

# **Отчёт по лабораторной работе 13**

**Задания для самостоятельного выполнения**

Наталья Андреевна Сидорова

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Задание</b>	<b>6</b>
<b>3</b>	<b>Теоретическое введение</b>	<b>7</b>
<b>4</b>	<b>Выполнение лабораторной работы</b>	<b>8</b>
<b>5</b>	<b>Выводы</b>	<b>15</b>
	<b>Список литературы</b>	<b>16</b>

# Список иллюстраций

4.1	Схема сети . . . . .	9
4.2	Достижимость состояний . . . . .	10
4.3	Декларации . . . . .	10
4.4	Сама сеть . . . . .	11
4.5	Моделирование . . . . .	11
4.6	Граф . . . . .	12
4.7	Отчет . . . . .	13
4.8	Характеристики . . . . .	14

## **Список таблиц**

# 1 Цель работы

Реализовать в CPN Tools задание для самостоятельного выполнения[1].

## 2 Задание

1. Используя теоретические методы анализа сетей Петри, провести анализ сети(с помощью построения дерева достижимости). Определить, является ли сеть безопасной, ограниченной, сохраняющей, имеются ли тупики[2].
2. Промоделировать сеть Петри с помощью CPNTools.
3. Вычислить пространство состояний. Сформировать отчёт о пространстве состояний и проанализировать его. Построить граф пространства состояний.

### 3 Теоретическое введение

Схема модели: Заявка (команды программы, операнды) поступает в оперативную память (ОП), затем передается на прибор (центральный процессор, ЦП) для обработки. После этого заявка может равновероятно обратиться к оперативной памяти или к одному из двух внешних запоминающих устройств (В1 и В2). Прежде чем записать информацию на внешний накопитель, необходимо вторично обратиться к центральному процессору, определяющему состояние накопителя и выдающему необходимую управляющую информацию. Накопители (В1 и В2) могут работать в 3-х режимах: 1. В1 — занят, В2 — свободен; 2. В2 — свободен, В1 — занят; 3. В1 — занят, В2 — занят.

## 4 Выполнение лабораторной работы

Сеть Петри моделируемой системы. Множество позиций:

P1 — состояние оперативной памяти (свободна / занята);

P2 — состояние внешнего запоминающего устройства B1 (свободно / занято);

P3 — состояние внешнего запоминающего устройства B2 (свободно / занято);

P4 — работа на ОП и B1 закончена;

P5 — работа на ОП и B2 закончена;

P6 — работа на ОП, B1 и B2 закончена;

Множество переходов:

T1 — ЦП работает только с RAM и B1;

T2 — обрабатываются данные из RAM и с B1 переходят на устройство вывода;

T3 — CPU работает только с RAM и B2;

T4 — обрабатываются данные из RAM и с B2 переходят на устройство вывода;

T5 — CPU работает только с RAM и с B1, B2;

T6 — обрабатываются данные из RAM, B1, B2 и переходят на устройство вывода.

Функционирование сети Петри можно рассматривать как срабатывание переходов, в ходе которого происходит перемещение маркеров по позициям:

1. работа CPU с RAM и B1 отображается запуском перехода T1 (удаление маркеров из P1, P2 и появление в P1, P4), что влечет за собой срабатывание перехода T2, т.е. передачу данных с RAM и B1 на устройство вывода;
2. работа CPU с RAM и B2 отображается запуском перехода T3 (удаление маркеров из P1 и P3 и появление в P1 и P5), что влечет за собой срабатывание перехода T4, т.е. передачу данных с RAM и B2 на устройство вывода;



3. работа CPU с RAM, B1 и B2 отображается запуском перехода T5 (удаление маркеров из P4 и P5 и появление в P6), далее срабатывание перехода T6, и данные из RAM, B1 и B2 передаются на устройство вывода;
4. состояние устройств восстанавливается при срабатывании: RAM — переходов T1 или T2; B1 — переходов T2 или T6; B2 — переходов T4 или T6. (рис. 4.1).

