n1, n2)

if abs(y + dy) > D or sin\_beta >= 1:

dy = -dy

sin\_beta = sin\_alpha

res += abs(dy) / (math.sqrt(1 - sin\_beta \*\* 2))

z += abs(dy) / math.sqrt(1 / sin\_beta \*\* 2 - 1)

n1 = n2

sin\_alpha = sin\_beta

points.append((z, y))

y += dy

return res, points

def show\_plot(points, D):

plt.axis([0, L+0.2, -D-0.2, D+0.2])

plt.grid(True)

plt.plot([point[0] for point in points], [point[1] for point in points], color='b')

plt.show()

def res\_to\_file(res):

with open("result.txt", 'w') as file:

file.write(str(res))

if \_\_name\_\_ == "\_\_main\_\_":

result, points = calculate\_trajectory(dy)

print("Result: ", result)

res\_to\_file(result)

show\_plot(points, D)