

Санкт-Петербургский государственный университет

**ПУШКО Федор Александрович**

**Выпускная квалификационная работа**

**Программное моделирование и тестовый контроль  
цифровых радиоэлектронных устройств со сложной  
логикой работы**

Уровень образования:

Направление *03.04.01 «Прикладные математика и физика»*

Основная образовательная программа *ВМ.5521.2017 «Математические  
и информационные технологии»*

Научный руководитель:

доктор физ.-мат. наук,

профессор

Овсянников Д. А.

Рецензент:

Ведущий инженер АО

«Научно-исследовательский  
институт точной механики»,

Кандидат технических наук

Антонов В. В.

Санкт-Петербург

2019

## Оглавление

1. Введение .....	3
2. Постановка задачи .....	5
3. История развития схемотехники .....	6
4. Современные методы тестирования радиоэлектронных устройств .....	9
5. Разработка моделей функционирования компонентов объекта контроля.....	12
5.1. 1533ЛА2 .....	13
5.2. 556РТ7А .....	16
5.3. 1533ТМ2.....	19
5.4. 1533ИР23 .....	22
5.5. КМ1802ВР4 .....	25
6. Программное моделирование и написание теста сложного радиоэлектронного устройства Субблок Н8.03.18.05 .....	31
7. Выводы .....	36
8. Заключение.....	37
9. Литература.....	38

## 1. Введение

В связи со стремительным развитием технологий, современный человек уже не способен представить свою жизнь без различных технических устройств, которые он использует в повседневной жизни. Корректная работа многих приборов, будь то холодильник или компьютер, тесно пересекается с электроникой. Помимо сказанного, наряду с совершенствованием технических устройств, функциональная электроника проникает практически во все научные отрасли. Благодаря этому, в обозримом будущем мы сможем стать свидетелями того, как наборы функциональных микрокомпонентов будут управлять искусственными органами и таким образом решат проблему донорства. Также невозможно представить современную науку без развития электроники еще и потому, что любые научные исследования требуют громоздких и трудоемких расчетов. Отдельно хочется упомянуть применение электроники в космической отрасли, где необходимо не только смоделировать сами элементы, но и сделать их применимыми в условиях открытого космоса и на поверхности других небесных и космических тел.

Именно большой практической значимостью может быть объяснено столь бурное развитие системотехники. Наряду с усложнением современных цифровых устройств появилась острая необходимость в новом подходе к контролю и анализу каждого элемента, позволяющему применять комплексный подход к организации процессов технической диагностики. Для этого создаются и применяются системы автоматизированного проектирования тестов, развиваются и внедряются новые комплексы тестового контроля. Одной из таких систем является разработка СПбГУ «SimTest» [6, 7, 11, 13].

В данной работе рассмотрены основные принципы разработки программных моделей в «Quartus II», а также процесс создания тестов для

отдельных микросхем и сложного цифрового радиоэлектронного устройства, с помощью системы «SimTest».

## 2. Постановка задачи

Основной задачей данной работы является создание модели радиоэлектронного устройства со сложной логикой работы Субблок Н8.03.18.05, а также создание тестовой программы для него, целью которой будет контроль работоспособности устройства. В состав субблок Н8.03.18.05, входят 70 элементов, представленных 17 различными типами микросхем, которые содержат между собой 204 внутренние связи. Таким образом, задачи данной работы выглядят следующим образом:

- Создать программную модель каждого элемента, входящего в состав радиоэлектронного устройства, на языке описания аппаратных средств в среде разработки ALTERA QUARTUS II.
- Создания модели радиоэлектронного устройства со сложной логикой работы Субблок Н8.03.18.05.
- Смоделировать тест для проверки работоспособности объекта контроля в САПР «SimTest».

Стоит обратить внимание, что итоговая тестовая последовательность должна иметь максимально возможное покрытие радиоэлектронного устройства, чтобы проверка компонентов и внутренних связей была полноценной.

### 3.История развития схемотехники

Началом развития схемотехники принято считать 1906 год, когда Д.А.Флеминг первым сумел получить ламповый диод или другими словами электронный прибор с односторонней проводимостью. Всего спустя год был поставлен опыт, в котором в диод между катодом и анодом вводился третий электрод («управляющая сетка»), что позволило получить первый триод. Дальнейшее развитие аналоговой и цифровой электроизмерительной техники было тесно связано с прогрессом в области электроники.

Следующий этап относится к рубежу 40-х и 50-х годов и характеризуется стремительным развитием дискретных полупроводниковых приборов, которому поспособствовали работы в области физики твердого тела, а также создание теории полупроводников.

В 1947 году инженеры Д.Бардин, В.Бреттейн и В.Шокли изобрели полупроводниковый трехэлектродный усилитель — биполярный транзистор, за создание которого годом позднее были удостоены Нобелевской премии по физике. Благодаря их исследованиям начался новый виток развития электроники, получивший название «кремниевый век». Первые образцы транзисторов, способных усиливать и генерировать электрические сигналы, в промышленных масштабах были предложены в 1948 г. Способность транзисторов работать при низком уровне напряжения и токе главным образом повлияла на размеры всех элементов, используемых в схемах, что в последствие открыло возможность создавать более миниатюрную радиоэлектронную аппаратуру.

Помимо изобретения новых типов приборов, разработки велись в направлении совершенствования технологий их изготовления. 50-е годы знаменуются открытиями в области физики твердого тела, а также переходом

к квантовой электронике, благодаря которой был дан толчок бурному развитию лазерной техники.

Очередным этапом развития электроники принято считать начало 60-ых годов прошлого века, отличительными чертами которого были разработка и практическое применение интегральных схем, позволивших в едином технологическом цикле производить активные и пассивные элементы функциональных устройств, значительно уменьшив их размеры одновременно увеличив функциональность. В дальнейшем это привело к появлению больших интегральных схем (БИС), которые были способны вмещать до нескольких тысяч элементов в одном кристалле. Позже это подтолкнуло к созданию совершенно новых цифровых устройств — микропроцессоров, позволяющих объединить обработку информации и управление по заданному алгоритму, а также рассчитанных на совместную работу с устройствами памяти.

В настоящее время электроника включает в себя несколько направлений исследований, а именно:

- Вакуумную электронику, специализирующуюся на изучении электронных ламп, например, диодов и триодов; ЭВП СВЧ, фотоэлектронных приборов, таких как фотоэлементы и фотоэлектронных умножителей; рентгеновских трубок, а также газоразрядных приборов (мощных преобразователей тока, источников света, индикаторов).
- Твердотельную электронику, занимающуюся разработкой и изготовлением различных видов полупроводниковых диодов (выпрямительных, смесительных, параметрических, стабилитронов), усилительных и генераторных диодов (туннельных, лавинно-пролетных, диодов Ганна), транзисторов (биполярных и униполярных), тиристоров, оптоэлектронных

приборов, интегральных схем, диэлектрической электроникой, а также криоэлектроникой, а также быстродействующих вычислительных и запоминающих устройств.

- Квантовая электроника, включающая в себя направления, связанные с созданием и использованием лазеров и мазеров: приборов квантовой электроники, систем оптической многоканальной связи, усилителей микроволн, оптоэлектроникой, голографией и многими другими.

Современный этап развития характеризуется появлением новых перспективных областей и направлений электроники, таких как наноэлектроника, связанная в первую очередь с разработкой функциональных устройств электроники с размерами элементов, не превышающими 100 нм и графеновая электроника, использующая технологию создания полупроводниковых приборов на основе графена.

Со временем стремительное развитие электроники и схемотехники привело к тому, что создаваемые устройства становились сложнее и функциональней, при этом пропорционально уменьшались их физические размеры. Сейчас нет ничего необычного в производстве больших интегральных схем и микропроцессоров, на основе которых создаются сложные и многофункциональные устройства, окружающие нас в современном мире. Серийное производство электронных приборов любой сложности несет за собой определенные риски появления брака(микросхем с дефектом), в основном вследствие человеческого фактора. Всё это приводит к необходимости применять автоматизацию контрольно-диагностических работ на каждом этапе производства.

## 4.Современные методы тестирования радиоэлектронных устройств

В серийном производстве радиоэлектронных устройств наиболее значимой частью является контроль функциональности и тестирование финальной сборки. В настоящее время наиболее распространенными видами тестирования являются: функциональное тестирование, визуальный контроль, внутрисхемное тестирование и периферийное/граничное сканирование. Рассмотрим отличия, а также плюсы и минусы каждого из них более детально.

Визуальный автоматизированный контроль – этот метод основан на возможностях зрения, другими словами компонент проверяется в видимом излучении. Для контроля используются простейшие инструменты, например, лупа. Метод позволяет проверить качество паянного соединения, наличие самого компонента и правильность его расположения. Также используется рентгеновское излучение для проверки мест, недоступных невооруженному взгляду. Из плюсов можно выделить простоту и достаточно высокую скорость проведения, из минусов - ограниченность тестирования только видимой частью схемы, а в случае с рентгеновским излучением, дороговизна и сложность оборудования.

Внутрисхемное тестирование – это технология, позволяющая проводить тестирование как схемы целиком, так и отдельных ее компонентов, с помощью специального оборудования. Из плюсов стоит выделить относительно высокую скорость тестирования, что позволяет применять данный метод в промышленных масштабах. Из минусов стоит отметить необходимость физического контакта щупов и тестируемых компонентов, что, очевидно, вызывает проблемы в тестировании многослойных печатных плат, а также в случае со слишком миниатюрными

элементами [1]. Существуют так называемые «летающие щупы» [2], которые решают данную проблему, но это значительно уменьшает производительность, что недопустимо для массового производства.

Периферийное/граничное сканирование или же JTAG-тестирование – метод, позволяющий выявлять брак еще до стадии функционального тестирования в отличие от предыдущих видов. Данный способ позволяет выявить западания, обрывы и короткие замыкания. В следствие большого распространения стандарта JTAG, который необходим для сканирования данным методом, он становится все более популярным в последнее время. Из минусов можно выделить невысокую скорость тестирования, а также невозможность проверки качества соединения [3-4].

