

Функциональное моделирование цифровых устройств, проектируемых на базе ПЛИС фирмы Xilinx в среде САПР WebPACK ISE

В предыдущих публикациях цикла [1–8] подробно рассмотрены основные этапы процесса проектирования цифровых устройств на базе ПЛИС фирмы Xilinx® в среде пакета САПР WebPACK™ ISE™ (Integrated Synthesis Environment), за исключением моделирования. Как упоминалось ранее, в состав пакета WebPACK ISE включена программа моделирования ModelSim™ (в виде редакции ModelSim XE Starter) корпорации Model Technology®, являющейся одним из подразделений компании Mentor Graphics®. Основные характеристики и пользовательский интерфейс этой системы HDL-моделирования представлены в предыдущей статье [9].

Валерий Зотов

walerry@euro.ru

В процессе проектирования средства пакета WebPACK ISE формируются несколько моделей разрабатываемого устройства. После создания модулей исходного описания проекта генерируется поведенческая модель, которая позволяет выполнить их функциональную верификацию. На этой стадии проектирования отсутствует информация о значениях задержек распространения сигналов, поэтому при функциональном моделировании можно обнаружить только логические и синтаксические ошибки в описании разрабатываемого устройства. Состав моделей, формируемых на последующих этапах процесса проектирования, зависит от типа семейства ПЛИС (FPGA, Field Programmable Gate Array или CPLD — Complex Programmable Logic Device), выбранного для реализации разрабатываемого устройства.

Настоящая статья посвящена изучению процессов подготовки описаний тестовых воздействий и функционального моделирования проектируемого устройства.

Создание тестового модуля проекта


Тестовый модуль проекта фактически содержит представление модели испытательного стенда для разрабатываемого устройства на языке описания аппаратуры HDL (Hardware Description Language), используемом в процессе проектирования. В дальнейшем предполагается, что процесс разработки выполняется при использовании средств синтеза XST (Xilinx Synthesis Technology) VHDL™. В этом случае тестовый модуль проекта имеет стандартную структуру VHDL-описания, которая включает в себя следующие элементы:

- ссылки на используемые библиотеки и пакеты;
- описание интерфейса объекта (ENTITY);
- описание архитектуры объекта;
- декларацию компонента, представляющего модуль описания верхнего уровня иерархии проектируемого устройства;

- декларацию сигналов, используемых для подачи входных тестовых воздействий и контроля выходных реакций моделируемой системы;
- выражение создания экземпляра компонента с подключением соответствующих сигналов;
- формирование поведения входных тестовых сигналов.

В данном случае под объектом описания понимается модель испытательного стенда в целом, поэтому он не имеет интерфейсных сигналов (портов). Архитектура этого объекта содержит единственный компонент, представляющий разрабатываемое устройство. На рис. 1 приведен сокращенный вариант модуля тестовых воздействий для счетчика Джонсона, проектирование которого, в качестве примера, рассмотрено в ранее опубликованных статьях серии [1–8].

«Мастер» подготовки модулей исходного описания проекта позволяет автоматически сформировать основу тестового файла, которая содержит все перечисленные выше элементы его структуры, за исключением секции, описывающей поведения входных сигналов. Формирование входных тестовых воздействий может выполняться в текстовом виде (на языке VHDL) при использовании встроенного HDL-редактора или в графической форме (в виде временных диаграмм) с помощью утилиты генерации тестов *HDL Benchner™*, которая входит в состав пакета WebPACK ISE. Последний способ является наиболее наглядным и не требует знаний языков HDL. Программа *HDL Benchner* автоматически транслирует созданные временные диаграммы тестовых сигналов при их сохранении в текстовый формат.

Для автоматического формирования основы файла тестовой последовательности необходимо активизировать режим создания нового модуля исходного описания проекта, воспользовавшись кнопкой  на оперативной панели или командой *New Source* из раздела *Project* основного меню *Навигатора проек-*

```
-- VHDL Test Bench for functional and timing simulation

-- ссылки на используемые библиотеки и пакеты
LIBRARY IEEE;
USE IEEE.std_logic_1164.all;

-- описание интерфейса объекта
ENTITY testbench IS
END testbench;

-- описание архитектуры объекта
ARCHITECTURE testbench_arch OF testbench IS
-- декларация компонента, представляющего модуль описания
верхнего уровня иерархии проекта
    COMPONENT jc2_top
        PORT (
            CLK : in STD_LOGIC;
            LEFT : in STD_LOGIC;
            RIGHT : in STD_LOGIC;
            STOP : in STD_LOGIC;
            Q : inout STD_LOGIC_VECTOR (3 DOWNTO 0)
        );
    END COMPONENT;
-- декларация сигналов, используемых для подачи входных
тестовых воздействий и контроля
    SIGNAL CLK : STD_LOGIC;
    SIGNAL LEFT : STD_LOGIC;
    SIGNAL RIGHT : STD_LOGIC;
    SIGNAL STOP : STD_LOGIC;
    SIGNAL Q : STD_LOGIC_VECTOR (3 DOWNTO 0);
BEGIN
-- создание экземпляра компонента с подключением
соответствующих сигналов
    UUT : jc2_top
    PORT MAP (
        LEFT => LEFT,
        RIGHT => RIGHT,
        STOP => STOP,
        CLK => CLK,
        Q => Q
    );
-- формирование поведения входных тестовых сигналов
    PROCESS
        BEGIN
            -----
            CLK <= transport '0';
            LEFT <= transport '1';
            RIGHT <= transport '1';
            STOP <= transport '1';
            -----
            WAIT FOR 110 ns;
            CLK <= transport '1';
            -----
            WAIT FOR 10 ns;
            CLK <= transport '0';
            -----
            WAIT FOR 10 ns;
            CLK <= transport '1';
            -----
            ...
            -----
            WAIT;
        END PROCESS;
END testbench_arch;
```

Рис. 1. Сокращенный вариант модуля тестовых воздействий для проекта счетчика Джонсона

та. В открывшейся диалоговой панели, показанной на рис. 2, в качестве типа нового модуля следует выбрать *VHDL Test Bench* для текстовой формы представления тестовых воздействий или *Test Bench Waveform* для формирования временных диаграмм в графическом режиме. Далее необходимо задать его имя и указать место расположения файла на диске.

