Санкт-Петербургский Государственный Электротехнический

Университет "ЛЭТИ"

кафедра физики

Задание №3 по дисциплине

"Физические основы информационных технологий"

Название: Прохождение сигнала через четырёхполюсник

|  |  |
| --- | --- |
| Фамилия И.О.: | Иванов А. С. |
| группа: | 1303 |
| Преподаватель: | Альтмарк А. М. |
| Итоговый балл: |  |
| Крайний срок сдачи: | 22.11.23 |

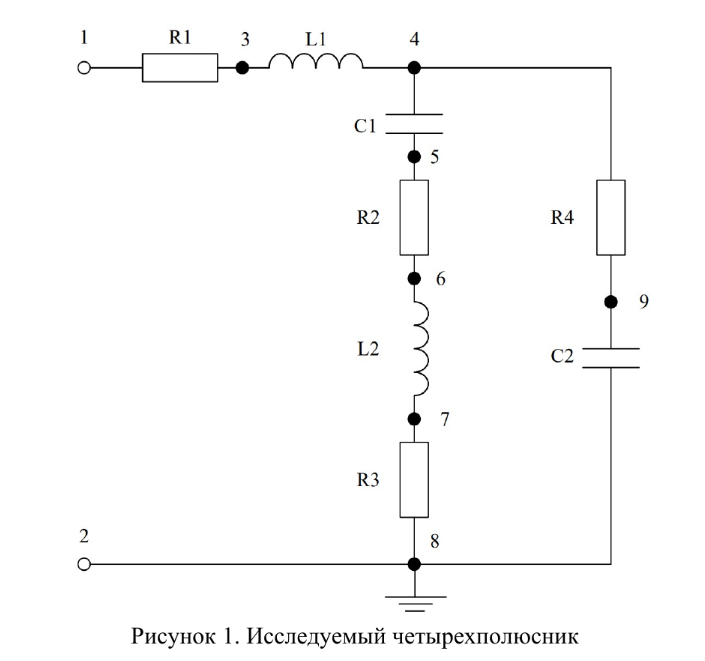
Санкт-Петербург 2023

Условие задания

Сигнал поступает на вход четырёхполюсника (контакты 1 и 2), показанного на рис.1. Четырехполюсник состоит из катушек индуктивности (L1 и L2), конденсаторов (C1 и C2) и резисторов (R1, R2, R3 и R4). Сигнал представляет собой суперпозицию гармонических функций:

Определить во сколько раз уменьшится амплитуда k-й гармоники на выходе четырехполюсника по сравнению с входным значением. В данном задании сигнал задается в виде массива из N элементов, который нужно взять из текстового файла. Каждый элемент массива соответствует определённому моменту времени ti. Интервал между соседними моментами времени также дается в задании и обозначается как δt=ti+1-ti. В данном задании требуется найти АЧХ передаточной характеристики четырехполюсника, а также применить алгоритм дискретного преобразования Фурье для определения спектрального состава сигнала. Необходимо построить график АЧХ в диапазоне циклических частот ω от 0 до 100 рад/с, а также графики сигнала и его спектра.

Параметры четырехполюсника, номер гармоники, номера выходных контактов четырехполюсника, массивы со входным сигналом можно найти в файле “FOIT\_IDZ3.xlsx”.



**Таблица с исходными данными**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Вар | L1, Гн | L2, Гн | С1, Ф | С2, Ф | R1, Ом | R2, Ом | R3, Ом | R4, Ом |
| 5 | 13.8628000224179 | 0.712836258342916 | 1.16422284290391E-05 | 1.47313522140348E-05 | 107.70872811802 | 30.4723663733368 | 1014.70686271098 | 501.625092437796 |

Количество отсчетов N (элементов массива): 8192

Время между соседними отсчетами (δt), c: 0.0196349540849362

Контакты выхода: 5 и 6

Номер гармоники: 4

Файл сигнала: 5.txt

**Теоретические сведения**

|  |  |  |
| --- | --- | --- |
| Резистор | ZR | R |
| Конденсатор | ZC |  |
| Катушка индуктивности | ZL | jωL |

**АЧХ** передаточной функции получается как |H(jω)|, и показывает изменение амплитуды сигнала в зависимости от его частоты.

**Преобразование Фурье** – операция, сопоставляющая для функции вещественной переменной другую функцию, описывающую коэффициенты ее разложения на элементарные гармонические колебания с разными частотами.