Функциональное тестирование – это тестирование в целях проверки реализуемости функциональных требований, то есть способности устройства в определённых условиях решать необходимые задачи. При правильно построенном тесте, этот метод способен охватить требуемую функциональность устройства в автоматическом режиме, при достаточно высокой скорости тестирования [5-6]. Из минусов стоит отметить высокую сложность разработки тестового программного обеспечения и возможность получить лишь общий результат проверки функционала устройства.

Очевидно, что при массовом производстве необходимо осуществлять контроль качества изделия, учитывая все плюсы и минусы каждого из методов автоматического тестирования, чтобы получить максимальное покрытие каждого устройства за минимально возможное время.

На данный момент наиболее перспективным считается функциональное тестирование, так как обладает рядом преимуществ перед другими видами тестирования радиоэлектронных устройств:

- Имеет значительное преимущество в скорости тестирования, относительно периферийного сканирования, а также не требует поддержки интерфейса JTAG.

- Не требует сложных, дорогостоящих устройств, в отличие от систем автоматизированного визуального контроля, в частности методов, использующих рентгеновское излучение.

- Не зависит от размеров и количества слоев тестируемого устройства, в отличие от внутрисхемного тестирования.

Разработанная в СПбГУ САПР «SimTest» позволяет автоматизировать процесс создания тестовой программы, необходимой для функционального тестирования цифрового устройства [6, 7, 11, 13].

## 5. Разработка моделей функционирования компонентов объекта контроля

Построение теста для любого элемента можно условно разбить на следующие этапы:

- Получение информации об элементе, а именно количество выводов, логика его работы и особенности построения.
- Описание логики работы элемента и особенности его функционирования на языке программирования Verilog.
- Моделирование блок-схемы элемента в среде разработки «Quartus II».
- Формирование его HDL-файла, для дальнейшей работы в «SimTest».
- Написание тестовой последовательности и контроль правильности функционирования элемента.

Далее рассмотрим некоторые элементы, входящие в состав устройства Субблок Н8.03.18.05.

## 5.1.1533ЛА2

Микросхема представляет собой достаточно примитивный элемент на восемь входов и один выход, выполняющий элементарную Булеву функцию  $y = \sim(d1 \& d2 \& d3 \& d4 \& d5 \& d6 \& d7 \& d8)$  [8]. Для моделирования и дальнейшего тестирования элемента, нам необходимы определенные данные, а именно:

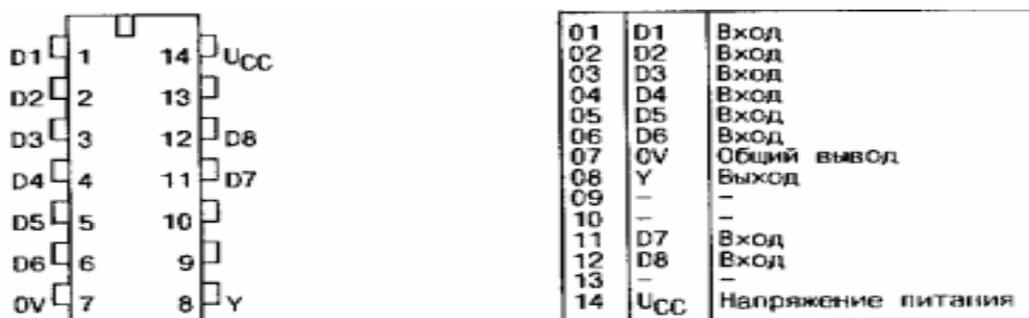


Рисунок 1 Расположение и назначение выводов 1533ЛА2

Данные рисунки несут в себе информацию о количестве входов и выходов, которые содержит микросхема и какое назначение имеет каждый из них.

D1	D2	D3	D4	D5	D6	D7	D8	Y
H	H	H	H	H	H	H	H	L
L	X	X	X	X	X	X	X	H
X	L	X	X	X	X	X	X	H
X	X	L	X	X	X	X	X	H
X	X	X	L	X	X	X	X	H
X	X	X	X	L	X	X	X	H
X	X	X	X	X	L	X	X	H
X	X	X	X	X	X	L	X	H
X	X	X	X	X	X	X	L	H

Рисунок 2 Таблица истинности 1533ЛА2

Данная таблица показывает алгоритм работы микросхемы, который пригодится для написания кода, с которым будет продолжаться дальнейшая работа.

Фрагмент кода 1533ЛА2 в среде разработки «Quartus II»:

```
module ic_1533la2(d1, d2, d3, d4, d5, d6, d7, d8, q);  
input d1, d2, d3, d4, d5, d6, d7, d8;  
output q;  
assign q =~(d1&d2&d3&d4&d5&d6&d7&d8);  
endmodule
```

После написания и компиляции полученного кода, он преобразуется в блок-схему, которая используется для формирования HDL-файла, используемый для написания теста:

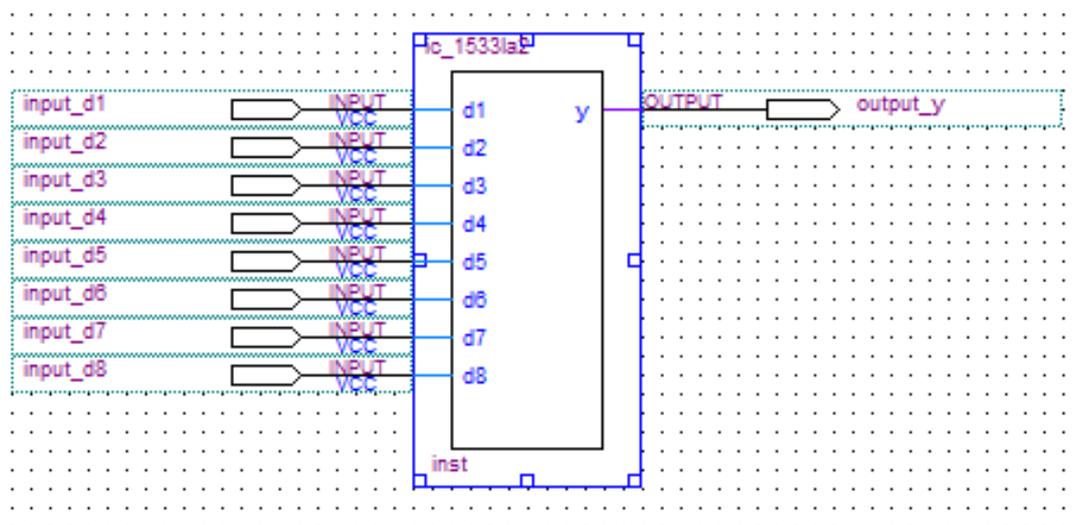
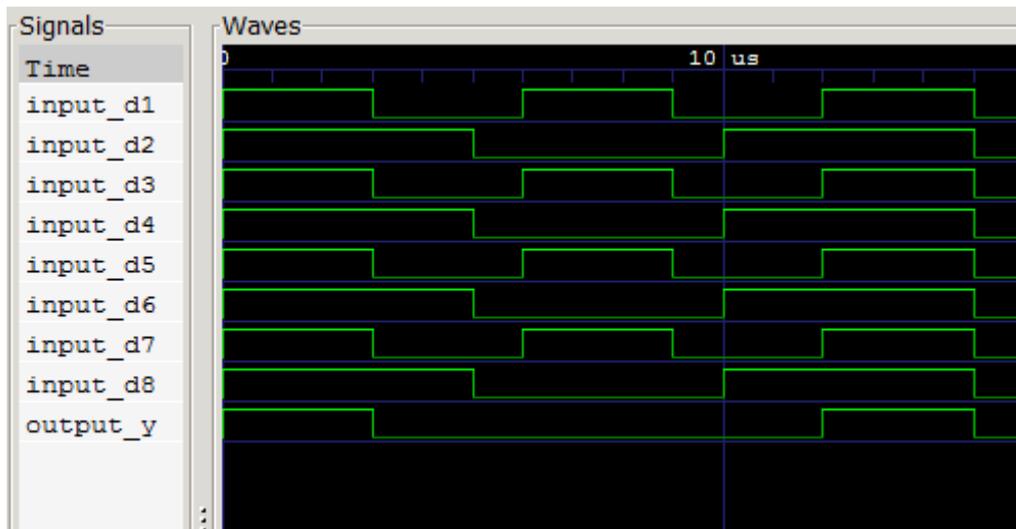


Рисунок 3 Блок-схема 1533ЛА2

Последним шагом будет написание теста и сравнение полученных результатов с таблицей истинности, как видно на рисунке алгоритм работает правильно:



*Рисунок 4 Результаты моделирования в "SimTest"*

## 5.2.556PT7A

Микросхема 556PT7A представляет собой программируемое постоянное запоминающее устройство (ПЗУ) с организацией 2048 8-разрядных слов. В исходном состоянии в незапрограммированных схемах в ячейках память записан логический «0». Запись логической единицы производится путем пережигания нихромовых перемычек. В программной реализации элемента считывание данных из \*.hex файла, содержащего прошивку микросхемы, ведется при наличии уровня низкого напряжения на входе CS1 и наличии уровней высокого напряжения на входах CS2, CS3 [11].

