

ИННОВАЦИОННАЯ ОБРАЗОВАТЕЛЬНАЯ ПРОГРАММА



Проект 2: индивидуальная траектория обучения
и качество образования

Цель: ориентированное на требования рынка
образовательных услуг улучшение качества
подготовки и переподготовки специалистов

Федеральное агентство по образованию
Государственное образовательное учреждение
высшего профессионального образования
Владимирский государственный университет
Кафедра вычислительной техники

ПРОЕКТИРОВАНИЕ ЗАКАЗНЫХ ИНТЕГРАЛЬНЫХ СХЕМ В СРЕДЕ САПР MENTOR GRAPHICS

МЕТОДИЧЕСКИЕ УКАЗАНИЯ
К ЛАБОРАТОРНЫМ РАБОТАМ

ВЛАДИМИР 2009

УДК 681.3 (621.3)

ББК 32.85

П79

Составители:

В.Н. Ланцов, С.Г. Мосин, В.С. Кухарук, С.В. Федоров

Рецензент

Доктор технических наук, профессор
заведующий кафедрой конструирования
и технологии радиоэлектронных средств
Владимирского государственного университета

В. П. Крылов

Печатается по решению редакционного совета
Владимирского государственного университета

Проектирование заказных интегральных схем в среде САПР
П79 Mentor Graphics : метод. указания к лаб. работам / Владим. гос. ун-т ;
сост. В. Н. Ланцов, С. Г. Мосин, В. С. Кухарук, С. В. Федоров. -
Владимир : Изд-во Владим. гос. ун-та, 2009. - 80 с.

Состоят из двух частей. Первая часть посвящена вопросам проектирования аналого-цифровых блоков интегральных схем, вторая часть – проектированию заказных ИС в среде САПР Mentor Graphics. Содержат методические разработки и материалы для проведения лабораторного практикума по дисциплине «Автоматизация проектирования» на тему «Проектирование заказных интегральных схем». В ходе лабораторных работ изучаются возможности САПР фирмы «Mentor Graphics» при проектировании интегральных схем по заказной технологии. Студенты приобретают практические навыки использования САПР фирмы «Mentor Graphics» при проектировании заказных цифровых и аналоговых блоков ИС на разных этапах: от системного уровня до топологического.

Предназначены для бакалавров и магистрантов, обучающихся по направлению 230100 – информатика и вычислительная техника.

Табл. 2. Ил. 82. Библиогр.: 4 назв.

УДК 681.3 (621.3)

ББК 32.85

ВВЕДЕНИЕ

С увеличением сложности разрабатываемых электронных устройств возрастает роль *автоматизации процесса проектирования*. Основное требование к современным САПР – реализация сквозного маршрута проектирования, обеспечивающего разработку электронного устройства в трех областях (поведенческой, структурной и физической) с различной степенью детализации в приемлемые сроки. Для обеспечения данного требования в состав САПР должны входить программные системы, реализующие в совокупности замкнутый цикл проектирования электронного устройства на всех этапах от получения технического задания до формирования масок топологии кристалла ИС. Пакеты прикладных программ (ППП) САПР должны быть совместимы на уровне используемых форматов представления исходных данных и промежуточных или окончательных результатов работы каждой программной системы.

Предлагаемый цикл лабораторных работ ориентирован на пошаговое изучение принципов проектирования цифровых интегральных схем (ИС) по заказной технологии и приобретение практических навыков применения средств САПР фирмы *Mentor Graphics* на каждом этапе от системного уровня до топологического.

В ходе выполнения лабораторных работ будут изучены следующие подсистемы САПР фирмы *Mentor Graphics*:

- *ModelSim* – функциональное моделирование и верификация выходных откликов ИС;
- *Leonardo Spectrum* – логический синтез ИС;
- *IC-Station* – проектирование топологии ИС;
- *xCalibre* – структурная верификация с учетом паразитных параметров.

Цикл лабораторных работ предназначен для бакалавров и магистрантов, обучающихся по направлению 230100 – «Информатика и вычислительная техника».

ЧАСТЬ I. ПРОЕКТИРОВАНИЕ АНАЛОГО-ЦИФРОВЫХ БЛОКОВ ИС

Лабораторная работа № 1

ФУНКЦИОНАЛЬНО-ЛОГИЧЕСКОЕ ПРОЕКТИРОВАНИЕ ЗИС

1 Цель работы: Изучение принципов функционально-логического проектирования цифровых схем в пакетах ModelSim и Leonardo Spectrum фирмы Mentor Graphics.

2 Система HDL-моделирования цифровых устройств (ModelSim)

Общий маршрут функционально-логического проектирования представлен на рисунке 1.1.

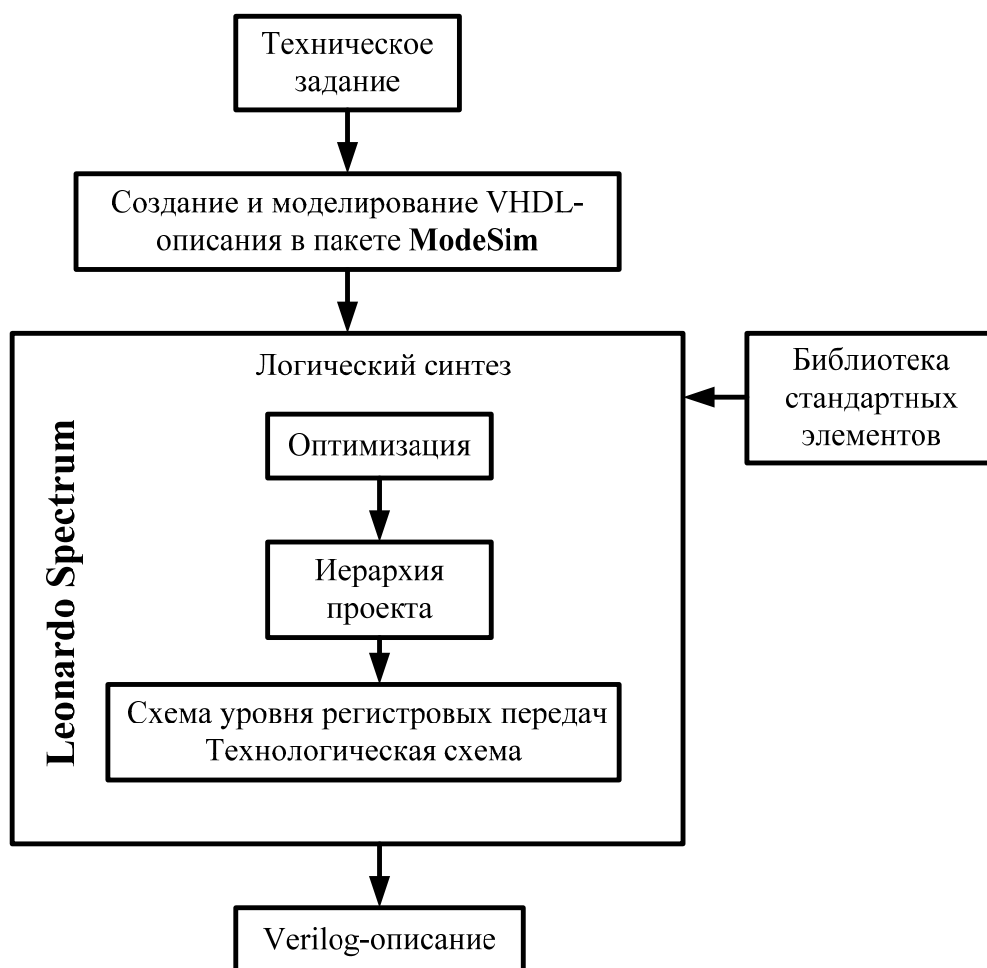


Рис. 1.1. Маршрут функционально-логического проектирования ЗИС

ModelSim представляет собой наиболее распространенное на сегодняшний день семейство средств моделирования цифровых схем на функционально-логическом уровне. Пользователь имеет возможность выбора между старшей версией ModelSim SE (Special Edition), работающей под управлением как ОС UNIX, так и Windows NT, и младшей версией ModelSim PE (Personal Edition), предназначенной только для использования под управлением ОС Windows. ModelSim SE и PE имеют опции, предназначенные для моделирования VHDL, Verilog и смешанных проектов (опция PLUS), предоставляя пользователю широкий выбор языков описания аппаратуры.

Среда проектирования обладает оконным интерфейсом, посредством которого пользователь управляет процессом проектирования и моделирования. Все операции производятся внутри рабочей области текущего окна.

Главное окно (ModelSim SE PLUS 5.8d) (рис. 1.2) содержит пункты меню, а так же набор инструментальных средств (иконок) для быстрого доступа к основным операциям.

Основные функции, выполняемые из главного окна:

- 1) Создание и управление проектами (**File**);
- 2) Редактирование проекта (**Edit**);
- 3) Графический интерфейс проекта (**View**);
- 4) Компилирование проекта (**Compile**);
- 5) Моделирование проекта (**Simulate**);
- 6) Инструментарий (**Tools**);
- 7) Настройка рабочего окна (**Window**);
- 8) Помощь (**Help**).

Пример выполнения задания: Создание и верификация VHDL-описания инвертора.

На рисунке 1.3, *а* показан КМОП инвертор или вентиль НЕ, который использует один р-МОП или n-МОП транзистор. Вверху схемы подключено питание VDD и внизу подключена земля GND. Когда на входе $A = 0$, n-МОП транзистор закрыт, а р-МОП транзистор открыт. Следовательно, выход $Y = 1$, так как соединяется с VDD. И наоборот, когда $A = 1$ n-МОП открыт, а р-МОП транзистор закрыт и $Y = 0$. Таблица истинности такого элемента показана в таблице 1.1. Символ вентили показан на рис. 1.3, *б*.

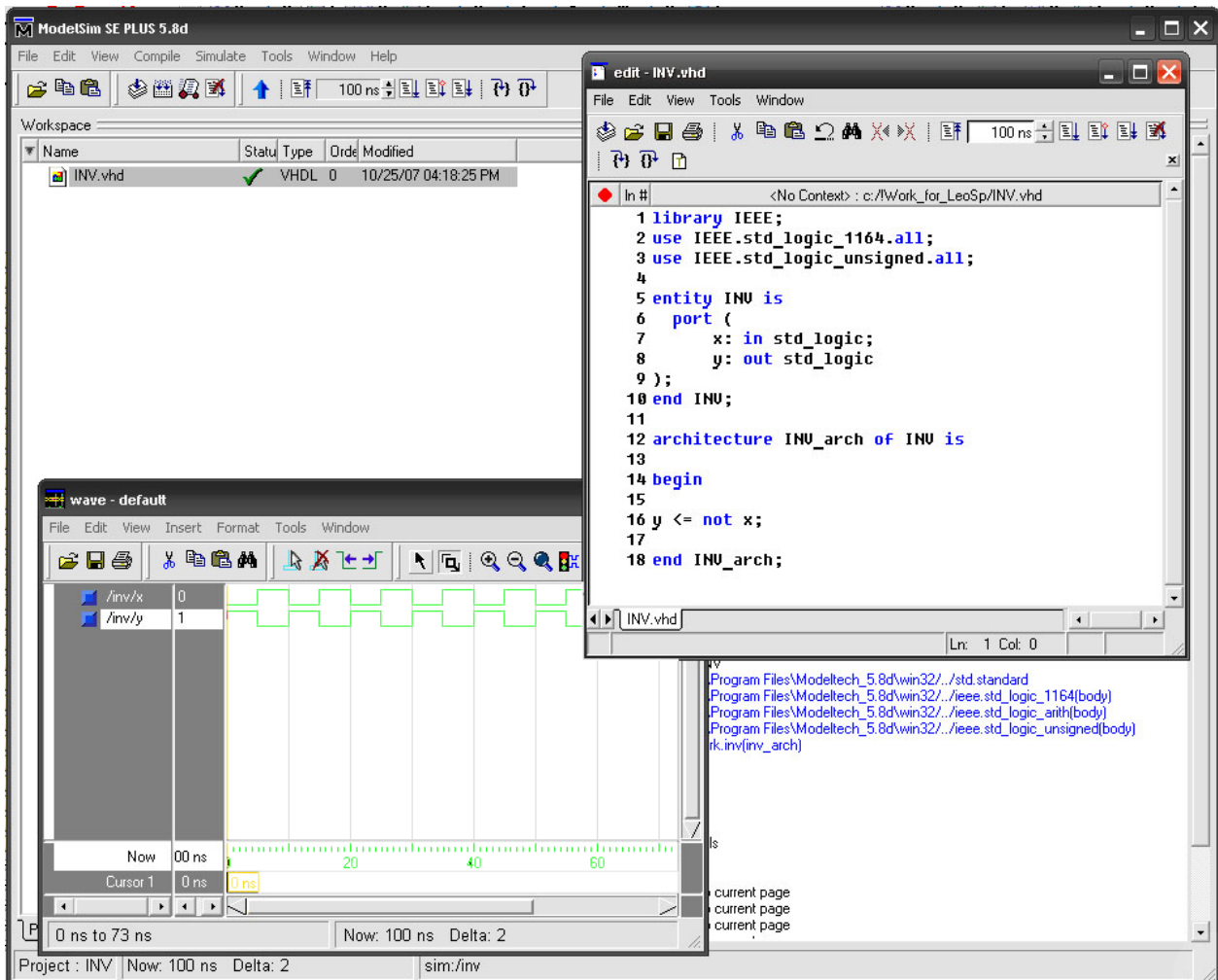


Рис. 1.2. Результаты моделирования VHDL-описания инвертора

Таблица 1.1. Таблица истинности инвертора

A	Y
0	1
1	0

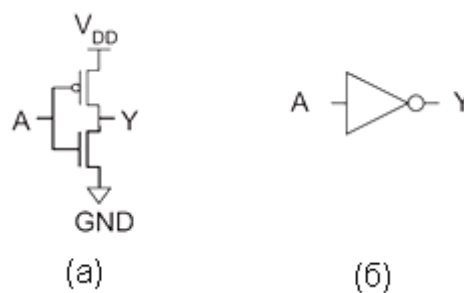


Рис. 1.3. Схема инвертора (а) и символ (б) $Y = \bar{A}$

1) Запустить программу **ModelSim SE 5.8d.exe**.

2) Для создания VHDL-описания необходимо создать проект «INV».

File > New > Project;

3) Создание VHDL-описания компонента INV.

```
library IEEE;
use IEEE.std_logic_1164.all;
use IEEE.std_logic_unsigned.all;
entity INV is
  port (
    x: in std_logic;
    y: out std_logic
  );
end INV;
architecture INV_arch of INV is
begin
  y <= not x;
end INV_arch;
```

4) Создание тестовых векторов «dofile» **File > New > Source > Do**, обеспечивающих определение правил изменения входных сигналов во времени.

```
vsim -t ns INV
restart -force
force -freeze INV/x 0 0, 1 5 -r 10
add wave -binary /*
configure wave -namecolwidth 200
configure wave -valuecolwidth 50
configure wave -justifyvalue left
configure wave -signalnamewidth 0
configure wave -snapdistance 10
configure wave -datasetprefix 0
configure wave -rowmargin 4
configure wave -childrowmargin 2
configure wave -signalnamewidth 2
set IgnoreWarning 1
run 100 ns
```

5) Компиляция проекта **Compile > Compile All**.

6) Моделирование запускают командой «do dofile» из командной строки.

3 Синтез VHDL-описания в пакете Leonardo Spectrum

Leonardo Spectrum – пакет логического синтеза и оптимизации CPLD/FPGA/ASIC на основе VHDL, Verilog и “смешанных” описаний проектов на базе библиотек всех ведущих фирм изготовителей (более 100). Основные функциональные возможности:

- единая среда для проектирования БМК, ПЛИС и ЗИС;
- поддержка проектов, использующих любую комбинацию VHDL, Verilog и EDIF, а так же стандартных решений в виде IP-блоков;
- высокое качество результатов (QoR – Quality of Results) в сочетании с высокой скоростью синтеза;
- возможность генерации оптимизированного HDL-кода с помощью модуля HDLInventor;
- использование оптимизирующего алгоритма F.A.S.T.;
- полная интеграция со средствами размещения и трассировки изготовителей кристаллов (P&R Integrator);
- усовершенствованные отладочные возможности (LeonardoInsight).

Уровни синтеза в пакете Leonardo Spectrum. Существует три уровня синтеза в Leonardo Spectrum, которые условно обозначают Level 1, Level 2, Level 3. Все три уровня используют одно ядро синтеза и оптимизации, что позволяет достичь высоких результатов с применением минимального количества инструментов, обеспечивая при этом контроль и управление процессом проектирования. Методология проектирования и управления, а также механизмы управления с каждым уровнем становятся более детальными: на уровне Level 1 создается базовый список соединений (netlist) схемы проекта, на уровне Level 2 добавляется больше настроек управления проектом, на уровне Level 3 доступно большое количество расширенных настроек.

Уровень Level 3 является легким в использовании, гибким и интерактивным пакетом логического синтеза, оптимизации и анализа. Level 3 позволяет пользоваться независимыми от технологии методами проектирования для FPGA и CPLD кристаллов, в отличии от Level 1и Level 2. Данный уровень дополнительно поддерживает разработку проектов для ASIC – Application Specific IC (микросхем, выполненных по заказной технологии). На данном уровне можно выполнять восходящее проектирование с использованием списка соединений, определенных выбранной технологией. Кроме этого, для работы могут использоваться файлы «сценариев», работающие интерактивно в режиме пакетной обработки. Поддержка работы над проектом как одним разработчиком, так и группой инженеров.

На уровне Level 3 используется большинство современных технологий оптимизации, которые позволяют достичь высококачественных результатов при работе как с FPGA, так и ASIC. Level 3 включает в себя следующие возможности:

- работа с проектами, реализуемыми в базе заказных микросхем;
- возможность перенастройки синтеза для работы с FPGA и ASIC;
- совместное использование при проектировании различных языков описания аппаратуры: Verilog, VHDL, EDIF;
- возможность работы с интерактивной командной строкой, для получения полного доступа к базе данных разработчика;
- поддержка работы над проектом команды разработчиков.

Выполнение логического синтеза. На рисунке 1.4 изображено главное окно пакета Leonardo Spectrum после запуска программы LeonardoSpectrum LS2007a_37.exe.

В левой части расположено окно, с помощью которого можно выполнить настройку синтеза:

- выбор целевой технологии, в базе в которой выполняется синтез (**Technology**);
- установка рабочей директории и выбор исходных файлов (**Input**);
- установка ограничений (частота тактового сигнала, специфические задержки) (**Constraints**);
- настройка параметров оптимизации для определенной технологии (**Optimize**);
- настройка параметров создания отчетов (**Report**);
- настройка формата выходного файла (**Output**).

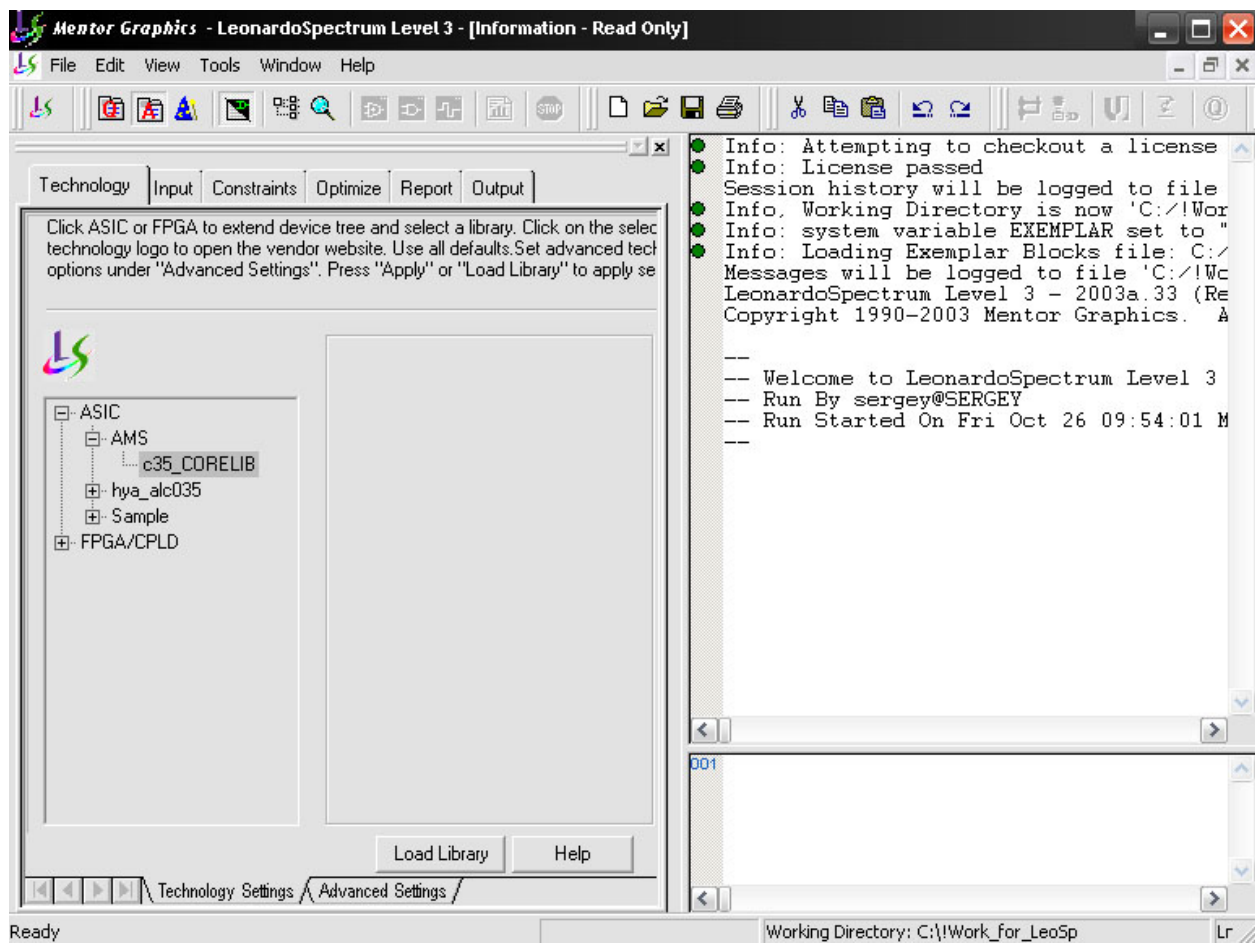


Рис. 1.4. Главное окно LeonardoSpectrum

Пример выполнения задания. Синтез VHDL-описания инвертора.

Выполнение логического синтеза, в определенном технологическом базисе, требует настройки Leonardo Spectrum на использование необходимой библиотеки. Это осуществляется выбором производителя ASIC в окне со списком производителей на первой закладке **Technology** (рис. 1.5), а также выбором необходимой технологии (c35_CORELIB).

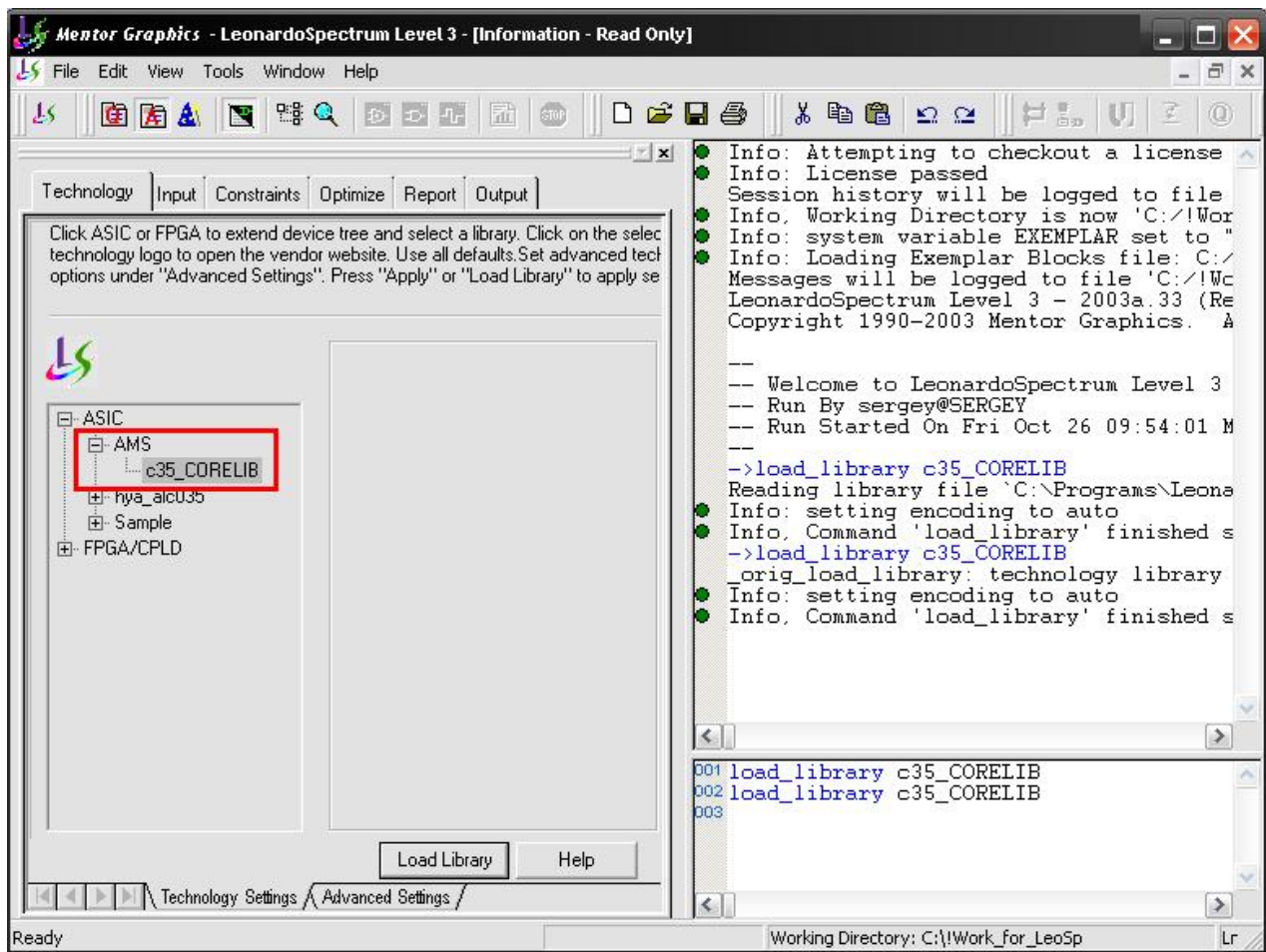


Рис. 1.5. Окно настроек целевой технологии

Вкладка **Input** используется для загрузки входных файлов и настроек стиля кодирования. Открываем VHDL-описание инвертора INV.vhd кнопкой **Open files** и устанавливаем рабочую директорию – кнопка **Working Directory** (рис. 1.6).

Выбор организации кодирования состояния конечных автоматов производят в поле **Encoding Style**. Стил **Binary**, обеспечивает обычный для двоичной арифметики способ кодирования (например, для трех бит 0 – 000, 1 – 001, 2 – 010, 3 – 011, 4 – 100 и т.д.). Подобный стиль позволяет экономно расходовать триггеры для хранения данных, однако уступает стилю **Onehot** по быстродействию, при котором для кодирования каждого состояния используют отдельный регистр (например, 0 – 0000, 1 – 0001, 2 – 0010, 3 – 0100, 4 – 1000). Как видно, при таком подходе расходуется значительно больше регистров, особенно для большого числа разрядов. Стили

Twohot и **Random**, являются разновидностями рассмотренных выше. Стиль **Gray** аналогичен **Binary**, однако использует для кодирования кода Грея. Основная его особенность заключается в том, что каждое последующее число отличается от предыдущего только одним битом. Это свойство полезно при построении помехоустойчивых схем. По умолчанию установлен режим **Auto**, при котором Leonardo Spectrum автоматически выбирает наиболее оптимальный стиль кодирования.

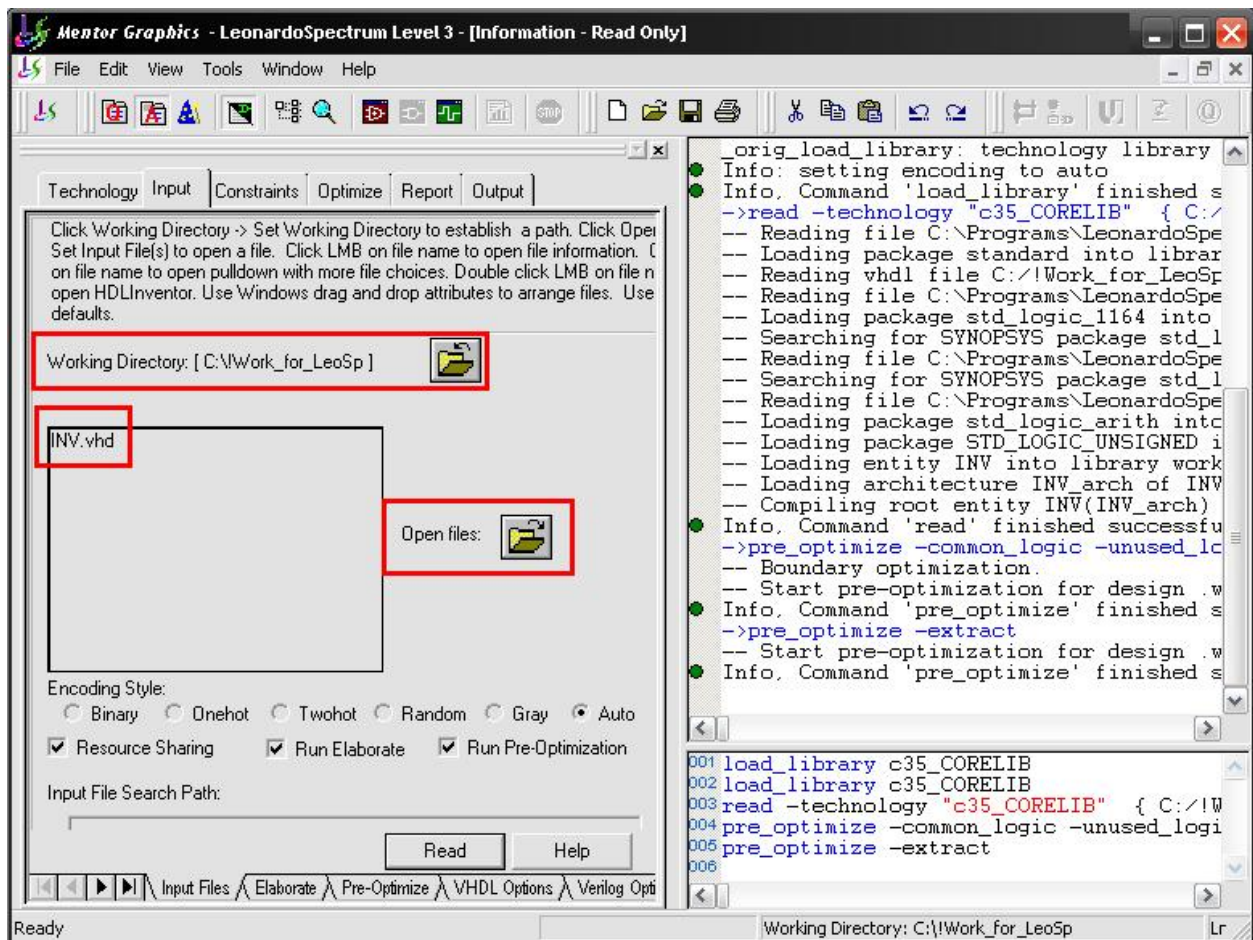


Рис. 1.6. Окно настройки входных файлов

Отметка полей **Run Elaborate** и **Run Pre-Optimization** обеспечивает выполнение предварительной оптимизации при считывании.

Вкладка **Constraints** определяет основные требования к результатам синтеза схемы устройства в определенном технологическом базисе (рис. 1.7).

В поле **Specify Clock Frequency** задается требование к частоте тактового сигнала. В случае если необходимо задать период сигнала, нужное значение заносится в поле **Specify Clock Period**.

Для того чтобы задать специфические задержки, служат поля **Input Ports to Registers**, **Registers to Registers**, **Registers to Output Ports**, **Inputs to Outputs**.

