

Eclipse/VDT – Руководство пользователя

Содержание

Ссылки на основные внешние компоненты	2
<i>Sun JDK 1.5.0.....</i>	2
<i>Eclipse</i>	2
<i>CVer.....</i>	2
Порядок установки и запуска Eclipse/VDT.....	3
Перспектива Verilog.....	4
Управление проектом.....	6
<i>Создание нового проекта.....</i>	6
<i>Создание нового Verilog-модуля.....</i>	6
<i>Добавление внешних модулей в проект.....</i>	8
<i>Экспорт VDT-проектов.....</i>	9
Графические компоненты	11
<i>Окно Design Menu.....</i>	11
<i>Окно Verilog Navigator.....</i>	13
<i>Окно Console</i>	13
Настройки инструментов.....	15
<i>Инсталляционные настройки.....</i>	15
<i>Пакетные настройки инструментов.....</i>	16
<i>Проектные настройки инструментов.....</i>	17
<i>Сеансовые настройки инструментов</i>	17
<i>Общий вид диалога настройки</i>	18
<i>Очистка настроек.....</i>	18
<i>Контроль версий.....</i>	19
Запуск инструментов	20
Редактирование TSL-спецификации параметров настройки.....	21

Ссылки на основные внешние компоненты

Sun JDK 1.5.0

<http://java.sun.com/j2se/1.5.0/download.jsp>

Eclipse

<http://eclipse.org>

Eclipse 3.2M4, Linux x86, GTK look and feel:

<http://download.eclipse.org/eclipse/downloads/drops/S-3.2M4-200512151506/download.php?dropFile=eclipse-SDK-3.2M4-linux-gtk.tar.gz>

Eclipse 3.2M4, Linux x86, Motif look and feel:

<http://download.eclipse.org/eclipse/downloads/drops/S-3.2M4-200512151506/download.php?dropFile=eclipse-SDK-3.2M4-linux-motif.tar.gz>

Eclipse 3.2M4, Win32:

<http://download.eclipse.org/eclipse/downloads/drops/S-3.2M4-200512151506/download.php?dropFile=eclipse-SDK-3.2M4-win32.zip>

Plug-in Verilog Editor

<http://veditor.sourceforge.net/>

http://sourceforge.net/project/showfiles.php?group_id=103963&package_id=111746&release_id=410197

CVer

<http://www.pragmatic-c.com/gpl-cver/downloads/gplcver-2.11a.src.tar.bz2>

<http://www.pragmatic-c.com/gpl-cver/downloads/gplcver-2.11a.linux.bin.tar.bz2>

Порядок установки и запуска Eclipse/VDT

Соглашение: Далее в этом документе плагин Eclipse/VDT для краткости называется VDT.

1) Скачать дистрибутив Sun JDK 1.5.0_x, установить.

Внимание: необходимо устанавливать именно JDK, а не JRE. Версия JDK должна быть не ниже 1.5.0. VDT не может работать под JDK 1.4.x и 1.3.x.

В системе Linux необходимо настроить переменную PATH до директории bin установленного JDK 1.5.0. Например, если пакет Sun JDK 1.5.0_06 установлен в директорию /usr/java/jdk1.5.0_06, команда будет выглядеть так:

```
export PATH=/usr/java/jdk1.5.0_06/bin:$PATH
```

2) Скачать дистрибутив Eclipse 3.2M4.

Внимание: VDT разрабатывается и тестируется именно с версией Eclipse 3.2M4. Все остальные версии Eclipse, возможно, не будут работать с VDT, либо будут работать некорректно.

Используя архиватор, развернуть дистрибутив где-нибудь в произвольном месте на локальном диске. Например, для Linux, это может быть директория /home/user:

```
cd /home/user
tar xvzf ~/downloads/eclipse-SDK-3.2M4-linux-gtk.tar.gz
```

или для Win32 – например, директория D:\user :

```
cd D:\user\
unzip D:\downloads\eclipse-SDK-3.2M4-win32.zip
```

После распаковки в выбранной директории появится новая директория eclipse.

3) Взять дистрибутив VDT и развернуть в директорию, куда был установлен Eclipse. Например, для Linux с Eclipse, установленным в директории /home/user/eclipse :

```
cd /home/user/eclipse
unzip ~/downloads/com.elphel.vdt_1.1.0.zip
```

или для Win32, если Eclipse был установлен в D:\user\eclipse :

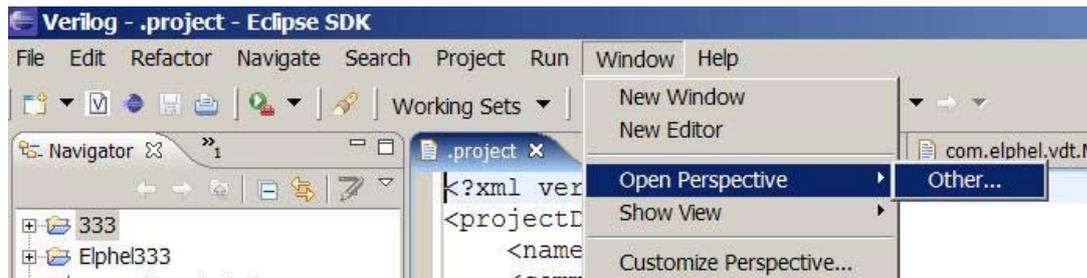
```
cd D:\user\eclipse
unzip D:\downloads\com.elphel.vdt_1.1.0.zip
```

4) Запустить Eclipse (исполняемый файл eclipse или eclipse.exe в корневой директории установки).

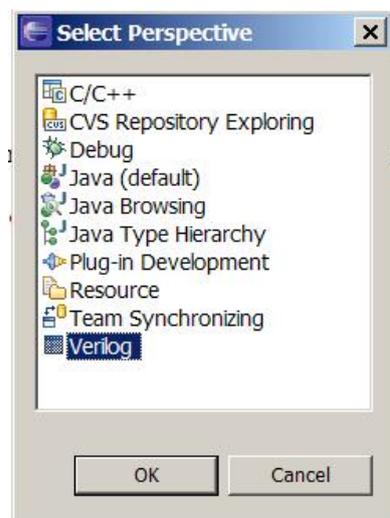
Перспектива Verilog

Компоненты, предоставляемые плагином VDT для разработки FPGA/Verilog программ в Eclipse, объединены в перспективу с именем Verilog. Перспектива (perspective) в Eclipse – это набор видимых в данный момент окон, управляющих элементов и их взаимное расположение. Перспектива позволяет оставить для использования только те компоненты из всего множества компонент, предоставляемых плагинами, которые требуются для решения конкретной задачи.