Условное графическое обозначение представлено на рисунке 5, назначение выводов в таблице 1:

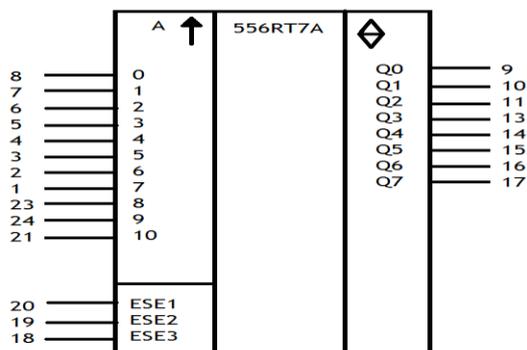


Рисунок 5 Расположение выводов 556PT7A

Вывод	Обозначение	Тип вывода	Функциональное назначение выводов
1-8, 21, 23, 24	A0-A10	Входы	Адресные входы
18-20	ESE1-ESE3	Входы	Входы разрешения выборки
9-17	Q0-Q7	Выходы	Информационные выходы
22	Ucc	-	Напряжение питания

25	GND	-	Общий
----	-----	---	-------

Таблица 1 назначение выводов 556PT7A

Ниже, как и в прошлом примере представлен фрагмент кода 556PT7A:

```

module ic_556rt7a (CS1, CS2, CS3, A0, A1, A2, A3, A4, A5, A6, A7, A8, A9, A10, Q0, Q1, Q2, Q3,
Q4, Q5, Q6, Q7, U);

input wire CS1, CS2, CS3;
input wire A0, A1, A2, A3, A4, A5, A6, A7, A8, A9, A10;
inout tri Q0, Q1, Q2, Q3, Q4, Q5, Q6, Q7;
input U;
wire CS;
wire [ 10 : 0 ] adr;
wire [ 7 : 0 ] Q;
reg [ 7 : 0 ] mem [ 0 : 2047 ];
assign { Q7, Q6, Q5, Q4, Q3, Q2, Q1, Q0 } = Q;
assign adr = { A10, A9, A8, A7, A6, A5, A4, A3, A2, A1, A0 };
assign CS = !CS1 & CS2 & CS3;
assign Q = !CS ? {8'bz} : mem[adr];
initial begin
$readmemh("mem.data", mem);
end
endmodule

```

Как видно из кода, при инициализации происходит чтение и запись данных из файла в массив памяти для дальнейшей корректной работы.

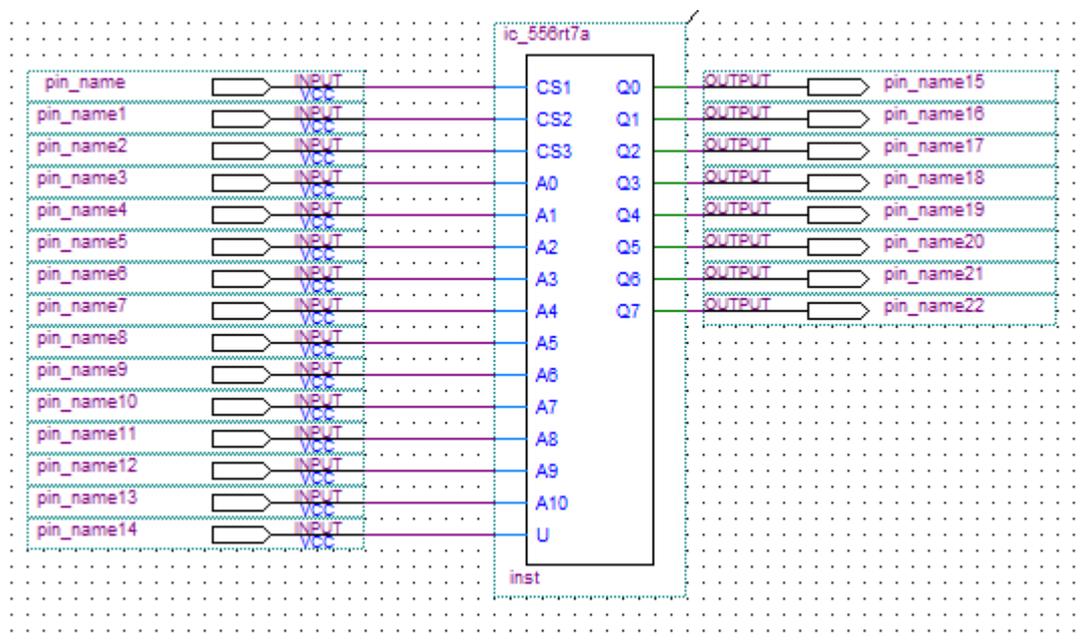


Рисунок 6 Блок-схема 556PT7A

Создаем тест в «SimTest» и получаем его результат:

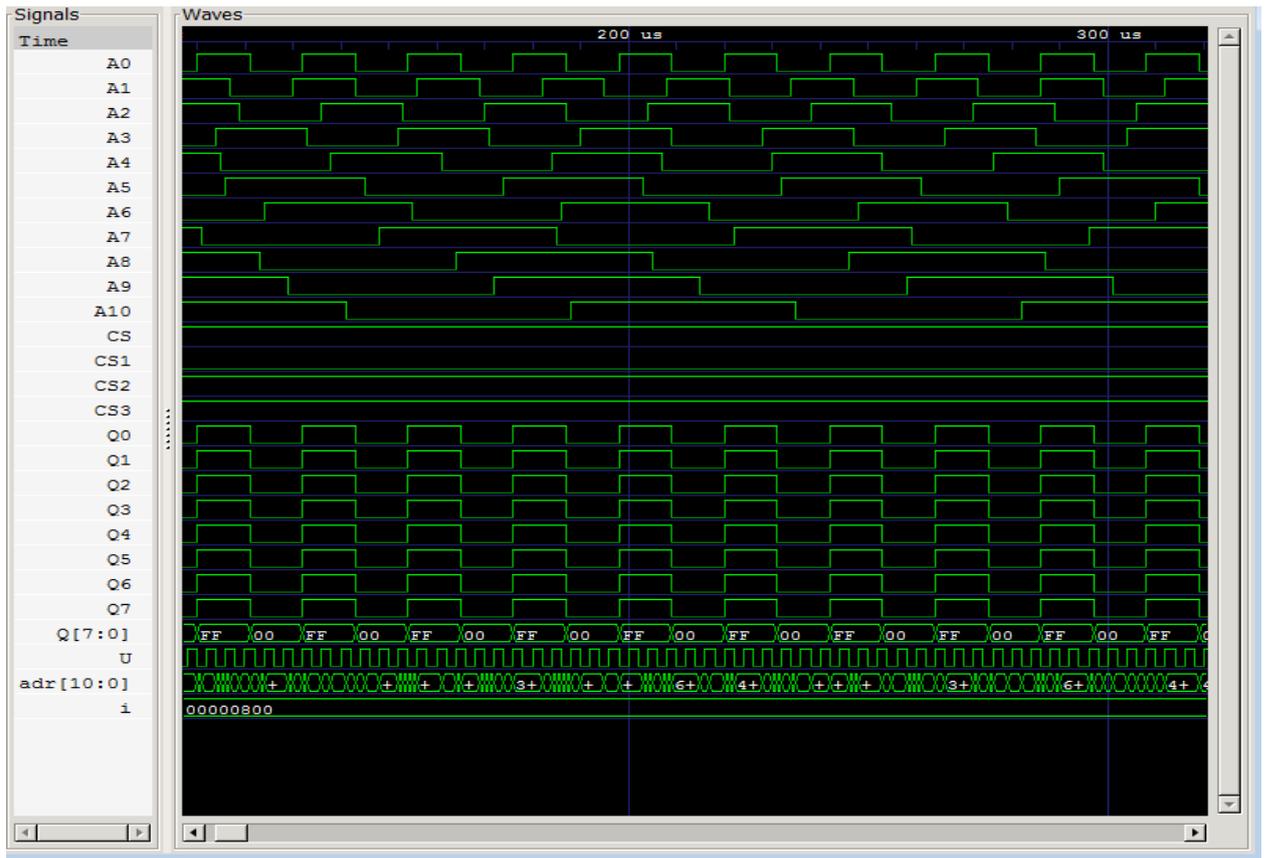


Рисунок 7 Результаты моделирования в "SimTest"

Как видно из теста, при наличии уровня низкого напряжения на входе CS1 и наличии уровней высокого напряжения на входах CS2, CS3, происходит считывание данных и их последующий вывод.

### 5.3.1533TM2

Микросхема представляет собой два независимых D-триггера, на четыре входа и два выхода каждый. Триггер - это элемент, имеющий два устойчивых состояния, который используется для запоминания одного разряда двоичного кода. [10].

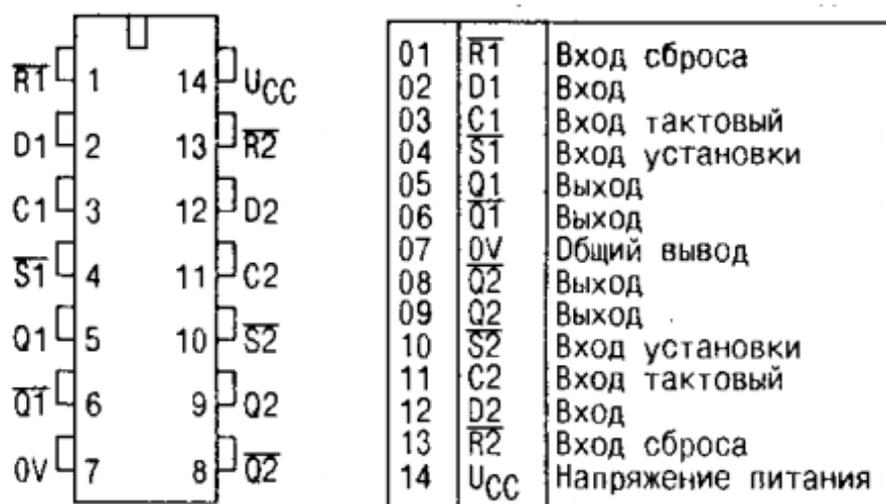


Рисунок 8 Расположение и назначение выводов 1533TM2

Как и в прошлых примерах на рисунке 10 показано, сколько входов и выходов имеет микросхема, а также их назначение.

Входы				Выходы	
S	$\overline{R}$	C	D	Q	$\overline{Q}$
L	H	X	X	H	L
H	L	X	X	L	H
L	L	X	X	H*	H*
H	H	┌	H	H	L
H	H	└	H	L	H
H	H	L	L	L	$\overline{Q_0}$
H	H	L	X	$Q_0$	$\overline{Q_0}$

Рисунок 9 Таблица истинности 1533TM2

Опираясь на описание элемента и данные рисунки, производится написание кода алгоритма работы схемы.

