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.: | Смирнов Д.Ю. |
| группа: | 1303 |
| Преподаватель: | Альтмарк А.М. |
| Итоговый балл: |  |
|  |  |

Крайний срок сдачи: 22.10.23

Санкт-Петербург 2023

Условие задания

Найти длину траектории светового луча *S* в прямолинейном дисперсионном оптоволоконном канале, Рис.1, c показателем преломления n1. Оптоволокно окружено средой с показателем преломления n2. Функцию распределения показателя преломления n1(y, ω) можно представить как:

,

где y – поперечная координата, ω – циклическая частота светового луча.

Функцию , функцию Zf(y), описывающую координату z выходного торца волновода, начальный угол ввода луча α в волновод, координату ввода луча в волновод *y*=y0, радиус канала R можно взять в файле FOIT\_IDZ1.xlsx. Все геометрические размеры даются в безразмерных координатах.

Необходимо построить график траектории луча, а также записать ответ *S* в текстовый файл IDZ1\IDZ1.txt. Помимо текстового файла IDZ1.txt в папке IDZ1 должен находиться Word-файл (Pdf-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 IDZ1.txt:

4.53258

n2

Y

R

Zf(y)

y0

n1

Z

α

0

n2

-R

n2

Рисунок.1

Исходные значения, согласно варианту:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Вар | R | n2 | f1(y) | Zf(y) | ɷ \* 10^14, рад/с | y0 | α, град |
| 25 | 0.6 | 1 | 1.2 + 0.3\*Cos[0.8\*y]^3 | 42 + 3\*Sin[17.951958020513104\*y] | 3.3 | 0.3 | 22 |

Теоретические положения

При переходе между средами с разным коэффициентом преломления, луч преломляется (см. рисунок 1). Новый угол можно найти по формуле:

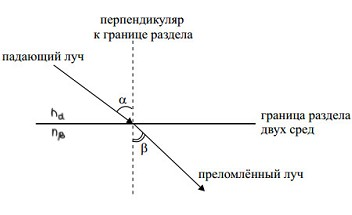


Рисунок 1 - преломление света

Плоские оптические волноводы — тонкие диэлектрические плёнки с малым поглощением на прозрачной подложке с относительно малым поглощением, пространственно неоднородные прозрачные структуры для направления света. Применяются в качестве передающей среды в системах оптической связи.

Показатель преломления: , где – скорость света в вакууме, – скорость света в среде.

Полное внутреннее отражение – это явление, при котором свет, падающий на границу двух сред из среды с большим показателем преломления () под углом, превышающим предельный угол , не преломляется, а полностью отражается.

**ПРИЛОЖЕНИЕ А**

**ИСХОДНЫЙ КОД ПРОГРАММЫ**

Файл main.py:

import numpy as np

import matplotlib.pyplot as plt

# Constants

def alpha():

return np.deg2rad(22)

def y0():

return 0.3

def omega():

return 3.3 \* 10 \*\* 14

def n2():

return 1

def R():

return 0.6

def dR():

return 0.0001

# Functions

def f1(y):

return 1.2 + 0.3 \* np.cos(0.8 \* y) \*\* 3

def Zf(y):

return 42 + 3 \* np.sin(17.951958020513104 \* y)

def n1(y, om=omega()):

return f1(y) \* (1 - ((0.35 \* 10 \*\* 14) / om) \*\* 2)

def findYZ(y, z, sinAngle, direction):

y += dR() \* np.sqrt(1 - sinAngle \*\* 2) \* direction

z += dR() \* sinAngle

return y, z

def main():

arrayZ = [0]

arrayY = [y0()]

trajectoryLen = 0

currentN = n2()

nextN = n1(y0())

currentY = y0()

currentZ = 0

currentAlpha = alpha()

sinAlpha = np.sin(np.pi / 2 - np.arcsin((np.sin(currentAlpha) \* currentN) / nextN))

verticalDirection = 1

currentN = nextN

while currentZ <= Zf(currentY):

currentY, currentZ = findYZ(currentY, currentZ, sinAlpha, verticalDirection)

nextN = n1(currentY)

if abs(currentY) >= R():

nextN = n2()

sinBeta = (currentN \* sinAlpha) / nextN

if sinBeta >= 1:

verticalDirection \*= -1

sinBeta = sinAlpha

currentN, sinAlpha = nextN, sinBeta

trajectoryLen += dR()

arrayZ.append(currentZ)

arrayY.append(currentY)

print(trajectoryLen)

plt.plot([0, 45], [R(), R()], "lime")

plt.plot([0, 45], [-R(), -R()], "lime")

plt.plot([0, 0], [-R(), R()], "lime")

exitArrayY = np.arange(-R(), R(), 0.0001)

plt.plot(Zf(exitArrayY), exitArrayY, "lime")

plt.plot(arrayZ, arrayY, 'red')

plt.show()

if \_\_name\_\_ == '\_\_main\_\_':

main()