При старте Eclipse открывает последнюю из ранее использованных перспектив. Для того, чтобы открыть Verilog-перспективу, в главном меню выберите **Window** → **Open Perspective** → **Other...**:



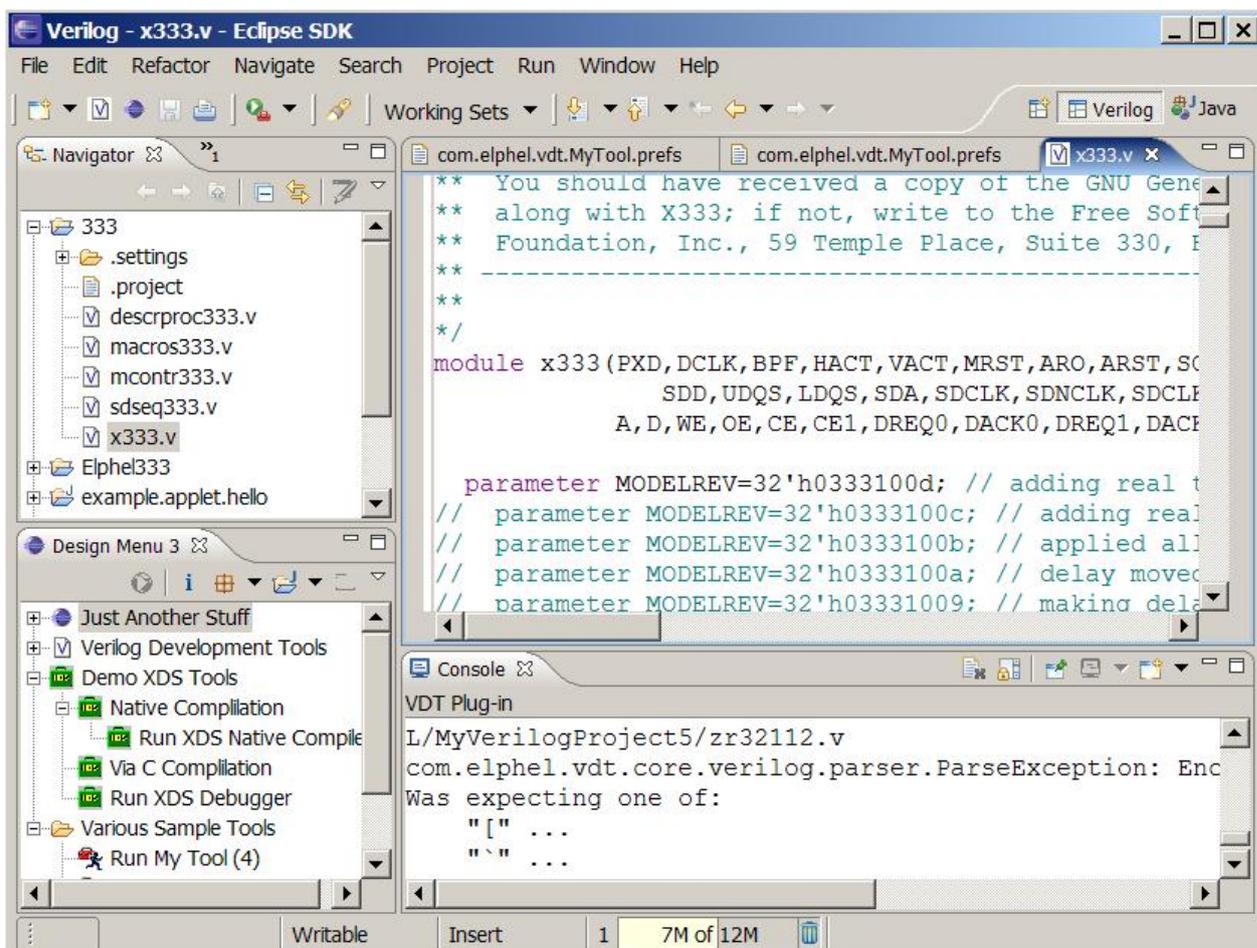
Затем в появившемся списке перспектив выберите перспективу **Verilog**:



Перспектива Verilog состоит из четырех окон-видов (views): Navigator, Verilog Navigator, Design Menu, Console и центрального окна с текстовыми редакторами.

Окно **Navigator** предназначено для навигации по каталогам и файлам проекта и манипуляции ими. Эти элементы изображаются в виде дерева. Узлы верхнего уровня – это проекты, которые могут содержать произвольное число файлов и каталогов. В окне Navigator можно удалять, добавлять, переносить и переименовывать проекты, каталоги и файлы. Для этого достаточно щелкнуть правой кнопкой мыши на нужном элементе и выбрать из всплывающего меню требуемое действие.

Окно **Verilog Navigator** визуализирует VDT-проект в виде списка Verilog-модулей, а не исходных файлов. Это окно удобно использовать, например, если необходимо указать утилите конкретный Verilog-модуль. По умолчанию окно Verilog Navigator скрыто за окном Verilog; перейти в него можно, нажав закладку с его именем или символ ">>", который показывается, если поле окна недостаточно широко, чтобы изобразить все имеющиеся закладки.



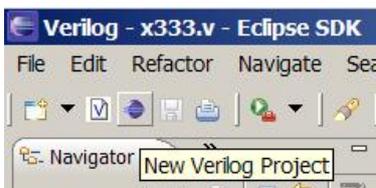
Окно **Design Menu** показывает иерархическое меню *инструментов* (вызовов внешних утилит) и служит для их настройки и запуска. Заголовок окна Design Menu может меняться в зависимости от текущего проекта.

Окно **Console** (консоль) принимает стандартные потоки вывода запускаемых внешних утилит. Отдельная консоль с именем **VDT Plug-in** служит для вывода сообщений самого плагина VDT.

Центральную область окна Eclipse занимает окно текстовых редакторов, в котором можно открыть для редактирования один или несколько файлов. В редактируемых текстах выполняется подсветка синтаксиса в соответствии с их типами. Например, файлы с расширением `.v` распознаются как исходные тексты программ на языке Verilog и раскрашиваются соответственно (если установлен плагин Verilog editor).

Кроме этих четырёх окон, в перспективе Verilog в панель инструментов добавлены кнопки:

-  New Verilog Project – для создания новых VDT-проектов;
-  New Verilog Module – для создания Verilog-модулей:



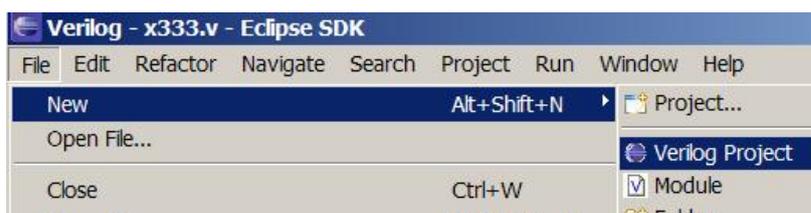
Управление проектом

Все возможности, предоставляемые плагином VDT по настройке и запуску внешних утилит, доступны только для VDT-проектов.

