

Санкт-Петербургский национальный исследовательский университет информационных технологий, механики и оптики

Кафедра информатики и прикладной математики

Цифровая схемотехника

Лабораторная работа №4

“Цифровые автоматы”

Вариант 7



Старались: **Шкаруба Н.Е.**

Суханова В.А.

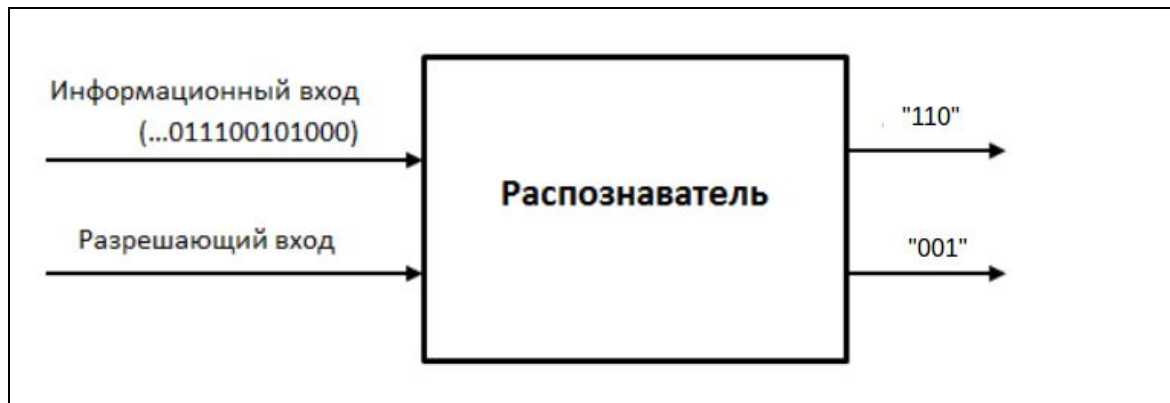
Проверил: **Пинкевич В.Ю.**

Группа **Р3218**

2016г

Конечный автомат:

Распознаватель последовательностей (110, 001) с двоичным кодированием.



Требования к работе:

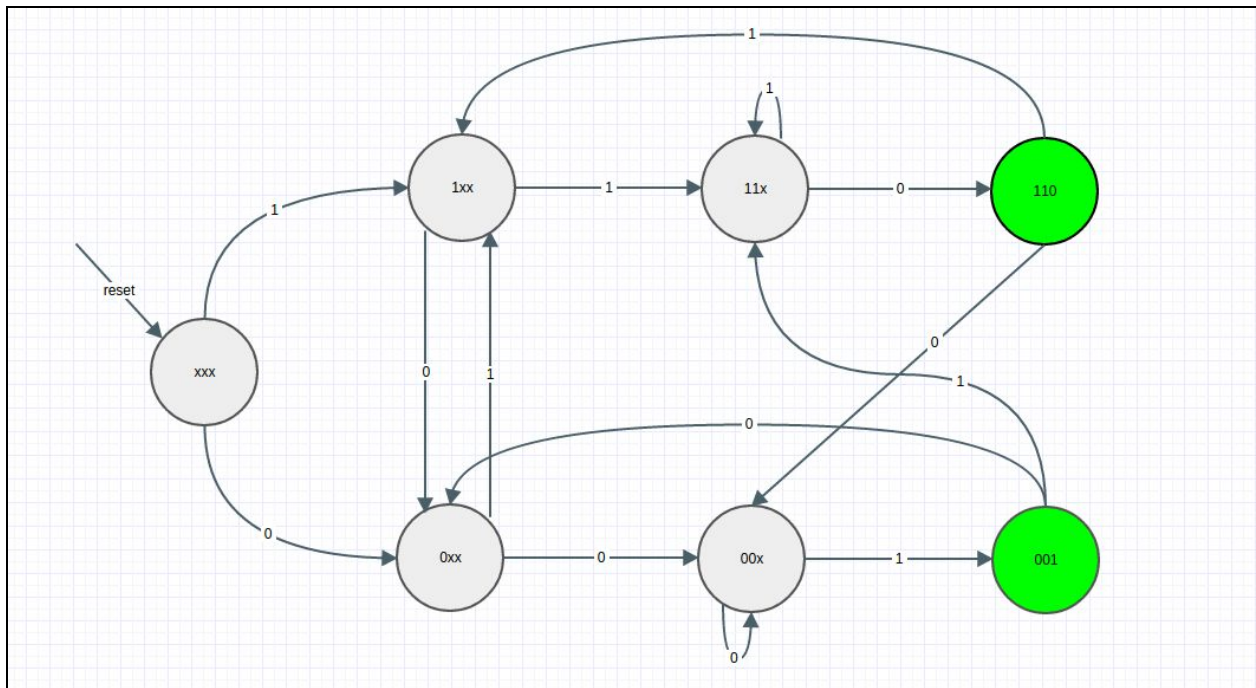
1. Разработать конечный автомат в соответствии с вариантом задания. В отчете привести граф переходов автомата, таблицы кодирования переходов, состояний, входных и выходных сигналов, а также словесное объяснение логики его работы.
2. Разработать реализацию конечного автомата в соответствии с вариантом задания с применением произвольных триггеров и комбинационных элементов (в т.ч. БОЭ). Оформить конечный автомат как иерархический элемент. В отчете привести схему конечного автомата.
3. Провести тестирование конечного автомата. Для этого разработать схему тестирования, которая должна обеспечивать проверку поведения автомата в различных условиях. В отчете привести схему тестирования и временные диаграммы, доказывающие корректную работу автомата

