

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Локтионова Оксана Геннадьевна

Должность: проректор по учебной работе

Дата подписания: 27.01.2022 22:54:50

Уникальный программный ключ:

0b817ca911e6668abb13a5d426d3e51fc11eabb175e9745d14a48911da36d089

МИНОБРАЗОВАНИЯ И НАУКИ РОССИИ

Федеральное государственное бюджетное
образовательное учреждение высшего образования
«Юго-Западный государственный университет»
(ЮЗГУ)

Кафедра космического приборостроения и систем связи

УТВЕРЖДАЮ

Проректор по учебной работе

О.Г. Локтионова

« 15 »

2017 г.



Проектирование иерархических блоков в среде графического редактора ORCAD CAPTURE

Методические указания по выполнению лабораторной работы
для студентов, обучающихся по направлению подготовки 11.03.03

Курск 2017

УДК 681.5

Составитель А.А. Гримов

Рецензент

Доктор технических наук, профессор А.Ф. Рыбочкин

Проектирование иерархических блоков в среде графического редактора ORCAD CAPTURE: методические указания по выполнению лабораторной работы по дисциплине «Информационные технологии конструирования электронных средств» / Юго-Зап. гос. ун-т.; сост.: А. А. Гримов. Курск, 2017. 11 с.: ил. 8.

Содержатся теоретические сведения, касающиеся реализации иерархических блоков методами восходящего и нисходящего проектирования в среде специализированного графического редактора ORCAD CAPTURE.

Указывается порядок выполнения лабораторной работы.

Текст печатается в авторской редакции

Подписано в печать . Формат 60×84 1/16.
Усл. печ. л. ____ . Уч.-изд. л. ____ . Тираж 100 экз. Заказ . Бесплатно.

Юго-Западный государственный университет.
305040, г. Курск, ул. 50 лет Октября, 94.

1. ЦЕЛЬ РАБОТЫ

Целью работы является изучение принципов восходящего и нисходящего проектирования на примере выполнения иерархических блоков в среде графического редактора OrCAD Capture.

2. ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

Основная идея проектирования *иерархических блоков* состоит в том, что в такой блок обычно помещается типовой фрагмент схемы, функция которого очевидна и повторяется в проекте неоднократно, например мультиплексор, счётчик или регистр.

Ещё одно применение *иерархических блоков* оказывается полезным при моделировании сложных объектов, когда требуемая модель отсутствует. В этом случае вы создаёте иерархическую модель (макромодель) из компонентов более низкого иерархического уровня с известными поведенческими описаниями.

Рассмотрим технологию создания и моделирования иерархических схем на примере мультиплексора $2 \rightarrow 1$. Внутренним описанием блока может быть *принципиальная схема (Schematic View)* или *программа на языке VHDL*, описывающая работу этой схемы, или *Net – файл этой схемы* в виде списка соединений в EDIF-формате. Рассмотрим сначала первый вариант.

Проектирование иерархического блока со схемой замещения

1. Создадим новый проект, набрав команду **File/New/Project**.

1.1. Введём его имя, например **test_mux2_HB**.

1.2. Укажем его место на диске, например **E:\OrCAD_9\mux2_HB**. Если названной папки не существует, она будет создана автоматически.

1.3. Зададим тип проекта, в нашем случае надо выбрать опцию: **PC Board Wizard** – моделирование и трассировка печатных плат (последней операцией мы заниматься не будем).

2. На панели **PCB Project Wizard** установим флажок **Enable Project Simulation** (разрешить моделирование проекта) и выберем нижнюю кнопку **Add VHDL-based digital simulation resources** (использовать VHDL-модели).

3. Подключим к проекту две библиотеки: **t1.olb** и **t1.vhd**.

4. В *менеджере проекта* сделаем активным окно **SCHEMATIC1:PAGE1** (оба имени устанавливаются умолчанием и могут быть изменены пользователем). В этом окне нам предстоит нарисовать схему, которая содержит всего один *иерархический блок мультиплексора HB1*.