Фрагмент кода 1533ТМ2:

```
module ic_1533tm2(r,s,c,d,q,q_);
input r,s,c, d;
output q, q_;
reg q, q_;
always @(posedge s)
begin
if (r==1'b0)
begin
q=1'b0;
q_=1'b1;
end
end
/*
always @(negedge s)
begin
if (r==1'b1)
begin
q=1'b1; q_=1'b0;
end
else
begin
q=1'b1; q_=1'b1;
end
end
always @(posedge r)
begin
if (s==1'b0)
begin
q=1'b1; q_=1'b0;
end
end
always @(negedge r)
begin
if (s==1'b1)
begin
q=1'b0; q_=1'b1;
end
else
begin
q=1'b1; q_=1'b1;
end
end
always @(posedge c)
begin
```

```

if ((r==1'b1)&&(s==1'b1))
begin
q=d; _=~q;
end
end
*/
Endmodule

```

Стоит обратить внимание на то, что один из блоков «always» закомментирован. Это связано с тем, что «Quartus II» не воспринимает несколько блоков always срабатывающих одновременно, выдавая ошибку компиляции, поэтому мы вынуждены прибегать в этой хитрости, а уже при создании теста в «SimTest», открываем все блоки обратно.

Далее идет преобразование кода в блок-схему, а также подведение входов и выходов, и их именование:

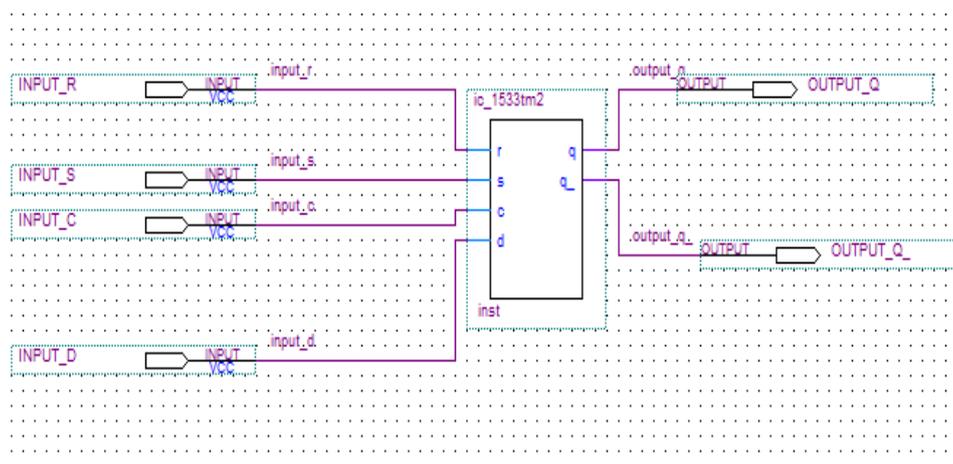


Рисунок 10 Блок-схема 1533TM2

Заходим в «SimTest», создаем новый тест и сверяем результат с таблицей истинности:

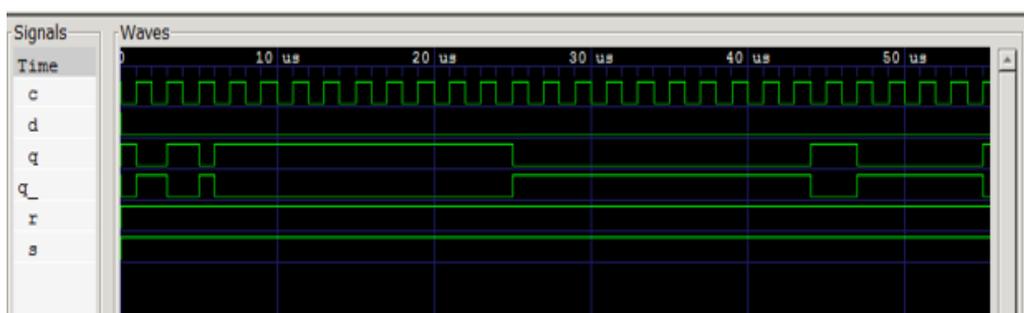


Рисунок 11 Результат теста 1533TM2

## 5.4.1533ИР23

Элемент представляет собой восьмиразрядный регистр на триггерах D-типа с тремя состояниями на выходе. Регистр представляет собой упорядоченный набор D-триггеров, число  $n$  которых соответствует числу разрядов в слове. Данная микросхема может быть использована в качестве регистра, буферного регистра, регистра ввода-вывода, магистрального передатчика и других [11].

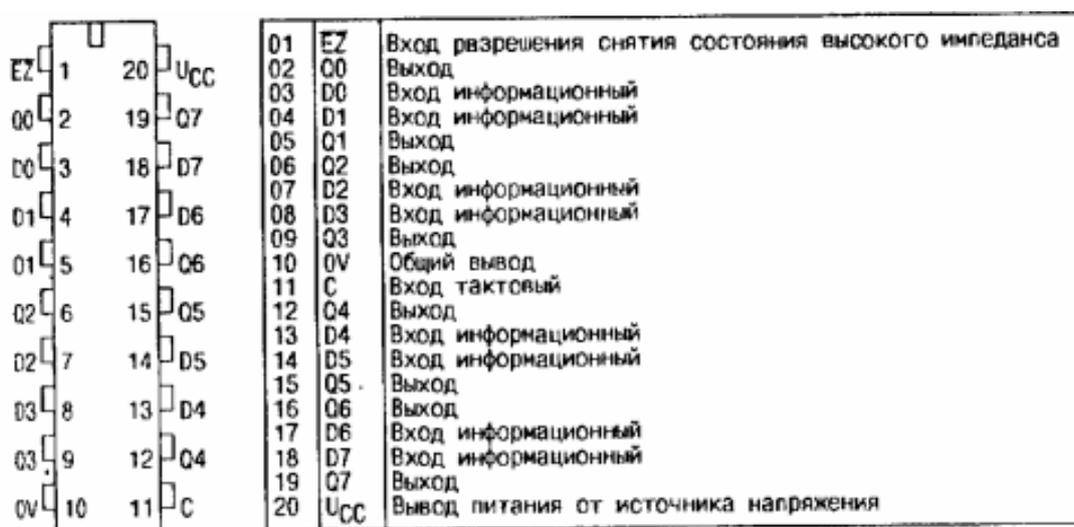


Рисунок 12 Расположение и назначение выводов 1533ИР23

Таким образом наша схема имеет 10 входов и 8 выходов.

Входы			Выход
$\overline{EZ}$	C	D	Q
L	┌	H	H
L	┌	L	L
L	L	X	Q <sub>0</sub>
H	X	X	Z

Рисунок 13 Таблица истинности 1533ИР23

Опираясь на описание элемента и данные рисунки, мы можем перейти к написанию самого кода.

Фрагмент кода 1533ИР23:

```
input ez, c, d0, d1, d2, d3, d4, d5, d6, d7;
output q0, q1, q2, q3, q4, q5, q6, q7;
reg [0:7] d;
always@(posedge c)
begin
d<= {d0, d1, d2, d3, d4, d5, d6, d7};
end
assign {q0, q1, q2, q3, q4, q5, q6, q7} = (ez = 1'b1)? 8'bz: d;
endmodule
```

Опираясь на написанный код, моделируется блок-схема в среде разработки «Quartus II»:

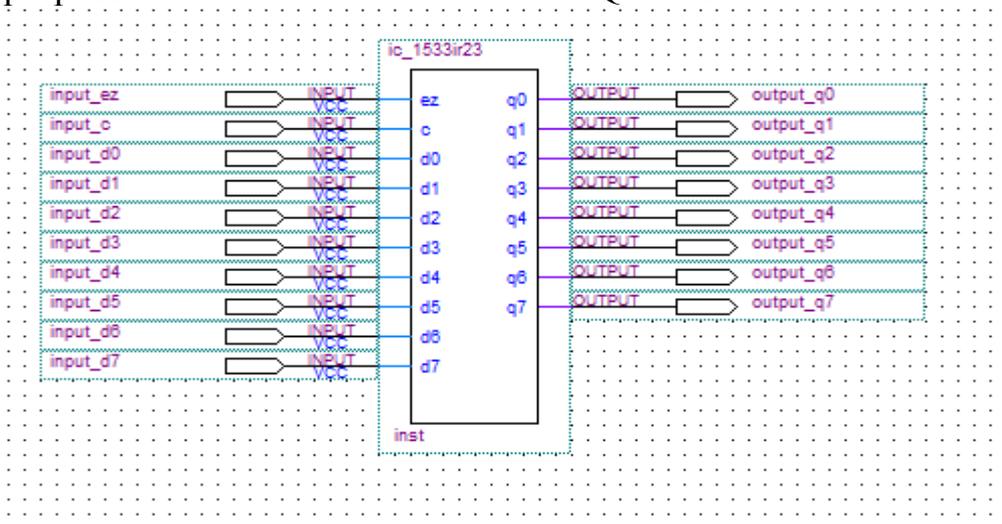
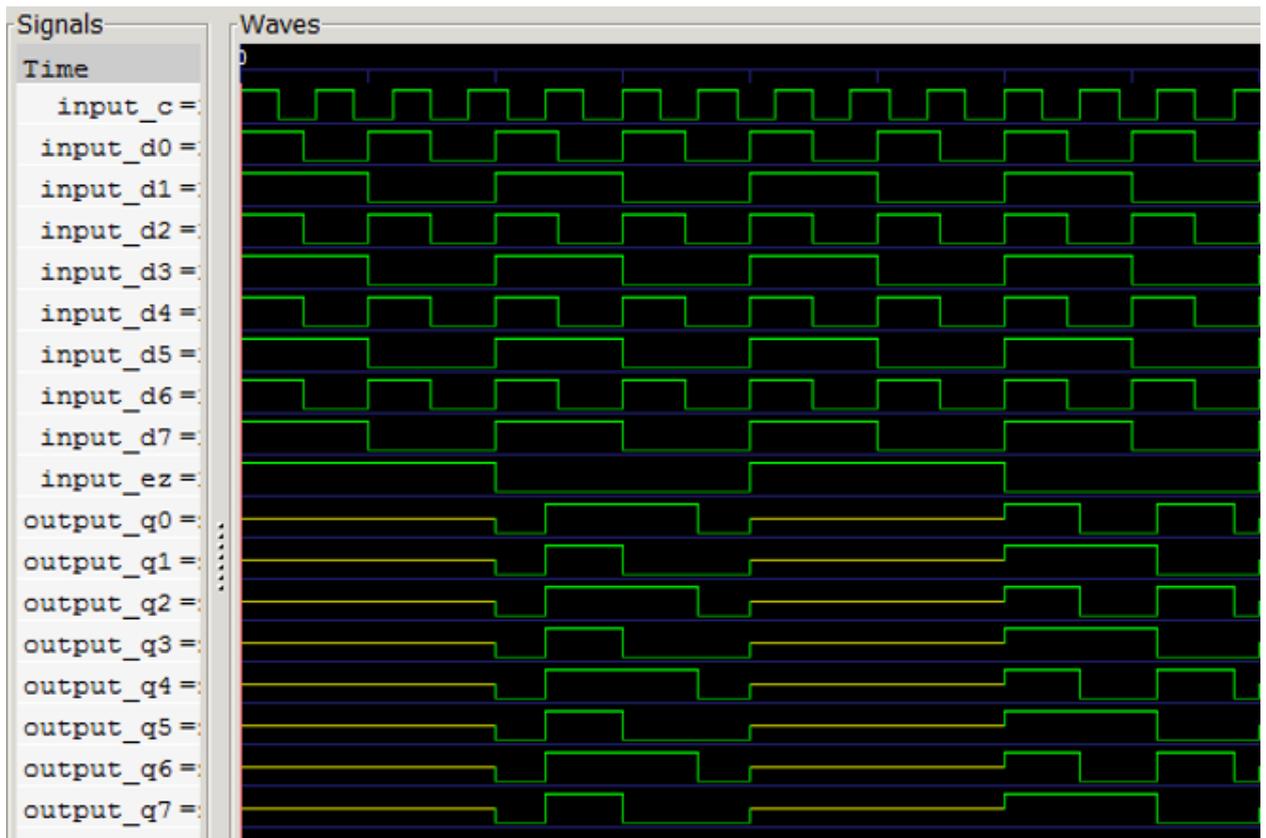