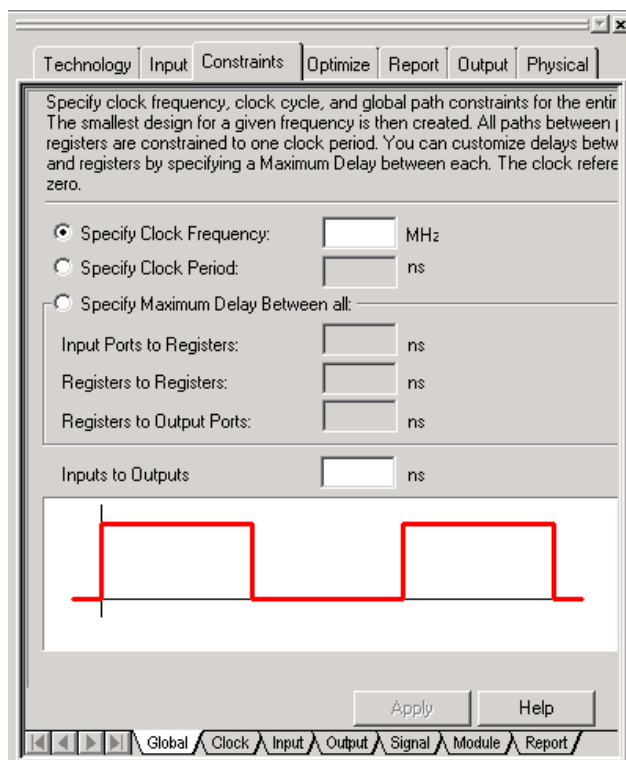


Рис. 1.7. Окно настройки требований к результатам синтеза

Если в схеме имеется несколько синхронизирующих сигналов, временные требования к каждому из них можно специфицировать на подзакладке **Clock**, выбираемой в нижней части поля закладки **Constraints**. Подзакладки **Input** и **Output** обеспечивают возможность задания ограничений для каждого из входных и выходных портов схемы соответственно.

На подзакладке **Signal** выделяются сигналы, которые не должны быть удалены при выполнении оптимизации схемы. На подзакладке **Module** аналогично отмечаются модули не подлежащие общей оптимизации и устанавливаются критерии оптимизации для каждого из них.

Подзакладка **Path** предназначена для выделения путей, подлежащих исключению из общих требований и ограничений. На подзакладке **Report**

имеется возможность просмотра всех установленных требований и сохранения их в файл для дальнейшего использования, а также загрузки ранее сохраненных файлов.

Для примера INV, настройки вкладки **Constraints** остаются выбранными по умолчанию.

Запуск оптимизации проекта в выбранном технологическом базисе выполняется на закладке **Optimize**, нажатием на кнопку **Optimize** (рис. 1.8).

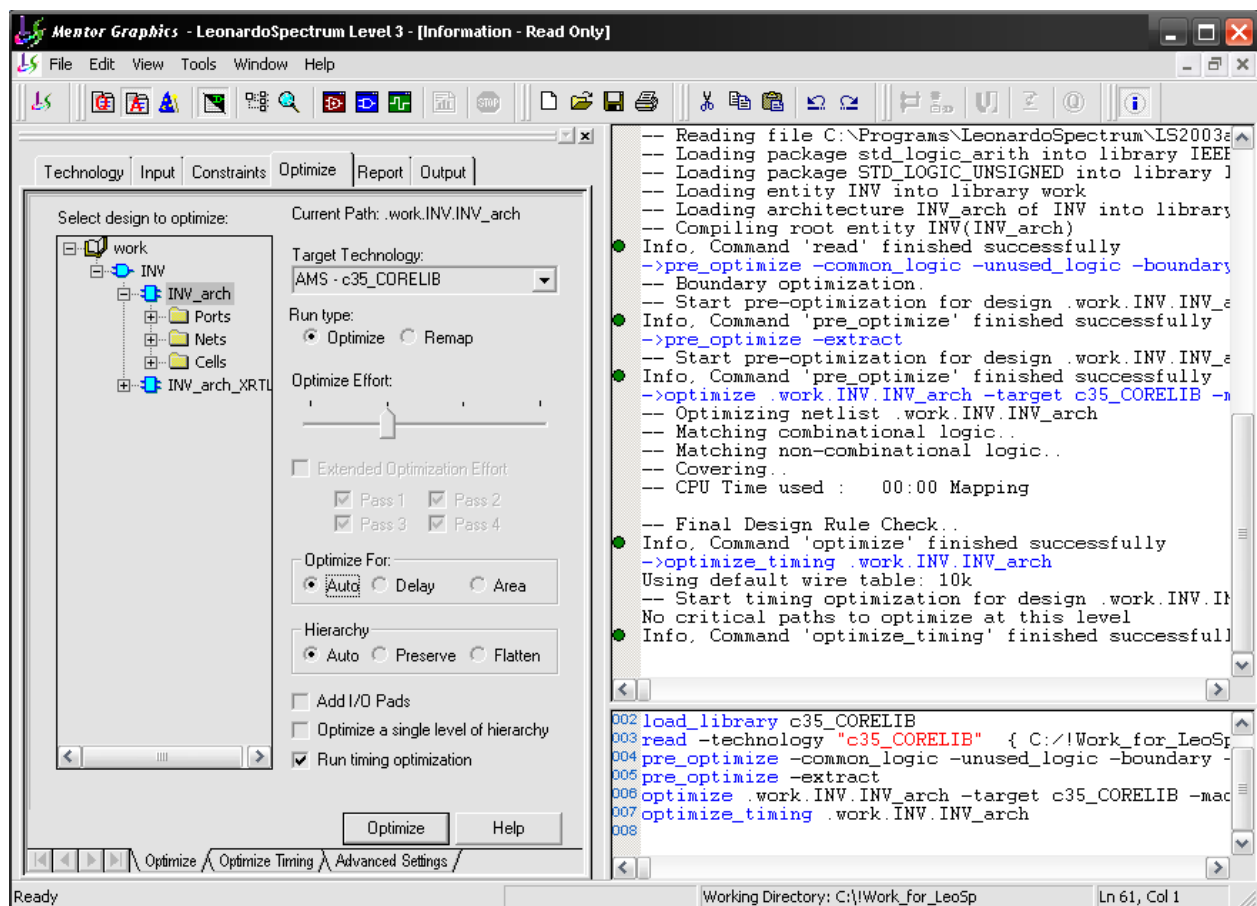


Рис. 1.8. Окно настройки оптимизации проекта

Если необходимо выполнить только переложение схемы в библиотечные элементы технологического базиса, в поле **Run type**, выбирается отметка **Remap**, иначе оптимизация не будет выполнена. Отметкой поля **Optimize Effort** настраивается количество итераций при выполнении оптимизации. В поле **Optimize For** указывается основной критерий оптими-

зации – достижение наибольшего быстродействия или наименьшей занимаемой площади кристалла.

Поле **Hierarchy** определяет возможность удаления иерархической структуры проекта. В случае если выбрана отметка **Preserve**, иерархия проекта сохраняется, то есть все компоненты схемы останутся представленными в виде отдельных блоков. Выбор значения **Flatten**, обеспечивает преобразование схемы устройства в один уровень иерархии, что обеспечивает достижение более качественных результатов оптимизации. Отметка **Auto** позволяет системе **Leonardo Spectrum** автоматически определить наиболее оптимальный выбор.

Поле **Add I/O Pads** позволяет включить в схему буферные библиотечные элементы, подключаемые к контактным площадкам. Это используется при синтезе полной схемы, размещаемой на одном кристалле.

Поле **Optimize a single level of hierarchy**, обеспечивает выполнение оптимизации только для верхнего уровня иерархии. Выбор этого свойства полезен при использовании тщательно отработанных технологически зависимых схемных решений отдельных структурных компонентов. Отметка поля **Run timing optimization**, обеспечивает выполнение всех временных и частотных требований на этапа синтеза и моделирования.

Для примера INV, настройки вкладки **Optimize** выбраны по умолчанию.

Закладка **Report** (рис. 1.9), служит для получения информации о результатах синтеза и оптимизации.

Кнопка **Report Area**, на одноименной подзакладке, служит для получения отчета о занимаемой площади, а кнопка **Report Delay** на подзакладке **Report Delay** – отчета о временных характеристиках схемы. На подзакладке **Report Delay**, имеется возможность настроить вывод информации в отчет таким образом, что будут отображены только те пути, в которых время прохождения сигнала отвечает заданным требованиям. Эти настройки доступны в группе настроек **Critical Path Reporting Options**.

Кроме того, имеется возможность настройки вывода путей в порядке возрастания времени задержки **Sort By Delay**, оценочной частоты тактовых сигналов **Report Clock Frequency**, имен входных портов компонентов **Report Input Pins** и ряда других характеристик. Как правило, для ПЛИС-реализаций оценки занимаемой площади выводятся по умолчанию, в то

время как для получения временных характеристик необходимо нажать кнопку **Report Delay**.

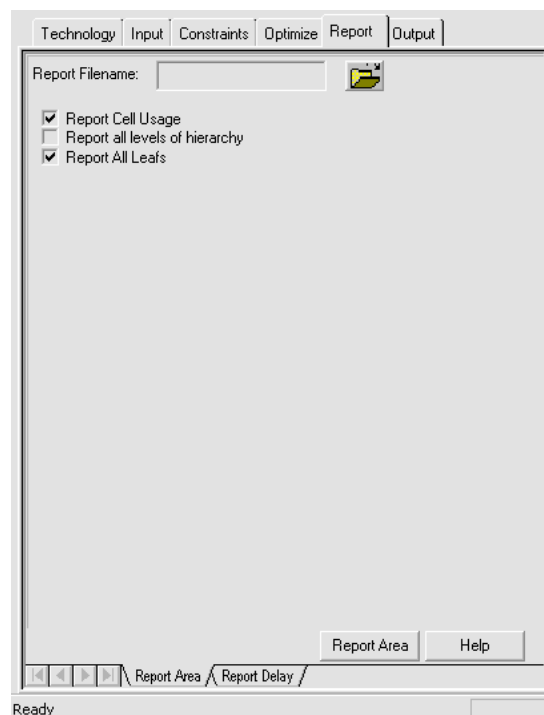


Рис. 1.9. Окно настройки вывода результатов

Система Leonardo Spectrum позволяет получить графическое отображение схемы устройства. Имеется возможность получить две разновидности схемы.

Первый вид показывает логическую схему устройства на уровне регистровых передач (RTL – Register Transfer Level), не выраженную библиотечными элементами целевой технологией. Эта схема подготавливается на этапе предварительной оптимизации при считывании файлов задания. Для получения схемы необходимо использовать кнопку **View RTL Schematic** (рис. 1.10).

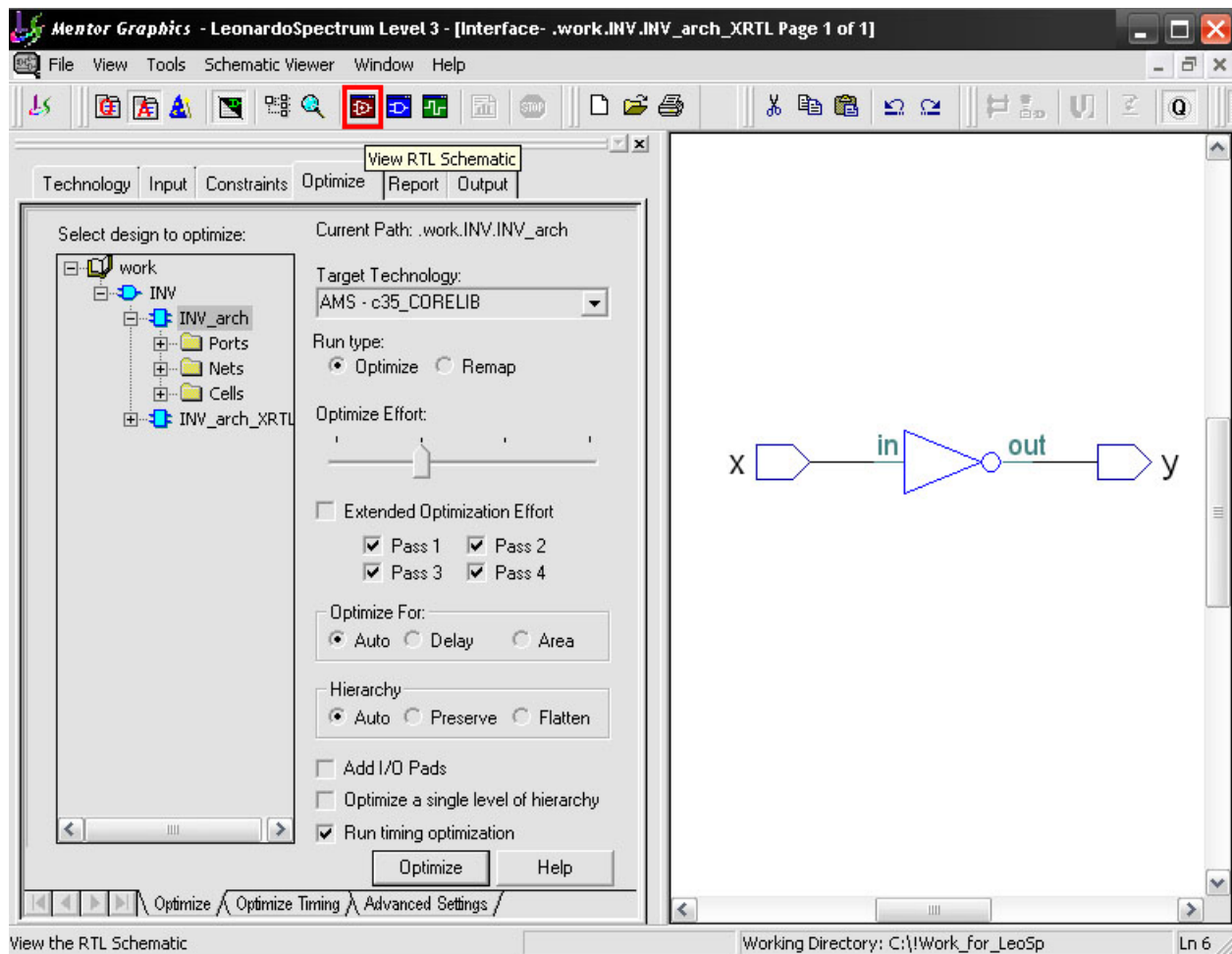


Рис. 1.10. Вид схемы RTL Schematic

Второй вид выражен библиотечными элементами целевой библиотеки и отображает результаты синтеза и оптимизации. Для получения этого вида схемы необходимо воспользоваться кнопкой **View Technology Schematic** (рис. 1.11).

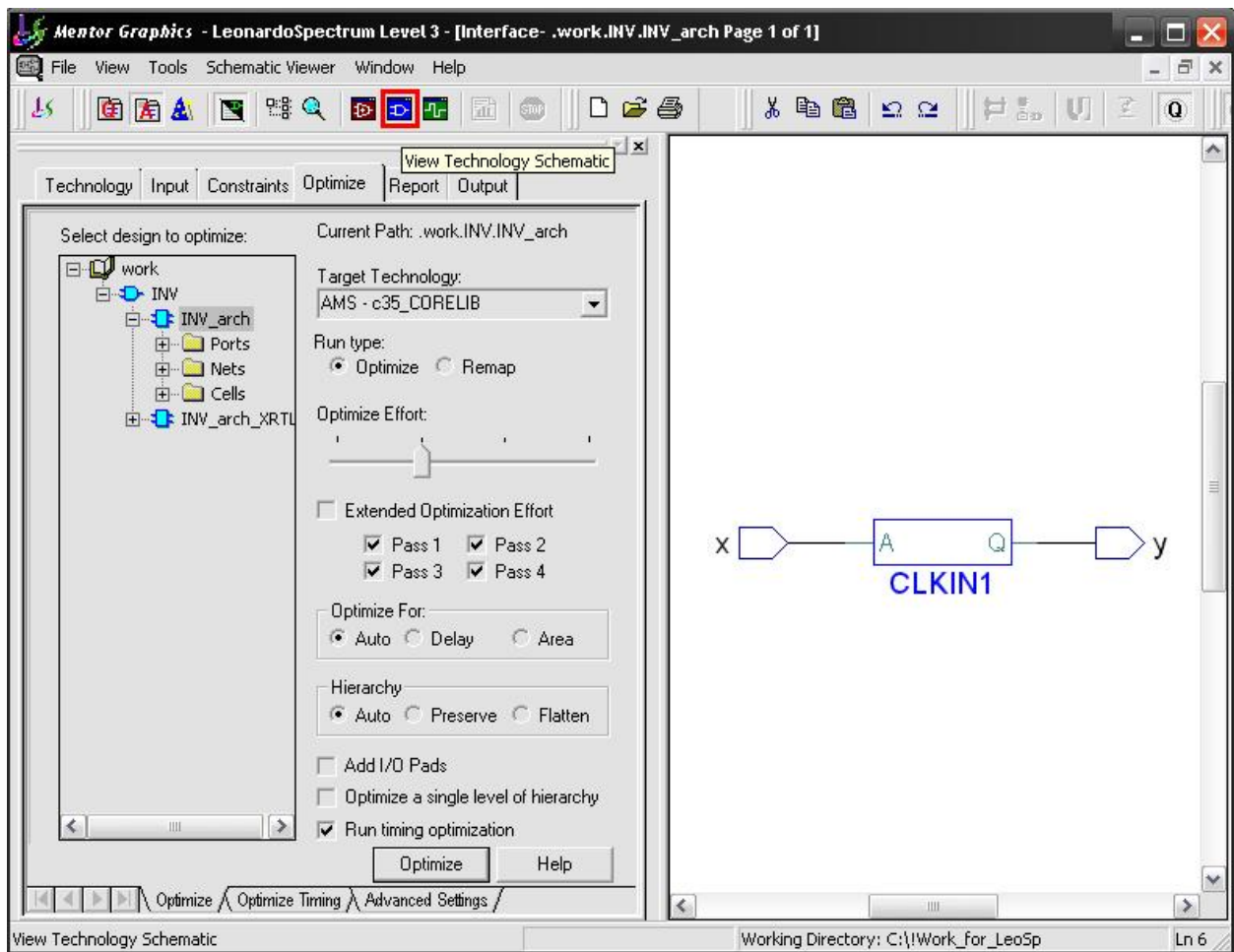


Рис. 1.11. Вид схемы Technology Schematic

Результаты синтеза и оптимизации должны быть сохранены (вкладка **Output**) в одном из стандартных форматов, описывающих схему устройств (рис. 1.12). В таком виде, данные могут быть прочитаны любой САПР, поддерживающей стандартные форматы файлов структуры.

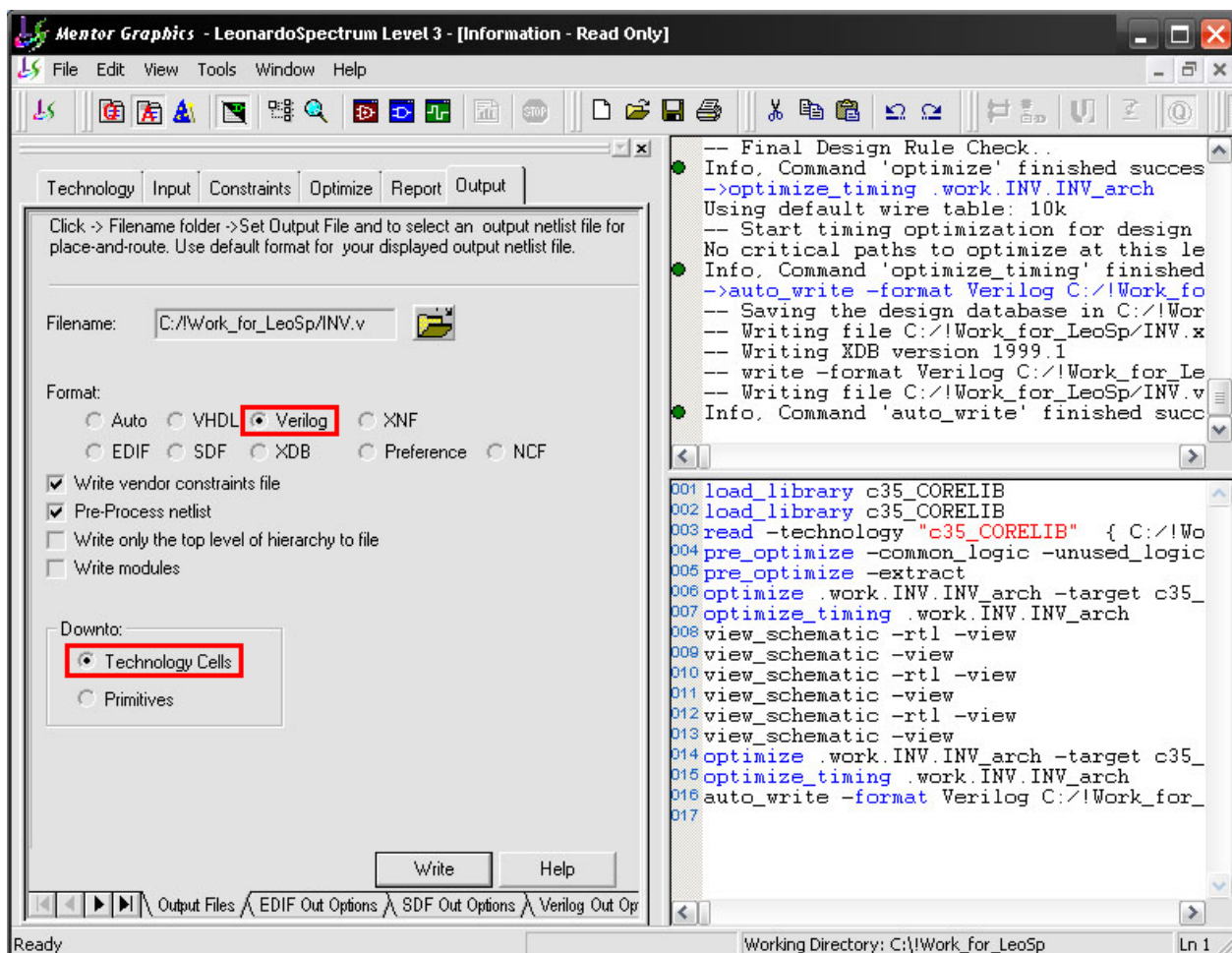


Рис. 1.12. Окно настройки формата полученных файлов

Таким образом, результаты синтеза могут быть переданы, например, для верификации в подсистему ModelSim или для выполнения функций размещения и трассировки в соответствующие САПР. Сохранение данных выполняется из закладки **Output**, с помощью кнопки **Write**. В поле **Format** имеется возможность выбрать один из выходных форматов:

- автоматический выбор формата (в зависимости от выбранной целевой технологии) (**Auto**);
- сохранение данных в структурном стиле языка (**VHDL**);
- сохранение данных в формате Verilog (**Verilog**);
- сохранение данных в формате САПР фирмы Xilinx (**XNF, NCF**);
- сохранение данных в наиболее распространенном структурном формате (**EDIF**);

- сохранение информации о задержках в структурных компонентах в специализированный файл sdf (**SDF**);
- сохранение файла внутреннего формата Leonardo Spectrum для сохранения как самого проекта, так и всех его настроек (**XDB**).

Поле **Write vendor constraints File**, служит для включения режима создания параллельно с файлом схемы соответствующего файла настроек, используемого в САПР конкретного производителя. Для ПЛИС-реализаций это поле должно быть отмечено. Поле **Pre-process netlist**, служит для включения режима записи, при котором выполняются все необходимые преобразования имен цепей, компонентов и т.п. Для ПЛИС-реализаций это поле должно быть отмечено. Поле **Write only the top level of hierarchy to file** позволяет записать в выходной файл только самый верхний уровень иерархии. Отметка в поле **Downto** значения **Technology Cells** обеспечивает запись в файл только структурной схемы, включающей библиотечные элементы, что достаточно для передачи данных в подсистемы размещения и трассировки. Значение **Primitives** позволяет получить в выходном файле помимо структурного описания, поведенческие фрагменты, обеспечивающие возможность моделирования полученного файла без привлечения библиотек элементов.

Для экспорта VHDL-описания проекта INV.vhd на уровень схемотехнического проектирования, поддерживаемый пакетом Design Architect-IC, необходимо сохранить данные в формате **Verilog**, выбрав значения **Technology Cells**.

4 Порядок выполнения работы

- 4.1. Ознакомиться с примером выполнения функционально-логического проектирования инвертора;
- 4.2. Получить вариант индивидуального задания у преподавателя;
- 4.3. Запустить программу ModelSim SE 5.8d.exe.
- 4.4. Подготовить VHDL-описание устройства из индивидуального задания.
- 4.5. Подготовить тестовые векторы «dofile».
- 4.6. Выполнить моделирование в пакете ModelSim и сравнить с заданными результатами моделирования.

- 4.7. Запустить программу LeonardoSpectrum LS2007a_37.exe.
- 4.8. Выполнить синтез VHDL-описания, получить RTL Schematic и Technology Schematic.
- 4.9. Получить результаты синтеза в пакете Leonardo Spectrum.
- 4.10. Провести анализ полученных результатов.

5 Содержание отчета

- 5.1. Цель работы и задание.
- 5.2. VHDL-описание индивидуального задания.
- 5.3. Таблица истинности для логических элементов из индивидуального задания.
- 5.4. Результаты моделирования в пакете ModelSim.
- 5.5. Схема уровня регистровых передач (RTL Schematic) и технологическая схема (Technology Schematic).
- 5.6. Результаты синтеза в пакете Leonardo Spectrum (Verilog-описание).
- 5.7. Анализ полученных результатов и выводы по работе.

6 Варианты индивидуальных заданий

Вариант №1. Регистр (рис. 1.13 – 1.15).

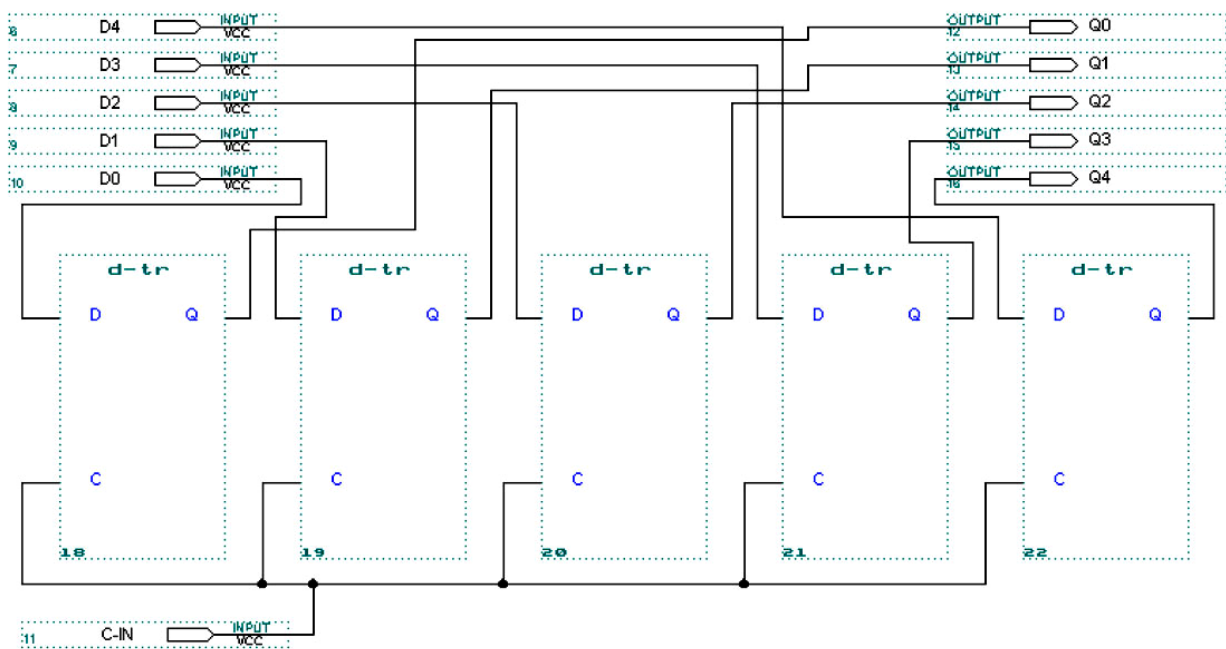


Рис. 1.13. Регистр

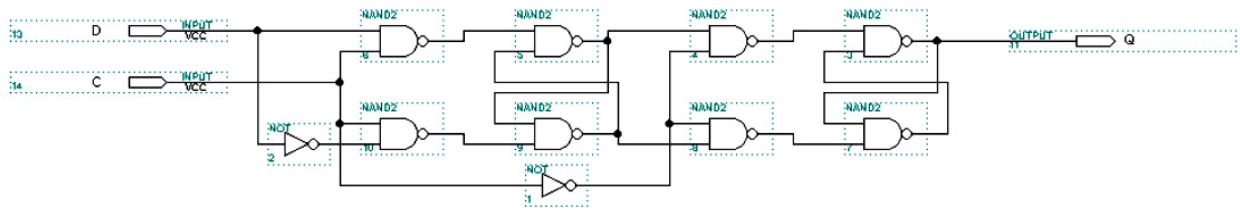


Рис. 1.14. D-триггер

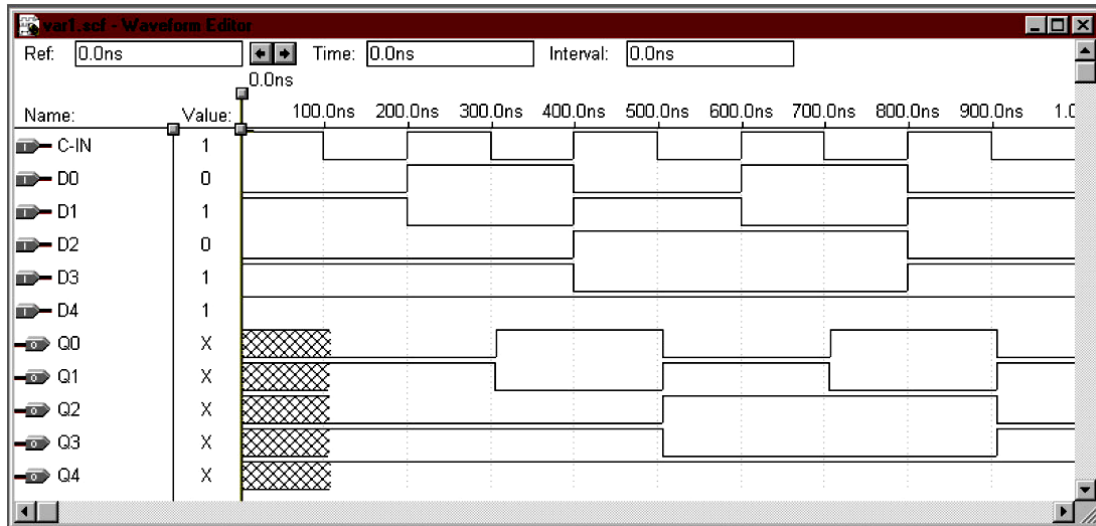


Рис. 1.15. Результаты моделирования регистра

Вариант №2. Трехразрядный сумматор на элементах исключающее ИЛИ, И-НЕ (рис.1.16-1.18).