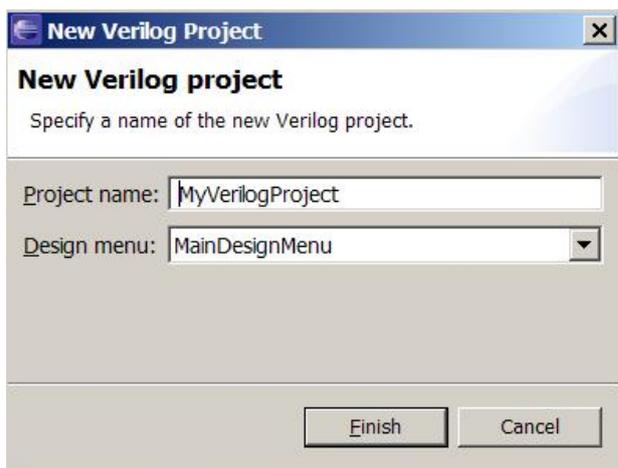
Создание нового проекта

VDT-проекты создаются при помощи мастера New Verilog Project.

1) Для создания нового VDT-проекта нажмите на панели инструментов кнопку **New Verilog Project** или выберите в главном меню **File** → **New** → **Verilog Project**:



2) В диалоге **New Verilog Project** введите в поле **Project Name** уникальное имя проекта и выберите из списка **Design Menu** подходящее меню для проекта (содержащее набор инструментов, необходимый для этого проекта). Название выбранного меню будет показано в заголовке окна Design Menu.



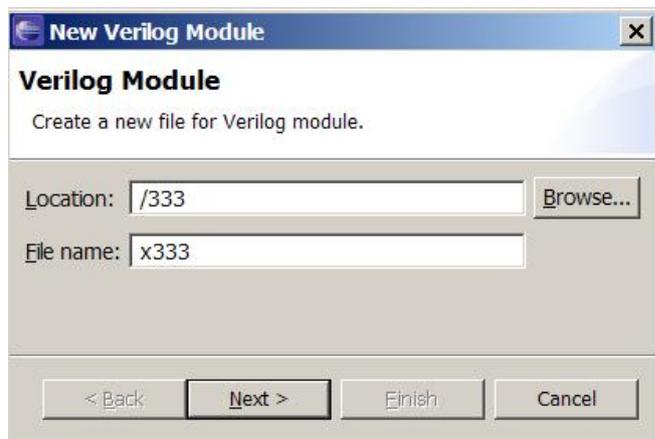
Создание нового Verilog-модуля

1) Для создания нового Verilog-модуля можно воспользоваться мастером New Verilog Module. Для этого нажмите на панели инструментов кнопку **New Verilog Module** или выберите в главном меню **File** → **New** → **Module**.



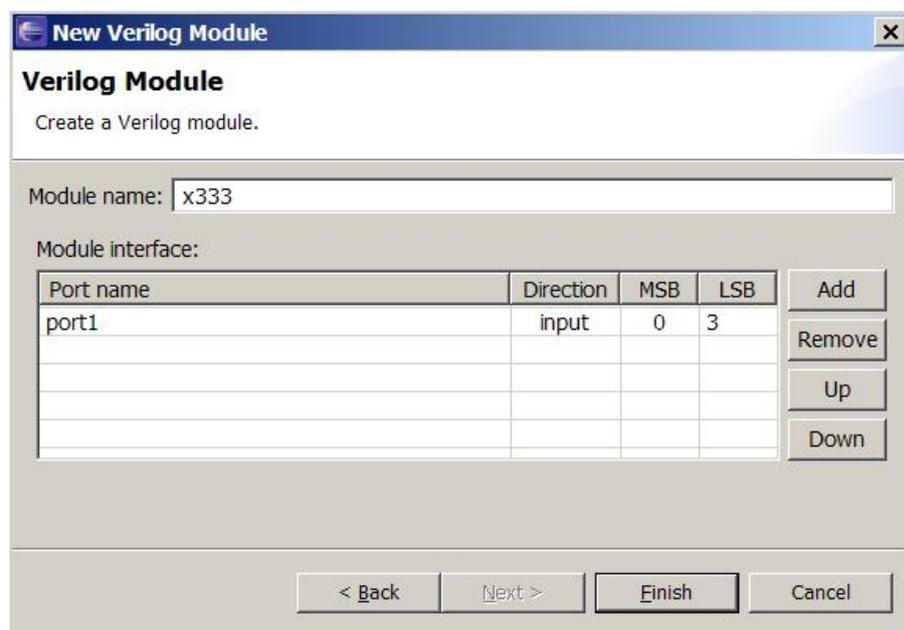
2) В диалоге **New Verilog Module** в поле **Location** укажите путь до каталога проекта, в котором при необходимости будет создан Verilog-файл. По умолчанию в этом поле стоит имя текущего проекта, выбранного в окне Navigator.

3) В поле **File name** укажите имя файла, в который будет помещен новый модуль (расширение “.v” может быть опущено).



4) Для перехода на следующую страницу нажмите кнопку **Next**.

5) На следующей странице диалога введите имя модуля и его интерфейс: количество, имена и свойства интерфейсных портов. Для добавления удаления и изменения порядка интерфейсных портов служат кнопки **Add, Remove, Up, Down**, находящиеся справа от списка портов.

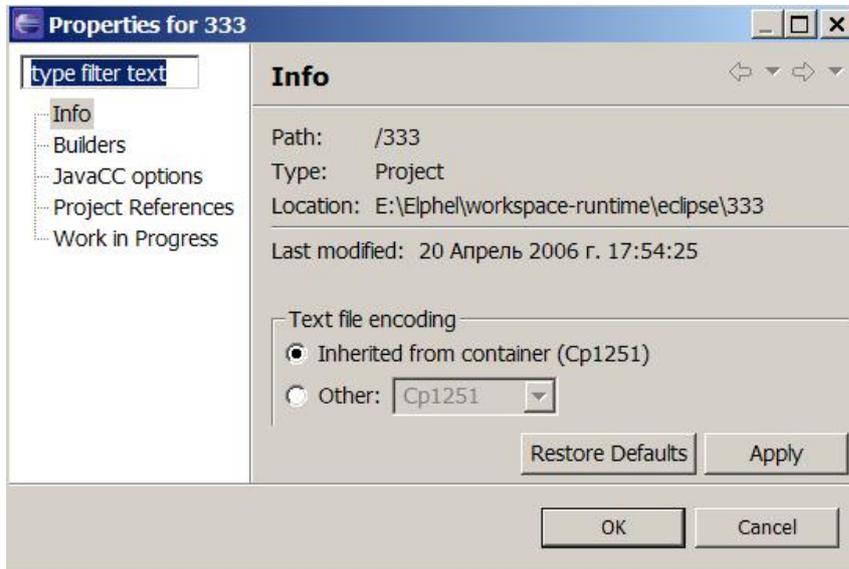


6) По окончании задания интерфейса нажмите кнопку **Finish**. В указанном каталоге проекта будет создан новый файл с указанным именем, в который будет вписана заготовка модуля, соответствующая заданному интерфейсу.