Требования к конечному автомату:

1. Все триггеры в составе автомата должны работать по положительному фронту.
2. Сигнал синхронизации на все триггеры должен подаваться напрямую, без промежуточных комбинационных элементов.
3. Необходимо предусмотреть сигнал сброса всех триггеров в начальное состояние. Для состояния «0» это сигналы RESET или CLEAR, для состояния «1» – PRESET; на триггерах вход этих сигналов может быть обратной полярности.
4. В начале работы схемы тестирования автомата необходимо выполнять сброс в начальное состояние.
5. Переход между состояниями автомата должен осуществляться только по фронту сигнала синхронизации
6. При запрещении работы все выходы должны быть сброшены в 0, а так-же вся предистория должна быть сброшена

Результат:

Граф переходов



Таблицы кодирования

Таблица переходов:

Current State	Input	Next State
"xx"	1	"1xx"
"xx"	0	"0xx"
"1xx"	1	"11x"
"1xx"	0	"0xx"
"0xx"	1	"1xx"
"0xx"	0	"00x"
"11x"	1	"11x"
"11x"	0	"110"
"00x"	1	"001"
"00x"	0	"00x"
"110"	1	"1xx"
"110"	0	"00x"
"001"	1	"11x"
"001"	0	"0xx"

Кодирование состояний:

State	Encoding
"xx"	000
"1xx"	001
"0xx"	010
"11x"	011
"00x"	100
"110"	101

"001"	110
-------	-----

Кодирование выходов:

Output	Encoding
"110"	10
"001"	01
-	00

Таблица переходов с двоичным кодированием:

Current State	Inputs	Next State
000	1	001
000	0	010
001	1	011
001	0	010
010	1	001
010	0	100
011	1	011
011	0	101
100	1	110
100	0	100
101	1	001
101	0	100
110	1	011
110	0	010

Таблица выходов:

Current State	Outputs
000	00
001	00
010	00
011	00
100	00
101	10
110	01

Формулы, вычисляющие текущее состояние

$$\begin{aligned}S_0' &= \neg S_2 \neg S_1 \neg S_0 T_A \vee \neg S_2 \neg S_1 S_0 T_A \vee \neg S_2 S_1 \neg S_0 T_A \vee \neg S_2 S_1 S_0 T_A \vee \neg S_2 S_1 S_0 \neg T_A \vee S_2 \neg S_1 S_0 T_A \vee S_2 S_1 \neg S_0 T_A \\S_1' &= \neg S_2 \neg S_1 \neg S_0 \neg T_A \vee \neg S_2 \neg S_1 S_0 T_A \vee \neg S_2 \neg S_1 S_0 \neg T_A \vee \neg S_2 S_1 S_0 T_A \vee \neg S_2 S_1 \neg S_0 T_A \vee S_2 \neg S_1 S_0 T_A \vee S_2 S_1 \neg S_0 \neg T_A \\S_2' &= \neg S_2 S_1 \neg S_0 \neg T_A \vee \neg S_2 S_1 S_0 \neg T_A \vee S_2 \neg S_1 \neg S_0 T_A \vee S_2 \neg S_1 S_0 \neg T_A \vee S_2 S_1 S_0 \neg T_A\end{aligned}$$

Словесное объяснение логики

Кружки на диаграмме переходов обозначают состояния, а дуги со стрелками между ними – переходы между этими состояниями. Переходы осуществляются по переднему фронту тактового импульса. Стрелка, обозначенная как Сброс, указывает на переход извне в состояние S_0 , отражая то, что система перейдет в это состояние сразу после сброса, независимо от того, в каком она была состоянии до этого. Если присутствует несколько стрелок, выходящих из некоторого состояния, то эти стрелки подписывают, чтобы показать, какой входной сигнал вызвал этот переход. Наш граф переходов имеет две ветки, т.к. конечный автомат должен распознавать две последовательности. Верхняя ветка “подозревает” последовательность “110”, а нижняя - “001”.