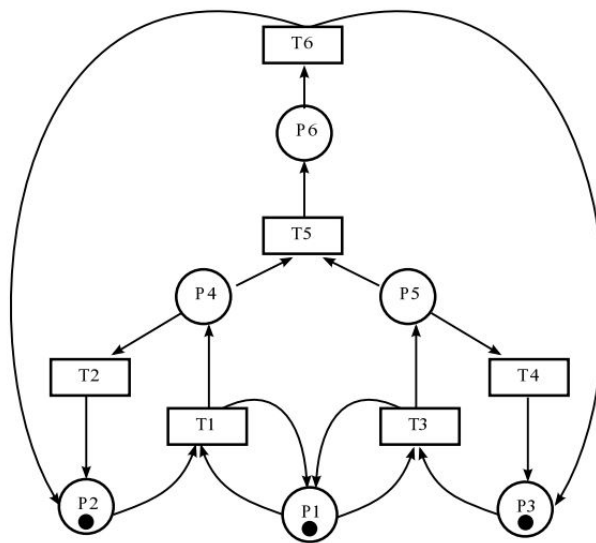


Рис. 4.1: Схема сети

Построила дерево достижимости. Можно увидеть, что наша сеть: 1. безопасна, поскольку в каждой позиции количество фишек не превышает 1; 2. ограничена, так как существует такое целое  $k$ , что число фишек в каждой позиции не может превысить  $k$  (в данном случае  $k=1$ ); 3. сеть не имеет тупиков; 4. сеть не является сохраняющей, так как при переходах  $t5$  и  $t6$  количество фишек меняется. (рис. 4.2).

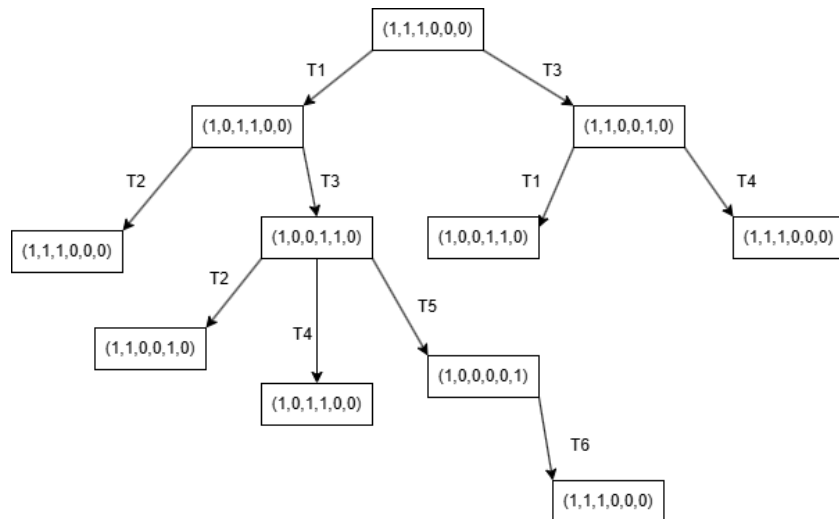


Рис. 4.2: Достижимость состояний

Написала декларации для своей сети Петри (рис. 4.3).

```

▼ Declarations
  ▼ var b2:B2;
  ▼ var b1:B1;
  ▼ var ram:RAM;
  ▼ colset B1xB2 = product B1*B2;
  ▼ colset B2 = unit with storage2;
  ▼ colset B1 = unit with storage1;
  ▼ colset RAM = unit with mem;
  ▼ Standard declarations
    ▶ colset UNIT
    ▶ colset INT
    ▶ colset BOOL
    ▶ colset STRING

```

Рис. 4.3: Декларации

Готовая сеть Петри (рис. 4.4).