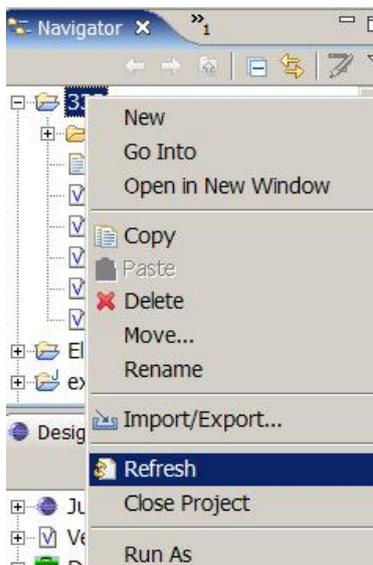
Добавление внешних модулей в проект

Для добавления уже существующих внешних модулей в проект:

- 1) Узнайте местоположение проекта на диске (в окне Navigator щелкните правой кнопкой мыши на проекте, выберите в контекстном меню **Properties** → **Info**, посмотрите значение поля **Location**):



- 2) Скопируйте внешние файлы и директории в директорию проекта.
- 3) В окне Navigator щелкните правой кнопкой мыши на проекте и в контекстном меню проекта выберите **Refresh**:

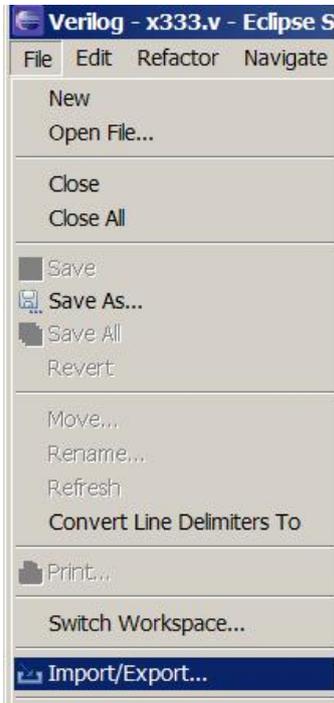


Экспорт VDT-проектов

Экспорт VDT-проектов нужен, например, для переноса проектов на другую машину.

Для экспорта VDT проектов можно воспользоваться мастером Eclipse Export Wizard, для этого:

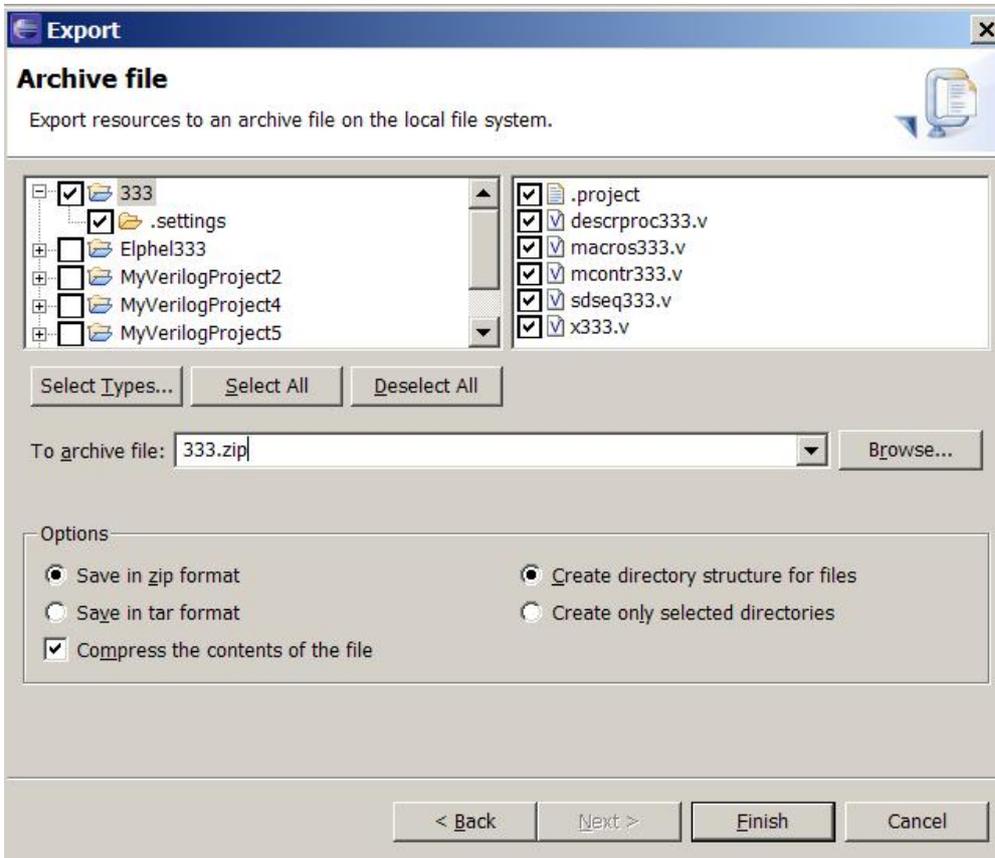
- 1) В главном меню выберите **File** → **Import/Export**:



- 2) В открывшемся диалоге выберите закладку **Export**, в меню выберите пункты **General** → **Archive File** и нажмите **Next**:



- 3) В левой панели диалога Export выберите проект, а в правой – файлы, которые надо экспортировать. В список экспортируемых файлов **обязательно добавьте файл .project !**
- 4) В левой панели выделите подкаталог .setting экспортируемого проекта и все его содержимое.
- 5) В поле **To archive file** введите имя архива.



- 6) Нажмите кнопку **Finish**.

Более подробное описание процесса экспорта можно найти по ссылке:

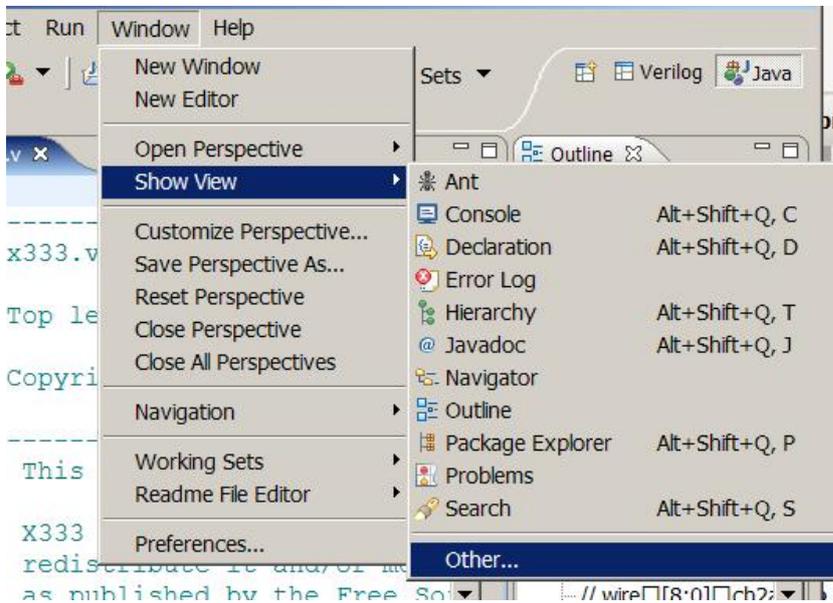
<http://www.cs.laurentian.ca/badams/c1047/eclipse-tutorials/export-tutorial.html>

Внимание: Небольшая тонкость заключается в том, что импортировать полученный архив надо не как Archive File, а как **Existing Project into Workspace**.