Например, система находится в состоянии xxx. Система перейдет в состояние 1xx, если на информационный вход подана 1. Когда система попадает в состояние 110 или 001, то на выходах появляются единички.

Схема конечного автомата

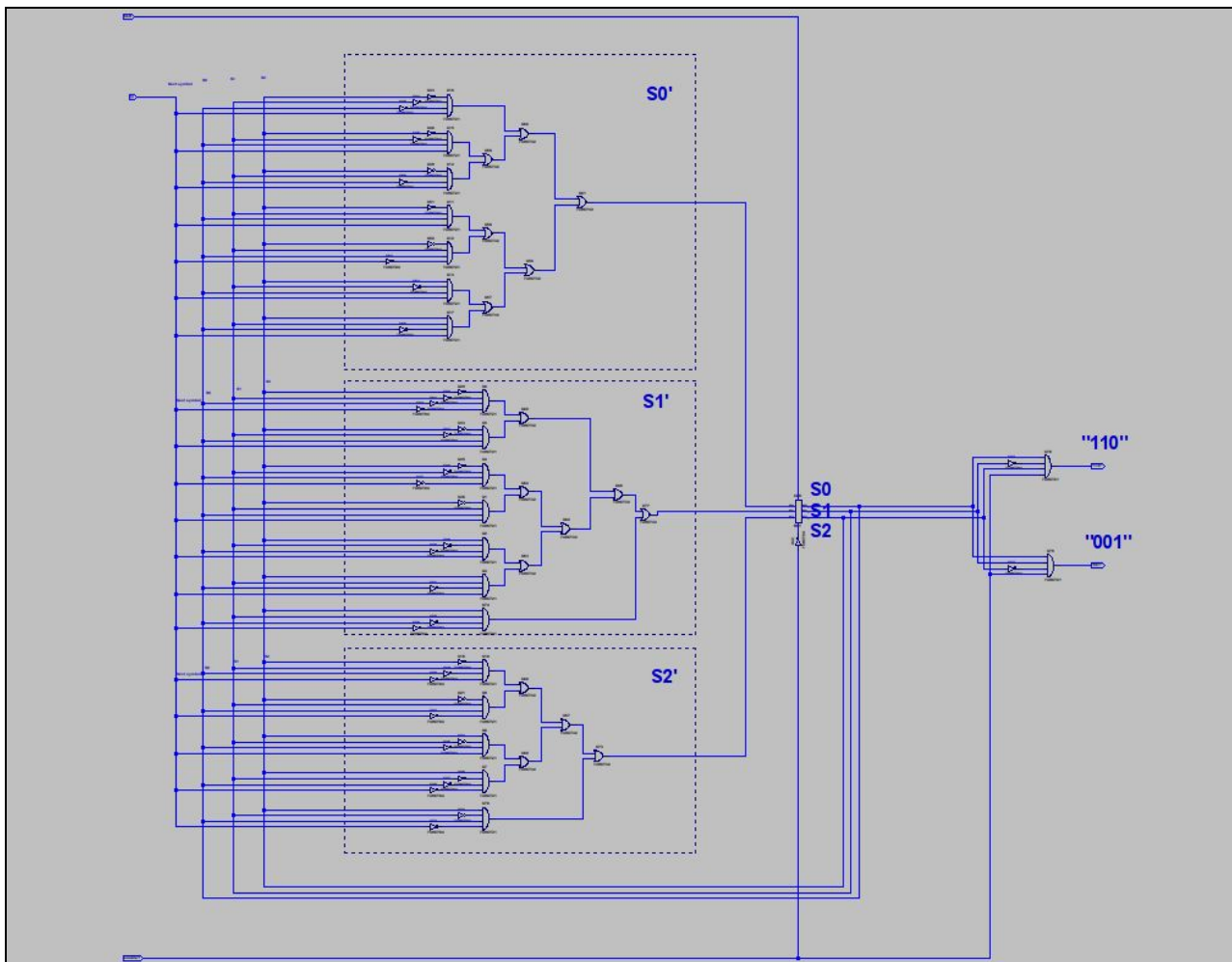


Схема хранителя состояний

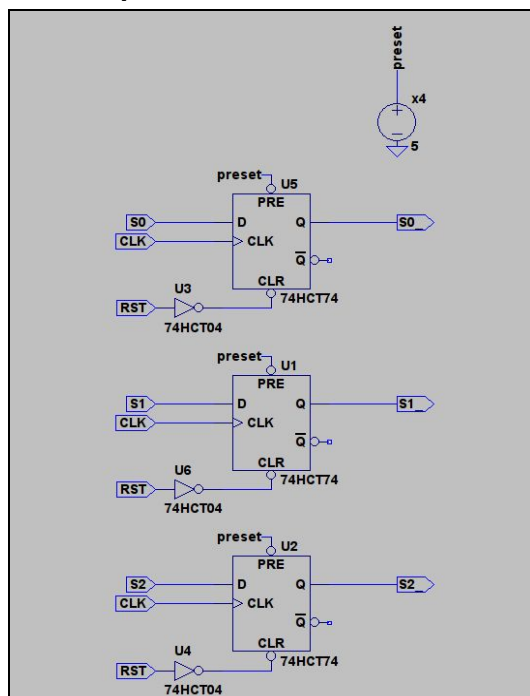
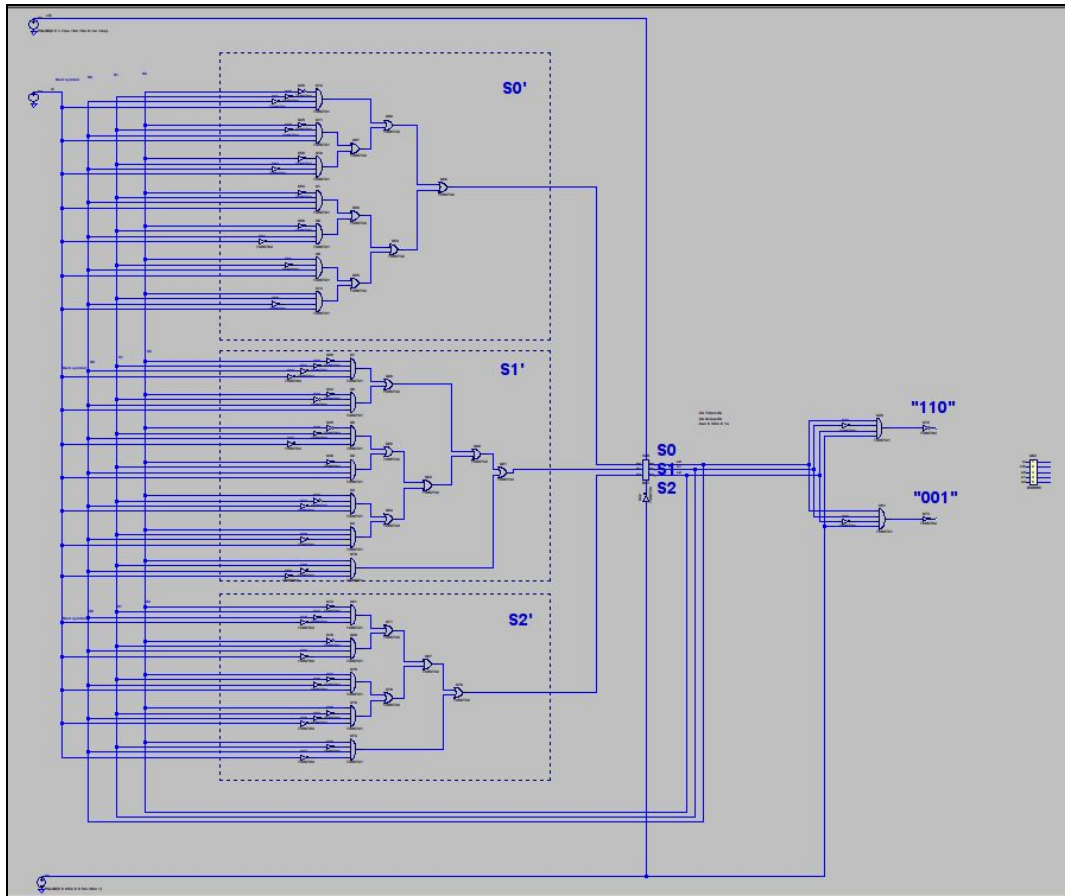