Рисунок 14 Блок-схема 1533ИР23

Финальным шагом будет проектирование тестовой последовательности в «SimTest» для проверки правильности логики работы элемента:



*Рисунок 15 Результаты теста 1533ИР23*

По результатам моделирования убеждаемся, что логика элемента описана верно.

## 5.5.КМ1802ВР4

Микросхема КМ1802ВР4 - параллельный умножитель 12х12 разрядов, предназначена для использования в радиоэлектронной аппаратуре. Каждый из операндов может быть либо кодом (числом без знака), либо числом со знаком. В последнем случае операнд представляется в дополнительном коде. Числа могут быть как целыми, так и меньше 1.

На выходе умножителя вырабатывается произведение двойной точности (24 разряда), которое может быть округлено до 12 разрядов (включая знаковый разряд). При умножении чисел со знаком в дополнительном коде произведение получается в дополнительном код. При действиях над числами со знаком предусмотрена возможность присвоения знака произведения младшей части произведения.

Умножитель может быть использован для построения быстродействующих процессоров цифровой обработки сигналов, реализующих алгоритмы быстрого преобразования Фурье, цифровую фильтрацию и т. п. Возможно применение также в специализированных и универсальных цифровых ЭВМ. [11]

Схема имеет 35 входов и 24 выхода. Условное графическое обозначение и назначение выводов представлено на рисунке 16:

Вывод	Обозначение	Тип вывода	Функциональное назначение выводов
1-8	X7-X0	Входы	Множимое, разряды 7-0, 11-8
61-64	X11-X8	Выходы	Произведение, разряды 0-11, 12-23
9-20	P0-P11		
29-40	P12-P23		
21	TRIL	Вход	Управление входными буферными каскадами младшей части произведения
22	TRIM	Вход	Управление выходными буферными каскадами старшей части произведения
23, 24	GND	-	Общий
25	FT	Вход	Управление «прозрачностью» регистров произведения
26	RS	Вход	Управление сдвигом вправо старшей части произведения
27	CLKL	Вход	Синхронизация регистра младшей части произведения
28	CLKM	Вход	Синхронизация регистра старшей части произведения
41	TCY	Вход	Знак весового коэффициента старшего разряда множителя
42-47	Y11-Y6	Входы	Множитель, разряды 11-6, 5-0
51-56	Y5-Y0		
48-50	Ucc	-	Напряжение питания
57	TCX	Вход	Знак весового коэффициента старшего разряда множимого
58	RND	Вход	Округление
59	CLKY	Вход	Синхронизация регистра множителя
60	CLKX	Вход	Синхронизация регистра множимого

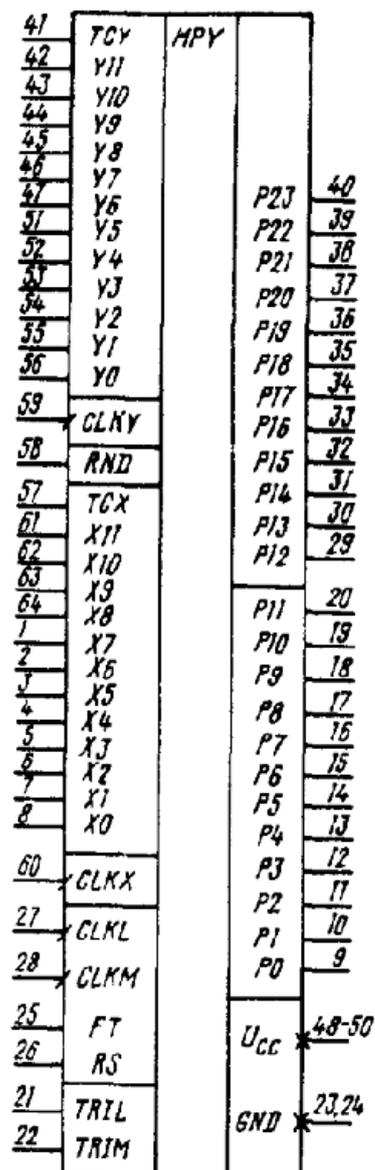


Рисунок 16 Расположение и назначение выводов KM1802BP4.

Структурная схема приведена на рисунке 21:

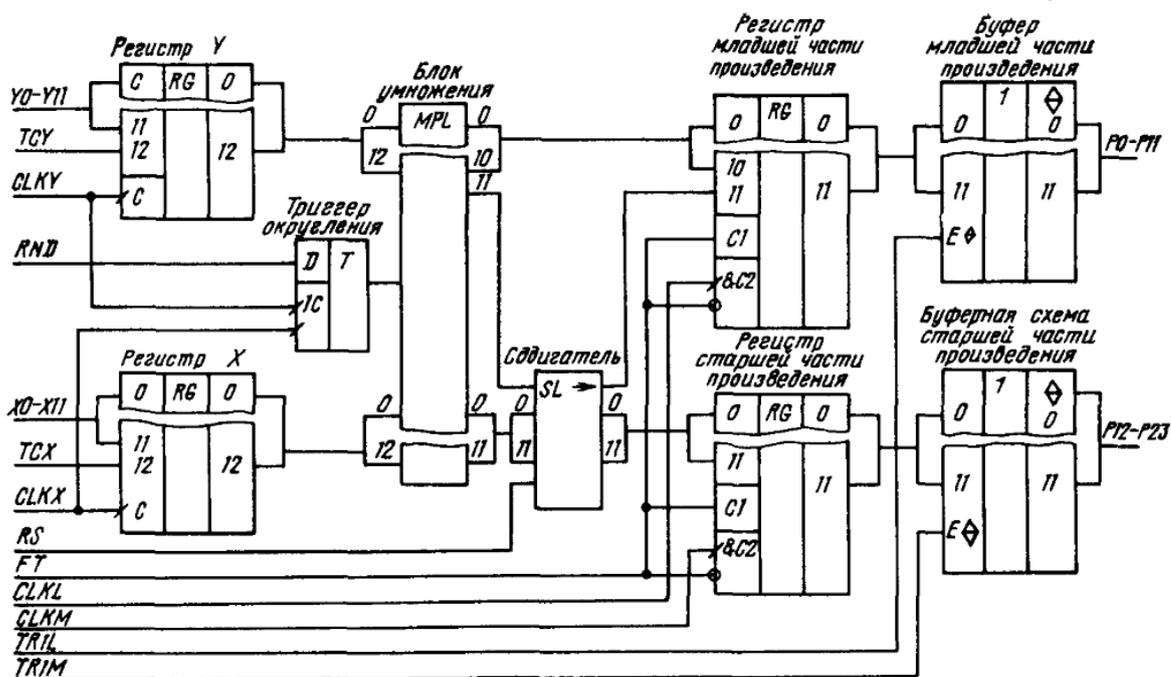


Рисунок 17 Структурная схема KM1802BP4.

Регистры X и Y выполнены на D-триггерах с одним тактирующим входом. Запись в них осуществляется по фронту сигналов CLKX и CLKY. Помимо 12 разрядов сомножителей в регистры X и Y заносятся признаки множимого и множителя TCX и TCY, которые имеют значения 1, если данный сомножитель - число со знаком, и 0, если сомножитель - число без знака. Другими словами, TCX и TCY - знаки весового коэффициента старшего разряда числа: при 1 - минус, при 0 - плюс.