4.1. Поместим блок в рабочем окне редактора **Schematic Page Editor**, для чего исполним команду **Place/Hierarchical Block**. На экране появится диалоговая панель с одноимённым названием (рис. 3.1).

4.2. Зададим позиционное обозначение блока, например **HB1**.

4.3. В окне «**Implementation Type**» (способ реализации иерархического блока) конкретизируем его тип – **Schematic View**.

4.4. В окне «**Implementation name**» укажем его имя – **mux2_HB**.

4.5. Нижнее окно **Path and filename** оставим пустым. В этом случае по умолчанию схема замещения блока будет размещена в каталоге текущего проекта, то есть внутри проекта **test_mux2_HB.dsn**.

4.6. В разделе **Primitive** поставим тип **No**, или сохраним заданное умолчанием значение **Default**, если вы уверены, что установки правильные. На них всегда можно взглянуть, набрав команду **Options/Design Template** и выбрав закладку **Hierarchy**. Заполненная панель должна выглядеть так, как показано на рис. 1.-

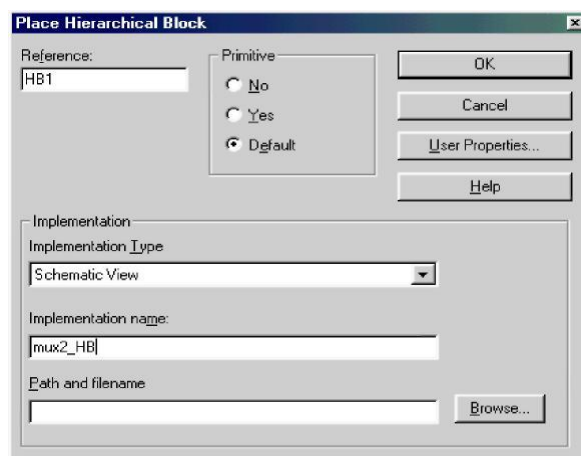


Рис. 1. Диалоговая панель для задания параметров иерархического блока **mux2_HB**

5. Нарисуем иерархический блок. Его размеры определяются числом входных или выходных контактов. Не следует делать блок излишне большим. Разместим внутри него иерархические контакты (команда **Place/Hierarchical Pin**), зададим им имена **D0**, **D1**, **A**, **Q** и тип выводов (**Input**, **Output**). Если названная команда не доступна для исполнения, значит, вы забыли выделить блок, для которого собираетесь задавать контакты.

6. Подведём к иерархическим контактам проводники (команда **Place/Wire**).

7. Зададим им пользовательские имена – алиасы (команда **Place/Net Alias**). Помните, контур с именем надо «прислонить» к тому проводнику, который вы именуете, чтобы он касался цепи.

Цепи могут иметь те же имена, что и контакты, или любые другие по вашему усмотрению. После всех выполненных действий схема должна выглядеть так, как показано на рис. 2, а.

8. Выделив блок, щёлкнем на нём правой кнопкой мыши и активизируем команду **DEscend Hierarchy** (понизить уровень). Редактор понизит уровень иерархии и откроет новую схемную страницу (рис. 3.2, б). На ней будут автоматически созданы порты, соответствующие входным и выходным контактам иерархического блока.