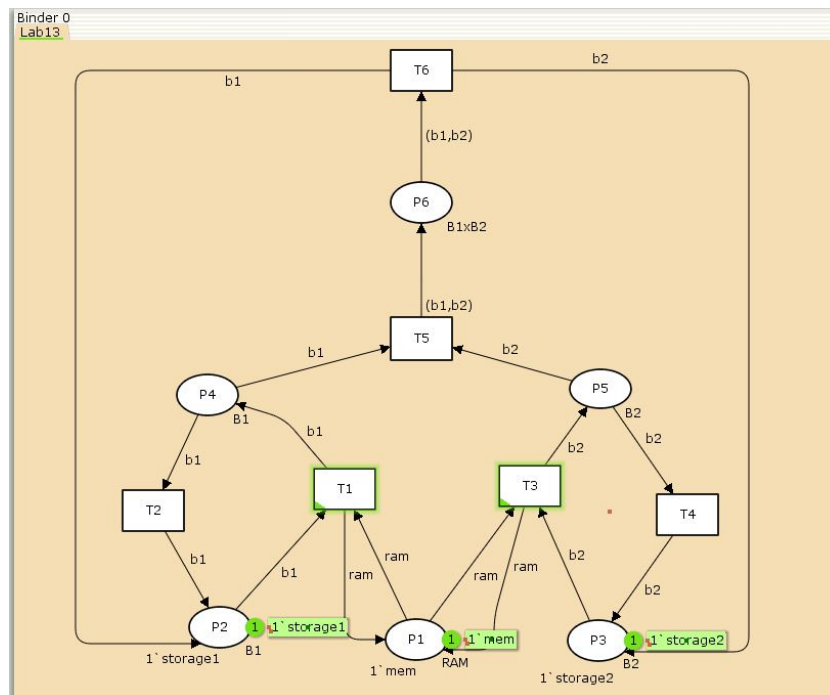


Рис. 4.4: Сама сеть

Процесс моделирования (рис. 4.5).

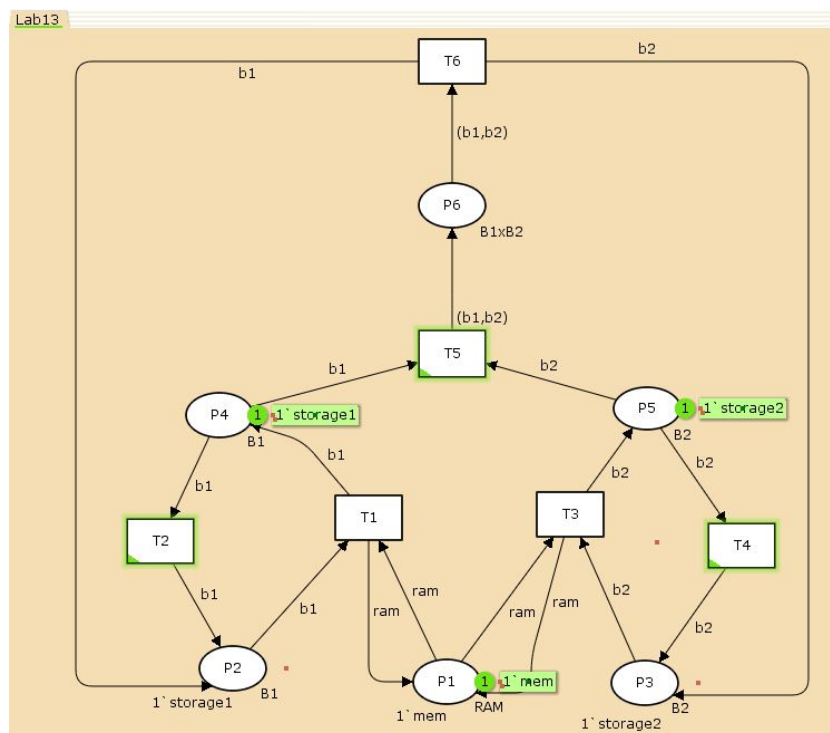


Рис. 4.5: Моделирование

Граф пространства состояний (рис. 4.6).