По фронту одного из сигналов CLKY и CLKX в триггер округления записывается сигнал RND, по которому производится округление произведения до 12 разрядов при RND = 1. [11]

Фрагмент кода KM1802BP4, описывающий алгоритм работы:

```
module ic_1802vr4 (
input SIY11, Y11, Y10, Y9, Y8, Y7, Y6, Y5, Y4, Y3, Y2, Y1, Y0, CY,
input SIX11, X11, X10, X9, X8, X7, X6, X5, X4, X3, X2, X1, X0, CX,
input RND, CPM, CPL, xEND, E_, EPL, EPM,
```

```

        output P23, P22, P21, P20, P19, P18, P17, P16, P15, P14, P13, P12, P11, P10, P9, P8, P7,
P6, P5, P4, P3, P2, P1, P0
    );
    wire [11:0] Xn;
    wire [11:0] Yn;
    reg [23:0] R;
    assign Xn = { X11, X10, X9, X8, X7, X6, X5, X4, X3, X2, X1, X0 };
    assign Yn = { Y11, Y10, Y9, Y8, Y7, Y6, Y5, Y4, Y3, Y2, Y1, Y0 };
    assign { P23, P22, P21, P20, P19, P18, P17, P16, P15, P14, P13, P12 } = EPM ? 12'bz : (
xEND ? TR[23:12] : R[23:12] );
    assign { P10, P9, P8, P7, P6, P5, P4, P3, P2, P1, P0 } = EPL ? 11'bz : ( xEND ? TR[10:0] :
R[10:0]);
    reg signed [11:0] DX;
    reg SX;
    reg signed [11:0] DY;
    reg SY;
    reg DRND;
    reg signed [23:0] TR;
    reg signed [23:0] TX;
    reg signed [23:0] TY;
    wire signed [23:0] DX11;
    wire signed [23:0] DY11;
    assign DX11 = !SX ? { DX[11], 11'b0 } : -{ DX[11], 11'b0 };
    assign DY11 = !SY ? { DY[11], 11'b0 } : -{ DY[11], 11'b0 };
    always @(posedge CX) DX <= Xn;
    always @(posedge CX) SX <= SIX11;
    always @(posedge CY) DY <= Yn;
    always @(posedge CY) SY <= SIY11;
    always @(posedge CX, posedge CY) DRND <= RND;
    always @(posedge CPL, posedge CPM, negedge xEND) R <= TR;
    always @( DX, DX11 ) TX <= DX[10:0] + DX11 ;
    always @( DY, DY11 ) TY <= DY[10:0] + DY11 ;
    always @( TX, TY, DRND ) TR = TX * TY + { DRND, 11'b0};
    assign P11_ = !E_ ? P23 : ( xEND ? TR[11] : R[11]);
    assign P11 = EPL ? 1'bz : P11_;
endmodule

```

Полученная после компиляции блок-схема:

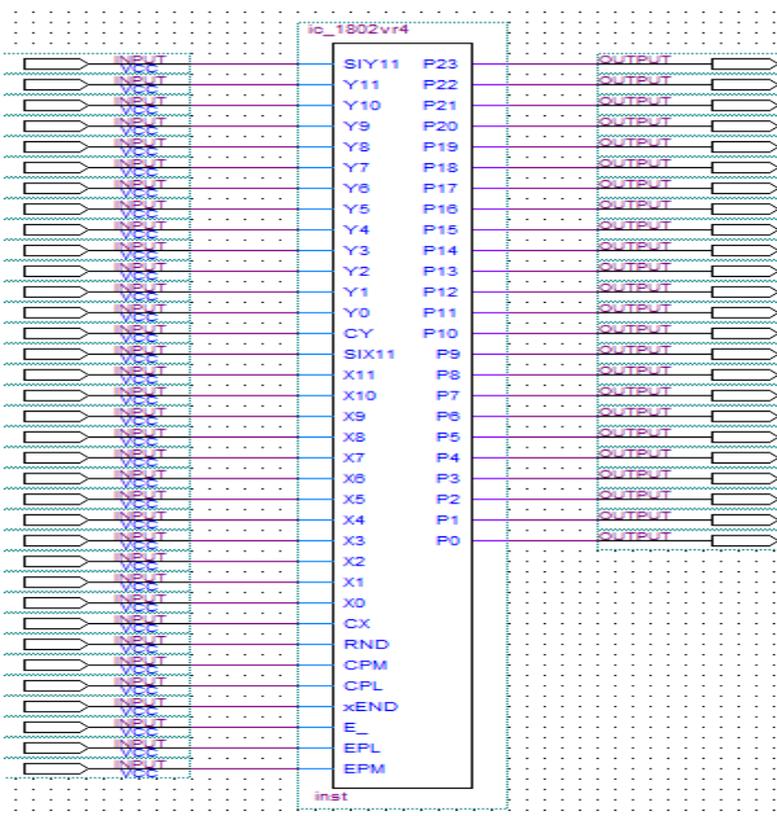


Рисунок 18 Блок-схема KM1802BP4

Как и в предыдущих примерах, на основе блок-схемы пишется тест и проверяется правильность работы модели в «SimTest»:

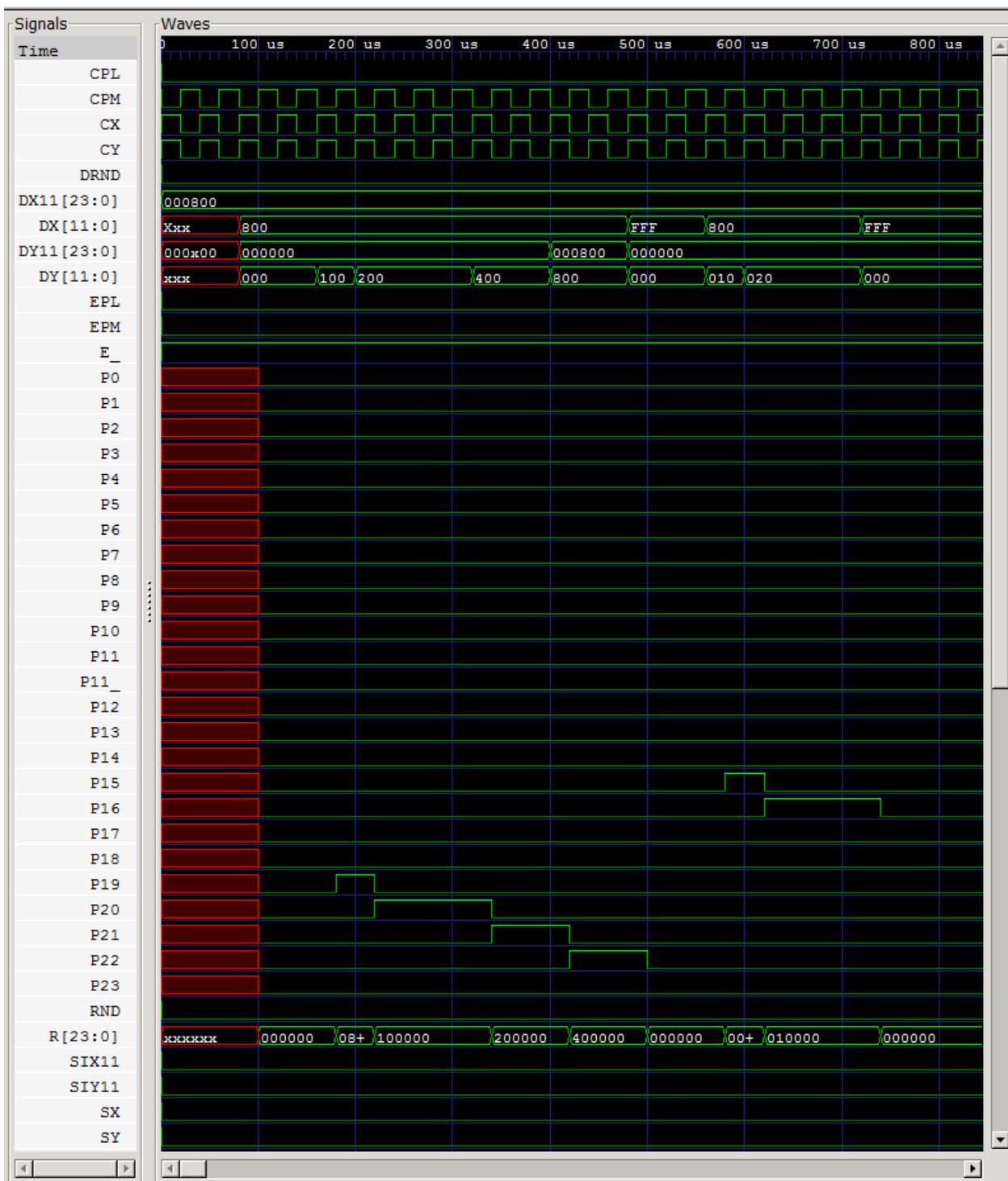


Рисунок 19 Результаты теста KM1802BP4

На рисунке показан результат проверки. Схема реализована правильно.

## 6. Программное моделирование и написание теста сложного радиоэлектронного устройства Субблок Н8.03.18.05

Субблок Н8.03.18.05 состоит из 17 различных элементов, а именно: 1533ИЕ7, 1533КП11, 1533КП13, 1533КП7, 1533ЛИ1, 1533ЛН1, 1533ЛП5, 1533ЛР11, 1533СП1, 1533ТМ2, 1533ТМ9, 1802ВР4, 533ИМ6, 533ИП5, 533ЛА12, 533ЛЛ1, 556РТ7А.

