Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Пермский национальный исследовательский

политехнический университет»

(ПНИПУ)

Кафедра вычислительной математики и механики

Лабораторная работа

по дисциплине: «Моделирование процессов и систем»

по теме: «Основы теории конечных автоматов»

Выполнил:

Студент группы ИСТ-22-1б

Братчиков З.С.

Проверил:

доцент кафедры ВМиМ

Максимов Петр Викторович

Пермь

2024 г

**Цель работы:** Составить модель конечного автомата, задав его начальное состояние. Посмотреть, как будет изменяться состояние системы при подаче на конечный автомат разных входных сигналов.

**Задание к лабораторной работе:**

1. В рамках выполнения лабораторной работы необходимо научиться выделять основную информацию из формулировки задания: все возможные состояния системы и выходные воздействия.
2. Получить навык составления таблицы переходов
3. Научиться изображать полученную информацию в виде графа переходов состояний.
4. Необходимо самостоятельно выполнить задание к лабораторной работе с учетом полученного у преподавателя варианта.

***Вариант 1.*** Применение метода конечных автоматов для моделирования работы умного электро-чайника. Чайник может быть выключен или находиться в состоянии нагрева или поддержания температуры.

**Основная часть:**

1. Из формулировки задания определим ***входные сигналы*** моделируемой системы, воздействующие на работу чайника:

* Включить
* Температура достигнута
* Охлаждение (вода остыла ниже заданной температуры)
* Выключить

1. Определим ***внутренние состояния***, в которых может находиться чайник:

* Горит оранжевая подсветка (вода нагревается)
* Горит синяя подсветка (чайник поддерживает температуру воды)
* Подсветка не горит (чайник выключен)

1. Составим таблицу переходов:

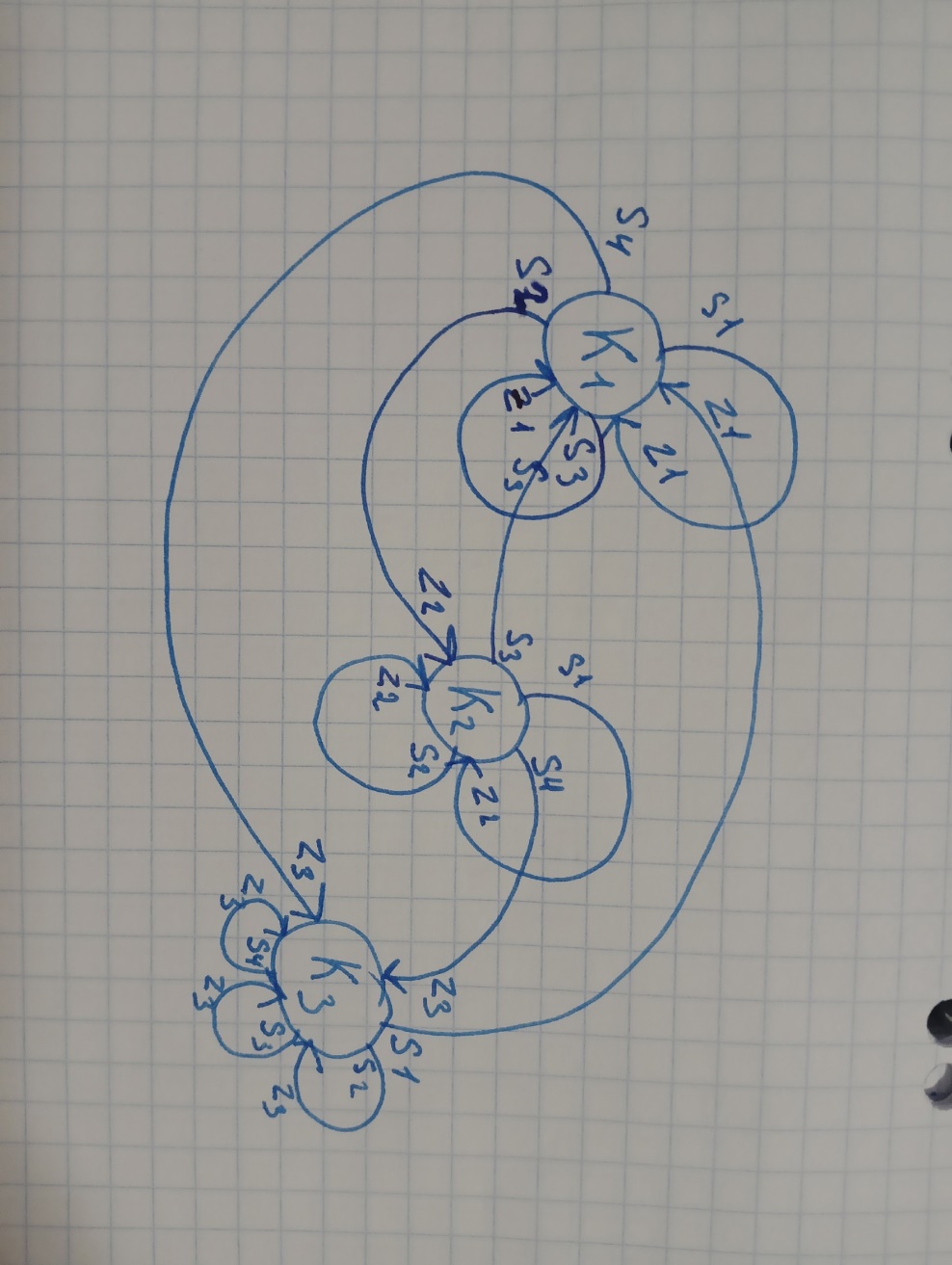
Определим ряд выходных сигналов, т.е. что будет делать собака в определенном состоянии, при наличии определенных входных воздействий:

|  |  |  |  |
| --- | --- | --- | --- |
| **Входные сигналы** | **Состояния** | | |
| Нагрев | Поддержание температуры | Выключен |
| Включить | Горит оранжевая подсветка, остается в состоянии нагрев | Горит синяя подсветка, остается в состоянии поддержания температуры | Включается оранжевая подсветка, переход в состояние нагрев |
| Температура достигнута | Горит синяя подсветка, переход в состояние поддержание температуры | Горит синяя подсветка, остается в состоянии поддержания температуры | Подсветка не горит, остается в состоянии выключен |
| Охлаждение (вода остыла ниже заданной отметки) | Горит оранжевая подсветка, остается в состоянии нагрева | Горит оранжевая подсветка, переход в состояние нагрев | Подсветка не горит, остается в состоянии выключен |
| Выключить | Подсветка не горит, переход в состояние выключен | Подсветка не горит, переход в состояние выключен | Подсветка не горит, остается в состоянии выключен |

1. Для построения графа сосредоточим все данные в общей таблице:

|  |  |  |
| --- | --- | --- |
| Входные сигналы | Состояния | Выходные сигналы |
| S1 – Включить | K1 – Нагрев | Z1 – Горит оранжевая подсветка |
| S2 – Температура достигнута | K2 – Поддержание температуры | Z2 – Горит синяя подсветка |
| S3 – Охлаждение (вода остыла ниже заданной отметки) | K3 – Выключен | Z3 – Подсветка не горит |
| S4 – Выключить |  |
|  |  |

В нашей системе у чайника три состояния, значит, граф будет иметь три вершины. Изобразим граф переходов (Мура), у начала дуги будем указывать входной сигнал, а на конце дуги – выходной сигнал:



**Заключение:**

В лабораторной работе мы изучили принцип работы конечного автомата. Описали автомат при помощи графов и таблиц. Написали компьютерную программу на языке Python3.12 (Приложение 1), наглядно показывающую работу конечного автомата. Научились составлять и описывать конечные автоматы.

Выполнив лабораторную работу, мы изучили основы теории конечного автомата, а также реализовали на практике модель простейшего конечного автомата на примере «Поведения собаки».

Модель включает в себя:

* Табличный способ записи работы алгоритма конечного автомата;
* Графический способ записи работы алгоритма конечного автомата;

Реализация программного кода на языке Python3.12.

# Приложение 1

states = ["Выключен", "Нагрев", "Поддержание температуры"]  
current\_state = states[0] *# Начальное состояние "Выключен"  
  
# Переходы (индексы состояний)*transitions = [  
 *# Переходы для состояния "Выключен"* {"Включить": 1},  
  
 *# Переходы для состояния "Нагрев"* {"Температура достигнута": 2, "Охлаждение": 1, "Выключить": 0},  
  
 *# Переходы для состояния "Поддержание температуры"* {"Охлаждение": 1, "Выключить": 0}  
]  
  
*# Выводы (индикаторы для каждого состояния)*outputs = [  
 "Подсветка не горит", *# Выключен* "Горит оранжевая подсветка", *# Нагрев* "Горит синяя подсветка" *# Поддержание температуры*]  
  
  
*# Функция для выполнения перехода*def transition(*signal*, *current\_state*):  
 state\_index = states.index(*current\_state*)  
 if *signal* in transitions[state\_index]:  
 return states[transitions[state\_index][*signal*]]  
 return *current\_state # Если сигнал не найден, состояние не меняется  
  
  
# Функция для получения текущего вывода*def get\_output(*current\_state*):  
 return outputs[states.index(*current\_state*)]  
  
  
*# Бесконечный цикл для приёма входных сигналов*while True:  
 *print*(f"Текущее состояние: {current\_state}")  
 *print*(f"Индикация(выходной сигнал): {get\_output(current\_state)}")  
  
 *# Получение сигнала от пользователя* signal = *input*("Введите сигнал (Включить, Температура достигнута, Охлаждение, Выключить, Выйти): ")  
  
 if signal == "Выйти":  
 break  
 *# Переход в новое состояние* current\_state = transition(signal, current\_state)  
  
 *print*("\n") *# Для визуального разделения шагов*