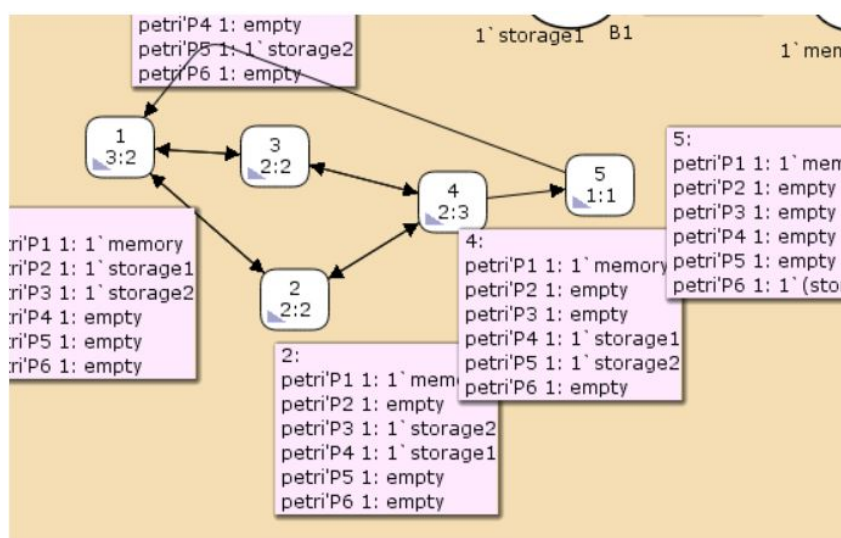


Рис. 4.6: Граф

Отчет о пространстве состояний, в котором можно увидеть: 1. 5 состояний и 10 переходов между ними, strongly connected components (SCC) graph содержит 1 вершину и 0 переходов. 2. Затем указаны границы значений для каждого элемента: состояние P1 всегда заполнено 1 элементом, а остальные содержат максимум 1 элемент, минимум – 0. 3. Также указаны границы в виде мультимножеств. 4. Маркировка home для всех состояний, так как в любую позицию мы можем попасть из любой другой маркировки. 5. Маркировка dead равная None, так как нет состояний, из которых переходов быть не может. 6. В конце указано, что бесконечно часто могут происходить переходы T1, T2, T3, T4, но не обязательно, также состояние T5 необходимо для того, чтобы система не попадала в тупик, а состояние T6 происходит всегда, если доступно. (рис. 4.7).

Statistics		
-----		
State Space		
Nodes:	5	
Arcs:	10	
Secs:	0	
Status:	Full	
Scc Graph		
Nodes:	1	
Arcs:	0	
Secs:	0	
Boundedness Properties		
-----		
Best Integer Bounds		
	Upper	Lower
petri'P1 1	1	1
petri'P2 1	1	0
petri'P3 1	1	0
petri'P4 1	1	0
petri'P5 1	1	0
petri'P6 1	1	0
Best Upper Multi-set Bounds		
petri'P1 1	1`memory	
petri'P2 1	1`storage1	
petri'P3 1	1`storage2	
petri'P4 1	1`storage1	
petri'P5 1	1`storage2	
petri'P6 1	1`(storage1,storage2)	
Best Lower Multi-set Bounds		
petri'P1 1	1`memory	
petri'P2 1	empty	
petri'P3 1	empty	
petri'P4 1	empty	
petri'P5 1	empty	
petri'P6 1	empty	

Рис. 4.7: Отчет

Вторая часть отчета (рис. 4.8).

Home Properties	
-----	
Home Markings	
All	
Liveness Properties	
-----	
Dead Markings	
None	
Dead Transition Instances	
None	
Live Transition Instances	
All	
Fairness Properties	
-----	
petri'T1 1	No Fairness
petri'T2 1	No Fairness
petri'T3 1	No Fairness
petri'T4 1	No Fairness
petri'T5 1	Just
petri'T6 1	Fair

Рис. 4.8: Характеристики

## 5 Выводы

В результате выполнения данной лабораторной работы я выполнила задание для самостоятельного выполнения, а именно провела анализ сети Петри, построила сеть в CPN Tools, построила граф состояний и провела его анализ.

## Список литературы

1. Королькова А.В., Кулябов Д.С. Лабораторная работа 13. Задание для самостоятельного выполнения передачи данных [Электронный ресурс].
2. Королькова А.В., Кулябов Д.С. Сети Петри. Моделирование в CPN Tools [Электронный ресурс].