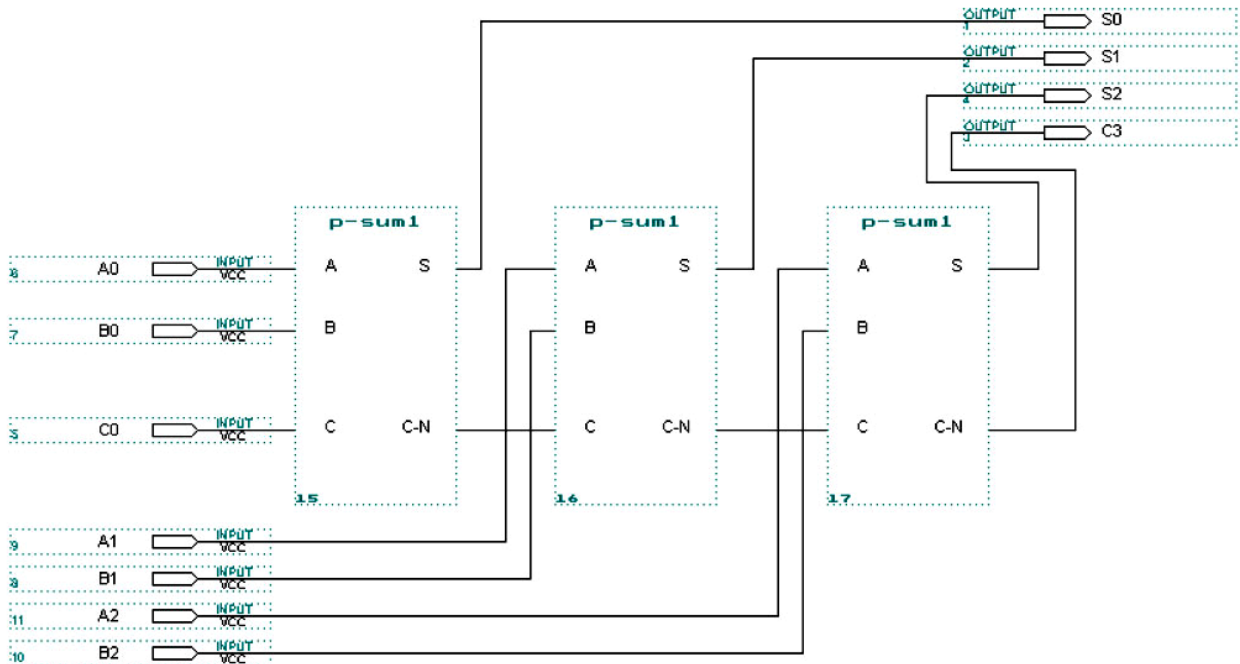


Рис. 1.16. Трехразрядный сумматор на элементах исключающее ИЛИ, И-НЕ

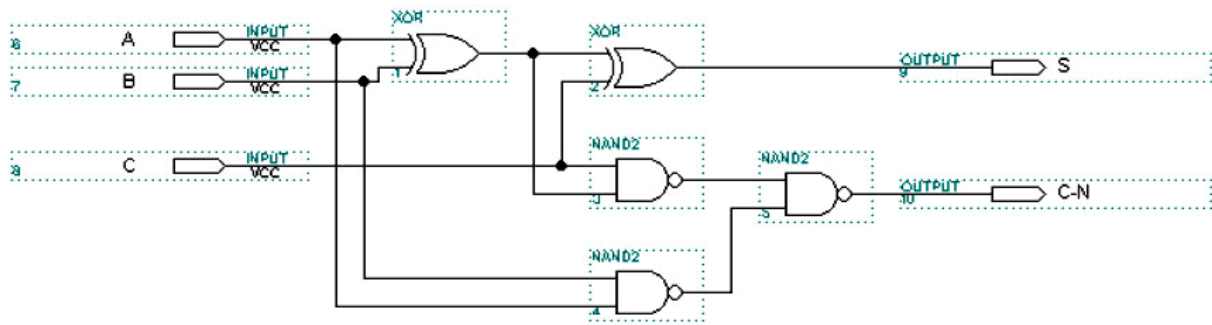


Рис. 1.17. Сумматор на элементах исключающее ИЛИ, И-НЕ

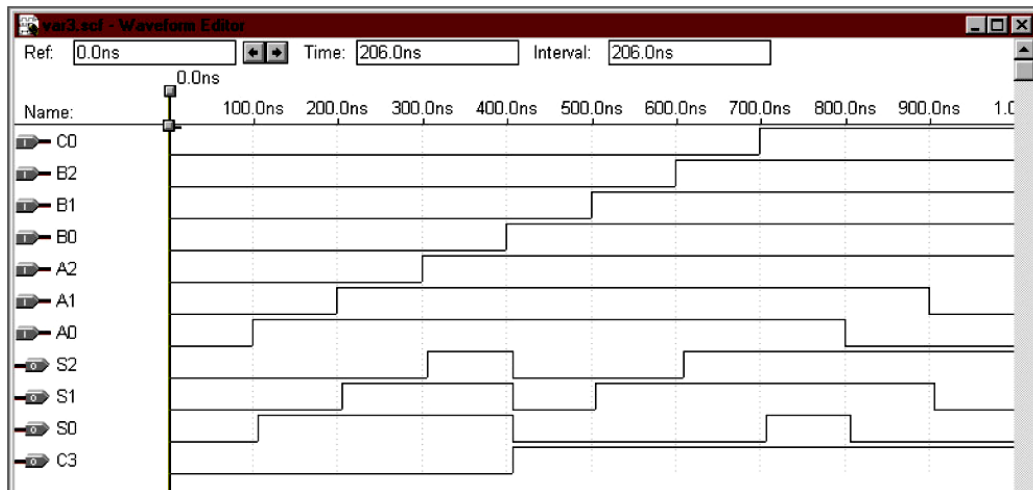


Рис. 1.18. Результаты моделирования трехразрядного сумматора на элементах исключающее ИЛИ, И-НЕ

Вариант №3. Трехразрядный сумматор на элементах И-НЕ (рис. 1.19-1.21).

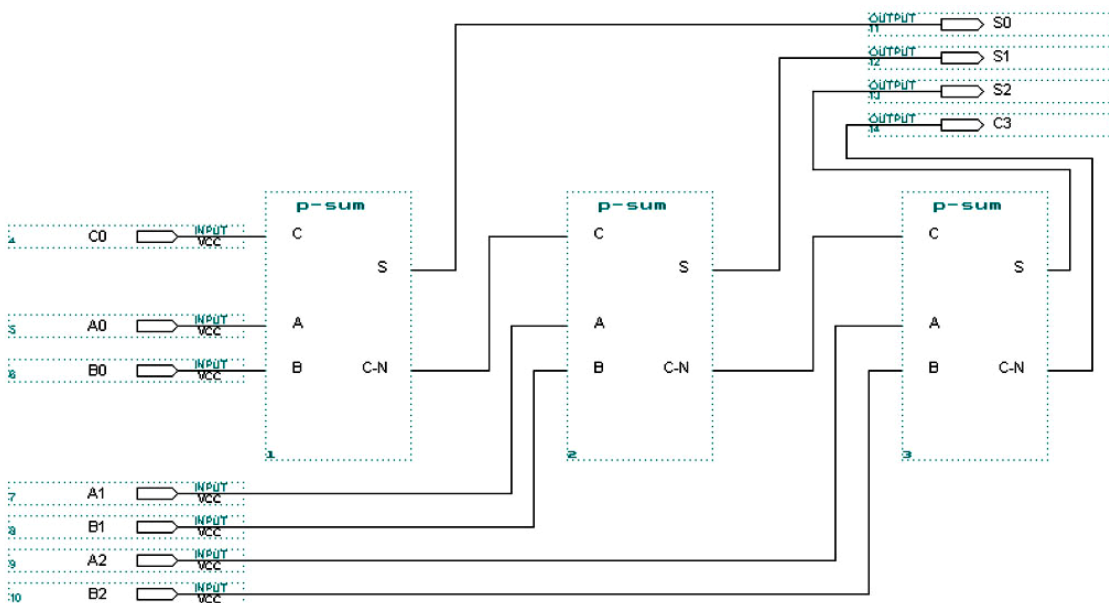


Рис. 1.19. Трехразрядный сумматор на элементах И-НЕ

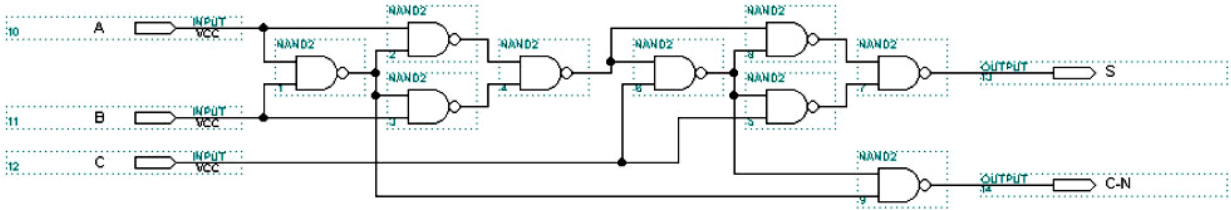


Рис. 1.20. Сумматор на элементах И-НЕ

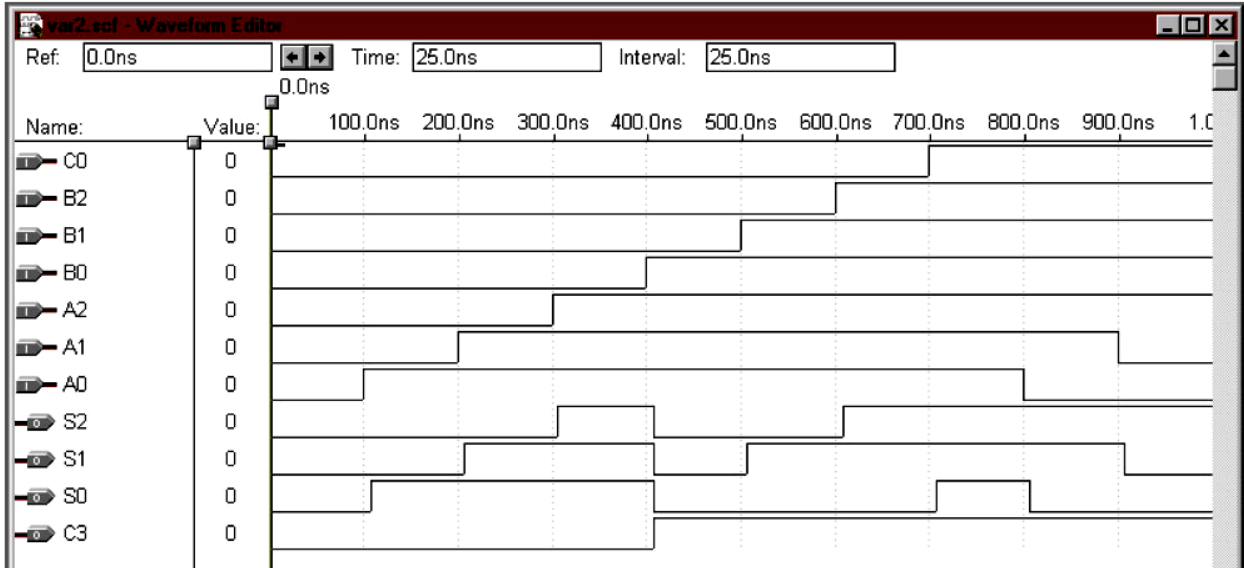


Рис. 1.21. Результаты моделирования трехразрядного сумматора на элементах И-НЕ

Вариант №4. Дешифратор на элементах И-НЕ (рис. 1.22-1.25).

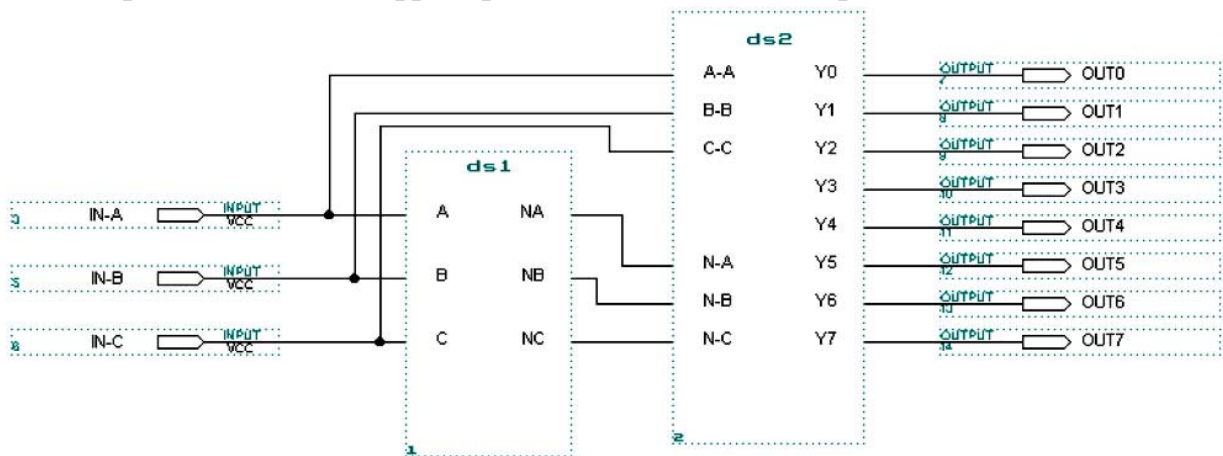


Рис. 1.22. Дешифратор на элементах И-НЕ

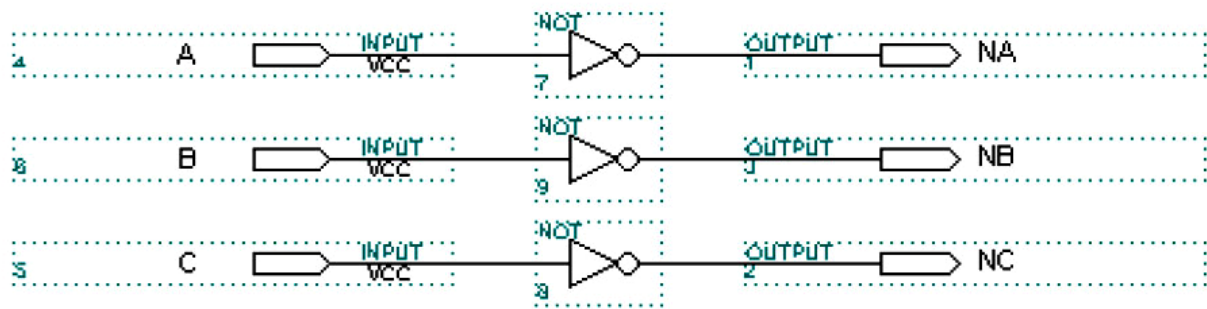


Рис. 1.23. Блок ds1

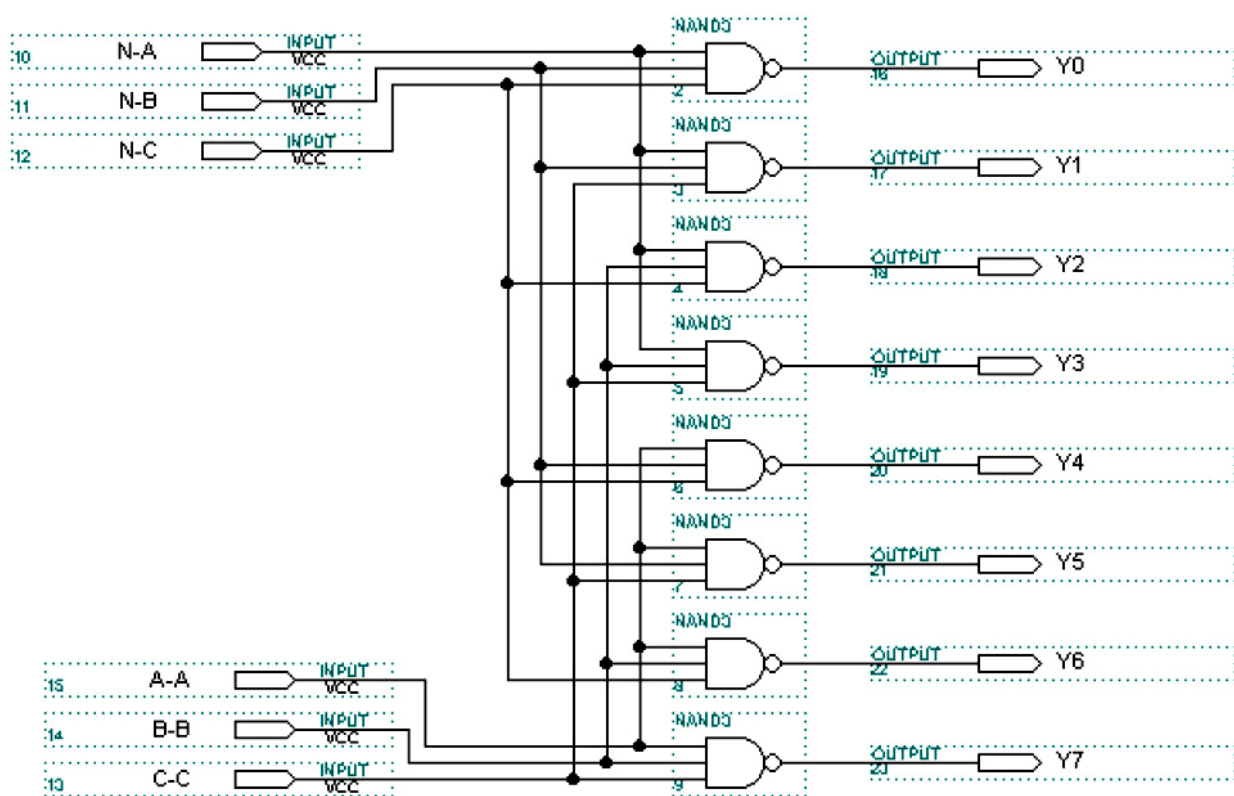


Рис. 1.24. Блок ds2

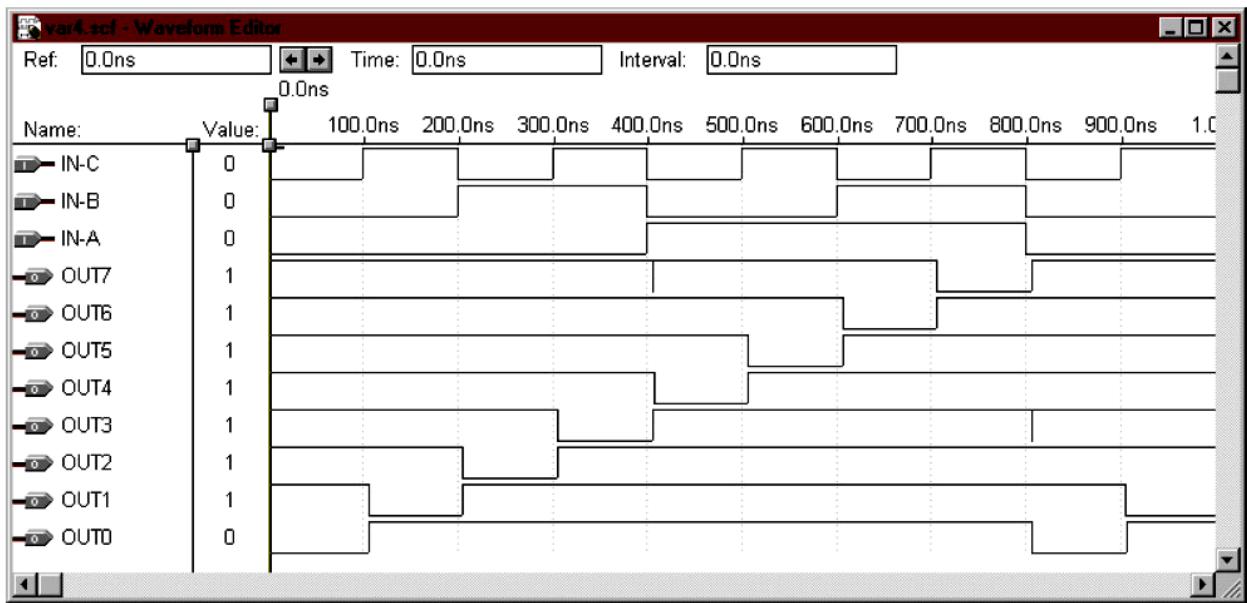


Рис. 1.25. Результаты моделирования дешифратора на элементах И-НЕ

Вариант №5. Дешифратор на элементах ИЛИ-НЕ (рис. 1.26-1.29).

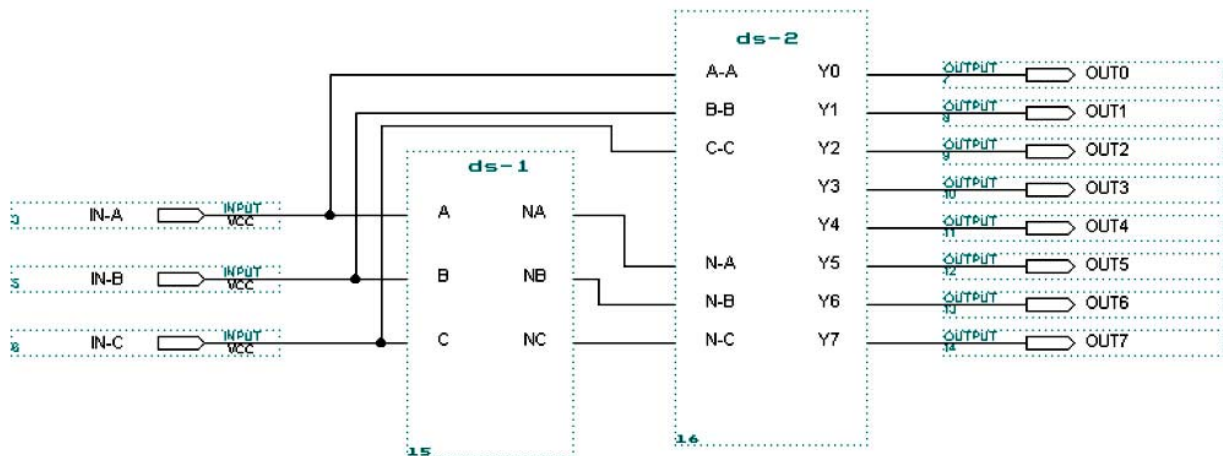


Рис. 1.26. Дешифратор на элементах ИЛИ-НЕ

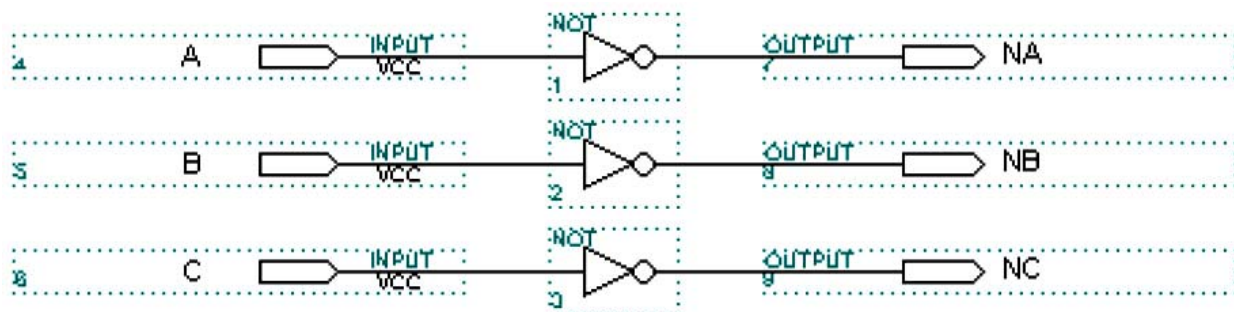


Рис. 1.27. Блок ds1

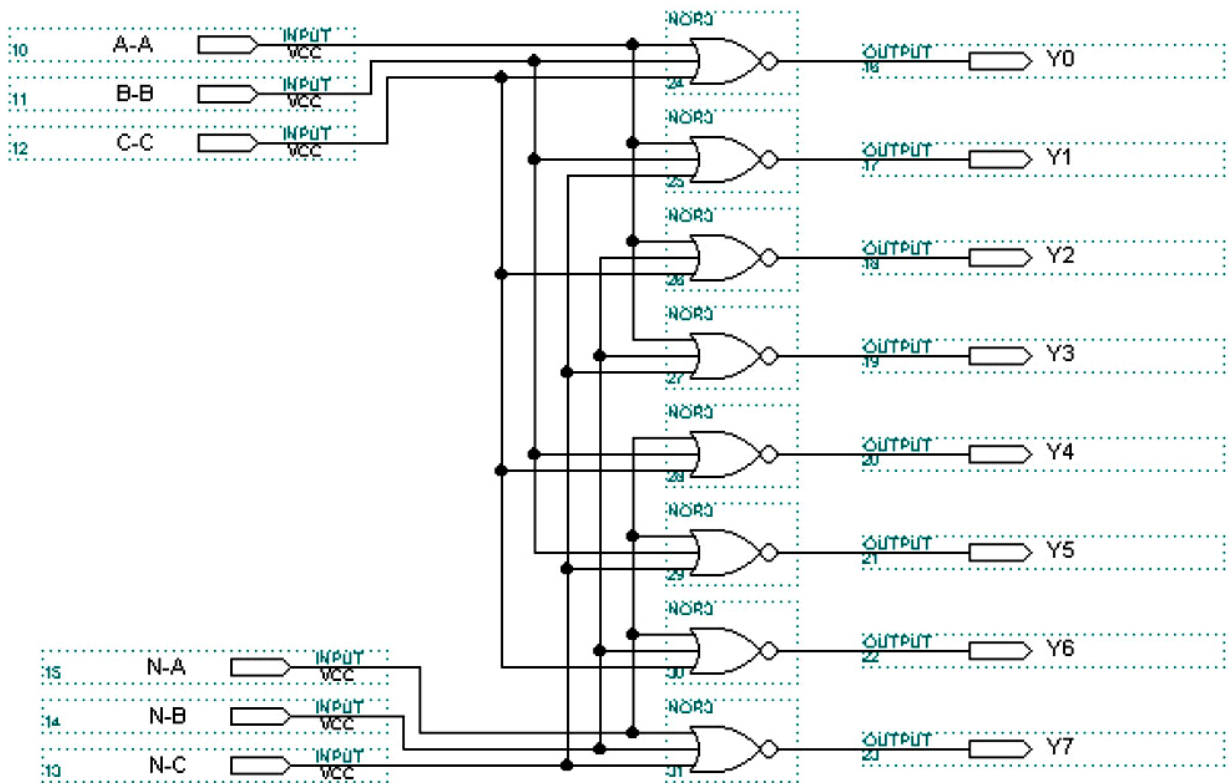


Рис. 1.28. Блок ds2

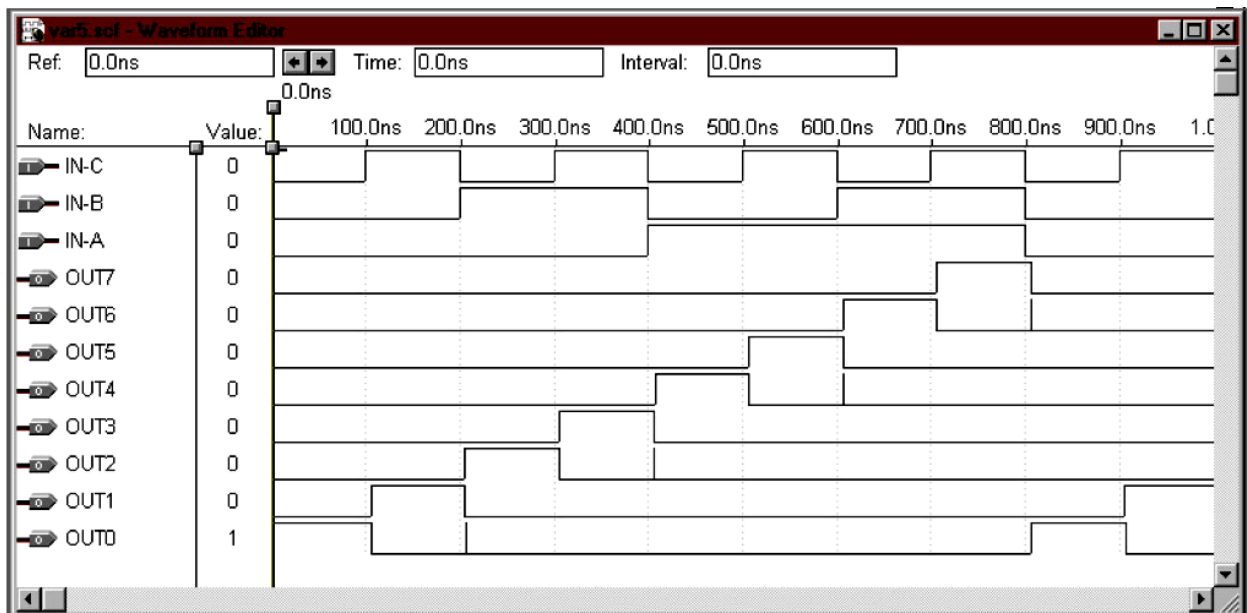


Рис. 1.29. Результаты моделирования дешифратора на элементах ИЛИ-НЕ

7 Контрольные вопросы и задания

7.1. Какого назначение и интерфейс пакета ModelSim?

7.2. Какого назначение и интерфейс пакета Leonardo Spectrum?

7.3. Поясните правила создания воздействующих сигналов в ModelSim.

7.4. Как выполнить моделирование со сравнением формы сигнала в пакете ModelSim?

7.5. Для чего необходимы целевые библиотеки в пакете Leonardo Spectrum?

7.6. Перечислить и охарактеризовать параметры, назначаемые в качестве требований к результатам оптимизации.

7.7. Какие HDL-форматы воспринимает система Leonardo Spectrum в качестве входных?

7.8. Каким образом можно оценить результаты синтеза и оптимизации логической схемы устройства?

7.9. Как получить графическое изображение схемы синтезированного устройства? Какие существуют разновидности схем в Leonardo Spectrum?

Лабораторная работа № 2

СХЕМОТЕХНИЧЕСКОЕ ПРОЕКТИРОВАНИЕ ЦИФРОВОЙ ЗИС

1 Цель работы: Исследование принципа построения и верификации проекта в пакетах Design Architect-IC, ICstudio и Eldo.

2 Создание и управление проектом (ICstudio)

Общий маршрут схемотехнического проектирования цифровой ЗИС представлен на рисунке 2.1.

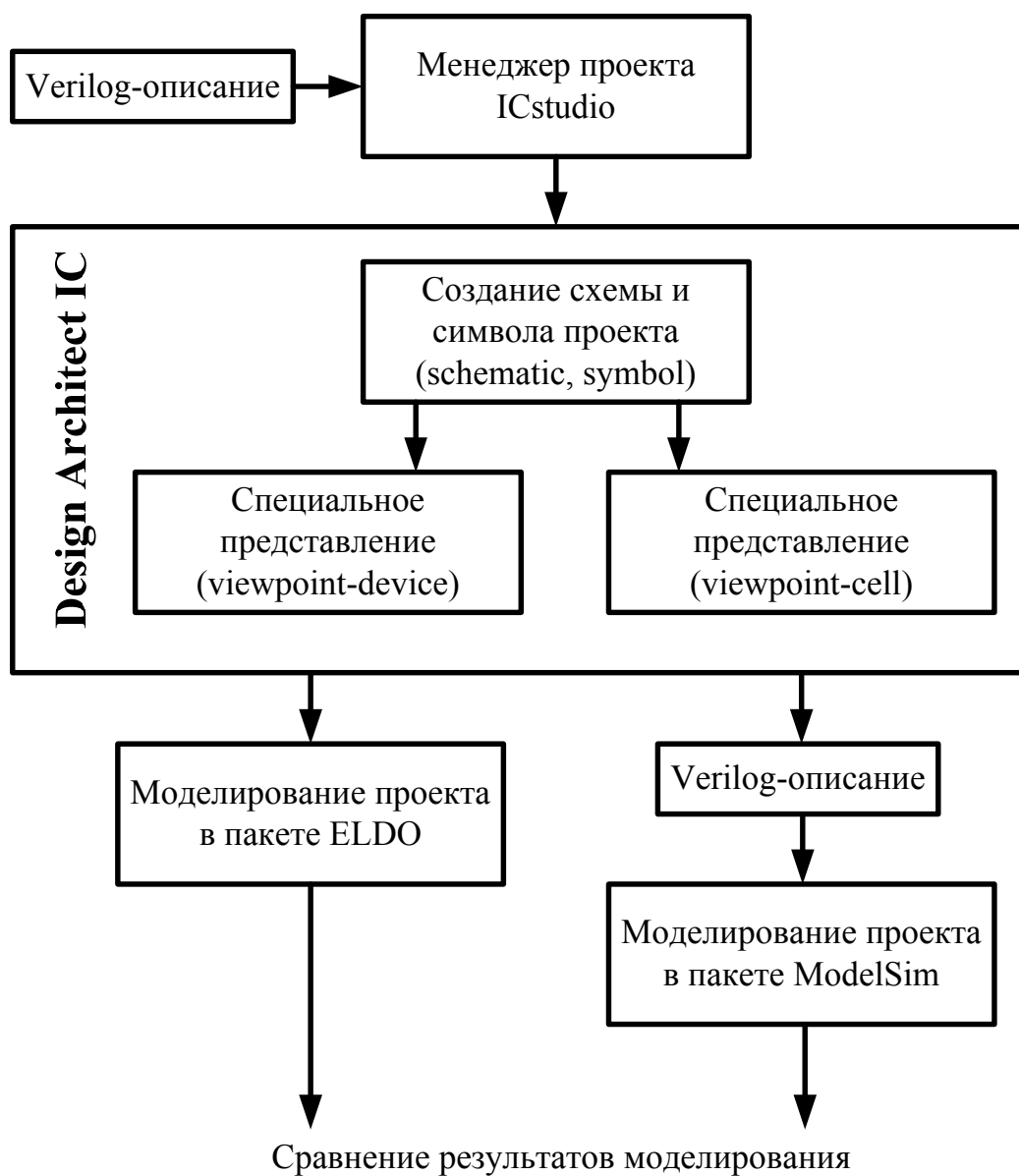


Рис. 2.1. Маршрут схемотехнического проектирования цифровой ЗИС

Создание и управление проектом выполняется в пакете ICstudio работающей под управлением операционной системы UNIX.

Интерфейс главного окна ICstudio представлен на рисунке 2.2.