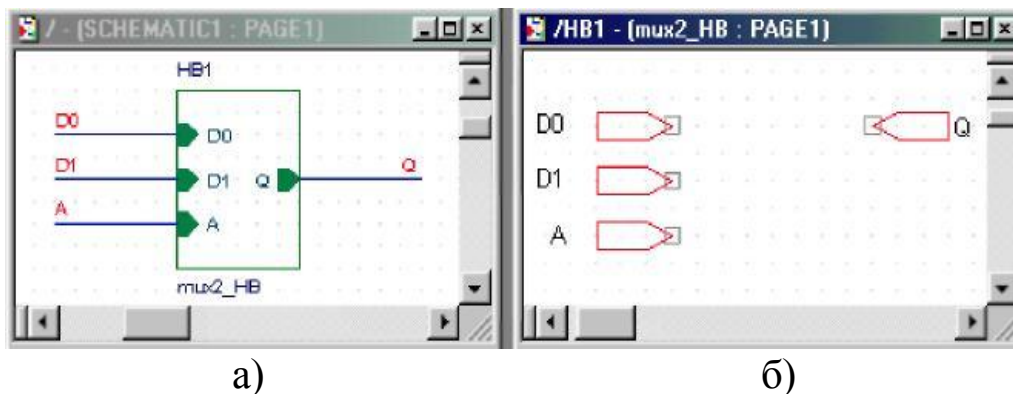


Рис. 2. Окна редактора страниц для проектирования внешнего (а) и внутреннего (б) описания иерархического блока

Саму схему замещения блока ещё предстоит нарисовать. Обратите внимание, редактор поместил в окне менеджера проекта новую папку с именем **mux2_HB**. Именно в ней будет храниться

схема замещения блока. В виде отдельного файла она не существует, что обеспечивает независимость проекта от внешней среды и гарантирует его мобильность.

9. 9. Рисуем схему замещения блока мультиплексора.

9.1. Из библиотеки **ttl.olb** извлекаем один за другим компоненты **7404** (инвертор), **7408** (2И), **7432** (2ИЛИ) и размещаем их в рабочем окне редактора.

9.2. Выполняем необходимые соединения в соответствии с принципиальной схемой мультиплексора.

9.3. Подключаем к внешним входам и выходу порты. Внутренние цепи можно не именовать, а внешние – автоматически получают имена, совпадающие с названиями портов.

Можно ли иерархический блок сделать доступным для любых проектов? Если вы спроектировали иерархический блок с типовой схемой замещения (как в нашем случае), то естественно возникает вопрос, а нельзя ли его применить в другом проекте. В виде отдельного файла она не существует. Этот факт мы уже отмечали. Кроме того, сам иерархический блок тоже «спрятан» внутри того же проекта. Можно ли иерархический блок (его внешнее и внутреннее описания) поместить в какую-нибудь библиотеку и сделать, таким образом, доступным для любых проектов? Оказывается, такая возможность есть.

Предположим, что объектом назначения является библиотека **my_lib.olb**. Двойным щелчком на иерархическом блоке вызовем редактор свойств **Property Editor** и в столбце **Implementation Path** укажем полный путь к этой библиотеке. Удобнее всего это сделать, щёлкнув на кнопке, справа от поля для ввода информации (рис. 3), отыскать нужную библиотеку и открыть её. Если такой библиотеки ещё не существует, то придётся ввести полный путь с клавиатуры, и *позднее* редактор автоматически создаст её в указанном месте.

Снова выделим иерархический блок **mux2_HB** и исполним команду **File/Export Selection**. Названная команда доступна только в том случае, если активно рабочее окно схемного редактора и выделен экспортируемый блок.

Появится диалоговая панель с тем же названием, в верхнее поле которой надо ввести имя иерархического блока, а в нижнее –

полный путь и имя библиотеки, куда он будет экспортирован (рис. 4). Последнюю операцию удобно выполнять с помощью кнопки **Browse**, расположенной справа от поля **Library**. В принципе можно указать имя несуществующей библиотеки (как в нашем случае), и тогда она будет создана автоматически. Нажмём на кнопку **OK**, чтобы поместить блок в указанную библиотеку.