Схема тестирования состояний



Временные диаграммы состояний

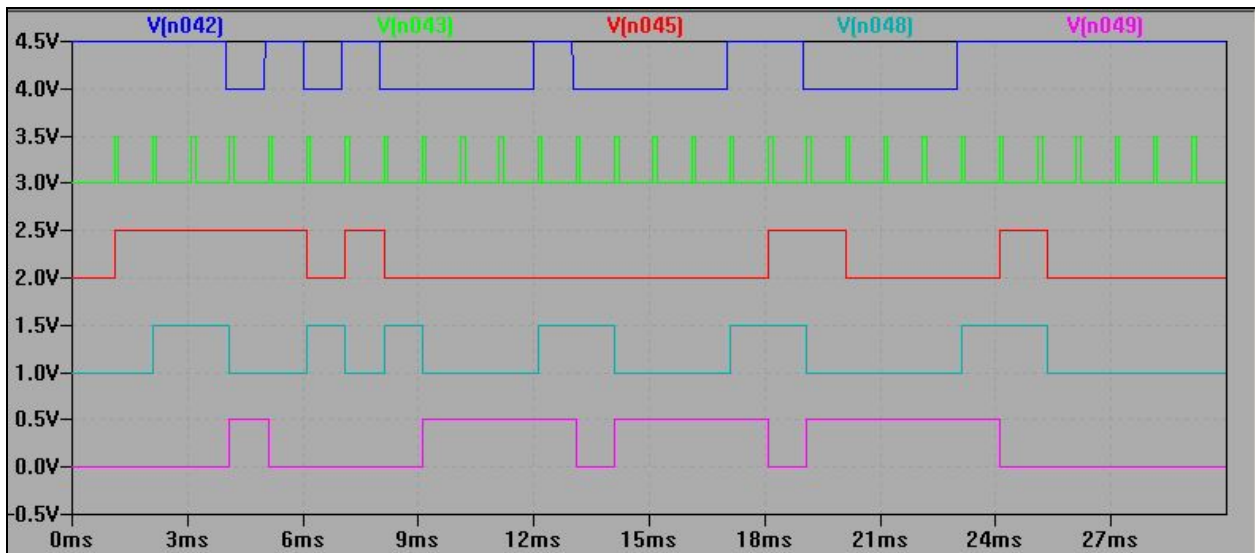
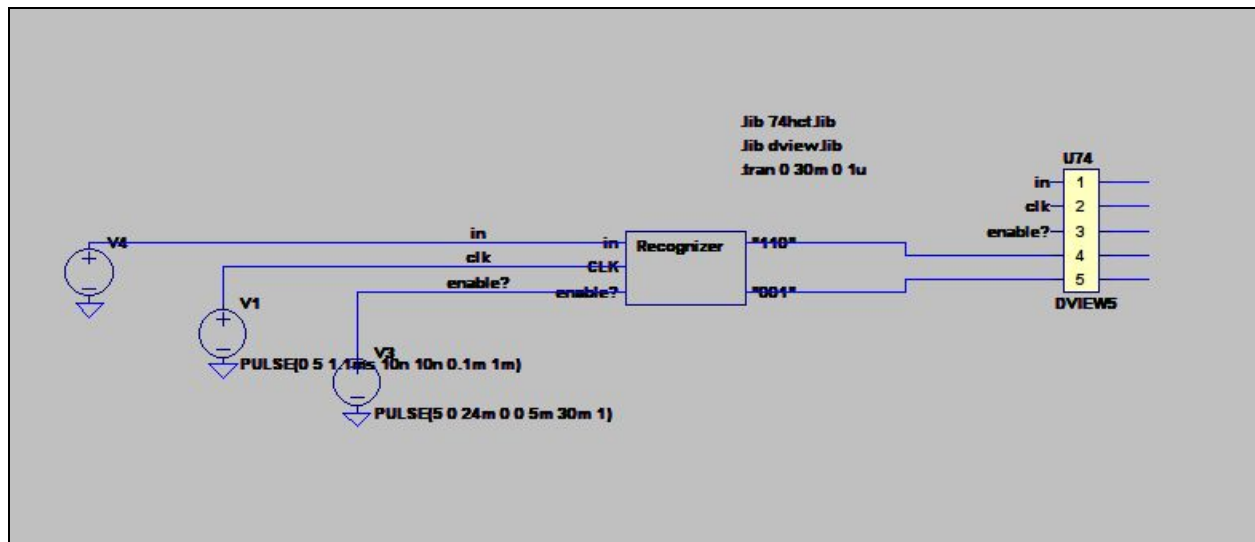


Схема тестирования выходов



Временные диаграммы выходов