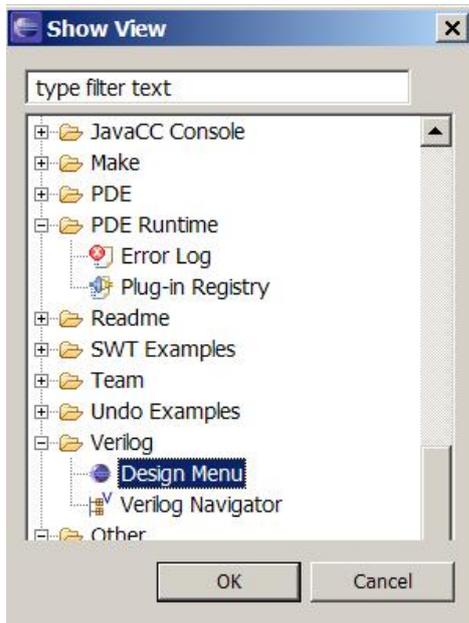
То же самое можно проделать “вручную”, просто скопировав каталог с проектом. Местонахождение проекта можно узнать следующим образом: в окне Navigator щелкните правой кнопкой мыши на проекте, в контекстном меню выберите **Properties** → **Info**, посмотрите значение поля **Location**.

Графические компоненты

Основными графическими компонентами плагина VDT являются окна Design Menu, Console и Verilog Navigator. По умолчанию все они доступны в перспективе Verilog, но могут быть открыты и в любой другой перспективе. Для этого в главном меню выберите **Window** → **Show View** → **Other** :



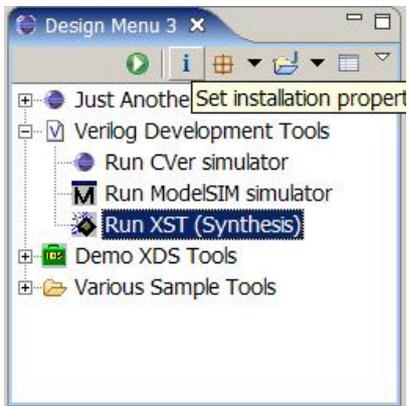
Далее в открывшемся диалоге выберите **Verilog** и имя нужного окна:



Окно Design Menu

Design Menu – это основное окно для работы с внешними утилитами. Оно содержит иерархический список инструментов (вызовов утилит) и предназначено для их настройки и запуска. Список доступных инструментов зависит от проекта и выбирается при создании VDT-проекта. Для не-VDT-проектов список инструментов всегда пуст.

По умолчанию окно Design Menu находится под окном Navigator. Заголовок окна содержит имя меню инструментов, которое было выбрано при создании проекта. Для разных проектов заголовок окна Design Menu может быть разным.



Локальная панель инструментов окна Design Menu содержит кнопки, позволяющие выполнить следующие действия:

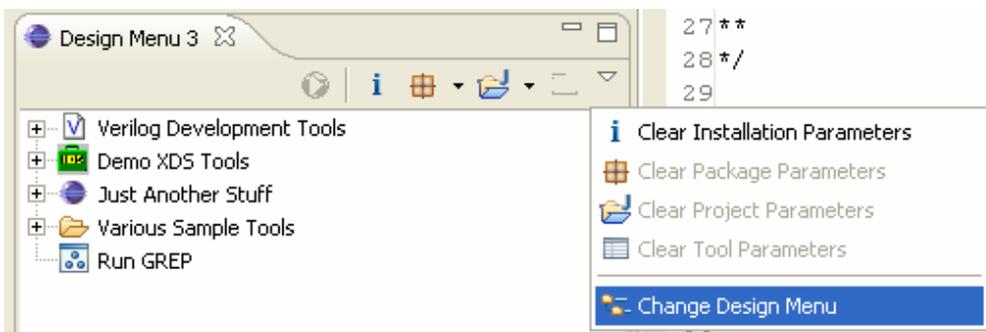
-  Запустить инструмент
-  Выполнить инсталляционные настройки
-  Настроить параметры пакетов
-  Настроить параметры проектов
-  Настроить параметры инструментов

Настройки параметров пакетов и проектов реализованы в виде выпадающих списков, содержащих имена всех пакетов/проектов, для которых есть возможность выполнить настройки параметров (см. *Настройки инструментов*).

Содержимое панели инструментов продублировано в контекстном меню, доступном по щелчку правой кнопки мыши в окне.

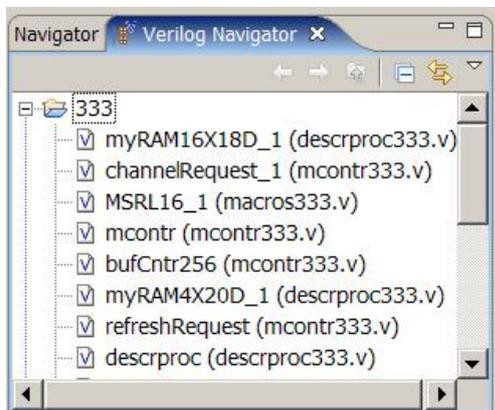
Набор инструментов, отображаемых в окне Design Menu, их иерархическая организация, способ запуска, набор параметров пакетов, проектов и утилит задаются TSL-спецификацией (см. *Редактирование TSL-спецификации параметров настройки*).

Для любого проекта в любой момент можно выбрать другое меню. Сделать это можно через специальное меню, которое вызывается при нажатии на белую стрелку на локальной панели инструментов окна Design Menu (самый правый элемент).

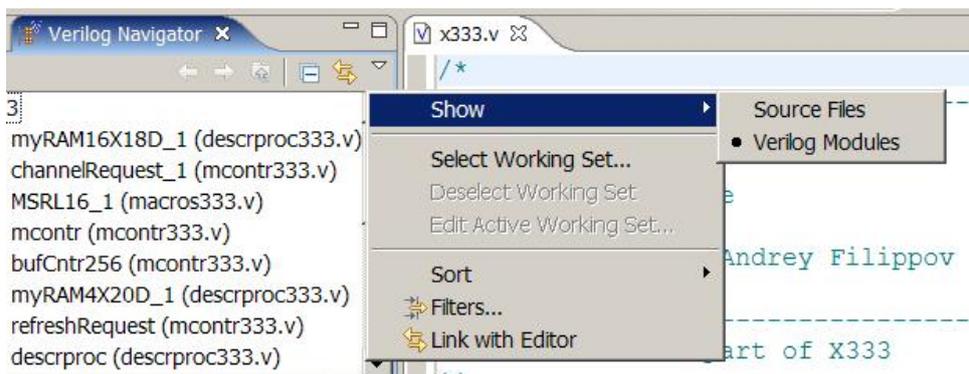


Окно Verilog Navigator

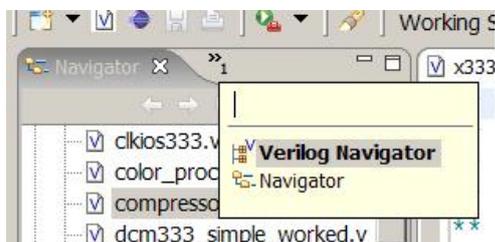
Окно Verilog Navigator отображает VDT-проекты в виде списка всех имеющихся в проекте Verilog-модулей, а не исходных файлов. После имени Verilog-модуля в круглых скобках указывается имя содержащего исходного файла:



Не-VDT-проекты показываются в виде файлов и каталогов в точности так же, как в окне Navigator. Более того, у окна Verilog Navigator есть режим, позволяющий и VDT-проекты показывать как набор файлов и каталогов. Для переключения режима нажмите белую стрелку в правой части локальной панели инструментов окна Verilog Navigator и выберите в пункте **Show** соответствующий режим:



По умолчанию окно Verilog Navigator находится за окном Navigator. Чтобы показать его, нажмите символ ">>", находящийся рядом с заголовком окна:



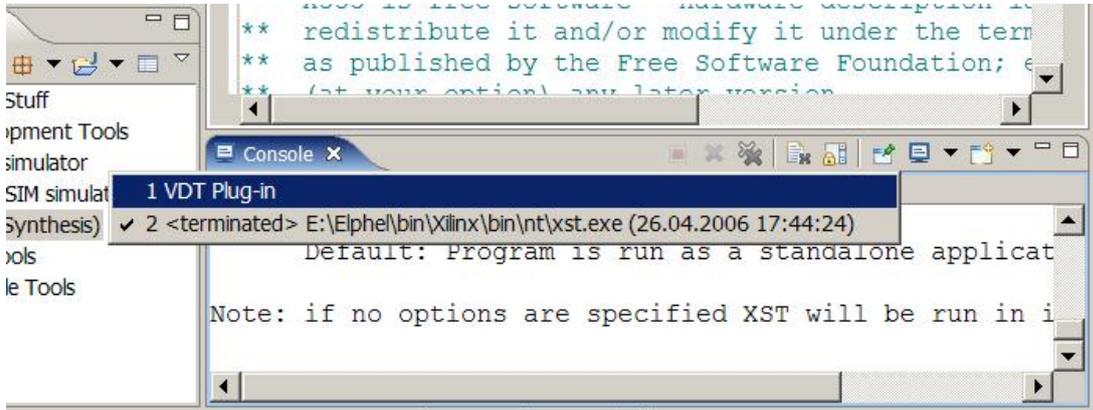
Окно Console

Окно **Console** (консоль) принимают стандартные потоки вывода запускаемых внешних утилит. Фактически оно поддерживает две консоли.

Одна консоль принимает вывод внешней утилиты. Ее содержимое сохраняется до запуска следующей утилиты.

Вторая консоль, с именем **VDT Plug-in**, принимает сообщения самого плагина VDT. Это, например, сообщения об ошибках синтаксического разбора Verilog-файлов проекта, который проводится для того, чтобы извлечь список модулей проекта для отображения в окне Verilog Navigator.

Для того чтобы переключить просмотр с одной консоли на другую, в локальной панели инструментов окна Console щелкните левой кнопкой мыши в стрелку справа от иконки «монитор» и в выпадающем списке выберите нужную консоль:



Настройки инструментов

Инструменты – это вызовы утилит, настроенные на различные режимы их работы. **Настройки инструментов** – это установки значений настроечных параметров, посредством чего пользователь управляет работой утилит.

Каждый параметр утилиты описывается следующими аспектами:

- типом и множеством возможных значений;
- форматом ввода, т.е. способом представления и ввода в диалоге настройки;
- форматом вывода, т.е. видом, в котором он передается утилите.

Набор и аспекты всех параметров полностью задаются в TSL-спецификации, в описаниях соответствующих TSL-**контекстов**. Эту *инсталляционную настройку* предварительно выполняет программист-настройщик среды программирования, который тем самым и определяет состав доступных в ней инструментов. Конечный пользователь выполняет только *оперативную настройку* инструментов, т.е. установку значений их параметров, управляющих режимом работы утилиты.

Оперативные настройки могут быть четырех уровней:

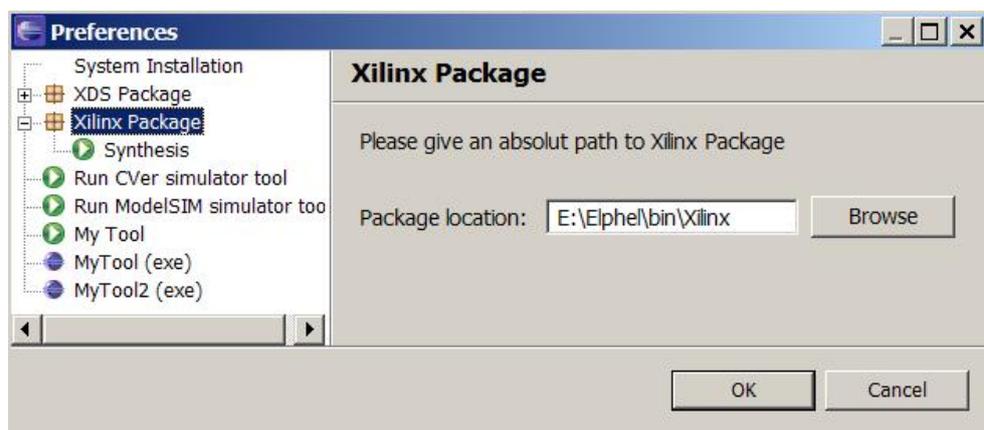
- инсталляционные – необходимые для того, чтобы утилита могла быть вообще запущена;
- пакетные (уровень конфигурационного файла);
- проектные (уровень проектного файла);
- сеансовые (уровень командной строки и командных файлов утилиты).

Инсталляционные настройки содержат некоторые предопределенные параметры, не указываемые в TSL-описании (например, путь до утилиты или директории пакета).

Инсталляционные настройки

Инсталляционные настройки задают пути до пакетов и исполняемых файлов утилит. Сюда же включаются параметры, заданные в TSL-контексте **installation**. Инсталляционные настройки едины для всех проектов; они сохраняются в метаданных плагина и не экспортируются вместе с проектом.

Диалог инсталляционных настроек открывается при нажатии кнопки  в локальной панели инструментов окна Design Menu. Диалог имеет две панели: в одной показывается список пакетов и утилит, в другой – набор параметров для выбранного элемента списка.