Как упоминалось ранее, для построения программной модели необходимо отдельно создать каждый элемент данной схемы, затем объединить их в общем проекте:

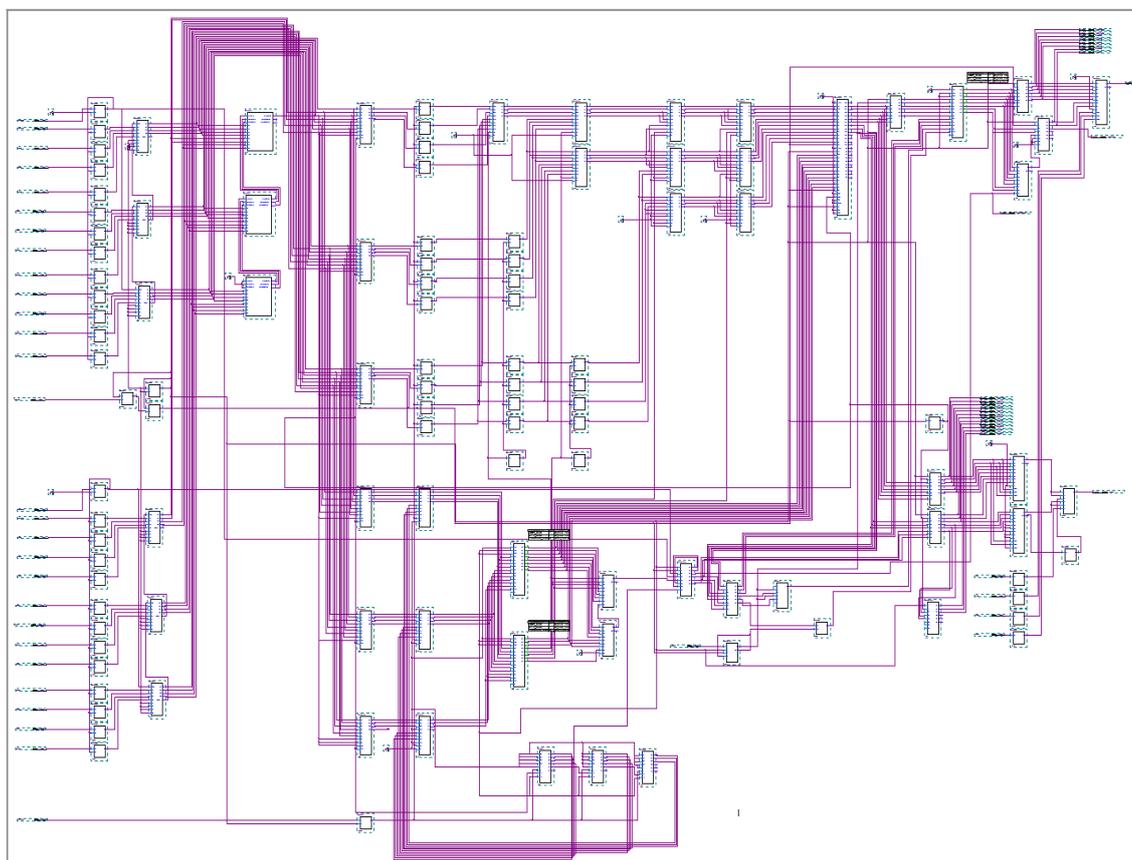


Рисунок 20 блок-схема Субблок Н8.03.18.05

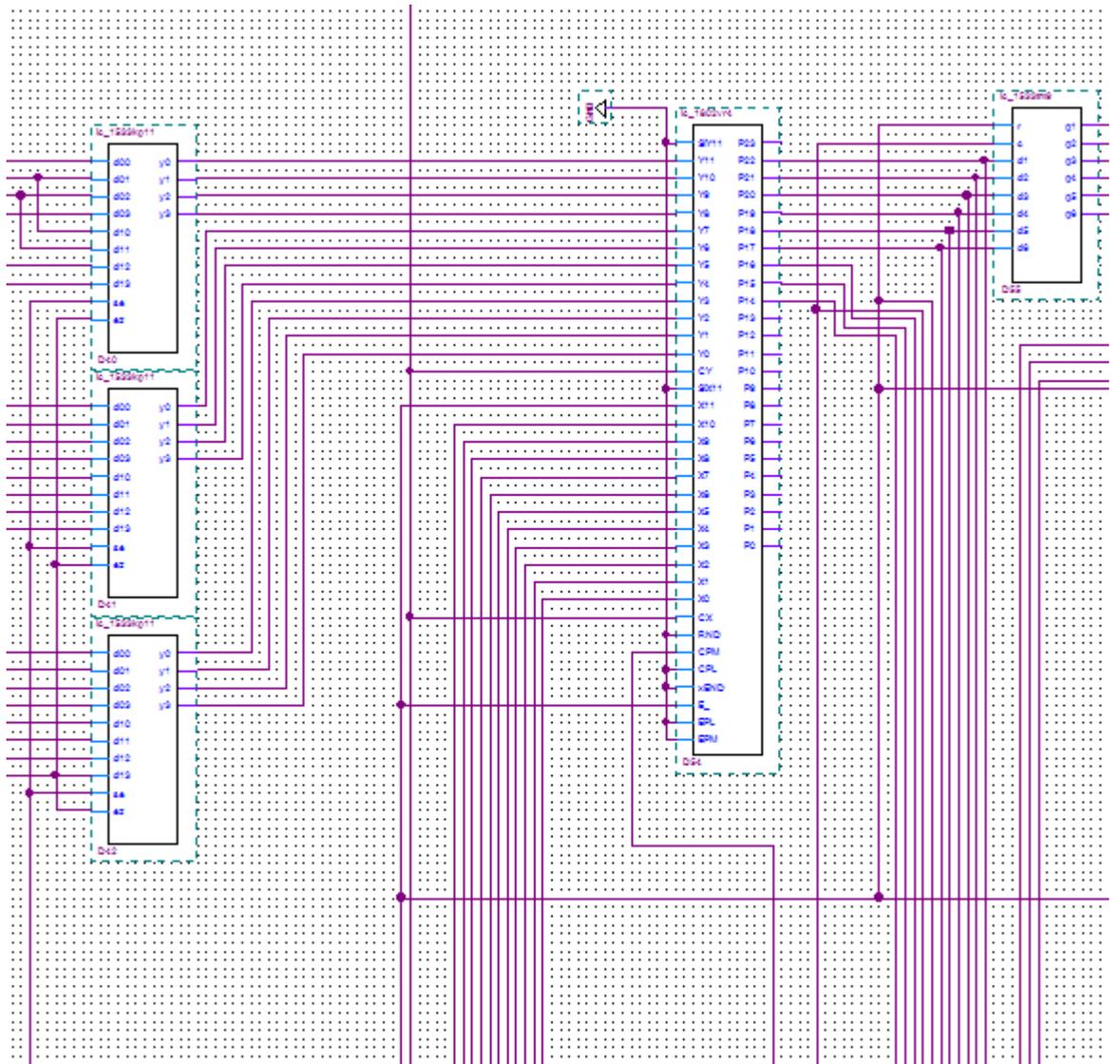
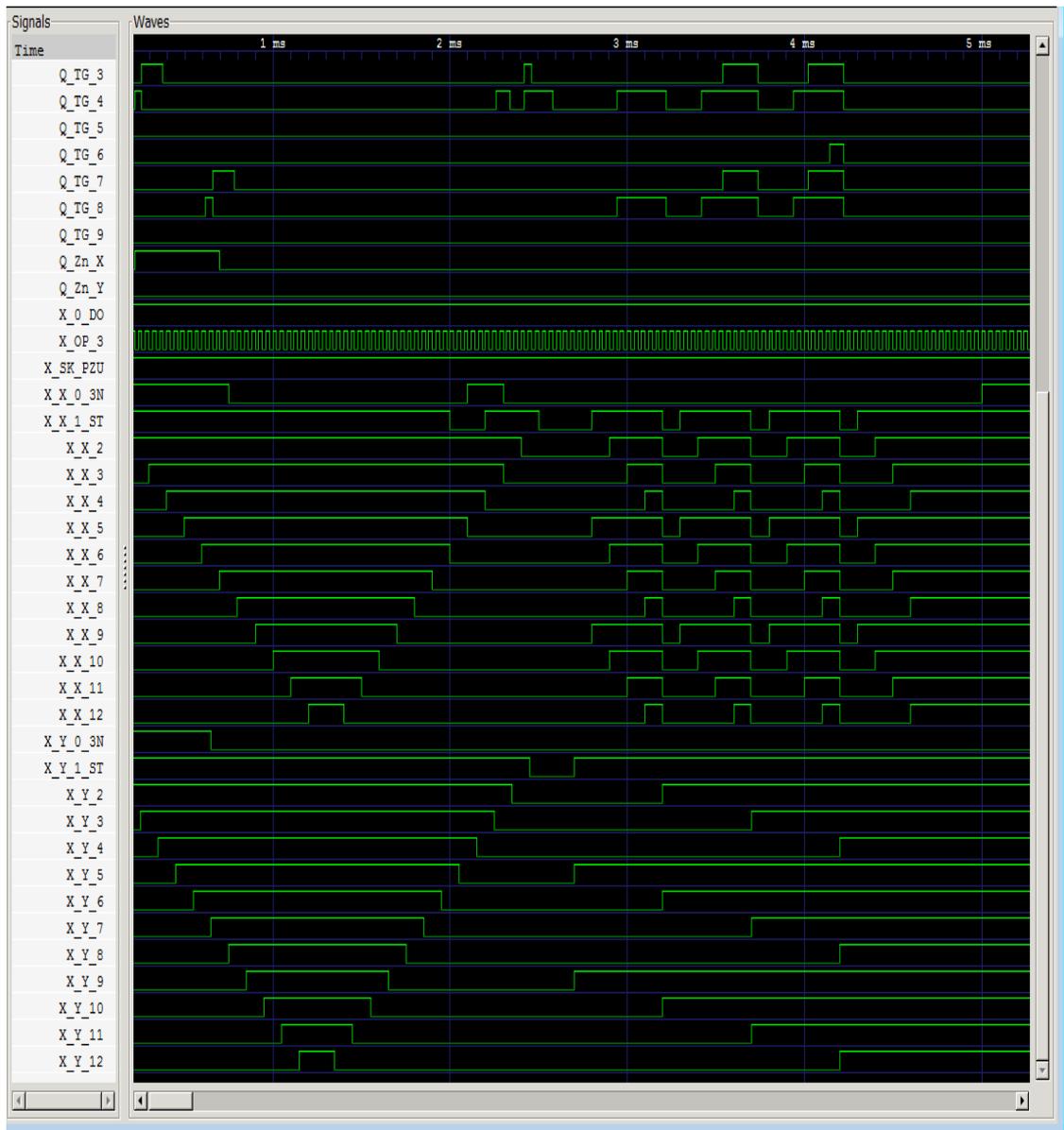


Рисунок 21 Фрагмент блок-схемы Субблок Н8.03.18.05

Далее, как и в предыдущих примерах, мы компилируем блок-схему, затем преобразуем ее в HDL-файл для дальнейшей работы в «SimTest».