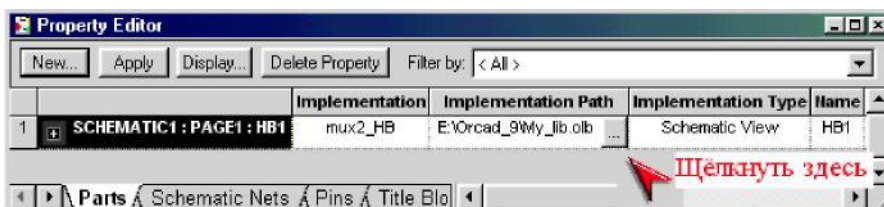


Рис. 3. Диалоговая панель редактора

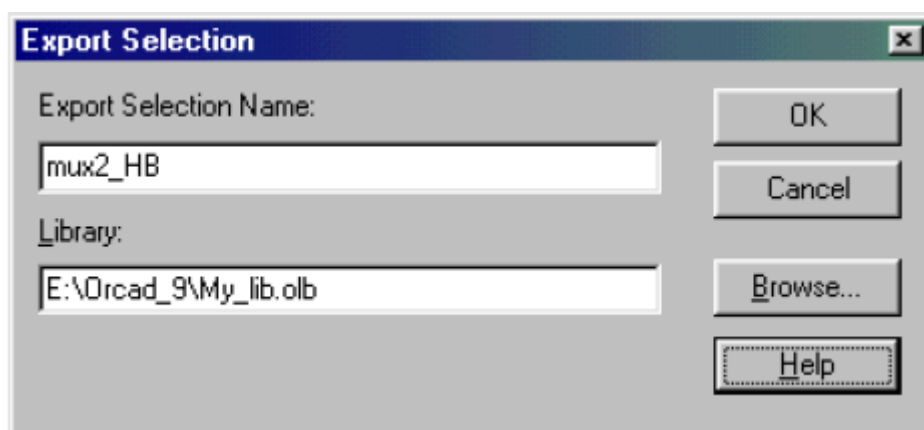


Рис. 4. Диалоговая панель для задания параметров экспортируемого иерархического символа

К сожалению, никакой информации о проделанной операции (удачной или нет) редактор не выдаёт. Поэтому приходится контролировать результат. Снимем выделение с иерархического блока и исполним команду **File/Import Selection**. Обратите внимание, команда экспорта сейчас не доступна, потому что не выделен никакой иерархический блок.

Появится диалоговая панель **Import Selection**. Нажмём на кнопку **Add Library** и добавим в пустой список библиотеку **my_lib.olb**. В эту библиотеку мы только что экспортировали иерархический блок.

Выделив его имя, увидим в правом окне изображение блока без графики (рис. 5). Нажав **ОК**, поместим его рядом с оригиналом.

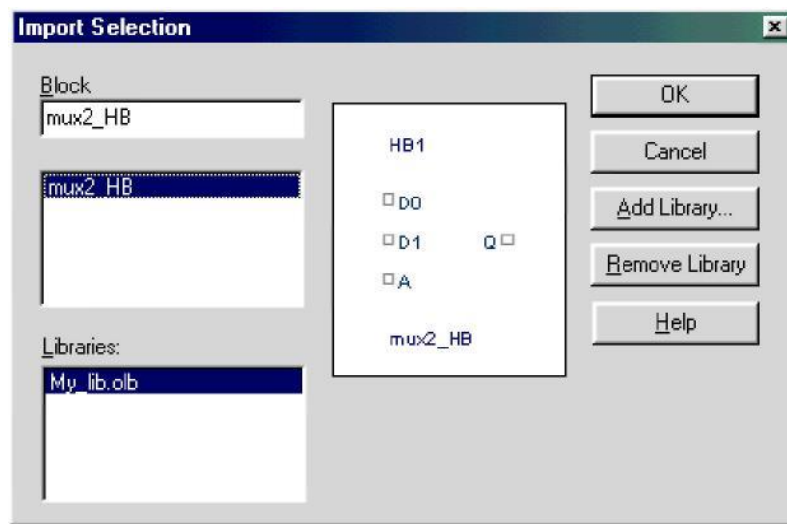


Рис. 5. Диалоговая панель для импорта (вставки в проект) иерархического блока

Открыв библиотеку **my_lib.olb**, куда был помещён иерархический блок, мы не обнаружим там никаких следов его присутствия, *хотя объём библиотечного файла увеличился*. Внутреннее описание иерархического блока (схему замещения) оказывается можно просто скопировать (или переместить) из текущего проекта в библиотеку. Это легко сделать, используя кнопки **Copy** и **Paste** или метод буксировки «**Drag-and-drop**». На рис. 6 показано, как отбуксировать папку со схемой замещения иерархического блока из текущего проекта **test_mux2_HB.opj** в библиотеку **my_lib.olb**.