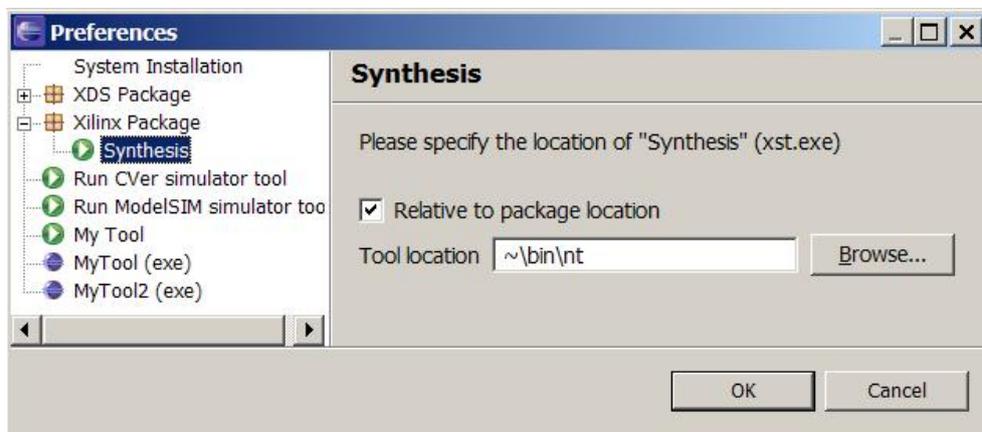
Первый элемент списка всегда соответствует параметрам, заданным в единственном TSL-контексте **installation**. Его название и вид панели настроек полностью определяются пользователем через описание этого контекста (см. документ *"Eclipse/ExDT – инструментарий для настройки интерактивных сред разработки (IDE)"*).

Далее в списке следуют пакеты, каждый из которых раскрывается в список инструментов, относящихся к этому пакету.

После пакетов следует список обособленных инструментов, не относящихся ни к какому пакету.

Для пакетов настраивается только один параметр – путь к местоположению пакета. Этот параметр может использоваться для настройки утилит пакета, которым нужно знать местоположение пакета. Указываемая директория должна, как минимум, существовать.

Для утилиты необходимо задать путь до соответствующего исполняемого файла. Если утилита входит в какой-либо пакет, это можно сделать двумя способами: либо задать абсолютный путь до исполняемого файла, либо путь относительно местоположения пакета, в который входит утилита:



Если указывается относительный путь, то путь до пакета, являющийся префиксом абсолютного пути, возвращаемого “Browse”, автоматически заменяется на ~.

Для утилит, не входящих в пакеты, всегда необходимо указывать абсолютный путь до исполняемого файла.

Пакетные настройки инструментов

Здесь, прежде всего, необходимо четко понять, что *пакетные настройки инструментов* относятся к утилитам, входящим в состав пакетов, а не к самим пакетам (т.к. пакеты не являются исполняемыми инструментами).

К **пакетным** относятся настройки утилиты, обычно задаваемые в её **конфигурационном файле** (если таковой предусмотрен), например:

- настройки параметров, характеризующих свойства инсталляции пакета на данной машине и инвариантные для всех проектов и вызовов утилиты;
- настройки умолчательных значений параметров утилиты; эти значения могут затем переопределяться на уровнях проектного файла и вызова.

Каждый настраиваемый параметр относится только к той утилите, для которой определен.

Для каждого пакета в VDT предусмотрен единственный *пакетный контекст* настройки (описываемый на TSL). Это означает, что:

- все параметры настройки пакетного уровня всех утилит из этого пакета, собраны воедино;
- все они настраиваются в общем диалоге настройки;
- все конфигурационные файлы утилит, содержащие эти параметры, порождаются на выходе из этого диалога.

Если в меню инструментов есть вызовы утилит из нескольких пакетов, может потребоваться отдельная настройка каждого из этих пакетов.

Настройки пакетов являются общими для всех проектов; они сохраняются в метаданных плагина и не экспортируются вместе с проектом.

Диалог для настройки пакета открывается при нажатии кнопки  в локальной панели инструментов окна Design Menu. При этом появляется выпадающий список имен пакетов, используемых в текущем меню инструментов. Диалог представляет собой набор закладок, содержащих группы параметров согласно TSL-спецификации.

Проектные настройки инструментов

Здесь, прежде всего, необходимо провести различие между *настройками проекта Eclipse* и *проектными настройками инструментов*.

К **настройкам проекта Eclipse** относится то, что знает о пользовательском проекте сама Eclipse: это список файлов проекта, меню утилит, список доступных утилит и пакетов, конфигурация и расположение окон, сохраняемые значения всех параметров и т.д. Эти настройки сохраняются в метаданных Eclipse, соответствующих текущему проекту; они не доступны напрямую ни пользователю, ни утилитами.

К **проектным настройкам инструмента** относится то, что знает о пользовательском проекте конкретная утилита. Это настройки, обычно задаваемые в её **проектном файле** (если таковой предусмотрен), например:

- настройки параметров, специфические для данного проекта и инвариантные для всех вызовов утилиты в данном проекте;
- настройки умолчательных значений параметров утилиты для проекта (в том числе, переопределения умолчательных пакетных настроек); эти значения могут затем переопределяться при вызове.

Каждый настраиваемый параметр относится только к той утилите, для которой определен.

В VDT для проекта может быть предусмотрено несколько проектных контекстов настройки (описываемых на TSL, где они называются **проектными шаблонами**, чтобы подчеркнуть отсутствие однозначного соответствия проектам Eclipse). Это означает, что:

- в одном контексте могут быть находиться параметры настройки проектного уровня как одной утилиты, так и нескольких; эти утилиты могут входить как в один пакет, так и в несколько;
- каждый контекст настраивается отдельным диалогом настройки;
- генерация каждого из проектных файлов связана с каким-либо из контекстов и происходит на выходе из соответствующего ему диалога настройки.

Проектные настройки утилит индивидуальны для каждого проекта Eclipse; они сохраняются в подкаталоге `.settings` текущего проекта и экспортируются вместе с ним.

Диалог для настройки проекта открывается при нажатии кнопки  в локальной панели инструментов окна Design Menu. При этом появляется выпадающий список имен проектных шаблонов, используемых в текущем меню утилит. Диалог представляет собой набор закладок, содержащих группы параметров согласно TSL-спецификации.

Сеансовые настройки инструментов

Утилиты вызываются из меню инструментов. Каждый элемент меню соответствует одному инструменту (т.е. вызову некоторой утилиты) и имеет индивидуальную настройку.

К **сеансовым** относятся настройки, передаваемые утилите непосредственно при вызове, исключая уже записанные в конфигурационном и проектном файлах (если они есть), например:

- настройки параметров, которые могут задаваться только в управляющей командной строке или командном файле;
- настройки параметров, специфические именно для этого вызова (т.е. характеризующие режим работы именно этого инструмента);
- настройки параметров пакетного и проектного уровня, переопределяемые для данного вызова.

Для каждого инструмента (вызова) в VDT предусмотрен единственный *сеансовый контекст* настройки (описываемый на TSL). Это означает, что:

- все параметры настройки вызова инструмента описываются вместе;
- все они настраиваются в одном диалоге настройки;
- командная строка и управляющие файлы утилиты, содержащие эти параметры, порождаются при запуске утилиты.

Сеансовые настройки индивидуальны для каждого проекта, сохраняются в каталоге `.settings` проекта и экспортируются вместе с ним.