Необходимо добиться максимального покрытия, чтобы проверить каждый компонент устройства, поэтому создаем новый проект и приступаем к написанию скрипта, с помощью которого шаг за шагом будет моделироваться последовательность тестовых воздействий.



*Рисунок 22 Фрагмент результата теста Субблок Н8.03.18.05*

Получаем анализ покрытия:

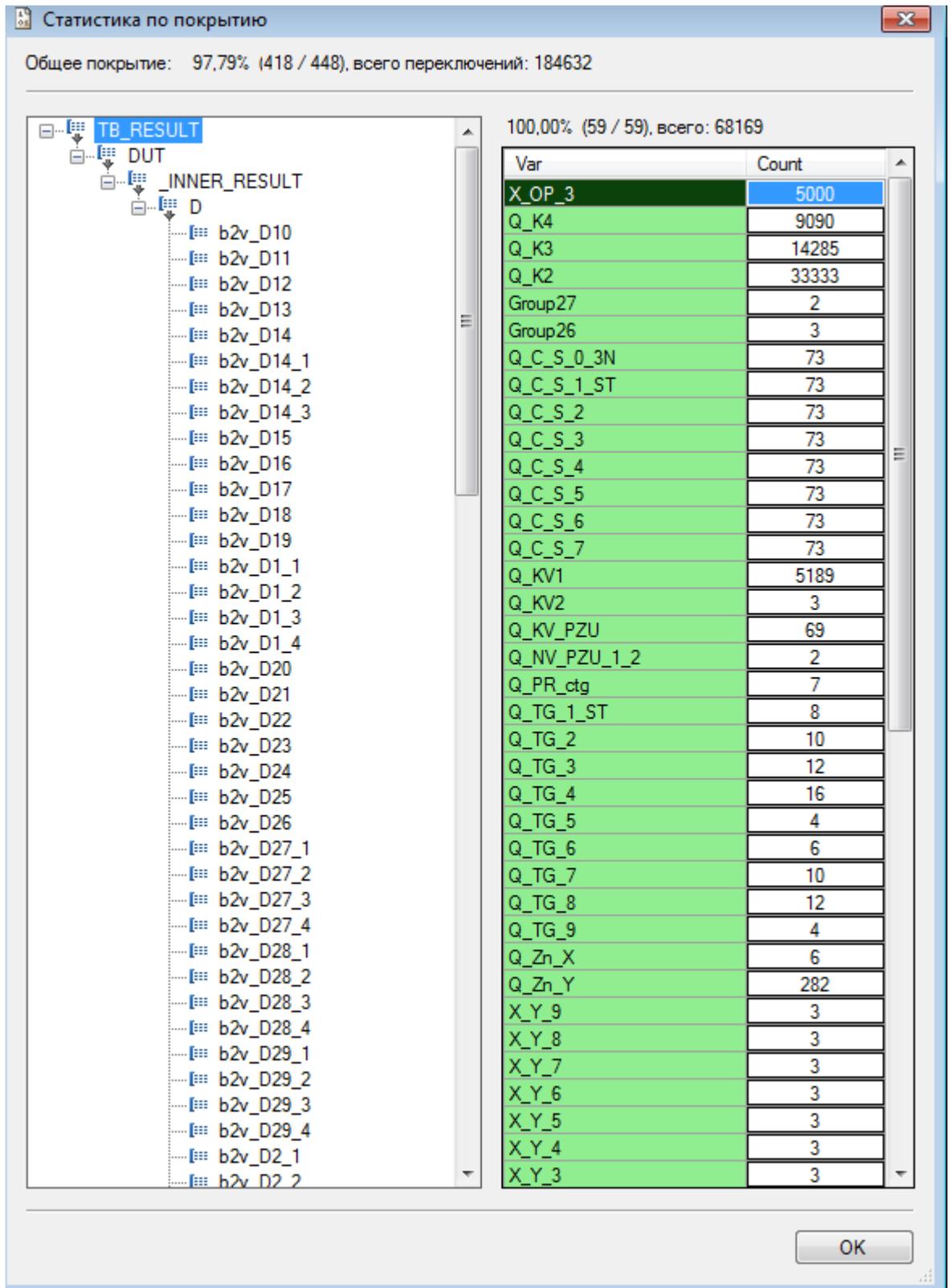
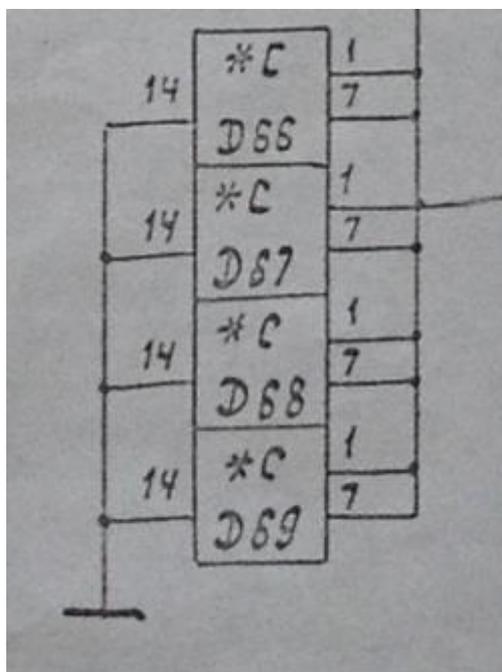


Рисунок 23 Анализ покрытия Субблок Н8.03.18.05

Как видно на рисунке 27, мы получили итоговое покрытие, близкое к 100%. Недостающие проценты составляют сигнальные линии, которые в силу особенностей схемы не имеют возможности изменения состояния, а также сигнальные линии, подключенные к питанию (постоянная лог. «1») и корпусу (постоянный лог. «0»), такие как на рисунке 28:



*Рисунок 24 Пример элемента с постоянным сигналом на входах*

Таким образом, покрытие в 97,79% является полным и максимальным для объекта контроля.

## 7.Выводы

1. Были созданы и протестированы программные модели для всех 17 различных типов микросхем, входящих в состав Субблок Н8.03.18.05
2. Создана программная модель Субблок Н8.03.18.05 в среде разработки «Quartus II» и произведено преобразование в HDL-модель.
3. С помощью САПР «SimTest» был смоделирован тест для проверки работоспособности объекта контроля, обладающий максимальным покрытием.

Таким образом были выполнены все поставленные задачи и достигнута цель работы.

## 8. Заключение

В данной работе рассмотрена история развития схемотехники, а также описаны и проанализированы современные методы автоматического тестирования.

Описанные в работе алгоритмы написания программы для функционального тестирования были применены для создания программной модели и разработки тестовой последовательности для контроля работоспособности цифрового радиоэлектронного устройства Субблок Н8.03.18.05.

## 9. Литература

1. Городецкий А. Снова о внутрисхемном тестировании ICT // Компоненты и технологии. 2011. №7. С. 58–59.
2. Holtzer M. In-circuit pin testing: An excellent potential source of value creation // SMT Surface Mount Technology Magazine, 2015, 30 (6). P. 68–71.
3. Renbi A., Delsing J. Application of Contactless Testing to PCBs with BGAs and Open Sockets // Journal of Electronic Testing: Theory and Applications, 2015, 31 (4). P. 339–347.
4. Renbi A., Delsing J. Contactless Testing of Circuit Interconnects // Journal of Electronic Testing: Theory and Applications, 2015, 31 (3). P. 229–253.
5. Thoulath Begam V. M., Baulkani S. Compact test set method for high fault coverage test pattern generation // International Journal of Applied Engineering Research, 2015, vol. 10, 55. P. 453–458.
6. Гришкин В. М., Лопаткин Г. С, Михайлов А. Н., Овсянников Д. А. Интерфейсный метод построения моделей входных воздействий для тестирования электронных цифровых модулей // Вопросы радиоэлектроники, серия ОТ. 2013. № 1. С. 80–88.
7. Степанов Ю. Л., Гришкин В. М., Большаков А. А., Лопаткин Г. С., Ким М. А. Автоматизированное построение тестов цифровых электронных модулей для комплекса тестового контроля и диагностики УТК-512 // Вопросы радиоэлектроники. 2012. Т. 1. № 1. С. 79–89.
8. Мащинский Н. С., Елаев Е. В., Федюкович П. А. Моделирование сложных цифровых устройств с целью их тестирования // Процессы управления и устойчивость. 2015. Т. 2. № 1. С. 452–457.

9. Елаев Е. В., Степанов Ю. Л., Ферсенков В. В. Подходы к моделированию микропроцессоров для построения контрольнодиагностических тестов // Процессы управления и устойчивость, 2015. Т. 2. № 1. С. 398–403.
10. Шахнова В. А. Микропроцессоры и микропроцессорные комплекты интегральных микросхем", том 2.
11. Гришкин В. М., Лопаткин Г. С, Михайлов А. Н., Овсянников Д. А. Интерфейсный метод построения моделей входных воздействий для тестирования электронных цифровых модулей // Вопросы радиоэлектроники, серия ОТ. 2013. № 1. С. 80–88.
12. Гусев О. А., Елаев Е. В., Машинский Н. С., Нуракунов А. Автоматизация генерации тестовых воздействий для комбинационных цифровых схем // Процессы управления и устойчивость. 2016. Т. 3. № 1. С. 389–393.
13. Мельник В. И., Гришкин В. М., Михайлов А. Н., Овсянников Д. А. Методика разработки тест-программ контроля и диагностики цифровых устройств с использованием САПР «SimTest» // Электроника: Наука, технология, бизнес. 2013. № S (128). С. 118–124.
14. Елаев Е. В., Степанов Ю. Л., Ферсенков В. В. Подходы к моделированию микропроцессоров для построения контрольно-диагностических тестов // Процессы управления и устойчивость, 2015. Т. 2. № 1. С. 398–403