Для этого надо «прижать» копируемую или перемещаемую папку левой кнопкой мыши и, не отпуская её, переместить (отбуксировать) на строку с названием библиотеки. Во время копирования папки операцию следует выполнять с нажатой клавишей **Ctrl**, при перемещении – с отпущенной клавишей.

Процесс иерархического проектирования может развиваться в двух противоположных направлениях – «сверху-вниз» (нисходящее проектирование) и «снизу-вверх» (восходящее проектирование). САПР **OrCAD** поддерживает обе стратегии иерархического проектирования.

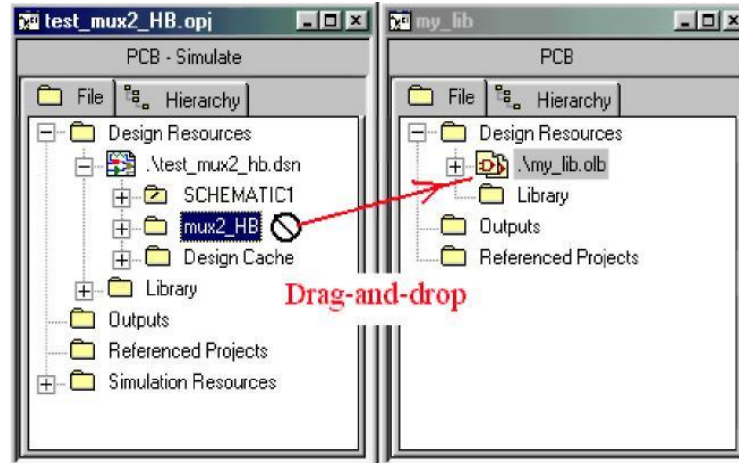


Рис. 6. Копирование папки со схемой замещения иерархического блока в библиотеку методом *drag-and-drop*

Вариант, который мы только что рассмотрели – это *нисходящее проектирование*: сначала появился иерархический блок, а уж потом его схема замещения. При переходе от блока к схеме замещения использовалась команда **DEscend Hierarchy** (понизить уровень иерархии).

Теперь рассмотрим *пример восходящего проектирования*. Создадим новый проект **test_dmux2_HB.opj** и нарисуем схему демультимплексора на два выхода. К внешним выводам схемы подключим иерархические порты. Это можно сделать командой **Place/ Hierarchical Port** или щёлкнув мышкой на иконке, показанной справа на экране.

Входным портам присвоим имена **D** и **A**, выходным – **Q0** и **Q1** (рис. 7).

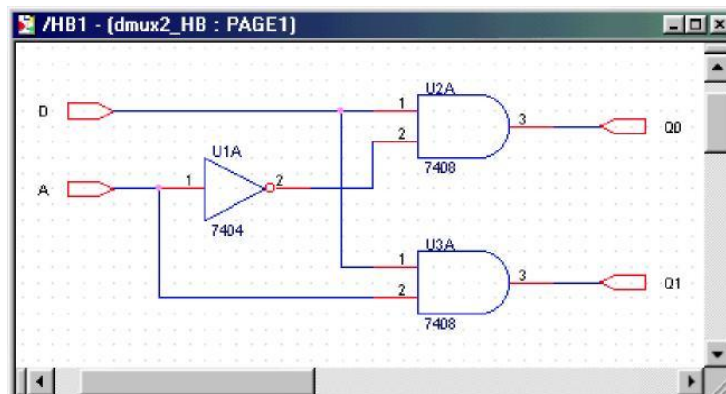


Рис. 7. Схема замещения несуществующего пока иерархического блока **dmux2_HB** (восходящее проектирование)