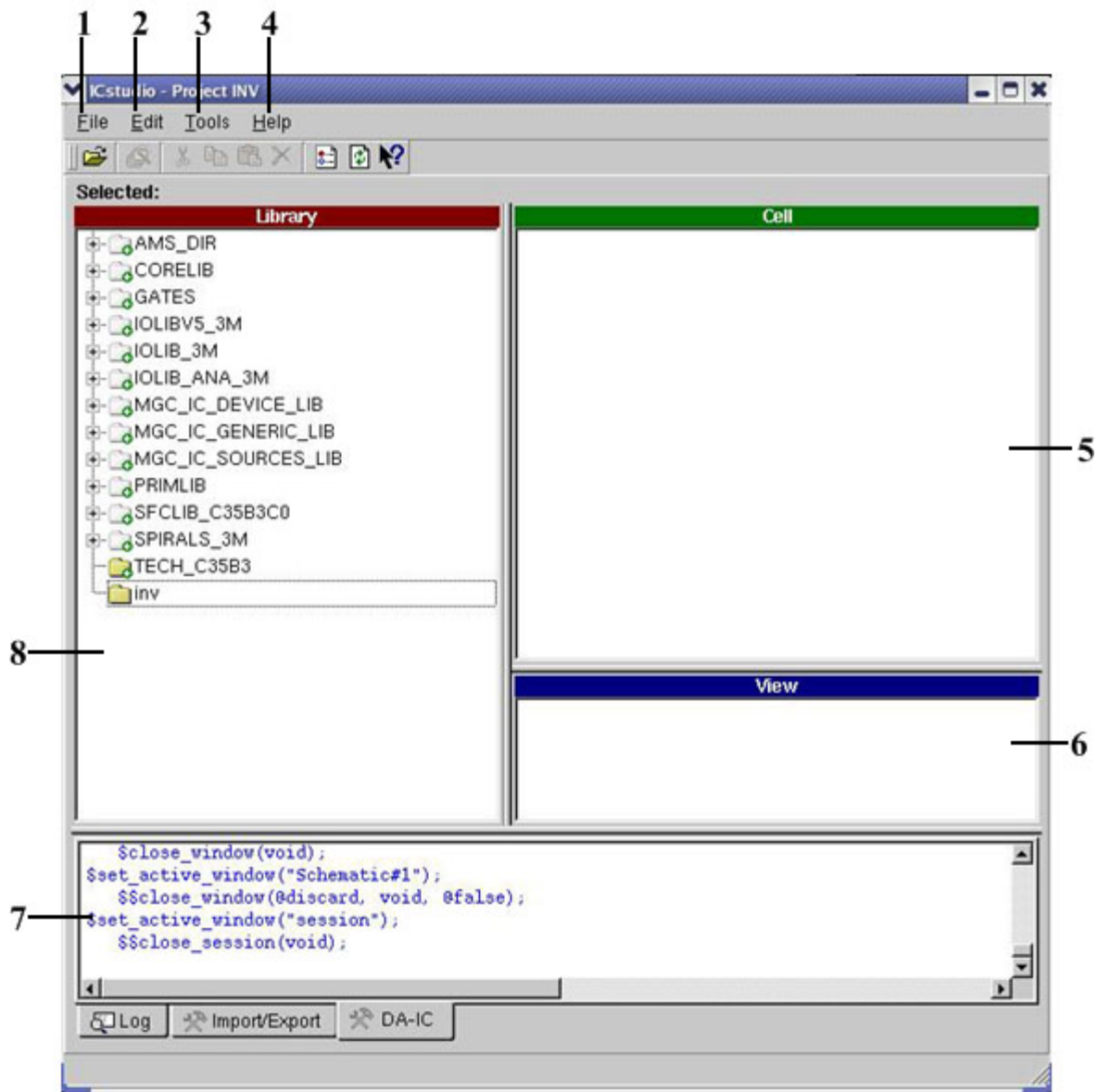


Рис. 2.2. Главное окно ICstudio

- 1 – Создание проектов и библиотек, экспорт/импорт файлов (**File**);
- 2 – Редактирование проекта (**Edit**);
- 3 – Подключение и настройка библиотек (**Tools**);
- 4 – Помощь (**Help**);
- 5 – Список ячеек в библиотеке (**Cell**);
- 6 – Содержание ячейки (**View**);

7 – Окно событий (**Log**);

8 – Список подключенных библиотек (**Library**).

3 Создание электрической – принципиальной схемы проекта (Design Architect-IC)

Создание электрической принципиальной схемы проекта выполняется в пакете Design Architect-IC (DA-IC).

В главном окне DA-IC расположены следующие пункты главного меню (**рис. 2.3**):

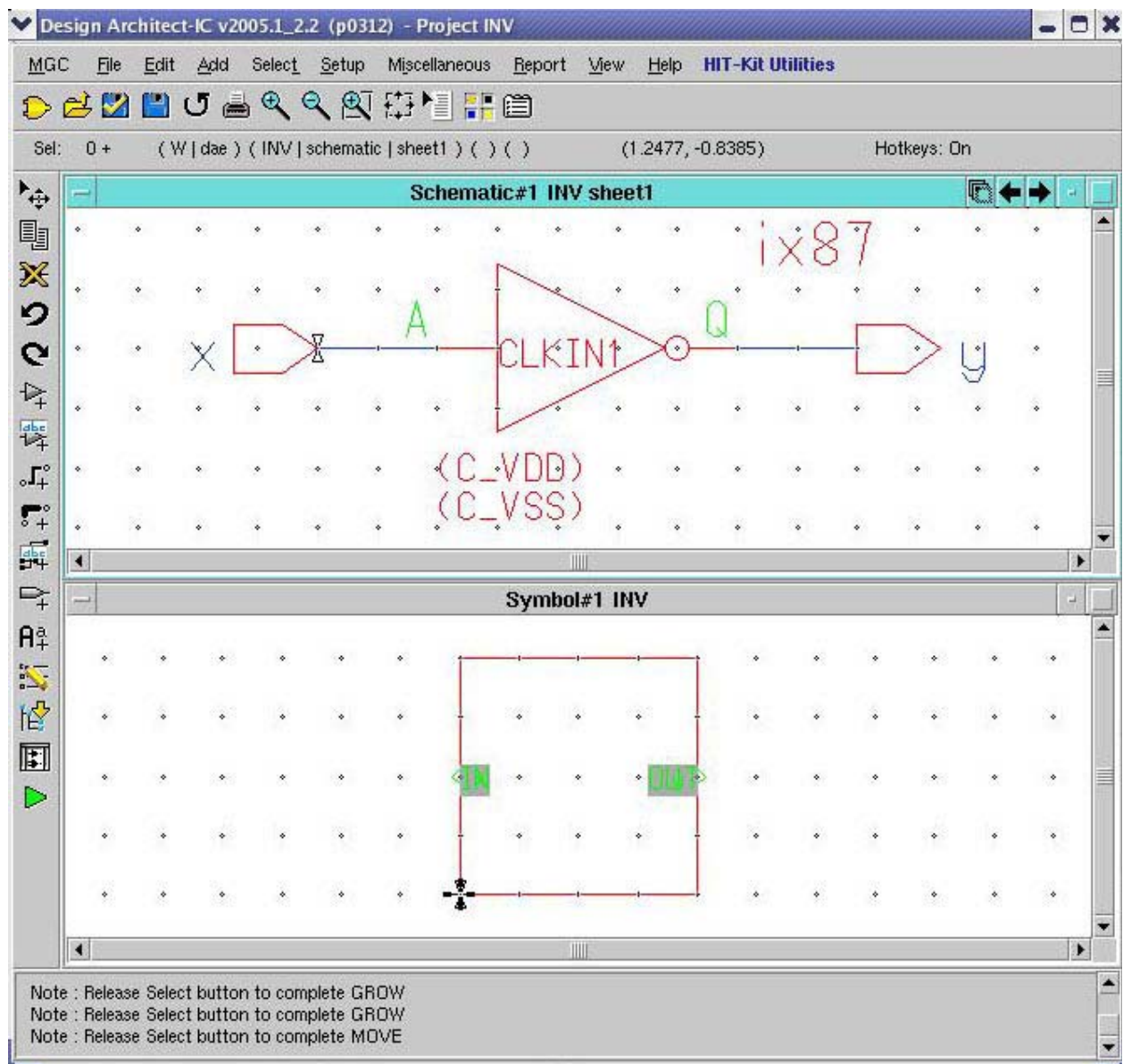















Рис. 2.3. Главное окно Design Architect-IC

- в меню **MGC** можно изменить пользовательские настройки для более удобной работы, задать рабочую директорию, показать используемые библиотеки, вызвать командную строку для отображения результатов DA-IC;
- открытие, сохранение и печать проекта (**File**);
- редактирование проекта (**Edit**);
- пункт меню **Add** служит для добавления различных электрических параметров (**Electrical**), комментариев и графических объектов (**Comment**) а также добавления данных схемы (**Sheet Data**);
- меню **Select** – служит для выбора объектов схемы;
- задания настроек DA-IC (**Setup**).
- выделение элементов схемы, создание символа (**Miscellaneous**);
- создание файла отчета (**Report**);
- выбор режима просмотра проекта (**View**);
- помощь (**Help**);
- библиотека стандартных технологических элементов (**HIT-kit Utilities**).

При работе со схемой в левой части окна DA-IC, располагается панель пиктограмм быстрого запуска:

-  – перемещение элементов схемы;
-  – копирование элементов;
-  ,  – удаление объектов, откат на один шаг назад, шаг вперед;
-  ,  – добавление созданных ранее элементов в схему;
-  ,  ,  ,  – добавление проводника, шины, имени проводника, портов, комментариев;
-  – добавление параметров объекта;
-  – открыть/скрыть панель библиотек;
-  – начало режима моделирования.

Пример выполнения задания. Синтезировать Verilog-описание инвертора в электрическую – принципиальную схему на основе стандартных технологических элементов.

1) Создать проект INV в пакете ICstudio командой в консоли:

```
ams_ics -p INV -t c35b3c0,
```

После этого запустится главное окно ICstudio (рис. 31).

2) Создать новую библиотеку (inv) **File > New > Library**.

3) Импорт Verilog-описания **File > Import > Verilog**, в появившемся окне в строке **Verilog netlist(s)** указать путь к Verilog-описанию инвертора */Project/INV.proj/INV.v*. В строке **Name map file(s)** указать путь к технологическому файлу */Mentor/TDK/mentor/c35/verilogin_cellmapfiles/c35b3_digital.cellmap* и нажать кнопку **Import** (рис. 2.4).

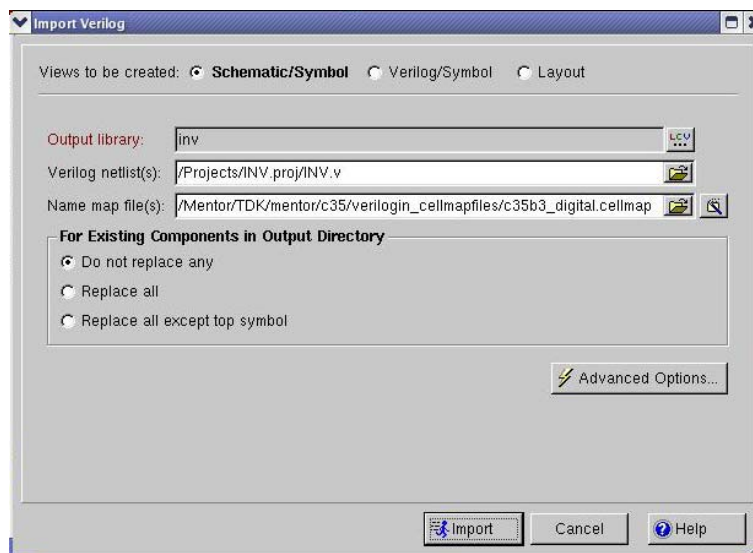


Рис. 2.4. Импорт Verilog-описания инвертора

В окне **Cell** появится новый компонент INV, а в окне **View** схема и символ инвертора. Для запуска DA-IC в окне **View** выбрать схему (Schematic) или символ (Symbol) соответственно (рис. 2.3).

3) Создать тестовую схему для моделирования инвертора. В поле Cell создать новый элемент INV_TestBench **File > New > View**. В появившемся окне (рис. 2.5), в поле **Cell Name** необходимо указать имя нового элемента (INV_TestBench), а в поле **View Type** выбрать Schematic.

В появившемся окне DA-IC необходимо собрать схему для моделирования инвертора (рис. 2.6). В появившемся окне необходимо выполнить следующую последовательность действий:

- добавить в схему символ компонента **INV Add > Instance > Choose Symbol**;
- добавить порты *portin* и *portout* из панели библиотек **Generic Lib**. Для этого выбрать элемент и нажать на клавиатуре клавишу **Q**. В появившемся окне **Edit Object Properties** в поле **Value**, значение **NET** заменить на *IN* и *OUT* соответственно;

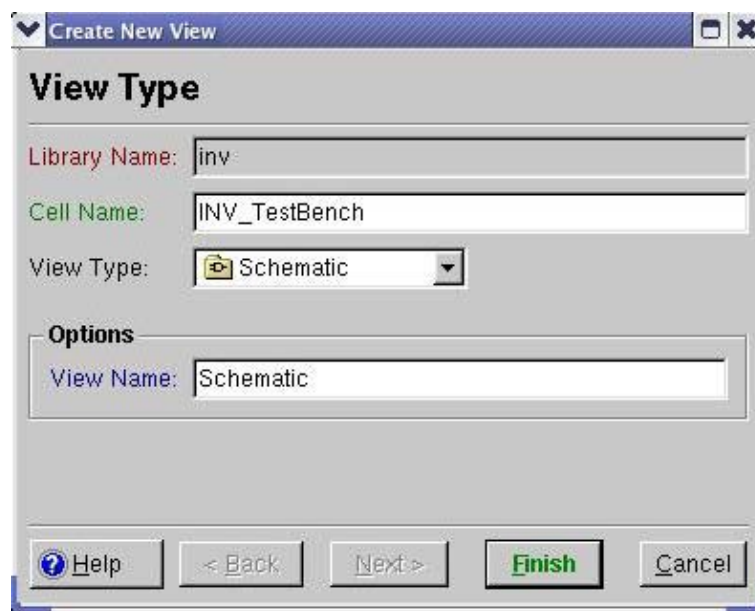


Рис. 2.5. Окно создания нового элемента

- элементы **VDD**, **VSS** и **Ground** выбрать из панели библиотек **Generic Lib**;
- элементы **V1 (DC)**, **V2 (DC)** и **V4(Pulse)** выбрать из панели библиотек **Sources Lib**.
- соединить, как показано на рисунке 2.6;
- для элемента **V1**, установить значение напряжения **3.3В**, в окне **Edit Object Properties** в поле **DC** задать значение **3.3V**;
- для элемента **V2**, установить значение напряжения **0V**;
- для элемента **V4** в окне **Edit Object Properties** установить следующие параметры:

delay 0us

initial_value 0V
 period 100ns
 pulse_value 3.3V
 t_fail 1ns
 t_rise 1ns
 width 49ns,

нажать **Ok**.

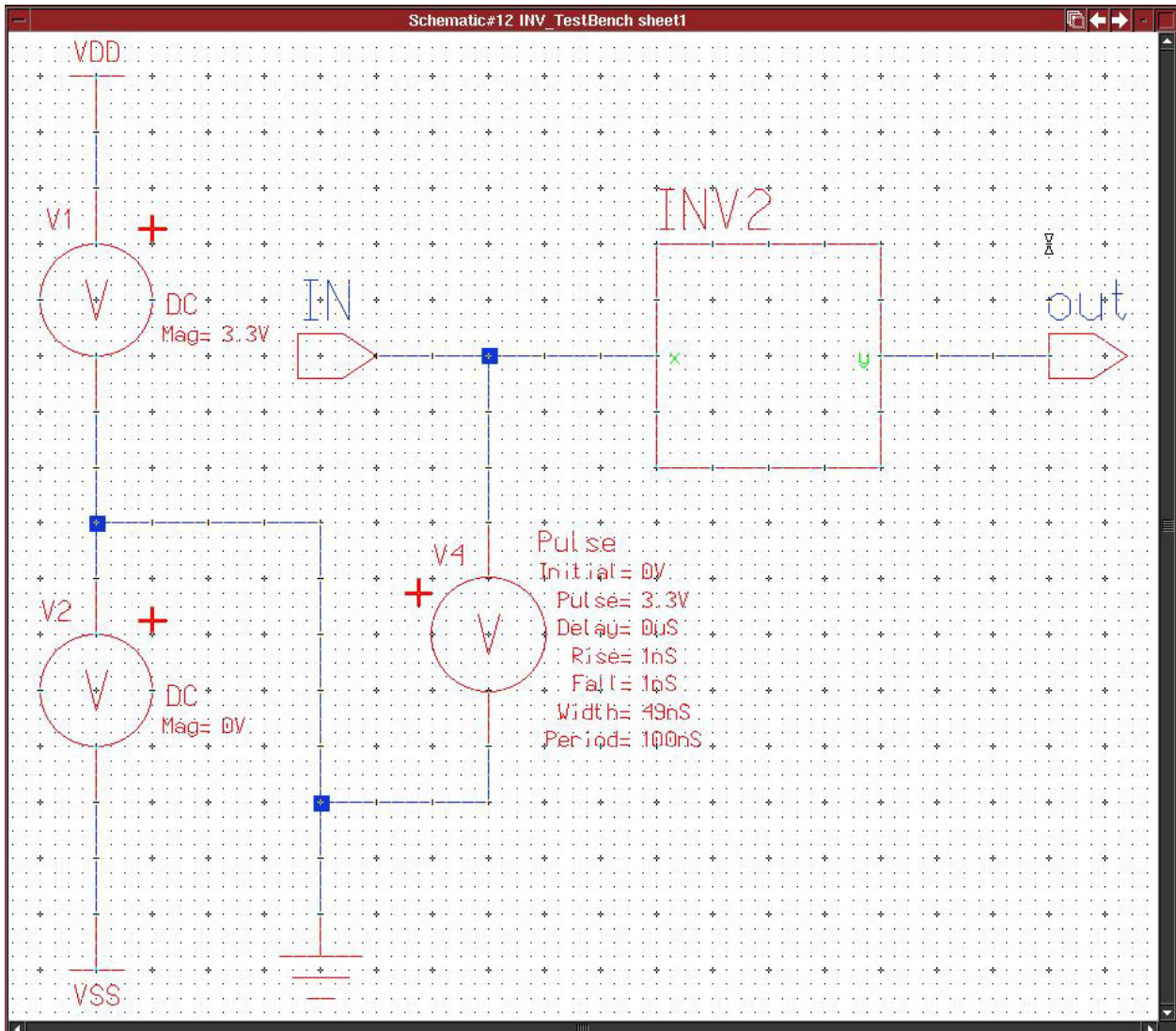


Рис. 2.6. Тестовая схема для моделирования инвертора

4) Для выполнения моделирования необходимо создать специальное представление проекта (viewpoint). В окне DA-IC выбрать **HIT-Kit Utilities > Create Viewpoint**. В появившемся окне (рис. 2.7) в строке **Design Path** с помощью кнопки **Navigator** необходимо выбрать логическое представление схемы инвертора *\$inv/default.group/logic.views/INV_TestBench*, в поле

Technology Name выбрать **C35B3**, а в поле **Viewpoint Level** выбрать **Device** и нажать кнопку **Ok**.

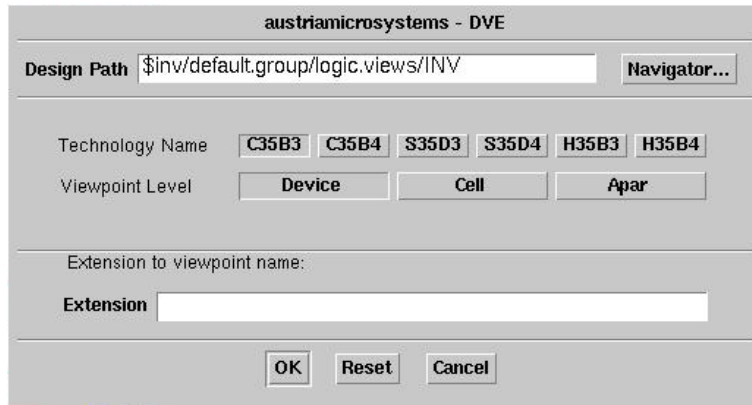


Рис. 2.7. Создание специального представления проекта (viewpoint)

5) Запустить режим моделирования.

- Из панели **Schematic Edit** выбрать **Simulation**. В появившемся окне **Entering Simulation Mode** выбрать `vpt_c35b3_device` и нажать **Ok**;
- В панели **Schematic Sim** нажать **Analyses > Setup Simulation Mode**. Выбрать режим моделирования **Transient** и с помощью кнопки **Setup** установить следующие параметры (рис. 2.8):

Stop Time (TSTOP) 1000N

Max Time Step (HMAX) 1n,

Нажать **Ok**.

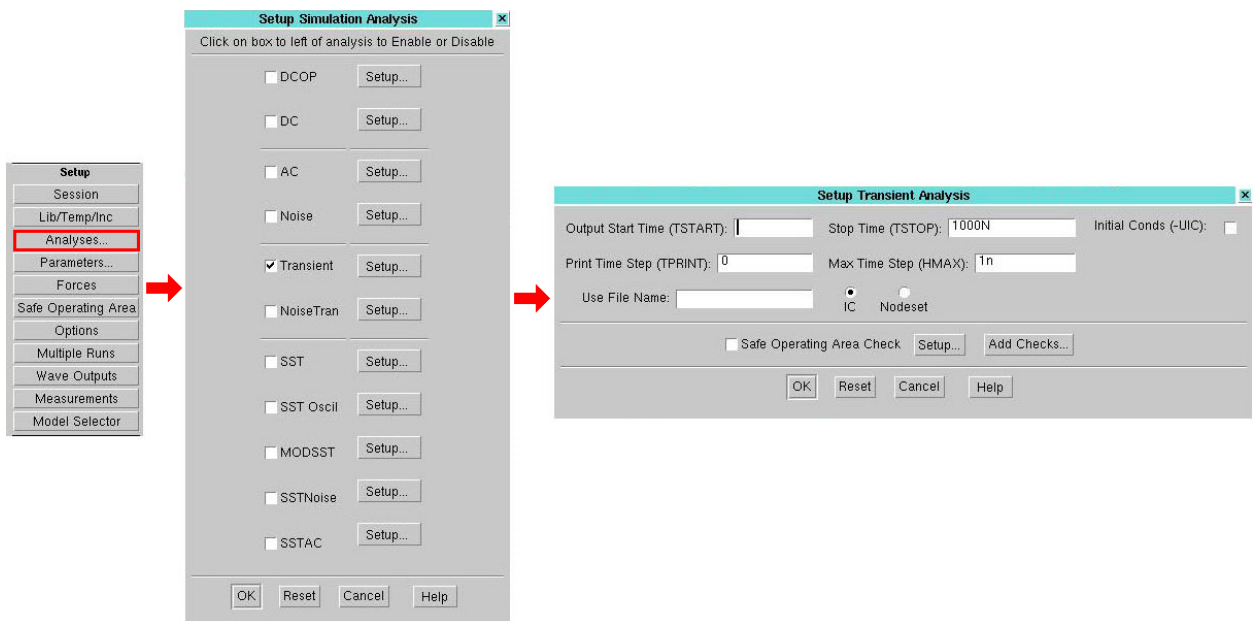


Рис. 2.8. Окно моделирования

- Из панели **Schematic Sim** выбрать **Wave Outputs > Save**. В появившемся окне **Setup Outputs Options** выбрать **Voltages** и нажать **Ok**.
- Выбрать **Schematic Sim > Run Eldo**. Появятся окна **Netlisting design** и **Simulation design** (рис. 2.9). При отсутствии ошибок нажать клавишу **Enter**.

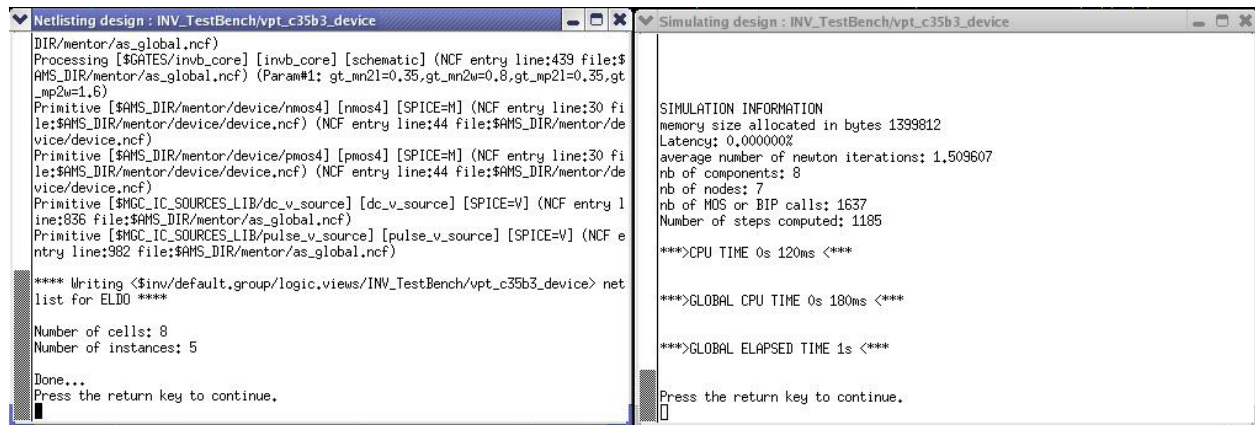


Рис. 2.9. Окна Netlisting design и Simulation design

- В панели **Schematic Sim** выбрать **View Waves > New Window**. В появившемся окне EZwave показаны результаты моделирования (рис. 2.10).
- б) Импортировать Verilog-описание из электрической – принципиальной схемы инвертора (DA-IC) для проверки в пакете ModelSim.
- В окне DA-IC выбрать **HIT-Kit Utilities > Create Viewpoint**. В появившемся окне (рис. 2.7) в строке **Design Path** с помощью кнопки **Navigator** необходимо выбрать логическое представление схемы инвертора *\$inv/default.group/logic.views/INV/*, в поле **Technology Name** выбрать **C35B3**, а в поле **Viewpoint Level** выбрать **Cell** и нажать кнопку **Ok**.
 - Выбрать **HIT-Kit Utilities > Create Verilog** (рис. 2.11) и нажать кнопку **Ok**. Созданное Verilog-описание находится по адресу */Projects/INV.proj/inv.lib/default.group/logic.views/INV/vpt_c35b3_cell/netlist.vrlg*.

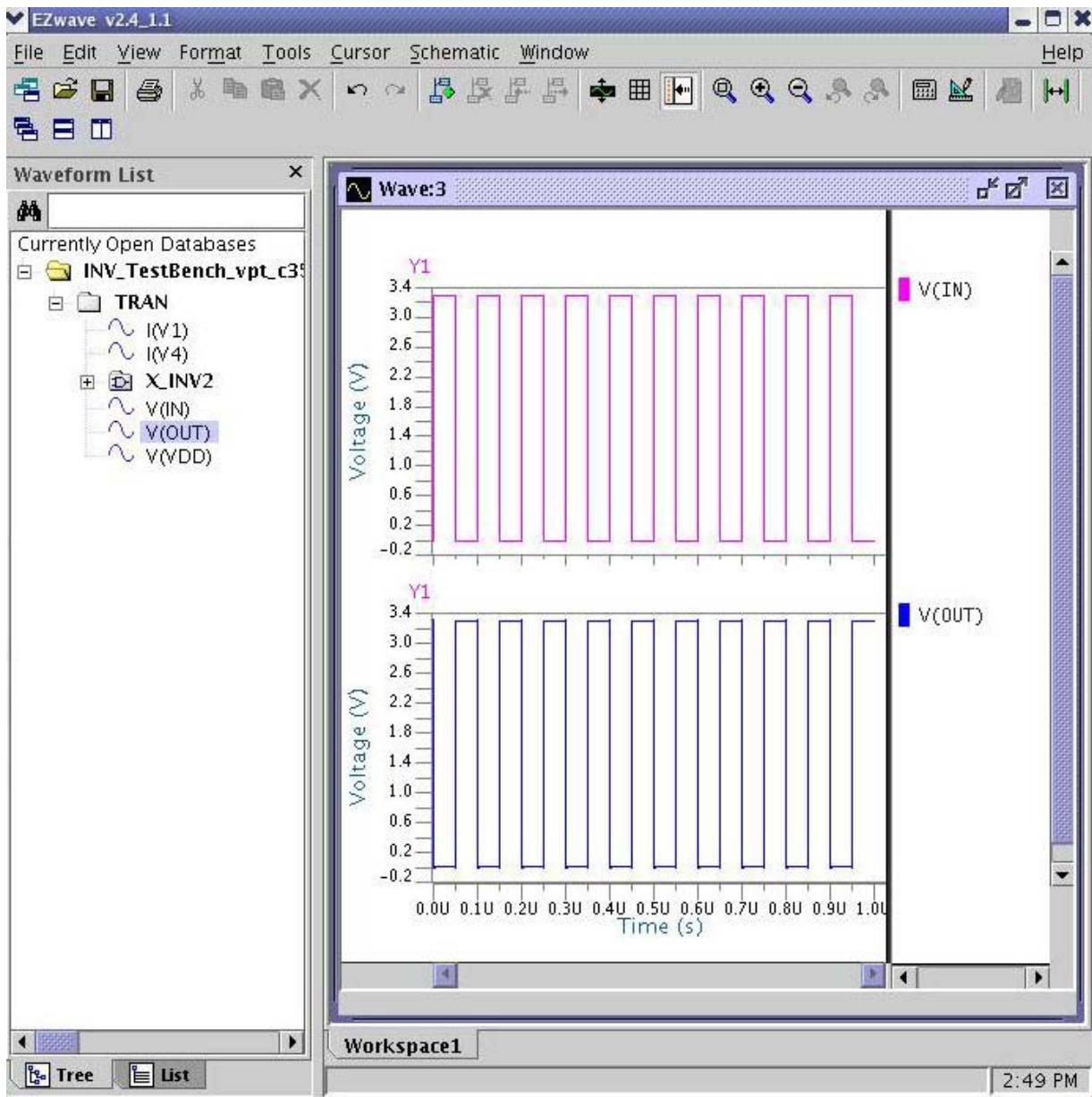


Рис. 2.10. Результаты моделирования

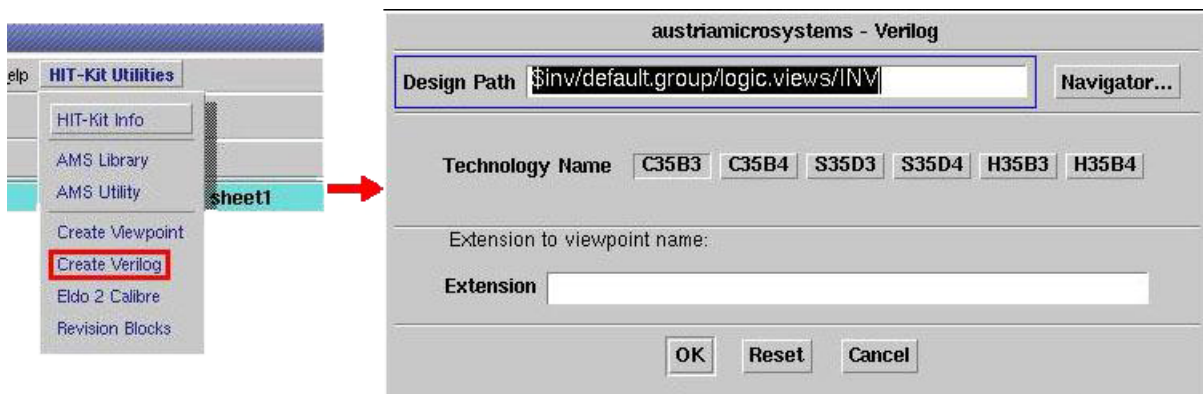


Рис. 2.11. Создание Verilog-описания

- Создать новый проект в пакете ModelSim. Изменить расширение netlist.vrlg на netlist.v и добавить в проект **File > Add to Project > Existing File**. Так же необходимо добавить технологический файл c35_CORELIB.v.
- Создание тестовых векторов «dofile» **File > New > Source > Do**, обеспечивающих определение правил изменения входных сигналов во времени.

```

vsim -t ns INV
restart -force
force -freeze INV/x 0 0, 1 5 -r 10
add wave -binary /*
configure wave -namecolwidth 200
configure wave -valuecolwidth 50
configure wave -justifyvalue left
configure wave -signalnamewidth 0
configure wave -snapdistance 10
configure wave -datasetprefix 0
configure wave -rowmargin 4
configure wave -childrowmargin 2
configure wave -signalnamewidth 2
set IgnoreWarning 1
run 100 ns

```

- Компиляция проекта **Compile > Compile All**.
- Моделирование запускается командой «do dofile» из командной строки (рис. 2.12).

4 Порядок выполнения работы

- 4.1. Ознакомиться с примером построения и верификации инвертора в пакетах Design Architect-IC, ICstudio и Eldo.
- 4.2. Получить вариант индивидуального задания у преподавателя
- 4.3. Запустить программу ICstudio и импортировать Verilog-описание.
- 4.4. Создать тестовую схему для моделирования проекта в DA-IC.
- 4.5. Выполнить моделирования в пакете Eldo.
- 4.6. Импортировать Verilog-описание из электрической – принципиальной схемы проекта в пакете DA-IC.
- 4.7. Запустить программу ModelSim SE 5.8d.exe.
- 4.8. Подключить импортированное Verilog-описание устройства.
- 4.9. Подготовить тестовые векторы «dofile».

4.10. Выполнить моделирование в пакете ModelSim и сравнить с результатами моделирования в пакете Eldo.

4.11. Провести анализ полученных результатов.