Диалог настройки инструмента открывается при нажатии кнопки  в локальной панели инструментов окна Design Menu. Диалог представляет собой набор закладок содержащих группы параметров согласно TSL-описанию:

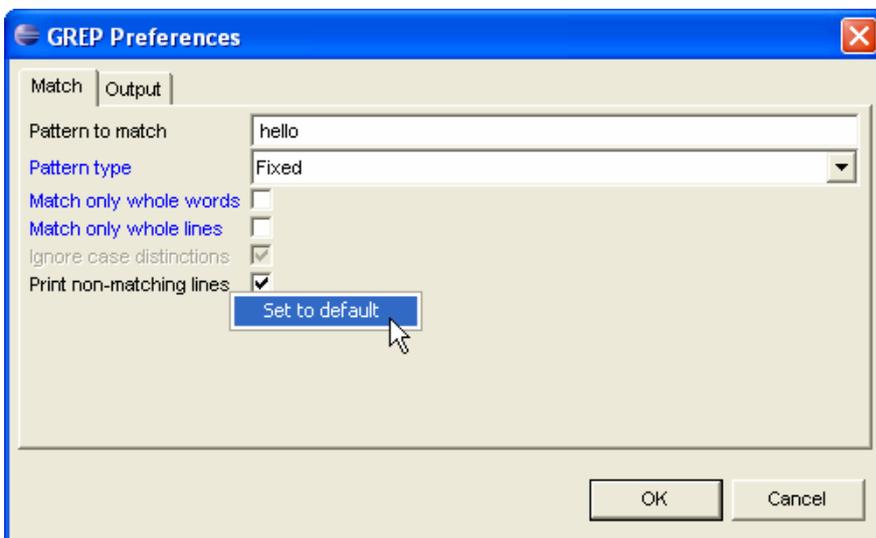
Общий вид диалога настройки

Общий вид всех диалогов настройки для всех контекстов примерно одинаков.

Закладки в диалоге соответствуют группам параметров, опеределенным в TSL-спецификации (см. закладки «Match» и «Output» в примере диалога ниже).

Каждая строка диалога задает один параметр и соержжит пояснительный текст (label) и поле ввода, вид которого зависит от типа параметра.

Обычно все тексты изображаются черным цветом. Неизменяемые параметры (с атрибутом “read-only”) затеняются серым цветом (и текст, и поле ввода). Если параметр находится в состоянии «по умолчанию», его текст выделяется синим цветом. Когда же его значение изменяется, текст снова становится черным. Для того, чтобы сбросить изменённое значение параметра в значение «по умолчанию», нужно щелкнуть по нему правой кнопкой мыши и в выпадающем контекстном меню выбрать “Set to default”.



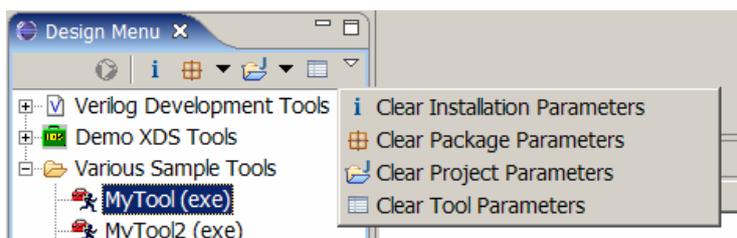
Есть ли разница между параметром, установленным в состояние «по умолчанию» и параметром, значение которого просто совпадает с умолчательным? Есть, и даже две:

- любое конкретное значение параметра сохранится до тех пор, пока не будет явно изменено, в то время как значение «по умолчанию» может быть унаследовано из некоторого другого контекста (например, пакетного) и будет автоматически изменено, как только его поменяют в этом «отчем» контексте;
- значение «по умолчанию» иногда может вовсе не передаваться утилите, и непередача значения может иметь смысл отличный от передачи любого конкретного значения (так, например, может быть задана логика трех состояний для флажков, со значениями «да», «нет» и «по умолчанию»).

Очистка настроек

Иногда бывает необходимо полностью удалить сохраненные значения параметров: например, если они стали некорректными после того, как в TSL-спецификации поменялись их форматы или множества допустимых значений.

Удалить сохраненные значения параметров можно через специальное меню, которое вызывается при нажатии на белую стрелку на локальной панели инструментов окна Design Menu (самый правый элемент):



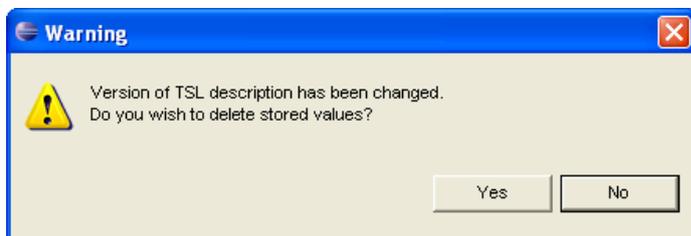
Пункты меню Clear Package Parameters и Clear Project Parameters удаляют сохраненные значения только для пакетов и проектов, доступных в данном Design Menu. Пункт Clear Tool Parameters удаляет сохраненные значения только для текущего инструмента.

Контроль версий

Ясно, что сохраненные значения параметров должны быть всегда согласованы со своими TSL-описаниями. Например, если значение некоторого параметра строкового типа сохранено, то при загрузке этого сохранённого значения параметр обязан иметь тот же самый строковый тип (включая длину и все атрибуты).

Однако, что будет, если TSL-описание данного инструмента изменилось и тип этого параметра стал логическим? Понятно, что сохранённое значение не соответствует новому типу и при загрузке значения будет обнаружена ошибка. В некоторых случаях (например, когда и старый, и новый типы имеют числовую базу) старое значение параметра может быть формально корректным значением нового типа, но семантически неверным.

В TSL предусмотрен механизм, позволяющий пользователю задавать версию TSL-файла. При сохранении любых настроек сохраняется также версия соответствующего TSL-файла. При считывании настроек сохранённая версия считывается и сверяется с текущей. В случае несовпадения версий пользователю выдаётся предупреждение с предложением удалить все предыдущие настройки, чтобы избежать возможных коллизий.



Если пользователь отклоняет предложение, то считается, что все сохранённые настройки верны, а возможные ошибки остаются на его совести.

Запуск инструментов

Для запуска инструмента:

1) Выберите в окне Navigator файл (либо в окне Verilog Navigator выберите модуль), для которого будет запущен инструмент.

2) В окне Design Menu два раза щелкните мышью на нужном инструменте. Если в окне Design Menu уже выбрана нужная утилита, то ее можно вызвать, нажав в локальной панели инструментов этого окна кнопку **Run** .

Если исполняемый файл, соответствующий утилите, не может быть найден (не были произведены инсталляционные настройки), то будет выдан диалог запроса директории.

Редактирование TSL-спецификации параметров настройки

Состав и внешний вид меню и диалогов настройки, а также способы вызова внешних утилит полностью описаны в TSL-спецификации динамической настройки Eclipse/VDT, которая хранится в отдельном каталоге в виде набора XML-файлов.

Язык спецификации TSL описан в отдельном документе.

В версии 1.0.2 директория с TSL-спецификацией находится в инсталляционной директории Eclipse и имеет имя `plugins/com.elphel.vdt_1.1.0/tools`. Для внесения изменений в спецификацию необходимо редактировать содержащие ее файлы любым редактором текстов. В дальнейшем предполагается создавать пользовательскую копию этой директории, предназначенную для редактирования, а также разработать специальный инструмент с графическим интерфейсом для удобства внесения дополнений и изменений.