Сохраним созданную схему, а затем изменим имя папки, где она находится. По умолчанию папка получила название **SCHEMATIC1**. Щёлкнем на этом имени правой кнопкой и исполним команду

Rename. Откроется диалоговая панель **Rename Schematic**, в единственное окно которой надо ввести новое имя – **dmux2_HB**.

Теперь откроем новую страницу для иерархического блока (его внешнего описания). Щёлкнем правой кнопкой на имени проекта **test_dmux2_hb.dsn** и выберем команду **New Schematic**. Редактор снова предложит назвать её **SCHEMATIC1**, и мы не станем возражать. Но вот незадача – она не является корневой папке! Корневую папку редактор помечает значком «/» и сейчас этот знак стоит на той папке, которая была создана первой, то есть на папке с именем **dmux2_HB**.

Придётся ещё раз вызывать всплывающее меню. Щёлкнем правой кнопкой на папке **SCHEMATIC1** и исполним команду **Make Root**. Теперь правильная иерархия восстановлена. Создадим в папке **SCHEMATIC1** новую страницу **PAGE1** и откроем её.

Нам осталось нарисовать иерархический блок, для которого уже имеется схема замещения. Не станем описывать процедуру создания такого блока, она уже хорошо известна. Более того, основную работу – ввод контактов редактор выполнит самостоятельно! Единственно за чем надо проследить, чтобы блок получил имя (в окне **Implementation Name**), совпадающее с названием папки, где хранится его схема замещения, то есть имя **dmux2_HB**.

Как только вы определите габариты будущего блока, он предстанет перед вами «во всём великолепии» – не просто пустой контур, а законченный блок со всеми иерархическими контактами (рис. 3.8)! Таким образом, компьютер за вас выполнил работу, которую при нисходящем проектировании приходилось делать вручную.

Нам осталось подвести проводники к иерархическому блоку и протестировать его. С этой работой вы можете справиться самостоятельно.

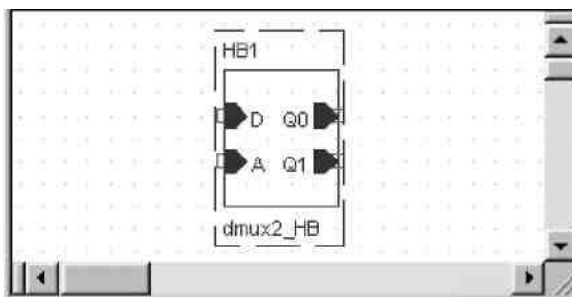


Рис. 8. Автоматическая генерация иерархического блока по его схеме замещения

3. ПОДГОТОВКА К РАБОТЕ

Изучить методические указания к данной работе.

Установить пакет OrCAD в собственном персональном компьютере или освоить работу с пакетом в лаборатории кафедры.

4. РАБОЧЕЕ ЗАДАНИЕ

– В справочнике по радиолетронным компонентам и учебнике (журнале) по радиоэлектронике выбрать самостоятельно или по указанию преподавателя принципиальную схему с пассивными (R, C, L) и активными (транзисторы, интегральные микросхемы и т.д.) электронными компонентами;

– Введите выбранную схему, используя графический редактор OrCAD Capture, выполните простановку позиционных обозначений и проверку простейших правил проектирования.

– Сформируйте список связей в любом пригодном для чтения формате и сравните его со схемой.

5. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Для чего применяются иерархические блоки?
2. Как сделать иерархический блок доступным для других проектов?
3. Что такое восходящее и нисходящее проектирование?
4. Как указать для блока схему замещения?
5. В каком виде может быть представлено внутренне описание иерархического блока?