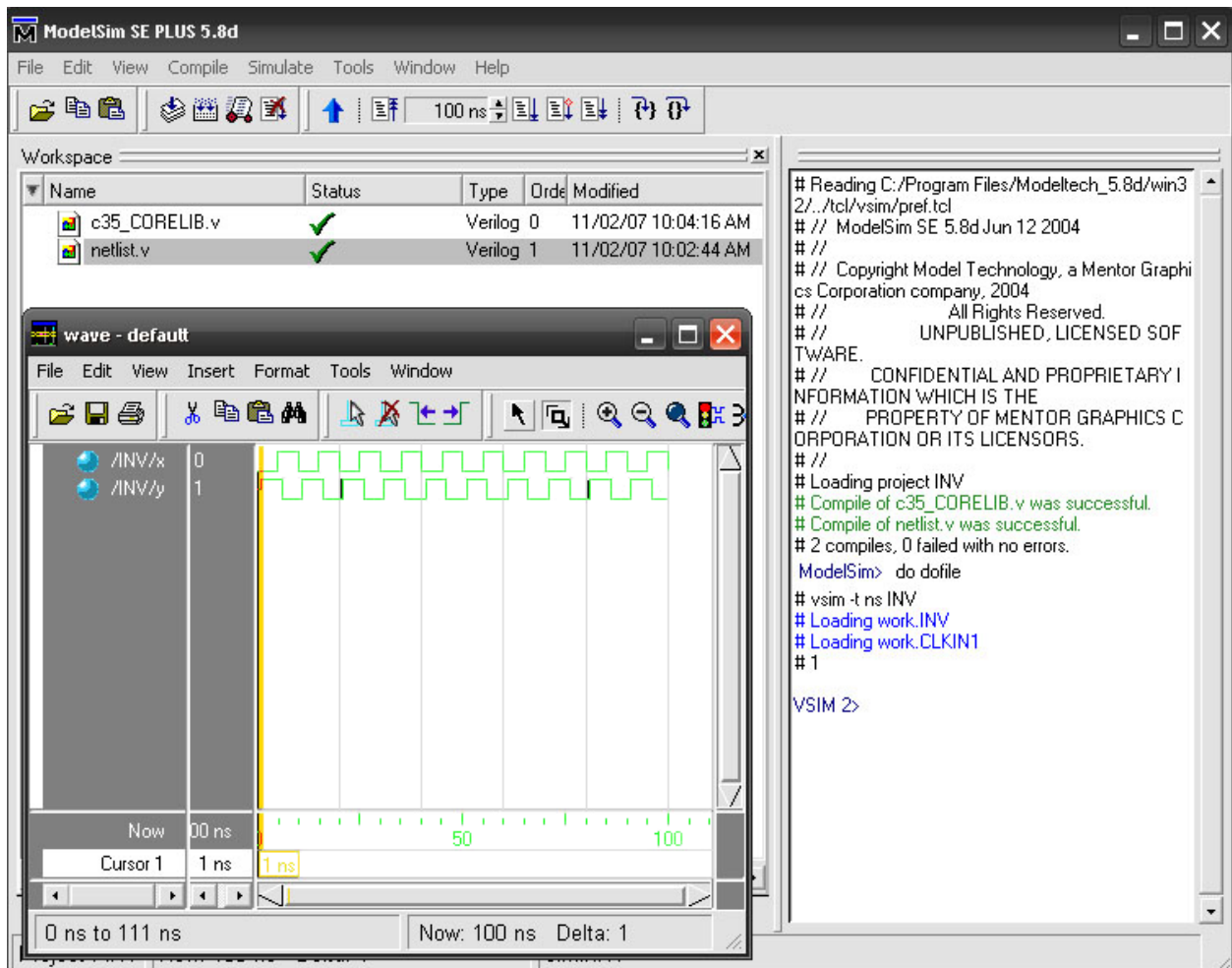


Рис. 2.12. Результаты моделирования Verilog-описания инвертора

5 Содержание отчета

5.1. Цель работы и задание.

5.2. Результаты синтеза в пакете Leonardo Spectrum (Verilog-описание).

5.3. Схема электрическая – принципиальная проекта в пакете DA-IC.

5.4. Тестовая схема для моделирования.

5.5. Результаты Netlisting design и Simulation design.

- 5.6. Результаты моделирования в пакете Eldo.
- 5.7. Verilog-описание полученное из DA-IC.
- 5.8. Результаты моделирования в пакете ModelSim.
- 5.9. Анализ полученных результатов и выводы по работе.

6 Варианты индивидуальных заданий

Вариант №1. Регистр.

Вариант №2. Трехразрядный сумматор на элементах исключающее ИЛИ, И-НЕ.

Вариант №3. Трехразрядный сумматор на элементах И-НЕ.

Вариант №4. Дешифратор на элементах И-НЕ.

Вариант №5. Дешифратор на элементах ИЛИ-НЕ.

7 Контрольные вопросы и задания

- 7.1. Опишите назначение и интерфейс пакета ICstudio;
- 7.2. Каково назначение и интерфейс пакета DA-IC?
- 7.3. Для чего нужны специальные представления проекта Device и Cell?
- 7.4. Для чего необходим пакет Eldo;
- 7.5. Как влияют элементы технологической библиотеки на выходные характеристики?

ЧАСТЬ II. ПРОЕКТИРОВАНИЕ ЗАКАЗНЫХ ИНТЕГРАЛЬНЫХ СХЕМ В СРЕДЕ САПР MENTOR GRAPHICS

Лабораторная работа № 3 ТОПОЛОГИЧЕСКОЕ ПРОЕКТИРОВАНИЕ ЦИФРОВОЙ ЗИС

1 Цель работы: Изучение средств топологического проектирования в пакете IC-Station.

2 Проектирование топологии

Общий маршрут топологического проектирования ЗИС показан на рисунке 3.1.

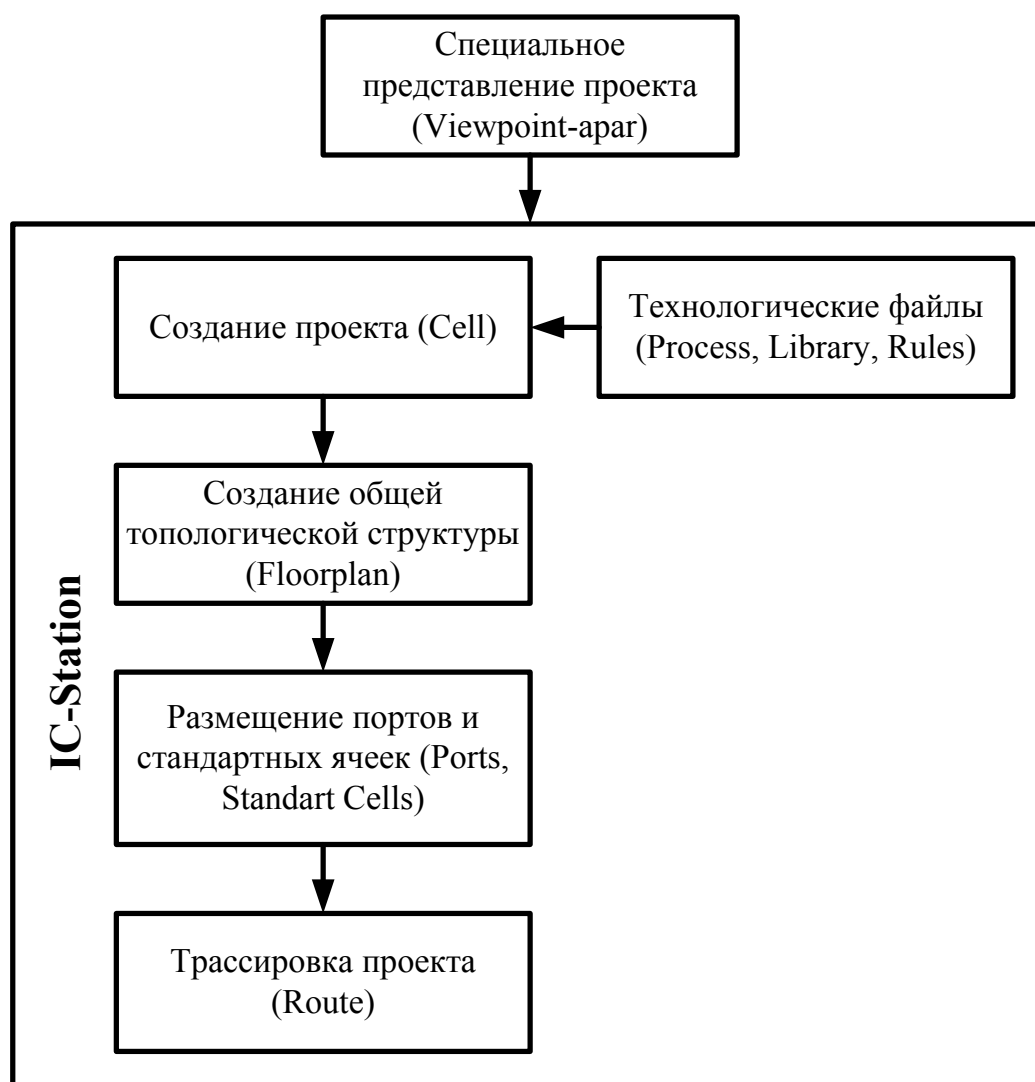


Рис. 3.1. Общий маршрут топологического проектирования ЗИС

Топология – это аналог электрической схемы в виде набора геометрических образов слоев кристалла. В каждом слое имеется определенное число фигур, расположенных относительно друг друга. Комбинация определенных фигур соответствует элементу электрической схемы (рис. 3.2).

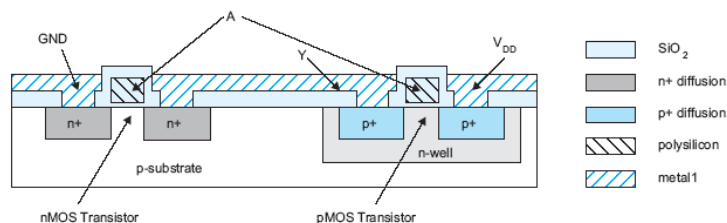


Рис. 3.2. Сечение инвертора

На рисунке 3.2 в разрезе показано устройство инвертора. Инвертор выполнен на подложке р-типа. Транзистору р-МОП необходима подложка n-типа, поэтому в существующую подложку встраивается «карман» n-типа. При проектировании КМОП технологии, возможно использование и подложки n-типа с р-карманами, которые содержат n-МОП транзисторы. Транзистор n-МОП имеет для истока и стока область n-типа и поликристаллический затвор из диоксида кремния. У транзистора р-МОП – похожая структура с р-типом истока и областями стока. Поликристаллический затвор этих двух транзисторов связан вместе, образуя вход А. Исток n-МОП транзистора соединен с металлической шиной земли (GND), а исток р-МОП транзистора соединен с металлической шиной питания (VDD). Стоки этих двух транзисторов соединены, образуя выход Y.

Толстый уровень SiO₂ называют защитным слоем оксида, предотвращающий от закорачивания металлический слой 1 с другими металлическими слоями, кроме тех мест, где эти контакты необходимы.

Технологические нормы для технологии КМОП 0,35мкм показаны в таблице 3.1.

3 Проектирование топологии в пакете IC-Station

Пакет IC-Station – комплексное решение в области разработки топологии заказных ИС. В состав пакета входит топологический редактор ICgraph SDL, включающий как интерактивный топологический редактор ICgraph Basic, так и автоматический генератор топологии из принципиальной схе-

мы, а также специализированные генераторы топологии цифровых и аналоговых устройств – ICdevice Digital и ICdevice Analog.

Таблица 3.1. Технологические нормы для технологии КМОП 0,35мкм

Слой металлизации	Минимальная толщина проводника (мкм)	Минимальный зазор между слоями металлизации (мкм)
Первый слой металлизации (Metal 1)	0,5	0,45
Второй слой металлизации (Metal 2)	0,6	0,5
Третий слой металлизации (Metal 3)	0,6	0,6

В главном окне IC-Station расположены следующие пункты главного меню (рис. 3.3):

- в меню **MGC** можно изменить пользовательские настройки для более удобной работы, задать рабочую директорию, показать используемые библиотеки, вызвать командную строку для результатов IC-Station;
- открытие схемы и топологии, просмотр иерархии проекта, установка технологических файлов (**File**);
- просмотр топологии по уровням иерархии проекта, изменение технологических настроек (**Context**);
- добавление, удаление и проверка элементов топологии (**Objects**);
- редактирование элементов (**Edit**);
- выбор элементов (**Select**);
- добавление и редактирование связей и портов (**Connectivity**);
- режим трассировки (**Routing**);
- изменение пользовательских настроек (**Setup**);
- создание файлов отчетов (**Report**);
- подключение «горячих клавиш», панели слоев, установка шага сетки (**Other**);
- выбор режима просмотра проекта (**View**);

- проверка наличия коротких замыканий и правил технологических норм (**Checking**);
- чтение/запись файла GDSII (**Translate**);
- авторазмещение, автотрассировка элементов, настройка средств DRC и LVS анализа (**Packages**);
- запуск средств верификации - DRC, LVS, PEX и RVE (**Colibre**);
- помощь (**Help**);
- библиотека стандартных технологических элементов (**HIT-kit Utilities**);
- показать/скрыть панель слоев (**Show LP/Hide LP**).

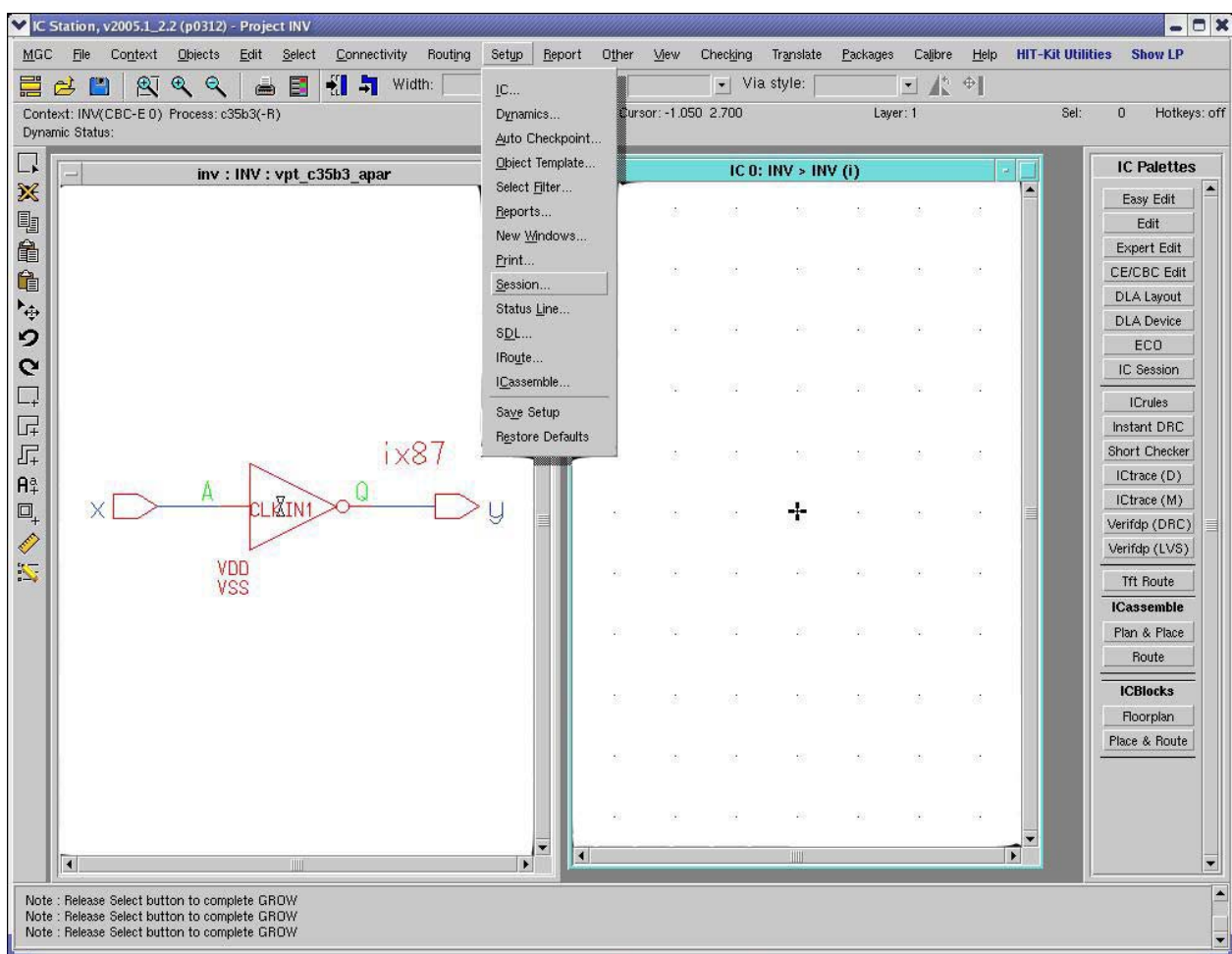


Рис. 3.3. Окно пакета IC-Station

Пример выполнения задания. Выполнить топологическое проектирование инвертора в пакете IC-Station.

1) Для создания топологии инвертора необходимо создать специальное представление проекта (viewpoint). В окне DA-IC выбрать пункт меню **HIT-Kit Utilities > Create Viewpoint**. В появившемся окне (рис. 3.4) в строке **Design Path** с помощью кнопки **Navigator** необходимо выбрать логическое представление схемы инвертора *\$inv/default.group/logic.views/INV_TestBench/*, в поле **Technology Name** выбрать **C35B3**, а в поле **Viewpoint Level** выбрать **Apar** и нажать кнопку **Ok**.

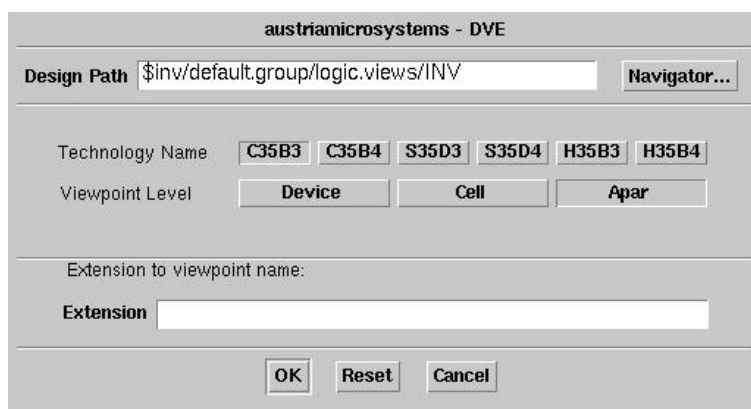


Рис. 3.4. Создание специального представления проекта (viewpoint)

2) В окне ICstudio в поле **View** необходимо создать новое представление для элемента **INV File > New > View** в появившемся окне **View Type** (рис. 3.5, а), в строке **Cell Name** будет указано имя элемента по умолчанию (INV), в поле **View Type** выбрать **Layout** и нажать кнопку **Next**. В окне **Cell Type** (рис. 3.5, б) в поле **Connectivity Source** необходимо выбрать **vpt_c35b3_apar** и нажать **Next**. В появившемся окне **Connectivity Loading Options** (рис.3.5, в) необходимо выбрать **Automatically create unplaced instances in layout** и нажать кнопку **Finish**, в результате автоматически запустится редактор топологии IC-Station.

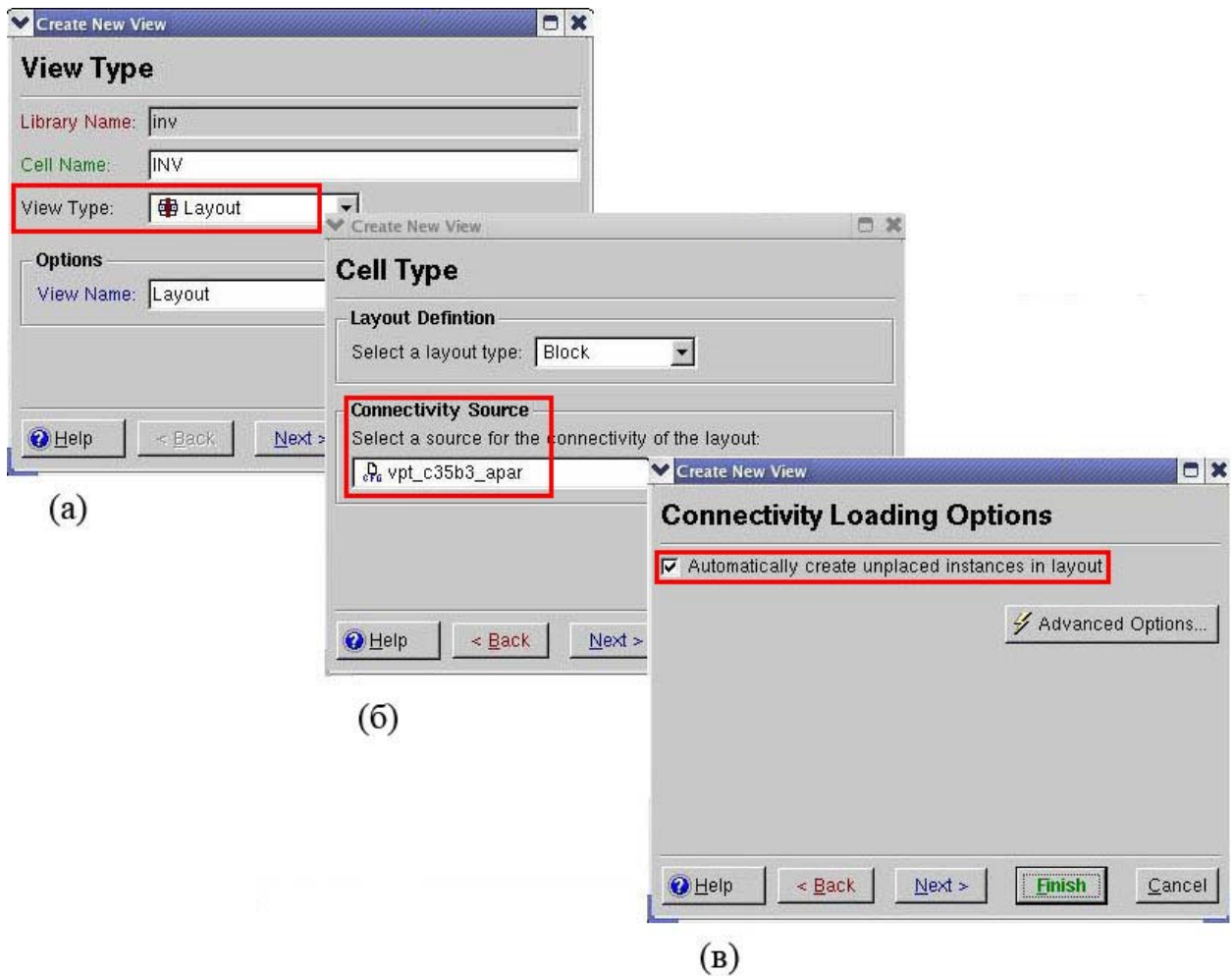


Рис. 3.5. Создания нового представления инвертора

3) В главном меню пакета IC-Station подключить «горячие» кнопки **Other > Hotkeys > Load**. В появившемся окне **Load Hotkey Settings** выбрать IC-Station и нажать **Ok** (рис. 3.6).

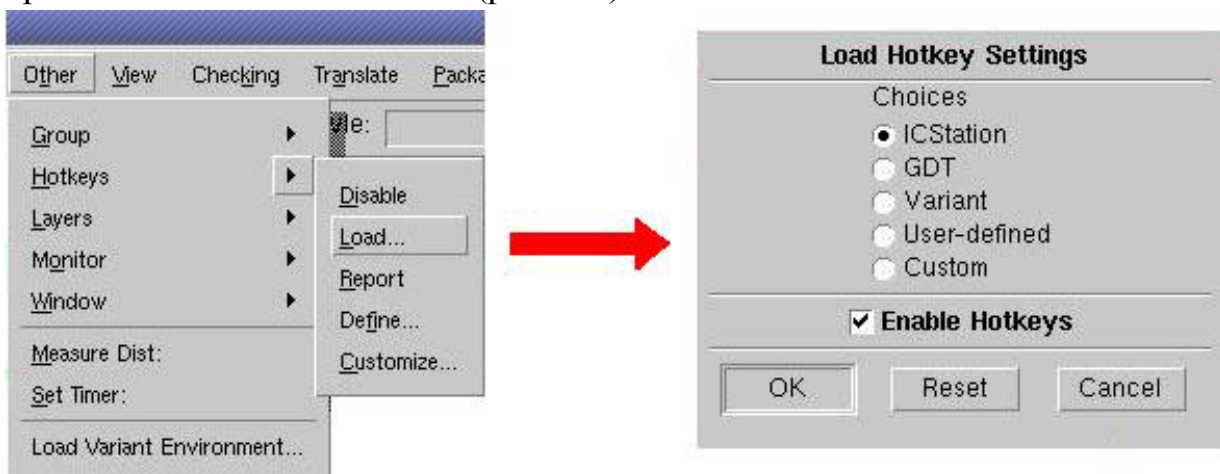


Рис. 3.6. Подключение «горячих» кнопок

4) В главном меню пакета IC-Station настроить рабочую область **Setup > Session >...** (рис.3.3).

5) Настроить общую топологическую структуру. Из панели IC Palettes выбрать **Floorplan > AutoFP**, оставить настройки по умолчанию и нажать **Ok** (рис. 3.7).

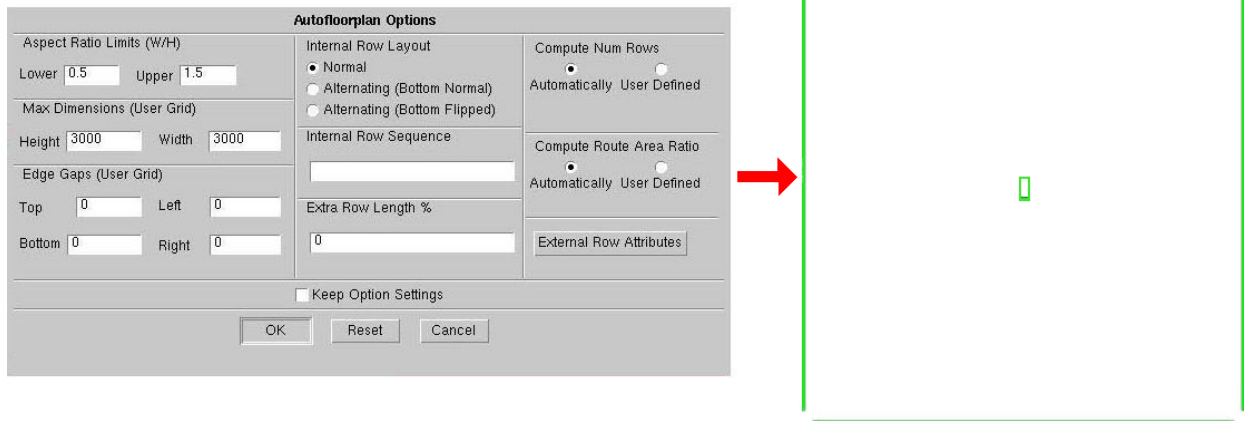


Рис. 3.7. Настройка общей топологической структуры

6) Разместить элементы. Из панели IC Palettes выбрать **Place & Route > StdCell**. В появившемся окне **AUTOPLA ST C** нажать **Ok** (рис. 3.8).

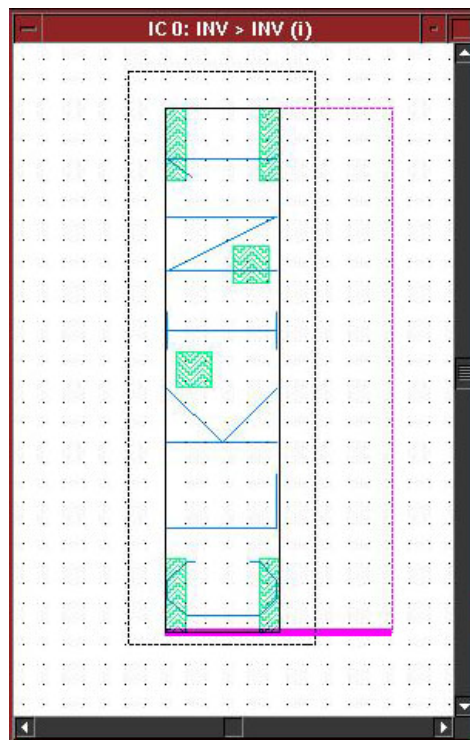


Рис. 3.8. Автоматическое размещение элементов в общей топологической структуре

7) Разместить порты. Из панели **IC Palettes** выбрать **Place & Route > Ports**. В появившемся окне **Autoplace Ports** в поле **Left/Right Level** выбрать **MET2**, остальные настройки оставить по умолчанию и нажать **Ok** (рис. 3.9).

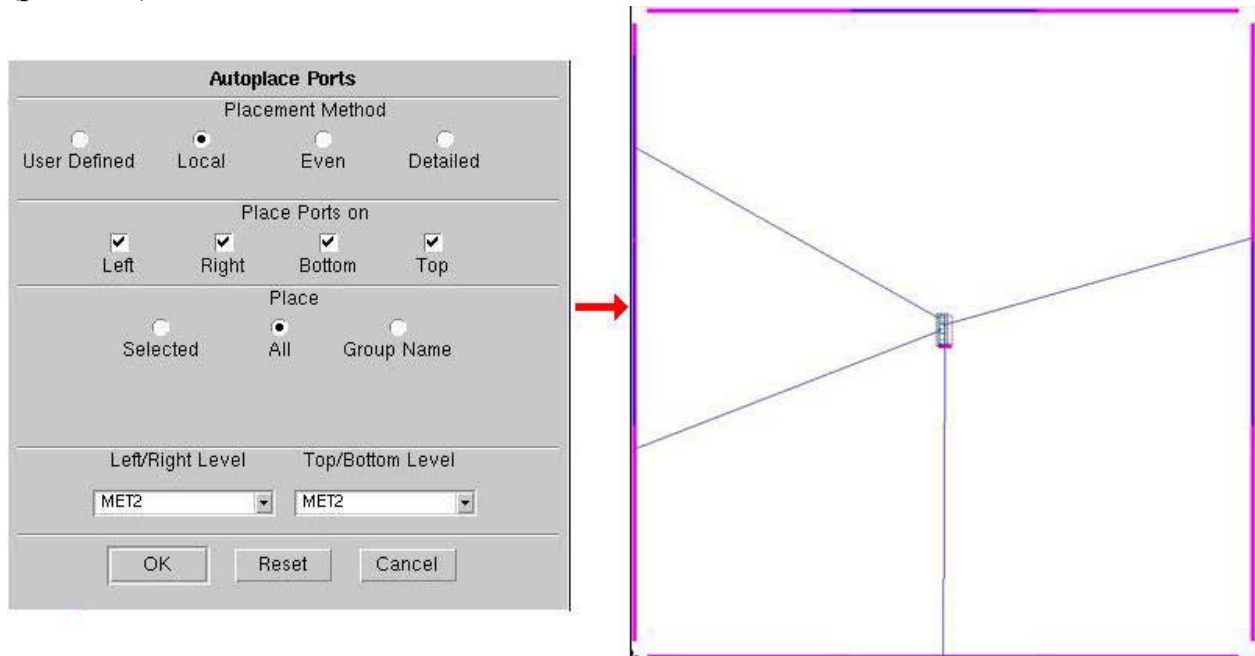


Рис. 3.9. Автоматическое размещение портов в общей топологической структуре

Удалить «ненужные» порты. Из главного меню пакета IC-Station выбрать **File > Open > Group Window**. В появившемся окне **Group Window for INV in IC 0** выбрать вкладку **PORTS**. В окне **GROUP MEMBERS** с помощью кнопки **Ctrl** на клавиатуре выбрать все порты, кроме X, Y, VDD и VSS (рис. 3.10). Правой кнопкой мыши или из панели **IC Groups** выбрать **Select**. В окне топологии выделенные порты будут подсвечены, их необходимо удалить с помощью клавиши **Delete** на клавиатуре.