Для построения спектра сигнала, заданного списком значений, используется дискретное преобразование Фурье:

Fi=

Абсолютное значение |F| и будет **АЧХ** спектра сигнала. В нем, как следует из названия, есть шкалы амплитуды и частоты. **АЧХ** спектра гармонического колебания будет представлена одной чертой, а ω - координата будет соответствовать ее частоте в исходном сигнале.

Четырехполюсник представляет собой устройство, описываемое четырьмя параметрами, которые связывают входные и выходные напряжения и токи. В данном случае мы имеем четырехполюсник, состоящий из катушек индуктивности (L1 и L2), конденсаторов (C1 и C2) и резисторов (R1, R2, R3 и R4).

Передаточная функция системы H =

Она показывает связь между входным и выходным напряжениями.

Для нахождения входного и выходного напряжения нужно знать импеданс Z – комплексное сопротивление для гармонического сигнала:

**ПРИЛОЖЕНИЕ А**

**ИСХОДНЫЙ КОД ПРОГРАММЫ**

import matplotlib.pyplot as plt  
import numpy as np  
  
class CircuitAnalyzer:  
 def \_\_init\_\_(self):  
 self.L1 = 13.8628000224179  
 self.L2 = 0.712836258342916  
 self.C1 = 1.16422284290391E-05  
 self.C2 = 1.47313522140348E-05  
 self.R1 = 107.70872811802  
 self.R2 = 30.4723663733368  
 self.R3 = 1014.70686271098  
 self.R4 = 501.625092437796  
 self.N = 8192  
 self.dt = 0.0196349540849362  
 self.df = 1 / (self.dt \* self.N)  
 self.spectreAbs = None  
 self.sequence = None  
 self.H = None  
  
 def read\_signal\_from\_file(self, filename='5.txt'):  
 with open(filename, 'r') as file:  
 signal = [float(line) for line in file]  
 return signal  
  
 def draw\_signal(self, signal):  
 time = [self.dt \* i for i in range(self.N)]  
 plt.title('Сигнал')  
 plt.xlabel('t')  
 plt.ylabel('s')  
 plt.plot(time, signal)  
 plt.show()  
  
 def draw\_amplitude(self, is\_stop):  
 plt.title('Амплитуда входа')  
 plt.xlabel('ω')  
 plt.ylabel('A')  
 plt.plot(self.sequence[:is\_stop], self.spectreAbs[:is\_stop])  
 plt.show()  
  
 def calculate\_H(self, w):  
 j = complex(0, 1)  
 Z\_C1 = 1 / (j \* w \* self.C1)  
 Z\_C2 = 1 / (j \* w \* self.C2)  
 Z\_L1 = j \* w \* self.L1  
 Z\_L2 = j \* w \* self.L2  
 R\_input = self.R1 + Z\_L1 + (self.R4 + Z\_C2) \* (self.R2 + self.R3 + Z\_C1 + Z\_L2) / \  
 (self.R2 + self.R3 + self.R4 + Z\_C1 + Z\_C2 + Z\_L2)  
 R\_output = (self.R4 + Z\_C2) \* self.R2 / (self.R2 + self.R3 + self.R4 + Z\_C1 + Z\_C2 + Z\_L2)  
 return R\_output / R\_input  
  
 def draw\_amplitude\_frequency\_response(self, is\_stop):  
 plt.title('H')  
 plt.xlabel('ω, рад/c')  
 plt.ylabel('H')  
 plt.plot(self.sequence[1:is\_stop], self.H[:is\_stop - 1])  
 plt.show()  
  
 def analyze\_circuit(self):  
 signal = self.read\_signal\_from\_file()  
 self.draw\_signal(signal)  
  
 self.spectre = np.fft.fft(signal)  
 self.spectreAbs = [abs(each) for each in self.spectre]  
 self.sequence = [2 \* np.pi \* self.df \* i for i in range(self.N)]  
 is\_stop = round(50 / (np.pi \* self.df)) + 1  
 self.H = [abs(self.calculate\_H(w)) for w in self.sequence[1:]]  
 result = self.calculate\_H(40)  
  
 self.draw\_amplitude(is\_stop)  
 self.draw\_amplitude\_frequency\_response(is\_stop)  
  
 print(abs(result))  
  
circuit\_analyzer = CircuitAnalyzer()  
circuit\_analyzer.analyze\_circuit()