8) Выполнить трассировку. Для редактирования топологии необходимо использовать панель **IC Palettes > Edit** (рис. 3.3) или «горячие» клавиши:

- группа кнопок **Select** служат для выбора элементов топологии;
- группа **Unselect** необходима для снятия выбора элементов;
- группа кнопок **Add**:
 - а) добавить проводник, используя кнопку **PAT**. В появившемся окне **ADD PA**, выбираем **Options** (рис. 3.11). В строке **Or Type In** выбрать слой металлизации MET2. В строке **Width** задать толщину проводника 0,5мкм и нажать **Ok**;

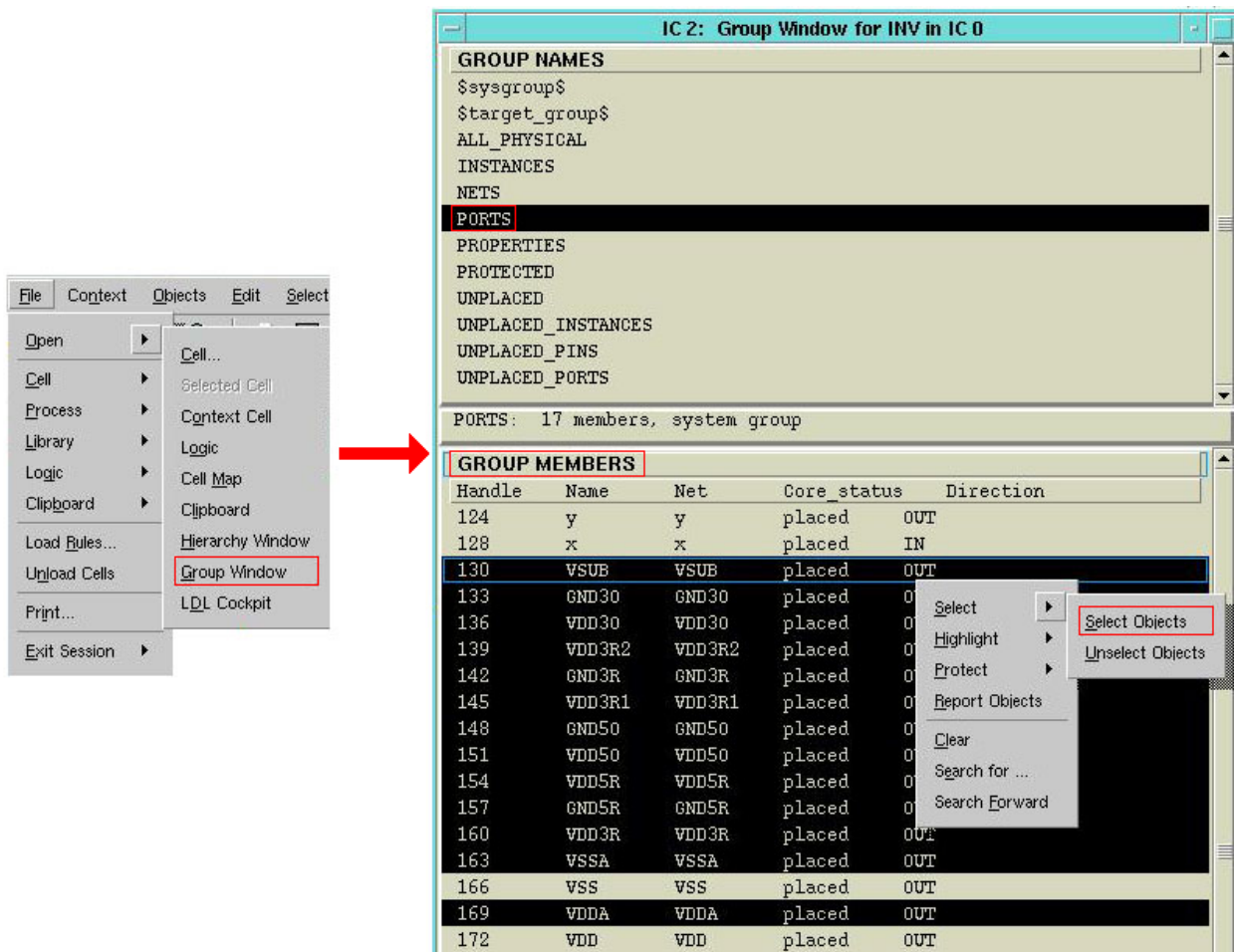


Рис. 3.10. Окно Group Window

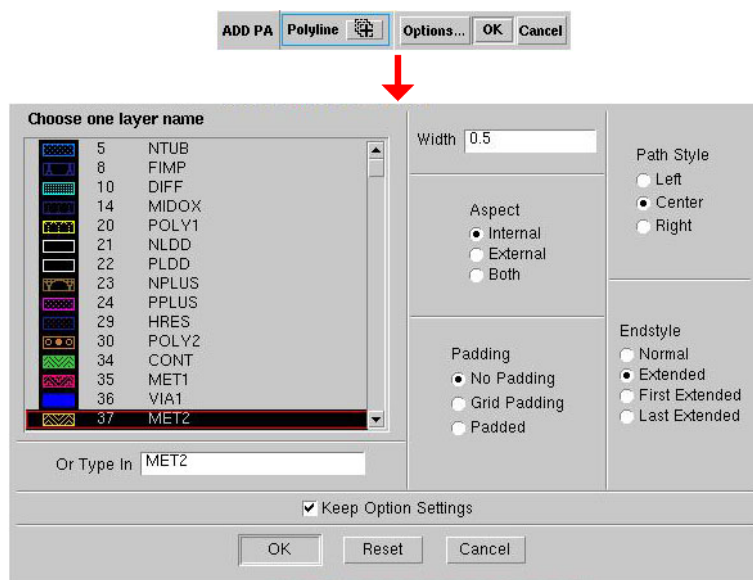


Рис. 3.11. Добавление и настройка проводника

- b) добавить переходные отверстия, используя кнопку **Via > Active Via**. Выбрать тип переходного отверстия и нажать **Ok** (рис. 3.12):
- переходное отверстие с первого слоя металлизации MET1 на второй MET2 – VIA1_C;
 - переходное отверстие со второго слоя металлизации MET2 на третий MET3 – VIA2_C.

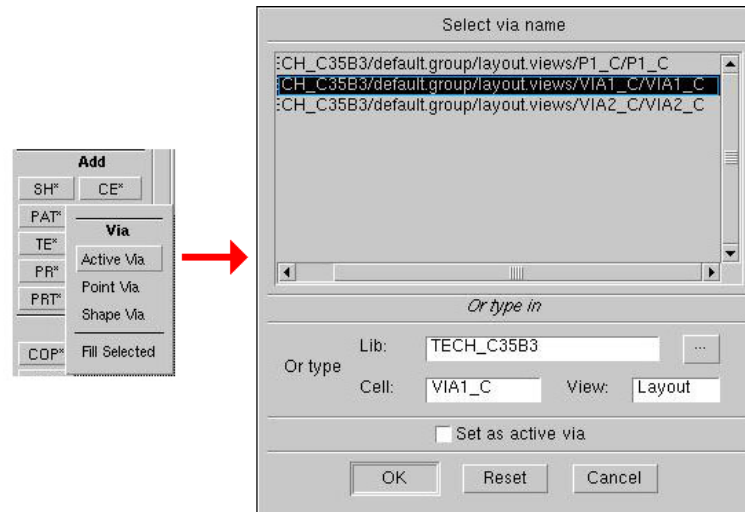


Рис. 3.12. Добавление переходных отверстий

Для добавления переходных отверстий заданной площади (обычно делается для цепей VDD, VSS) нажать **Via > Shape Via** и выбрать тип переходного отверстия и нажать **Ok** (рис. 3.13):

- переходное отверстие с первого слоя металлизации m1 на второй m2 – m1m2;
- переходное отверстие со второго слоя металлизации m2 на третий m3 – m2m3.

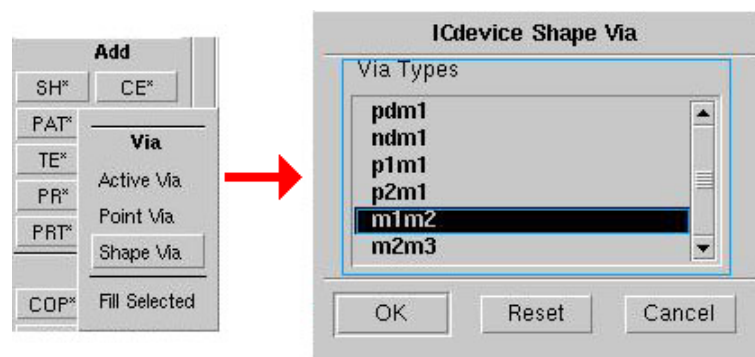


Рис. 3.13. Добавление переходных отверстий заданной площади

- с) С помощью кнопки **TE** можно добавить текстовые обозначения элементов (рис. 3.14);

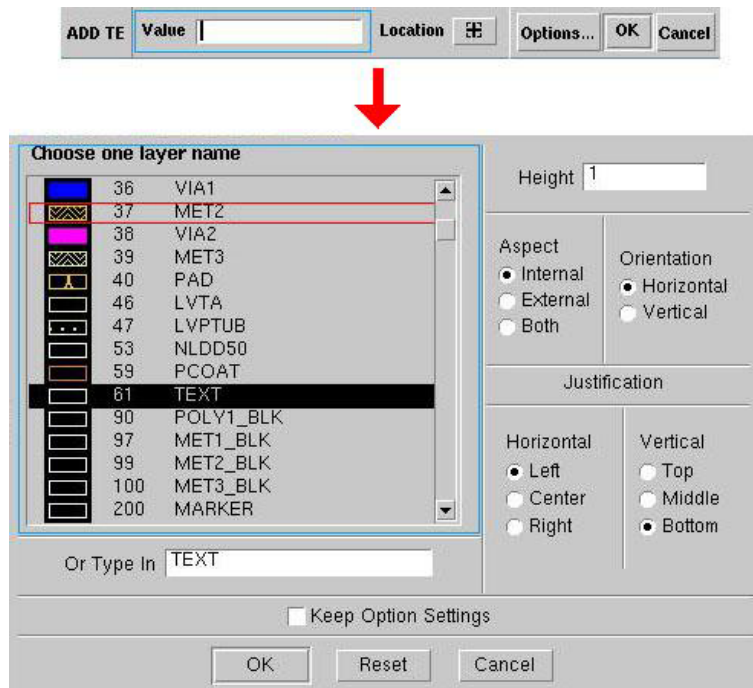


Рис. 3.14. Добавление текстовых обозначений элементов

- Группа **Edit**:
 - а) копирование элементов **COP**;
 - б) зеркальное отображение **FLI**;
 - в) модификация элементов **MOD**;
 - г) перемещение элементов **MOV**;
 - д) измерение расстояния **MEAS**;
 - е) возвращение к предыдущей команде **UNDO**;
 - ж) удаление **DEL**;
 - з) вращение **ROT**;
 - и) отсечение по плоскости **SLI**;
 - к) удлинение/укорачивание элементов **STR**;
 - л) автовыделение **ALGN**;
 - м) включение/выключение защиты от изменений выделенных элементов **PRO/UNPRO**;
 - н) задание шага сетки.

После трассировки всех элементов открыть все слои топологии. В главном меню IC-Station выбрать **Context > Hierarchy > Peek Area** (рис. 3.15).

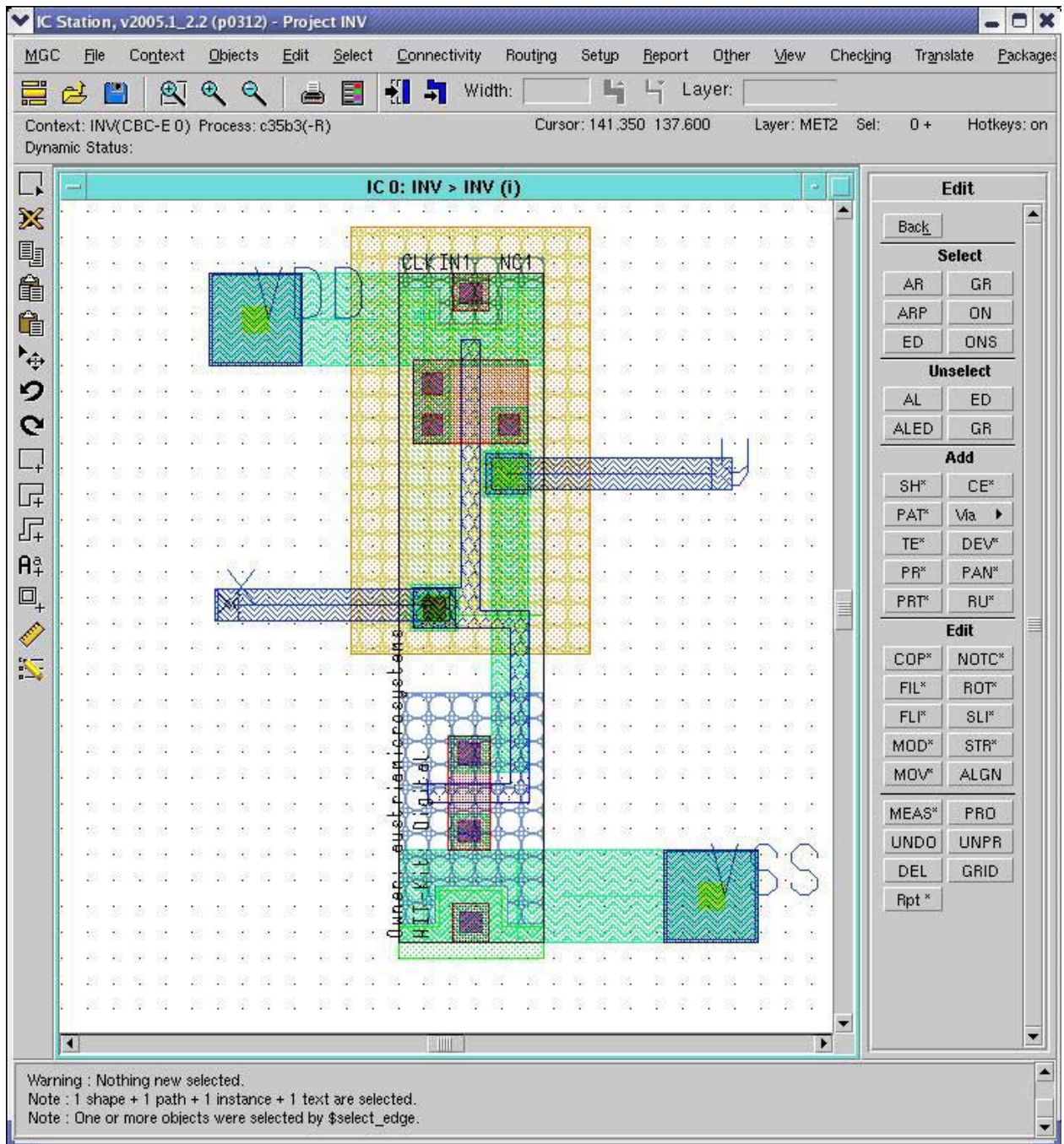


Рис. 3.15. Топология инвертора

9) Проверка на «короткие замыкания». В главном меню IC-Station выбрать **Checking > Shorts – All Nets**, в появившемся окне

\$scheck_shorts_all нажать **Ok**. В окне сообщений появится информация о количестве ошибок (рис. 3.16). При наличии ошибок из главного меню IC-Station выбрать **Checking > First/Next Error** и исправить их.

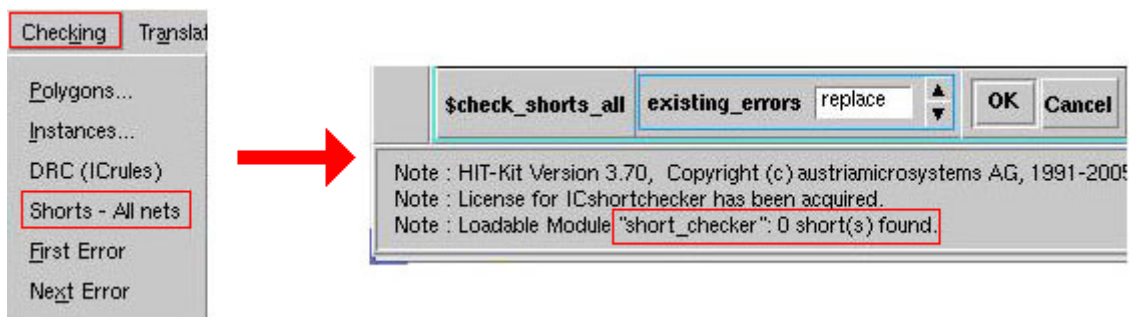


Рис. 3.16. Проверка на «короткие замыкания»

10) Проставить обозначения портов. Из главного меню выбрать **Connectivity > Port > Add Text On Ports**.

4 Порядок выполнения работы

4.1. Изучить технологические нормы для технологии КМОП 0,35мкм.

4.2. Ознакомиться с примером топологического проектирования инвертора в пакете IC-Station.

4.3. Получить вариант индивидуального задания у преподавателя.

4.4. Запустить пакет DA-IC и создать контрольную точку проекта.

4.5. Запустить редактор топологии IC-Station из пакета ICstudio.

4.6. Выполнить топологическое проектирование индивидуального задания.

4.7. Провести анализ полученных результатов.

5 Содержание отчета

5.1. Цель работы и задание.

5.2. Топология проекта.

5.3. Анализ полученных результатов и выводы по работе.

6 Варианты индивидуальных заданий

Вариант №1. Регистр.

Вариант №2. Трехразрядный сумматор на элементах исключающее ИЛИ, И-НЕ.

Вариант №3. Трехразрядный сумматор на элементах И-НЕ.

Вариант №4. Дешифратор на элементах И-НЕ.

Вариант №5. Дешифратор на элементах ИЛИ-НЕ.

7 Контрольные вопросы и задания

7.1. Опишите назначение и интерфейс пакета IC-Station.

7.2. Для чего нужны специальные представления проекта Device и Apar?

7.3. Какие технологические нормы применимы для технологии КМОП 0,35мкм?

Лабораторная работа № 4 ВЕРИФИКАЦИЯ ТОПОЛОГИИ ЗИС

1 Цель работы: Изучение правил проверки и верификации проекта средствами DRC (Design Rule Check) и LVS (Layout Versus Schematic) анализа.

2 Физическая верификация

Маршрут верификации топологии проекта показан на рисунке 4.1.

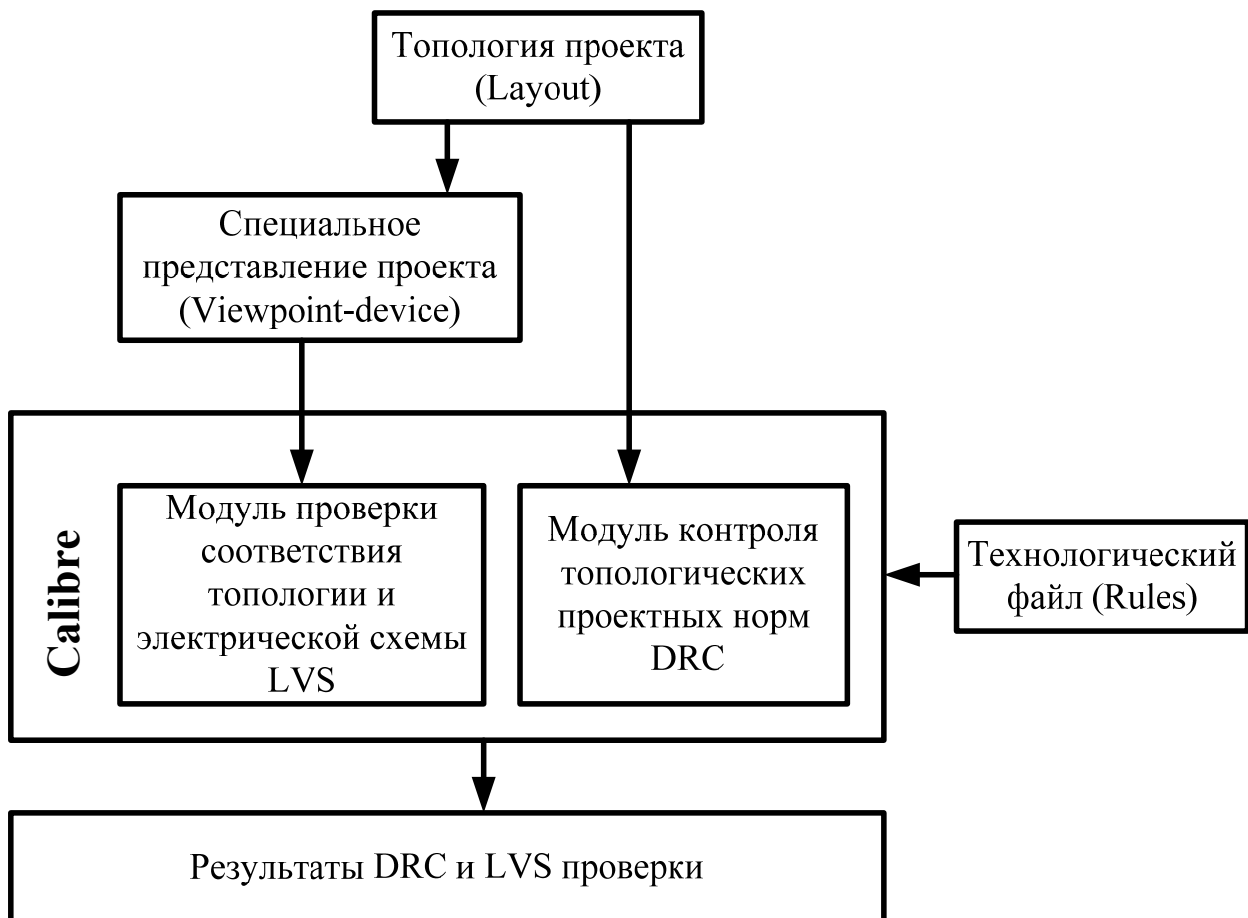


Рис. 4.1. Маршрут верификации проекта

Проектирование топологии завершается этапом физической верификации и экстракции паразитных параметров. Для этого САПР фирмы Mentor Graphics использует пакет Calibre – фактически промышленный стандарт верификации топологии ASIC. Пакет включает модуль контроля то-

пологических проектных норм Calibre DRC, модуль проверки соответствия топологии и электрической схемы Calibre LVS, модуль интерактивной верификации ячеек и блоков, работающий непосредственно в среде топологического редактора – Calibre Interactive, модуль визуализации результатов верификации и отладки Calibre RVE/QDB, модуль экстракции паразитных параметров для ячеек, блоков и кристаллов Calibre xCR. Результаты экстракции могут быть использованы для более точного моделирования с учетом реальных физических параметров и соответствующей модификации схемы проекта.

В главном окне Calibre Interactive DRC/LVS расположены следующие пункты меню (рис. 4.2):

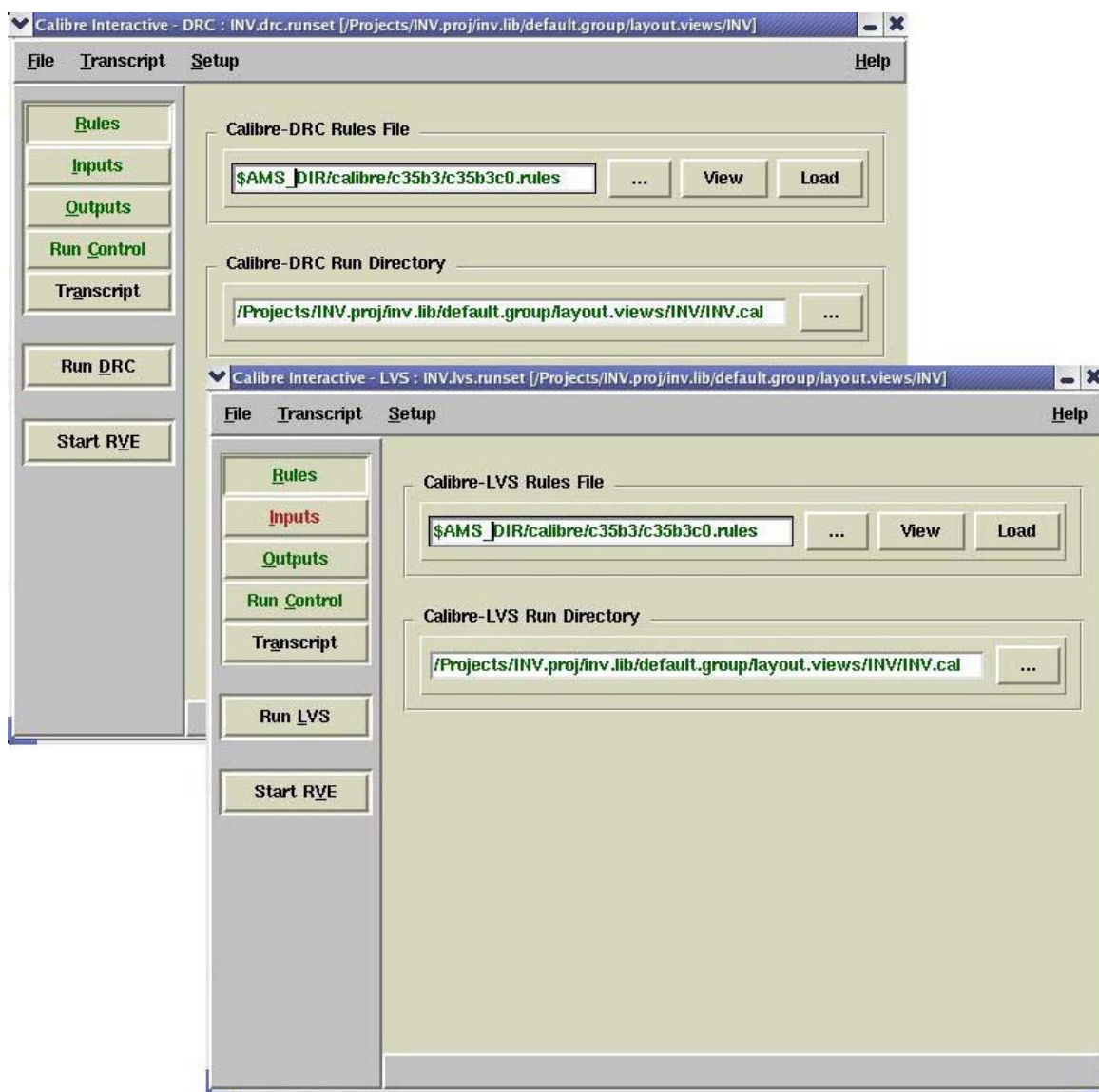


Рис. 4.2. Окно Calibre DRC/LVS

- задание технологического файла и рабочей директории (**Rules**);
- задание формата входных файлов (**Inputs**);
- задание формата выходных файлов (**Outputs**);
- настройка запуска проекта (**Run Control**);
- описание выполнения проекта (**Transcript**);
- запуск DRC/LVS проверки (**Run DRC/LVS**);
- запуск модуля визуализации результатов верификации DRC/LVS RVE (**Start RVE**).

Пример выполнения задания. Выполнить верификацию проекта средствами DRC и LVS анализа.

1) Проверка топологических норм Calibre DRC.

- Из главного меню IC-Station запустить **Calibre > Run DRC** (рис. 4.3), в появившемся окне нажать **Run DRC**;
- В окне Calibre – DRC RVE появится список ошибок (рис. 4.4). Ссылки на ошибки являются активными.

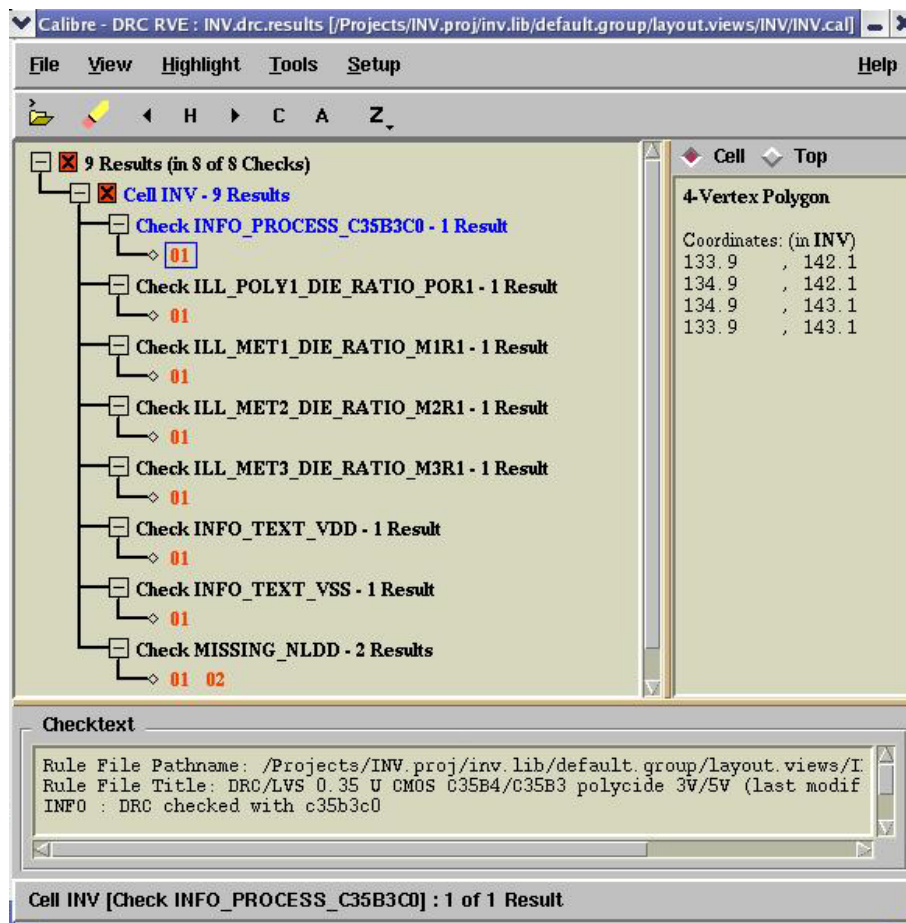


Рис. 4.4. Окно Calibre – DRC RVE

Типы ошибок, которые носят информационный характер и не требуют исправления:

- а) Check INFO_PROCESS_C35B3C0 – технология проектирования (информация);
- б) Check ILL_MET1_DIE_RATIO_M1R1 – процент занимаемой площади слоя MET1;
- в) Check ILL_MET2_DIE_RATIO_M2R1 – процент занимаемой площади слоя MET2;
- г) Check ILL_MET3_DIE_RATIO_M3R1 – процент занимаемой площади слоя MET3;
- д) Check ILL_MISS_MET3BLOCK_AMTS1 – минимальное расстояние между металлом верхнего уровня;
- е) Check INFO_TEXT_VDD – информация о порте VDD;
- ж) Check INFO_TEXT_VSS – информация о порте VSS;
- з) Check MISSING_NLDD – пропущен слой, который необходимо добавить на верхнем уровне проектирования кристалла.

Типы ошибок, которые необходимо исправить:

- а) Check SPAC_MET1_NOTCH_M1S1 – превышено минимальное расстояние между слоями MET1;
- б) Check SPAC_MET2_NOTCH_M2S1 – превышено минимальное расстояние между слоями MET2;
- в) Check SPAC_MET3_NOTCH_M3S1 – превышено минимальное расстояние между слоями MET3;
- г) Check ILL_MET1_NOT_VIA_CONT_ERC – не соединенный проводник MET1;
- д) Check ILL_MET2_NOT_VIA_CONT_ERC – не соединенный проводник MET2;
- е) Check ILL_MET3_NOT_VIA_CONT_ERC – не соединенный проводник MET3;
- ж) Check ILL_FLAOTING_NET_ERC – все не соединенные проводники.

После исправления ошибок в окне Calibre – DRC RVE выбрать **Highlight > Clear highlights**.

2) Проверка соответствия топологии и электрической схемы Calibre LVS.

- Для выполнения проверки LVS необходимо создать специальное представление проекта (viewpoint). Загрузить пакет DA-IC и в окне выбрать **HIT-Kit Utilities > Create Viewpoint**. В появившемся окне в строке **Design Path** с помощью кнопки **Navigator** необходимо выбрать логическое представление схемы инвертора *\$inv/default.group/logic.views/INV*, в поле **Technology Name** выбрать **C35B3**, а в поле **Viewpoint Level** выбрать **Device** и нажать кнопку **Ok**.
- Из панели Schematic Edit выбрать **Simulation**. В появившемся окне **Entering Simulation Mode** выбрать *vpt_c35b3_device* и нажать **Ok**;
- Из панели Schematic Sim выбрать **Session > Netlister**. В поле **Generate a .subckt for the Top Cell** нажать **Yes** (рис. 4.5) и нажать **Ok**;

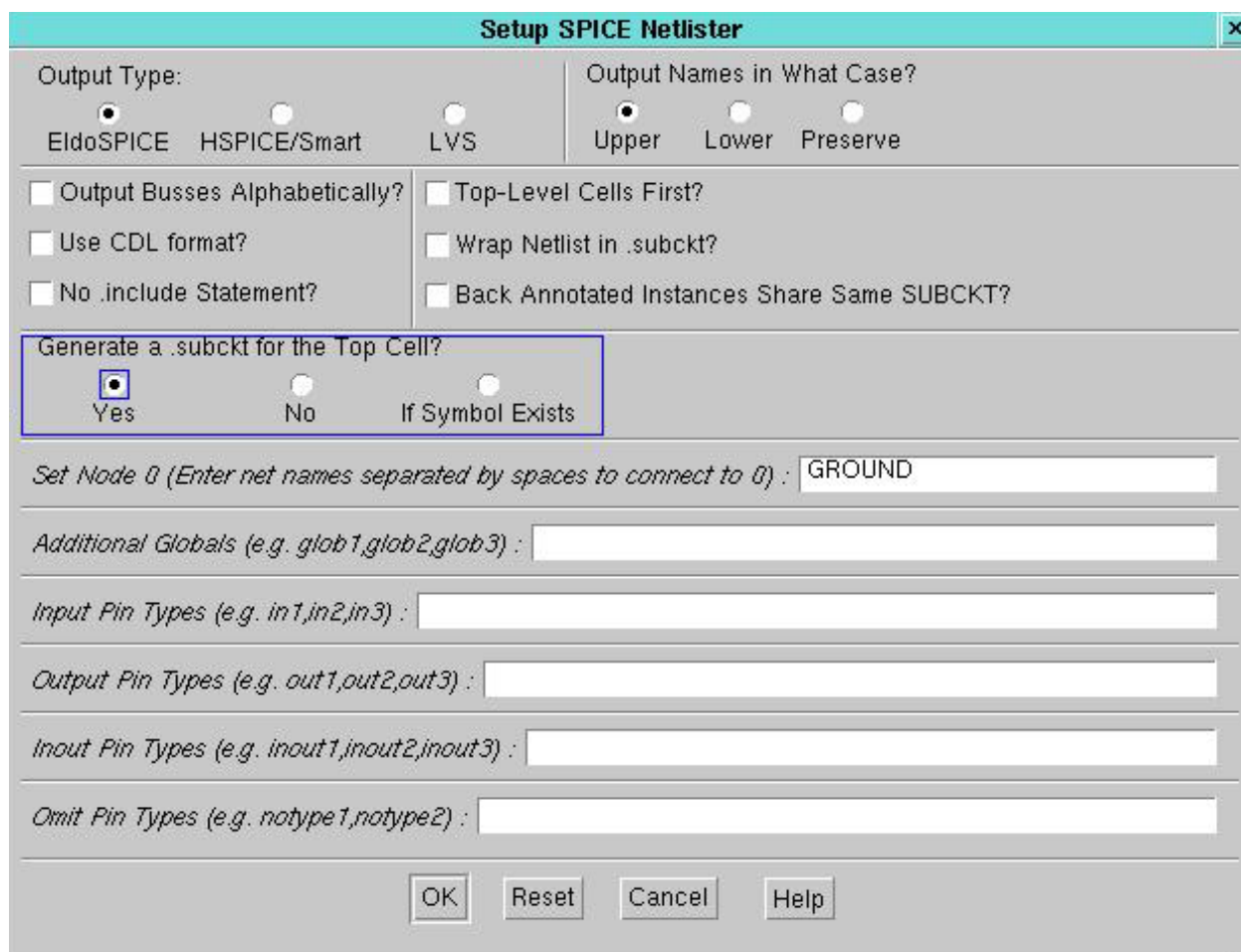


Рис. 4.5. Окно Setup SPICE Netlister

- Из панели Schematic Sim выбрать **Netlist**, в появившемся окне Netlisting Design нажать **Ok**;
- Из главного окна DA-IC в режиме моделирования выбрать **HIT-Kit Utilities** > **Eldo 2 Calibre**. В появившемся окне **Eldo2Calibre** нажать **Ok** (рис. 4.6);

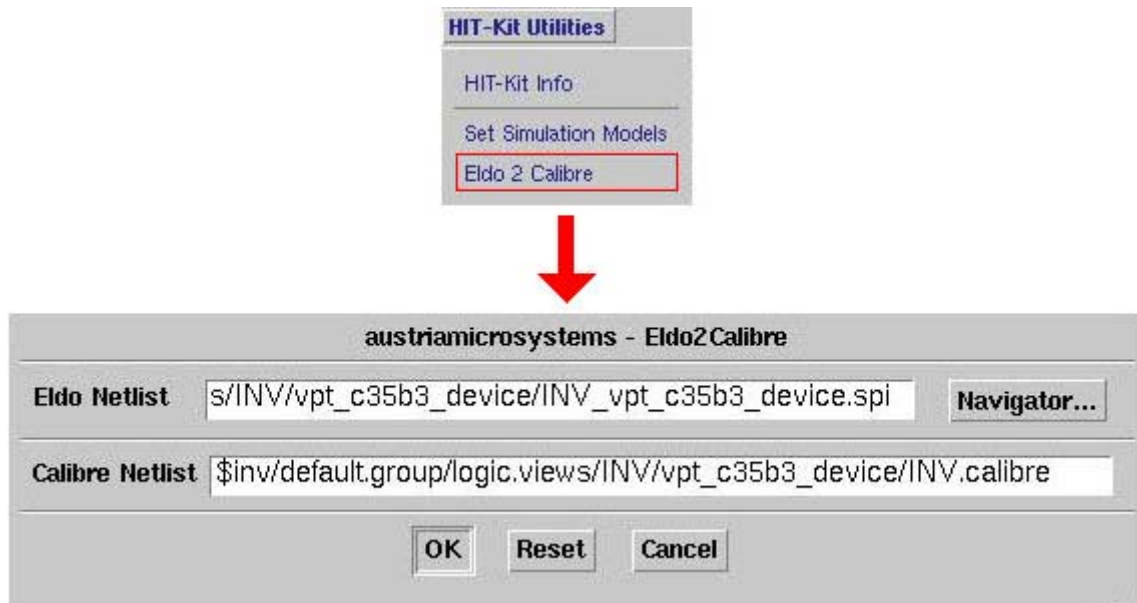


Рис. 4.6. Создание списка соединений для Calibre LVS

- Из главного меню IC-Station запустить **Calibre** > **Run LVS**, в появившемся окне нажать **Run LVS**. Нажать **Input**, выбрать вкладку Netlist. В поле **Files** выбрать *INV.proj/inv.lib/default.group/logic.views/INV/vpt_c35b3_device/INV.calibre* и нажать **Run LVS** (рис. 4.7);

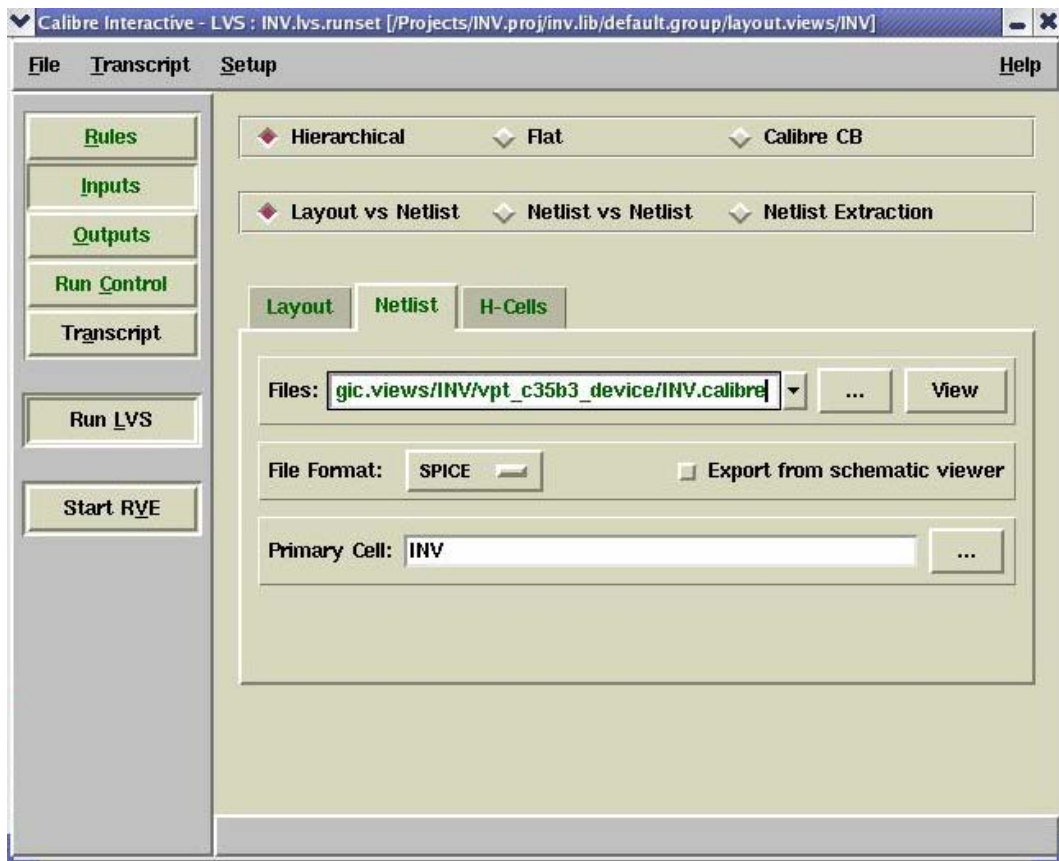


Рис. 4.7. Окно Calibre LVS

После этого, появится окно LVS Report File содержащее отчет о проекте и окно Calibre LVS RVE содержащее результат проверки списка цепей схемы и топологии проекта (рис. 4.8).

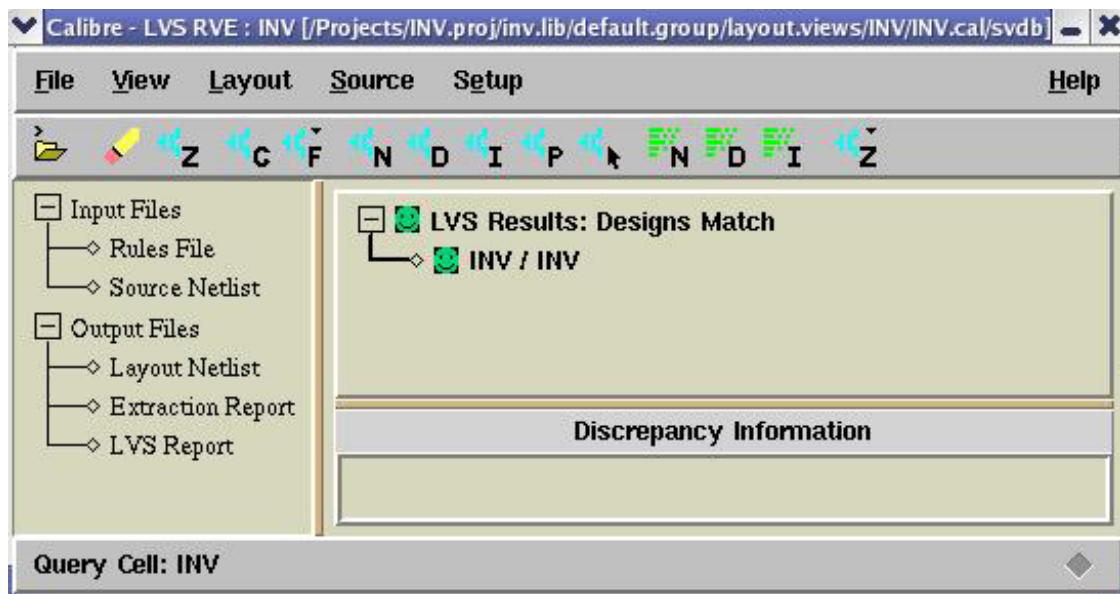


Рис. 4.8. Результат проверки Calibre LVS

4 Порядок выполнения работы

- 4.1. Ознакомиться с примером верификации инвертора средствами DRC и LVS анализа.
- 4.2. Получить вариант индивидуального задания у преподавателя.
- 4.3. Запустить программу IC-Station.
- 4.4. Запустить Calibre DRC и выполнить проверку топологических проектных норм.
- 4.5. Запустить пакет DA-IC и создать список цепей для Calibre LVS.
- 4.6. Запустить Calibre LVS и выполнить проверку соответствия топологии и электрической схемы.
- 4.7. Провести анализ полученных результатов.

5 Содержание отчета

- 5.1. Цель работы и задание.
- 5.2. Топология проекта после верификацию проекта средствами DRC и LVS анализа.
- 5.3. Отчет проверки Calibre DRC (из DRC Summary Report, пункт RULECHECK RESULTS STATISTICS).
- 5.4. Отчет проверки Calibre LVS (LVS Report File).
- 5.5. Анализ полученных результатов и выводы по работе.

6 Варианты индивидуальных заданий

- Вариант №1. Регистр.
- Вариант №2. Трехразрядный сумматор на элементах исключаящее ИЛИ, И-НЕ.
- Вариант №3. Трехразрядный сумматор на элементах И-НЕ.
- Вариант №4. Дешифратор на элементах И-НЕ.
- Вариант №5. Дешифратор на элементах ИЛИ-НЕ.

7 Контрольные вопросы и задания

- 7.1. Каково назначение и интерфейс пакета Calibre DRC/LVS?
- 7.2. Для чего нужны специальные представления проекта Device?
- 7.3. Назовите технологические нормы для технологии КМОП 0,35мкм.

Лабораторная работа № 5

ПОДГОТОВКА МИКРОСХЕМЫ ДЛЯ ИЗГОТОВЛЕНИЯ НА КРЕМНИЕВОЙ ФАБРИКЕ

1 Цель работы: Изучение правил создания технологического файла в формате GDSII.

2 Помещение кристалла в корпус

На рисунке 5.1 показан маршрут проектирования для подготовки микросхемы к изготовлению на кремниевой фабрике. В маршрут не включен этап моделирования, так как на этом этапе проектирования ИС добавляются только стандартные периферийные ячейки (Pads, Peri Spacer Cells, Corner Cells), а сам проект считается подготовленным для изготовления на кремниевой фабрике.

Для подготовки микросхемы к изготовлению на кремниевой фабрике необходимо в топологии кристалла включить буферные библиотечные элементы (Pad), которые подключают к выводам схемы проекта для соединения с контактными площадками корпуса микросхемы. Существуют следующие типы буферных элементов:

- двунаправленный буфер КМОП (Pad-limited CMOS Bidirectional Buffers);
- двунаправленный буфер с триггером Шмитта (Pad-limited Schmitt-Trigger Bidirectional Buffers);
- двунаправленный буфер ТТЛ (Pad-limited TTL Bidirectional Buffers);
- выходной тристабильный буфер (Pad-limited Tri-State Output Buffers);
- выходной буфер (Pad-limited Output Buffers);
- внутренний буфер (Pad-limited Internal Buffers);
- «земляной» контакт (Pad-limited Ground Pads);
- входной буфер КМОП (Pad-limited CMOS Input Buffers);
- входной буфер с триггером Шмитта (Pad-limited Schmitt-Trigger Input Buffers);
- входной буфер ТТЛ (Pad-limited TTL Input Buffers);

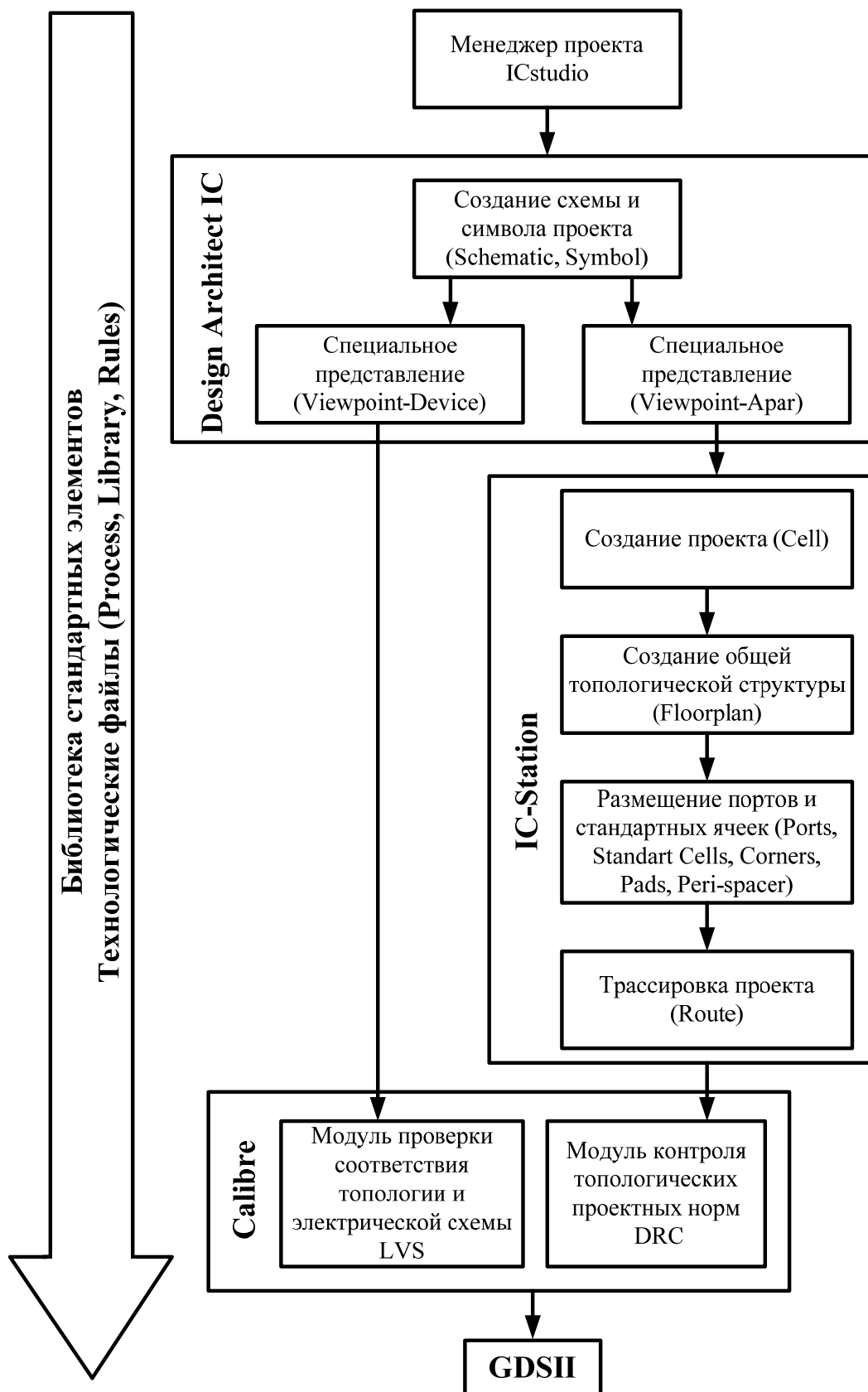


Рис. 5.1. Маршрут проектирования ИС (для подготовки микросхемы к изготовлению на кремниевой фабрике)

- контакт «питания» (Pad-limited Power Pads).

Зазор между буферными элементами должен быть заполнен ячейками Peri Spacer Cells, которые имеют различную толщину (0.1, 1, 2, 5, 10, 20, 50, 100 мкм). Вершины углов должны содержать элементы - Corner Cells (рис. 5.2).

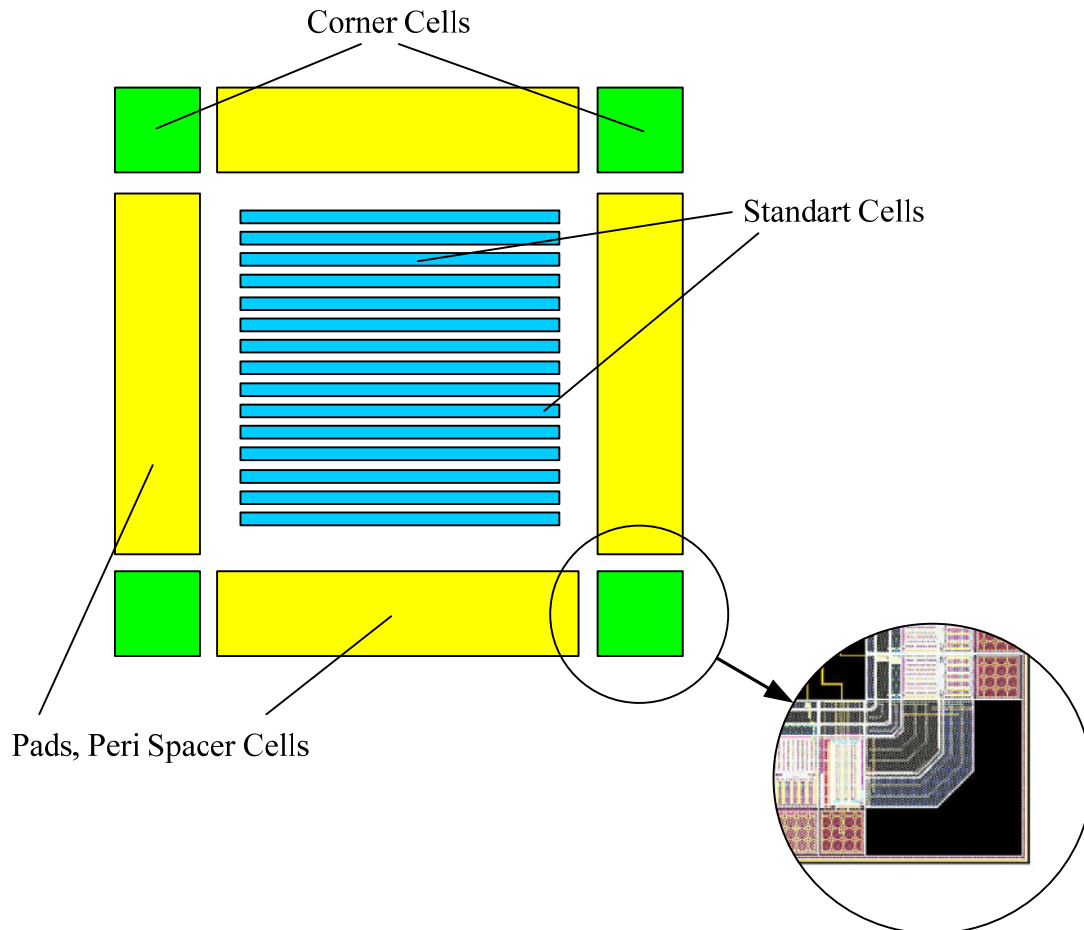


Рис. 5.2. Общая топологическая структура микросхемы

Для соединения выводов кристалла и корпуса используют разварку проволокой (золотой или алюминиевой), столбиковые выводы или непосредственное формирование выводов микросхемы с помощью дополнительной фотолитографии (рис. 5.3).

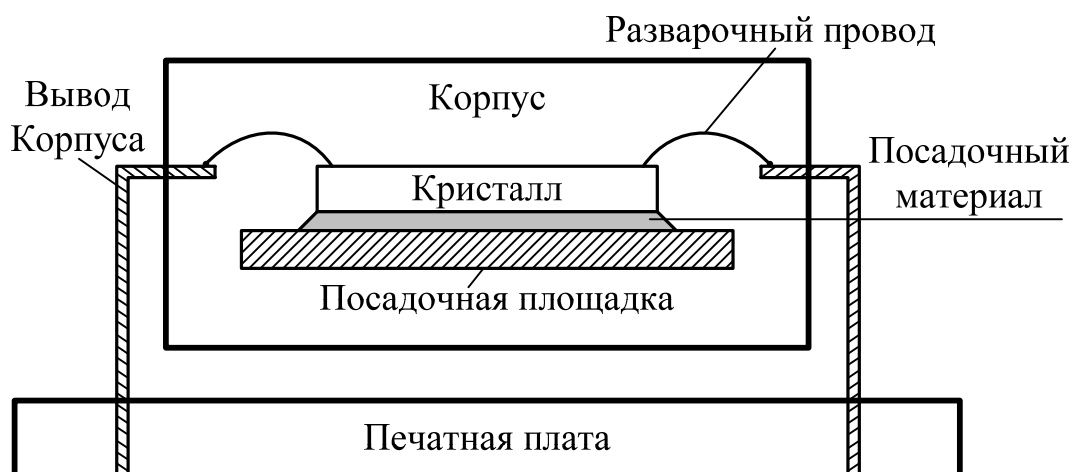


Рис. 5.3. Микросхема, посаженная на печатную плату (в разрезе)

Пример выполнения задания. Включить в проект буферные библиотечные элементы для подготовки микросхемы к изготовлению на кремниевой фабрике. Создать технологический файл в формате GDSII.

1) Запустить пакет ICstudio. Создать новую схему с использованием элемента инвертора (INV). В поле **Cell** создать новый элемент INV_chip **File > New > View** в появившемся окне, в поле **Cell Name** необходимо указать имя нового элемента (INV_chip), в поле **View Type** выбрать **Schematic**.

2) В появившемся окне DA-IC необходимо собрать схему.

- Добавить в схему символ компонента INV **Add > Instance > Choose Symbol**.
- Включить панель библиотек .
- Из панели **ic library** выбрать **HIT-Kit Utilities**. В панели **AMS Library** выбрать **OILIB_3M**. Добавить входные контакты **INPUT PADS > ICP**, выходные контакты **OUTPUT PADS > Output buffer > BU1P**, контакты «питания» **POWER PADS > VDD3ALLP** и «земли» **POWER PADS > GND3ALLP**.
- Элементы VDD, VSS выбрать из панели библиотек **Generic Lib**.
- Добавить порты portin и portout из панели библиотек **Generic Lib**. Для этого необходимо выбрать элемент и нажать на клавиатуре клавишу **Q**. В появившемся окне Edit Object Properties в поле **Value**, значение NET заменить на IN и OUT соответственно.

- Соединить элементы схемы, как показано на рисунке 5.4.

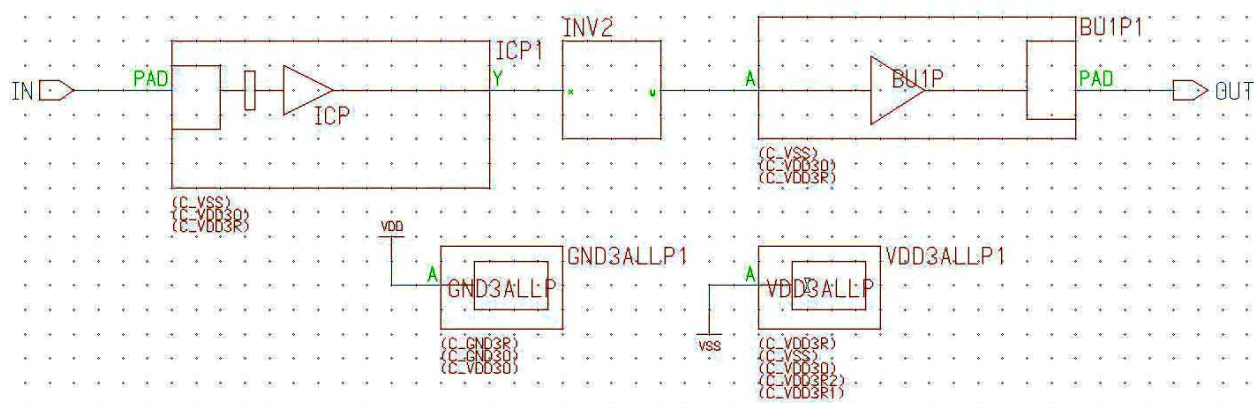



Рис. 5.4. Схема с использованием элемента инвертора и буферными библиотечными элементами

3) Для создания топологии ячейки `INV_chip` необходимо создать специальное представление проекта (viewpoint). В окне пакета DA-IC выбрать пункт меню **HIT-Kit Utilities > Create Viewpoint**. В появившемся окне в строке **Design Path** с помощью кнопки **Navigator** необходимо выбрать логическое представление схемы `INV_chip $inv/default.group/logic.views/INV_chip`, в поле **Technology Name** выбрать **C35B3**, а в поле **Viewpoint Level** выбрать **Apar** и нажать кнопку **Ok**.

4) В окне ICstudio в поле **View** необходимо создать новое представление для ячейки `INV_chip` **File > New > View**. В появившемся окне View Type в строке **Cell Name** будет указано имя элемента по умолчанию (`INV_chip`), в поле **View Type** выбрать **Layout** и нажать кнопку **Next**. В окне Cell Type в поле **Connectivity Source** необходимо выбрать `vpt_c35b3_apar` и нажать **Next**. В появившемся окне Connectivity Loading Options необходимо выбрать **Automatically create unplaced instances in layout** и нажать **Next**. В окне Automatic Layout Options выбрать **Only load the top level** и нажать кнопку **Finish**.

5) В главном меню пакета IC-Station подключить «горячие» кнопки **Other > Hotkeys > Load**. В появившемся окне Load Hotkey Settings выбрать IC Station и нажать **Ok**.

6) Появятся два окна: окно редактора электрической схемы и окно редактора топологии. В главном меню пакета IC-Station настроить рабочую область **Setup > Session >...**

7) В окне редактора топологии добавить топологию компонента INV **Objects** > **Add** > **Cell** (рис. 5.5). В появившемся окне в поле **Lib** выбрать кнопку . В окне **Open Layout View** выбрать топологию компонента верхнего уровня. В поле **Library** выбрать INV в поле **Cell** выбрать INV и в поле **View** выбрать **Layout**.

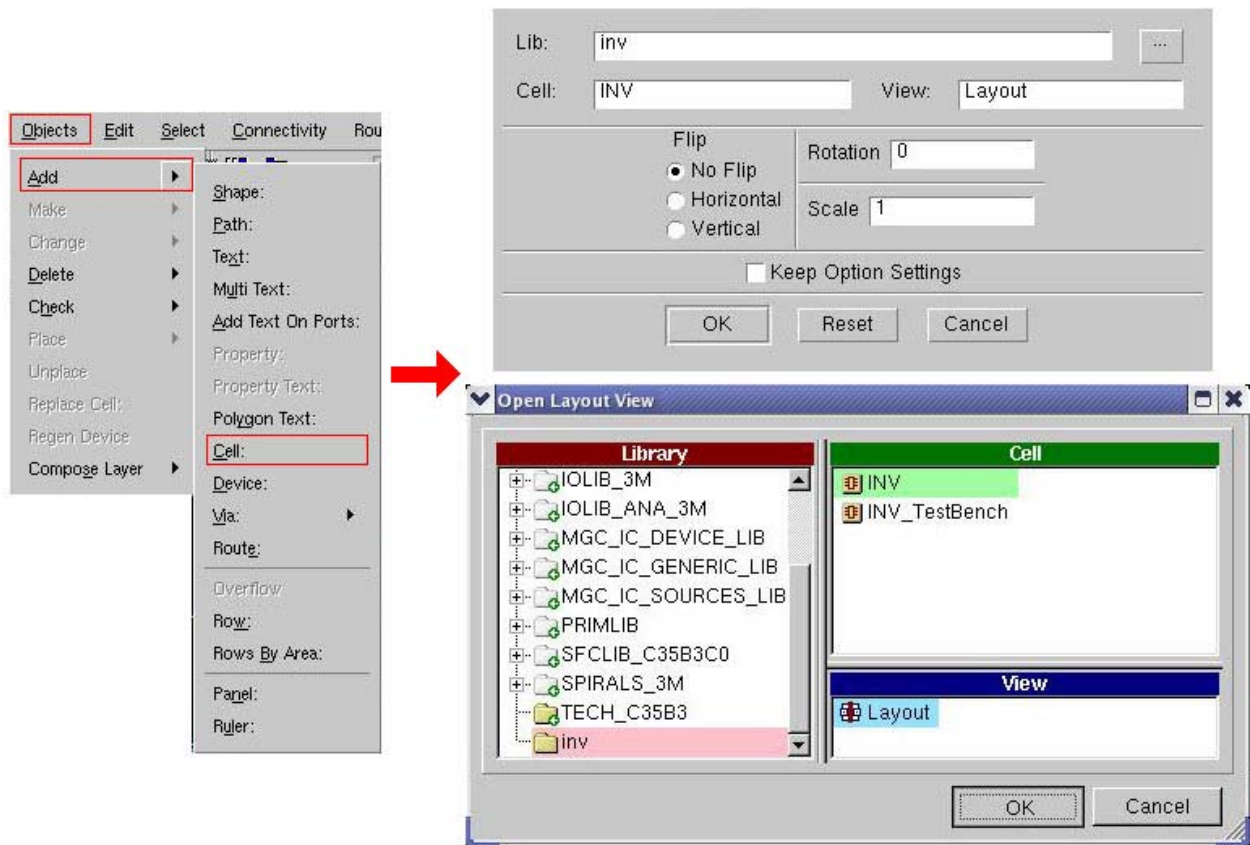


Рис. 5.5. Добавление топологии компонента INV

8) Разместить топологию компонента верхнего уровня INV в окне редактора топологии (для загрузки топологии ячейки в пакете IC-Studio) и затем удалить её.

9) Выбрать окно редактора электрической схемы в панели **DLA Logic**. Выбрать функцию авторазмещения элементов **Auto Inst** (рис. 5.6).

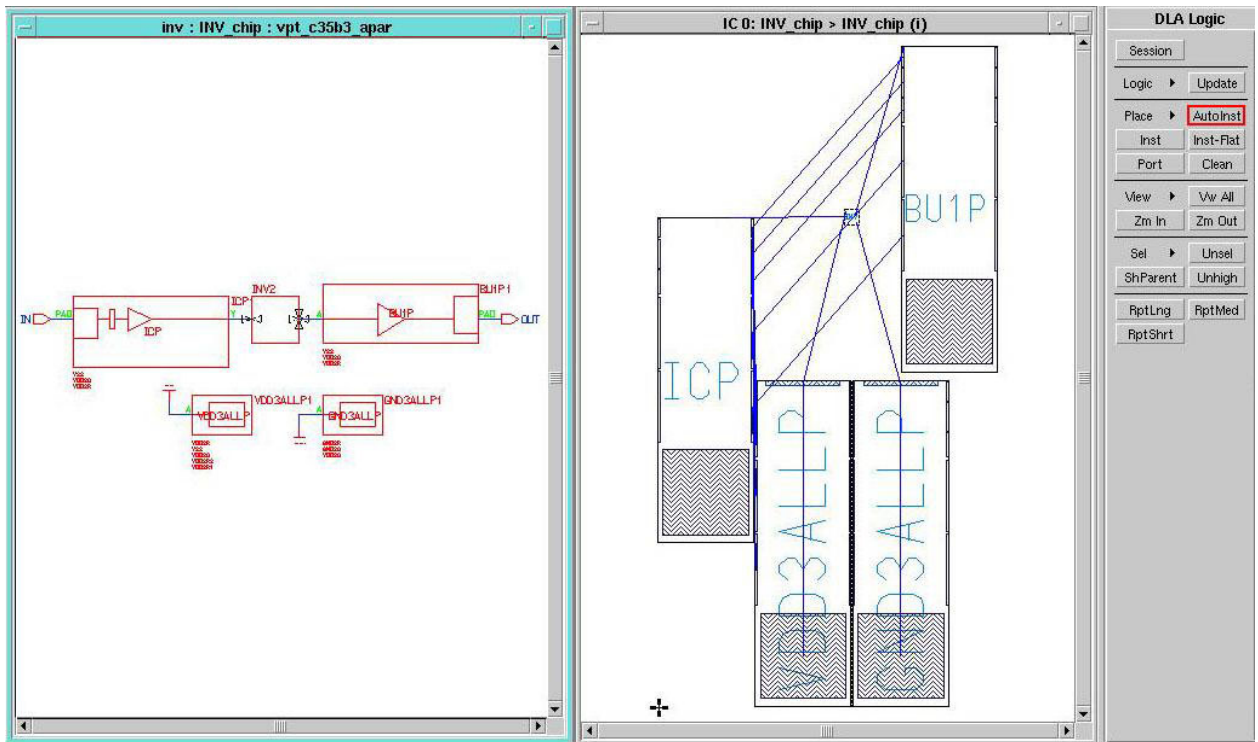


Рис. 5.6. Авторазмещение элементов

10) Настроить общую топологическую структуру. Из панели IC Palettes выбрать **Floorplan > AutoFP**, оставить настройки по умолчанию и нажать **Ok** (рис. 5.7).

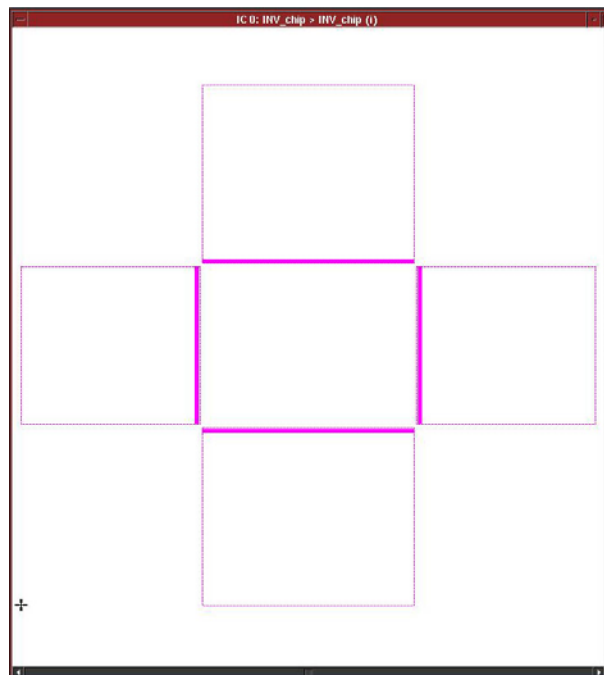


Рис. 5.7. Общая топологическая структура

11) Разместить компонент INV. Из панели IC Palettes выбрать **Place & Route > Blocks**. В появившемся окне Autoplace Blocks оставить настройки по умолчанию и нажать **Ok**.

12) Разместить площадки (Pads). Из панели IC Palettes выбрать **Place & Route > Ports**. В появившемся окне Autoplace Ports оставить настройки по умолчанию и нажать **Ok**.

13) Разместить элементы INV и Pads как показано на рисунке 5.8.

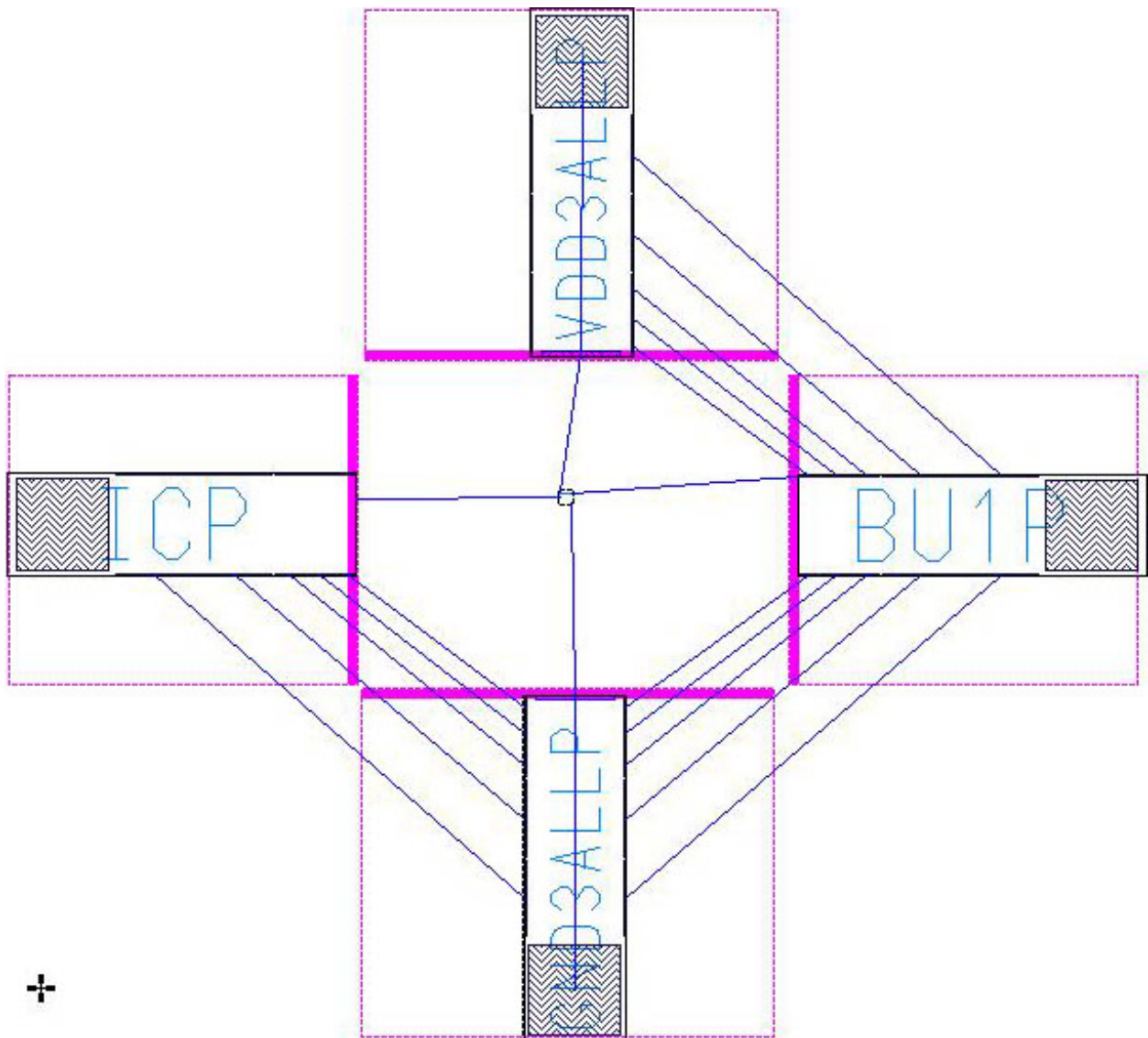


Рис. 5.8. Размещение элементов INV и Pads

14) В появившемся окне редактора топологии добавить 4 элемента Corner Cells **Objects > Add > Cell**. В появившемся окне в поле **Lib** в окне Open Layout View выбрать Corner Cells. В поле **Library** выбрать

IOLIB_3M, в поле **Cell** выбрать **CORNERP**, в поле **View** выбрать **Layout** и нажать **Ok**.

15) В окне редактора топологии добавить 8 элементов Peri Spacer Cells **Objects > Add > Cell**. В появившемся окне в поле **Lib**. В окне Open Layout View выбрать Peri Spacer Cells. В поле **Library** выбрать **IOLIB_3M**, в поле **Cell** выбрать **PERI_SPACER_10_P**, в поле **View** выбрать **Layout** и нажать **Ok**.

16) Разместить все элементы (INV, Pads, Corner Cells и Peri Spacer Cells) как показано на рисунке 5.9.

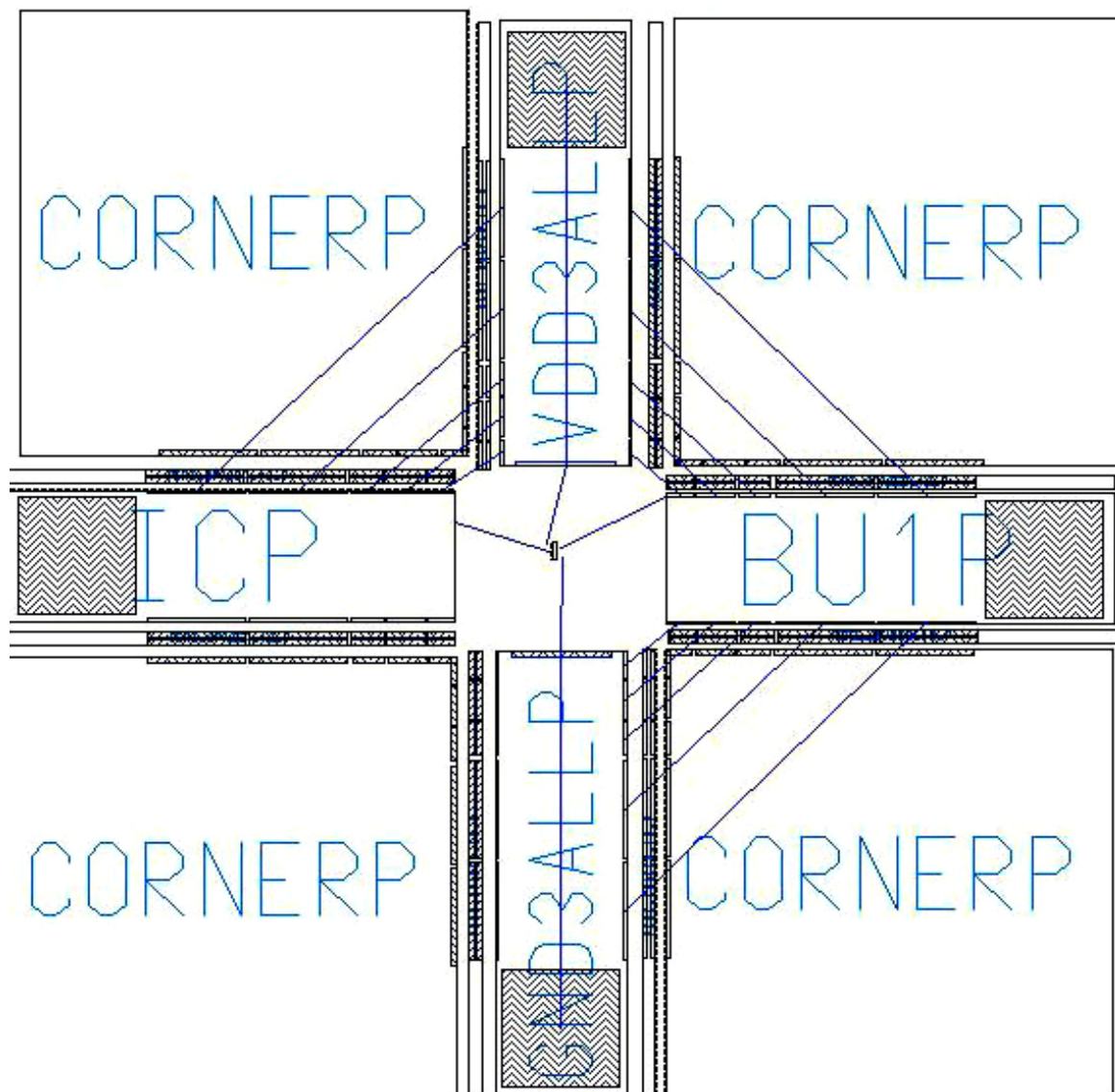


Рис. 5.9. Размещение элементов INV, Pads, Corner Cells и Peri Spacer Cells

17) Выбрать окно редактора электрической схемы. В панели **DLA Logic** выбрать функцию размещения портов **Port**. Разместить их на соответствующие площадки (Pads) и поменять значение слоев на MET3 (рис. 5.10).

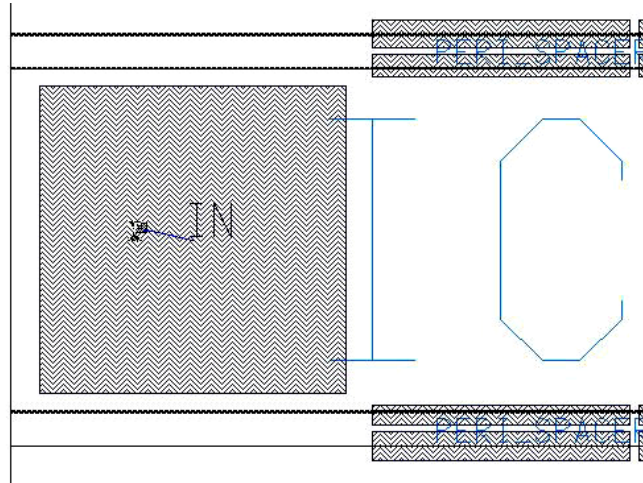


Рис. 5.10. Размещение портов

18) Выполнить трассировку и открыть все слои топологии (рис. 5.11). В главном меню IC-Station выбрать **Context > Hierarchy > Peek Area**.

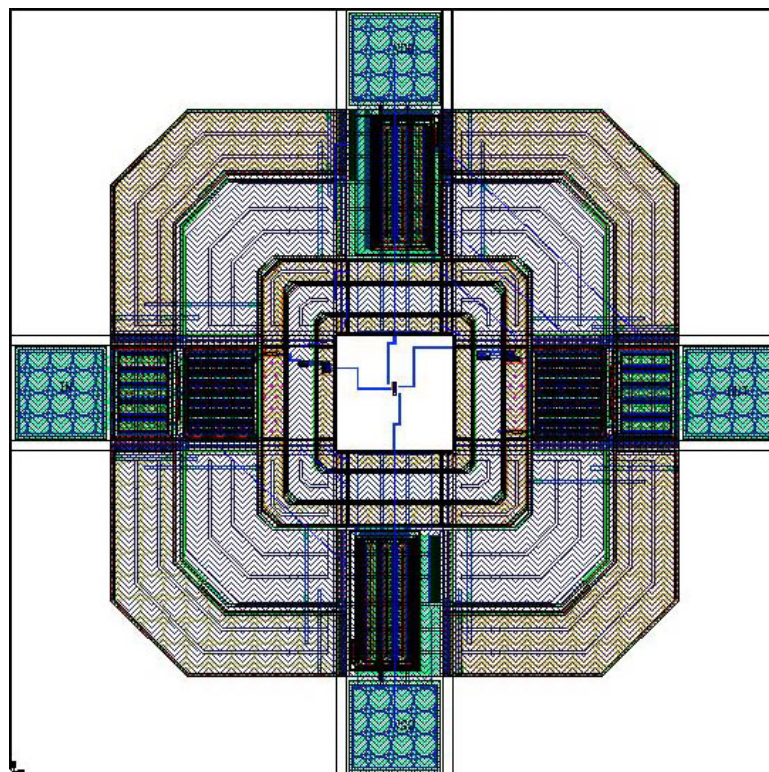


Рис. 5.11. Топология проекта INV_chip

19) Добавить слои N-LDD Implant Layer (NLDD) и Field-Implant Layer (FIMP). В главном меню пакета IC-Station нажать **HIT-Kit Utilities > Generated Layers**. В появившемся окне Load Hotkey Settings выбрать IC-Station и нажать **Ok** (рис. 5.12).

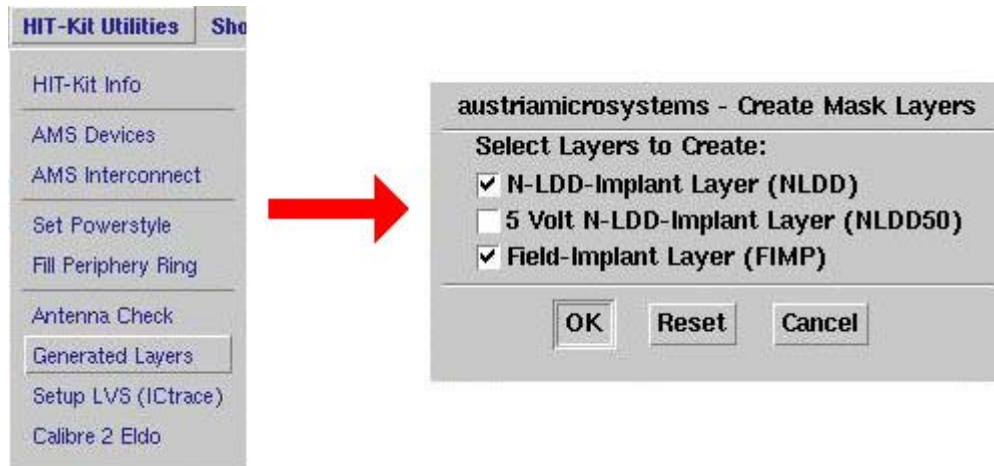


Рис. 5.12. Добавление слоев NLDD и FIMP

20) Выполнить проверку DRC. В главном меню пакета IC-Station выбрать **Calibre > Run DRC**.

21) Выполнить проверку LVS.

- Для этого необходимо создать специальное представление проекта (viewpoint). В окне DA-IC выбрать **HIT-Kit Utilities > Create Viewpoint**. В появившемся окне в строке **Design Path** с помощью кнопки **Navigator** необходимо выбрать логическое представление схемы INV_chip \$inv/default.group/logic.views/INV_chip, в поле **Technology Name** выбрать **C35B3**, а в поле **Viewpoint Level** выбрать **Device** и нажать кнопку **Ok**.
- Из панели **Schematic Edit** выбрать **Simulation**. В появившемся окне Entering Simulation Mode выбрать vpt_c35b3_device и нажать **Ok**;
- Из панели **Schematic Sim** выбрать **Session > Netlister**. В поле **Generate a .subckt for the Top Cell** нажать **Yes** и нажать **Ok**;
- Из панели **Schematic Sim** выбрать **Netlist**, в появившемся окне Netlisting Design нажать **Enter**;
- Из главного окна DA-IC в режиме моделирования выбрать **HIT-Kit Utilities > Eldo 2 Calibre**. В появившемся окне Eldo2Calibre нажать **Ok**;

- Из главного меню IC Station запустить **Calibre > Run LVS**, в появившемся окне нажать **Run LVS**. Нажать **Input**, выбрать вкладку Netlist. В поле **Files** выбрать *INV.proj/inv.lib/default.group/logic.views/INV_chip/vpt_c35b3_device/INV_chip.calibre* и нажать **Run LVS**;

22) Создать технологический файл в формате GDSII. Из главного меню IC-Station запустить **Translate > Write GDS**. В появившемся окне Write GDSII в поле **Output GDS File** задать имя INV_chip.gds (рис. 5.13).

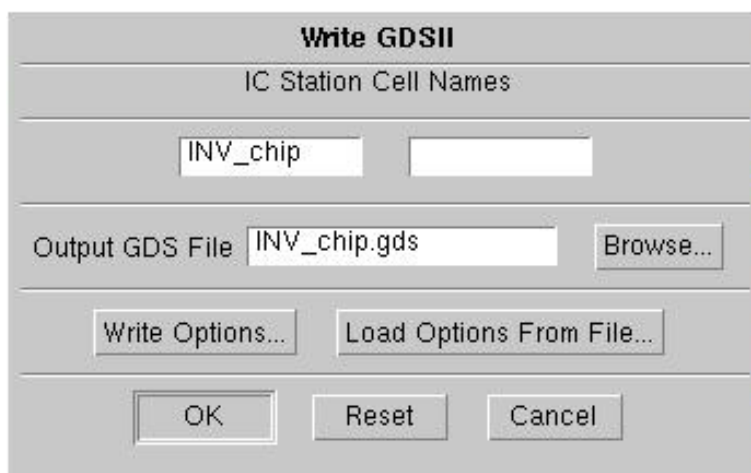


Рис. 5.13. Создание технологического файла в формате GDSII

Созданный технологический файл будет находиться в директории *INV.proj/inv.lib/default.group/layout.views/INV_chip/INV_chip.gds*.

23) Транслировать технологический файл в формате GDSII в топологию.

- В окне ICstudio необходимо создать новую библиотеку (inv_translate) **File > New > Library** (рис. 5.14).

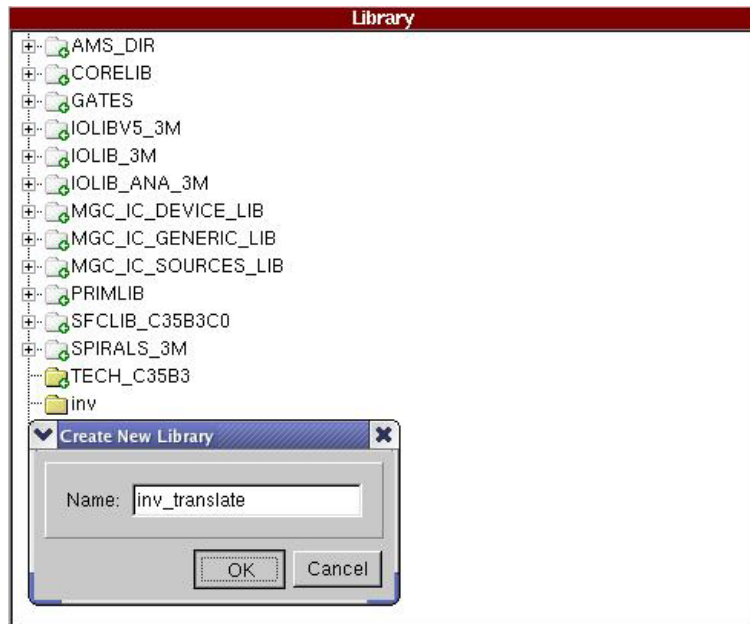


Рис. 5.14. Создание библиотеки inv_translate

- Импорт технологического файла. Выбрать библиотеку inv_translate и нажать **File > Import > GDSII** (рис. 5.15).

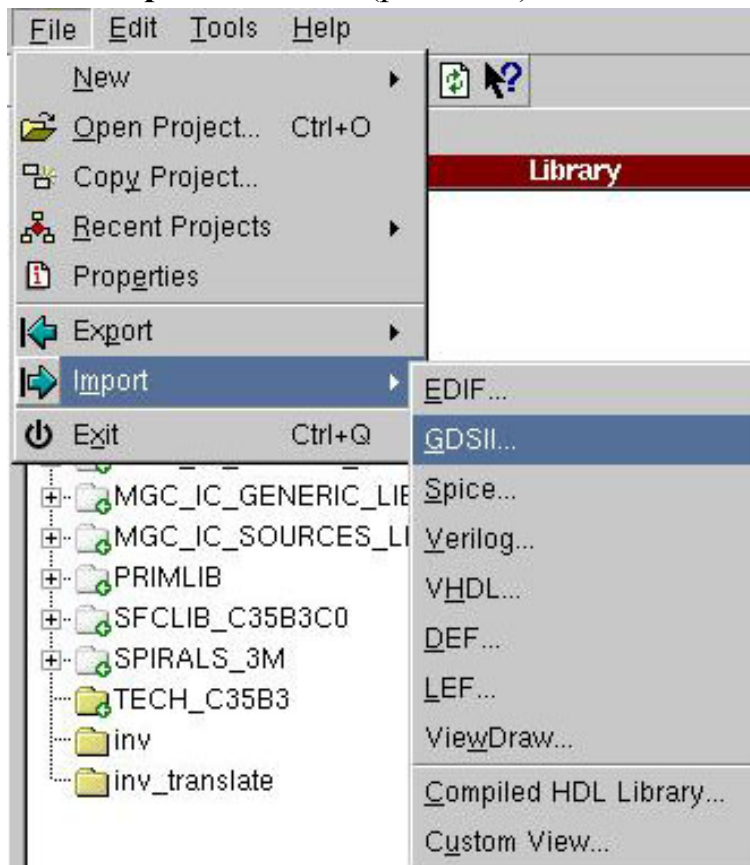


Рис. 5.15. Импорт технологического файла

- В появившемся окне **Import GDSII** указать путь к технологическому файлу (INV_chip.gds) и нажать **Import** (рис. 5.16).

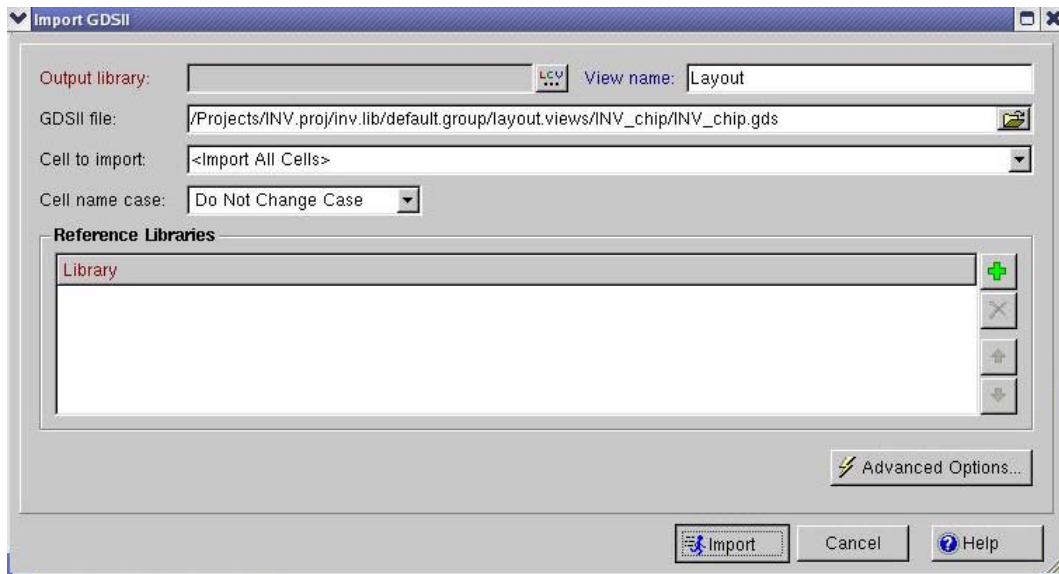


Рис. 5.16. Окно Import GDSII

- После импорта технологического файла в библиотеке inv_translate появятся все ячейки, используемые при создании топологии (рис. 5.17).

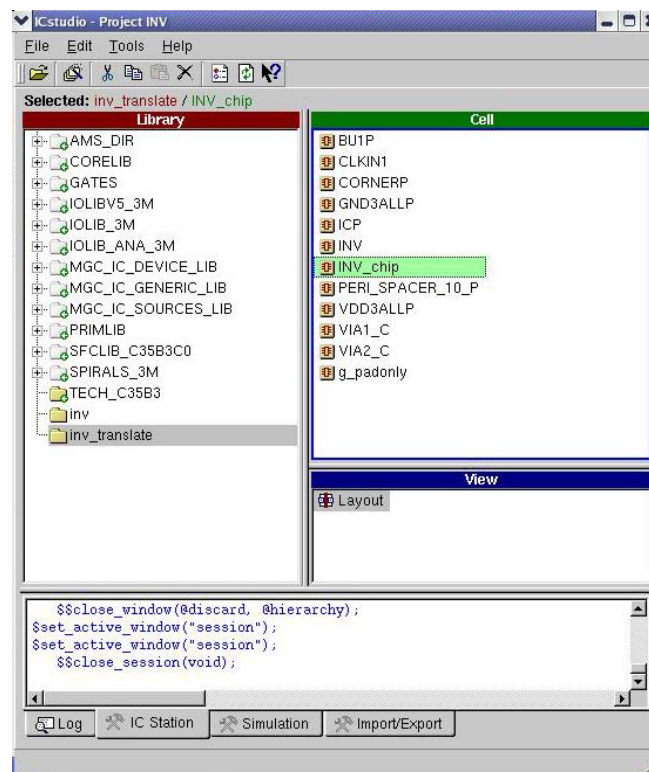


Рис. 5.17. Импорт из технологического файла INV_chip.gds

4 Порядок выполнения работы

- 4.1. Изучить технологические нормы для технологии КМОП 0,35мкм.
- 4.2. Ознакомиться с примером создания технологического файла в формате GDSII.
- 4.3. Получить вариант индивидуального задания у преподавателя.
- 4.4. Запустить пакет DA-IC и создать схему с применением буферных библиотечных элементов.
- 4.5. Запустить IC-Station из пакета ICstudio.
- 4.6. Выполнить топологическое проектирование индивидуального задания.
- 4.7. Получить технологический файл GDSII из топологии.
- 4.8. Получить топологию из технологического файла GDSII.
- 4.7. Провести анализ полученных результатов.

5 Содержание отчета

- 5.1. Цель работы и задание.
- 5.2. Схема электрическая принципиальная с буферными библиотечными элементами.
- 5.3. Топология проекта с буферными библиотечными элементами.
- 5.4. Отчет проверки Calibre DRC (из DRC Summary Report, пункт RULECHECK RESULTS STATISTICS).
- 5.5. Отчет проверки Calibre LVS (LVS Report File).
- 5.6. Элементы технологического файла GDSII после транслирования в топологию.
- 5.7. Анализ полученных результатов и выводы по работе.

6 Варианты индивидуальных заданий

- Вариант №1. Регистр.
- Вариант №2. Трехразрядный сумматор на элементах исключаящее ИЛИ, И-НЕ.
- Вариант №3. Трехразрядный сумматор на элементах И-НЕ.
- Вариант №4. Дешифратор на элементах И-НЕ.
- Вариант №5. Дешифратор на элементах ИЛИ-НЕ.

7 Контрольные вопросы и задания

7.1. Опишите маршрут проектирования ИС для подготовки микросхемы к изготовлению на кремниевой фабрике.

7.2. Перечислите стандартные периферийные ячейки.

СПИСОК РЕКОМЕНДУЕМОЙ ЛИТЕРАТУРЫ

1. С. Г. Мосин, В. С. Кухарук, С. В. Федоров. Методика проектирования цифровых ЗИС в САПР Mentor Graphics. Проектирование и технология электронных средств. Всероссийский научно-технический журнал. – В.: ВлГУ, 1/2006.

2. www.austriamicrosystems.com

3. www.mentor.com

4. А. Лохов. Средства проектирования компании Mentor Graphics. Электроника: Наука, Технология, Бизнес, 7/2000, с. 28-30.

Оглавление

ВВЕДЕНИЕ	3
ЧАСТЬ I. ПРОЕКТИРОВАНИЕ АНАЛОГО-ЦИФРОВЫХ БЛОКОВ ИНТЕГРАЛЬНЫХ СХЕМ	4
<i>Лабораторная работа № 1</i> ФУНКЦИОНАЛЬНО-ЛОГИЧЕСКОЕ ПРОЕКТИРОВАНИЕ ЗИС	4
<i>Лабораторная работа № 2</i> СХЕМОТЕХНИЧЕСКОЕ ПРОЕКТИРОВАНИЕ ЗИС	29
ЧАСТЬ II. ПРОЕКТИРОВАНИЕ ЗАКАЗНЫХ ИНТЕГРАЛЬНЫХ СХЕМ В СРЕДЕ САПР MENTOR GRAPHICS	42
<i>Лабораторная работа № 3</i> ТОПОЛОГИЧЕСКОЕ ПРОЕКТИРОВАНИЕ ЦИФРОВОЙ ЗИС	42
<i>Лабораторная работа № 4</i> ВЕРИФИКАЦИЯ ТОПОЛОГИИ ЗИС	56
<i>Лабораторная работа № 5</i> ПОДГОТОВКА МИКРОСХЕМЫ ДЛЯ ИЗГОТОВЛЕНИЯ НА КРЕМНИЕВОЙ ФАБРИКЕ	64
СПИСОК РЕКОМЕНДУЕМОЙ ЛИТЕРАТУРЫ	79

ПРОЕКТИРОВАНИЕ ЗАКАЗНЫХ ИНТЕГРАЛЬНЫХ СХЕМ В СРЕДЕ САПР MENTOR GRAPHICS

Методические указания к лабораторным работам

Составители

ЛАНЦОВ Владимир Николаевич
МОСИН Сергей Геннадьевич
КУХАРУК Вячеслав Степанович
и др.

Ответственный за выпуск – зав. кафедрой профессор В.Н. Ланцов

Подписано в печать 10.02.09.
Формат 60x84/16. Усл. печ. л. 4,56. Тираж 50 экз.
Заказ
Издательство
Владимирского государственного университета.
600000, Владимир, ул. Горького, 87