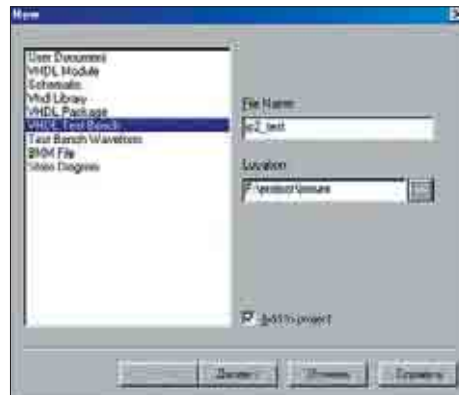


Рис. 2. Диалоговая панель установки исходных параметров тестового модуля проекта

Место расположения создаваемого модуля на диске указывается в поле редактирования *Location* диалоговой панели (рис. 2). Для хранения тестового файла рекомендуется использовать рабочий каталог текущего проекта, предлагаемый по умолчанию. При необходимости можно выбрать иной каталог, воспользовавшись клавиатурой или стандартной панелью навигации по дискам компьютера, как подробно рассмотрено ранее [2]. Чтобы созданный тестовый модуль был автоматически включен в состав текущего проекта, флаг индикатора *Add to project* должен находиться в установленном состоянии (поле индикатора помечено маркером). Для модификации этого параметра достаточно щелкнуть левой кнопкой мыши, поместив курсор на поле индикатора. При этом состояние индикатора изменится на противоположное.

Установка значений всех необходимых исходных параметров создаваемого модуля завершается нажатием клавиши «Далее», которая находится в нижней части диалоговой панели (рис. 2). На экране отображается следующая диалоговая панель, содержащая список модулей исходного описания проекта (рис. 3). В списке следует выделить элемент, с которым ассоциируется создаваемый тестовый модуль, поместив курсор на соответствующую строку и щелкнув левой кнопкой мыши.



Рис. 3. Диалоговая панель установки соответствия тестового файла и модуля исходного описания проекта

При успешном создании основы тестового модуля открывается информационная панель, показанная на рис. 4. В этой панели отображается информация обо всех параметрах созданного модуля. Если необходимо изменить значение какого-либо параметра, то кнопка «Назад» позволяет вернуться к предыдущей диалоговой панели. Процесс создания основы тестового модуля завершается нажатием кнопки «Готово», находящейся в нижней части информационной панели (рис. 4).

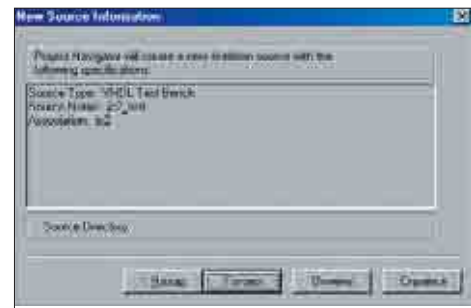


Рис. 4. Информационная панель параметров создаваемого тестового модуля проекта

После этого в окне исходных модулей *Навигатора проекта* добавляется пиктограмма, сопровождаемая названием созданного тестового модуля. Одновременно автоматически открывается окно встроенного HDL-редактора, в котором отображается код сформированного описания модели испытательной системы, если была выбрана текстовая форма представления тестовых воздействий (строка *VHDL Test Bench* — рис. 2). Далее необходимо дополнить сформированный модуль выражениями, описывающими изменение во времени входных сигналов. После завершения редактирования следует обязательно выполнить команду сохранения тестового файла.

Если была выбрана графическая форма представления тестовых воздействий в виде временных диаграмм (строка *Test Bench Waveform*, рис. 2), то автоматически производится запуск программы *HDL Benchner*. Прежде чем приступить к редактированию диаграмм тестовых сигналов, необходимо установить значения исходных временных параметров, которые представлены в диалоговой панели (рис. 5). Эта панель автоматически выводится на экран при активизации утилиты *HDL Benchner*.

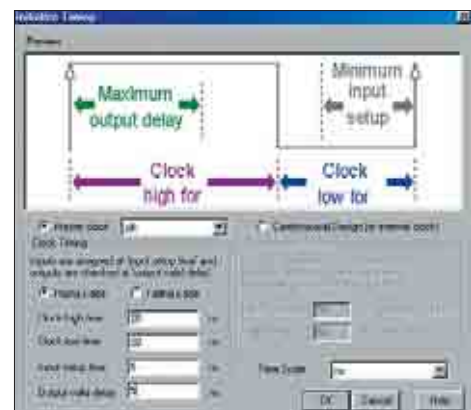


Рис. 5. Диалоговая панель инициализации временных параметров

Временные параметры, необходимые для формирования тестовых сигналов, различаются для синхронных и асинхронных комбинационных устройств. Программа *HDL Bench* автоматически определяет тип устройства, для которого разрабатываются тестовые последовательности. Для синхронного устройства необходимо, прежде всего, проконтролировать и при необходимости изменить выбор основного тактового сигнала в поле *Master clock*, используя выпадающий список интерфейсных сигналов устройства. Затем следует выбрать единицы измерения временных параметров, используя выпадающий список возможных значений поля *Time scale*. Далее нужно определить длительность состояний высокого и низкого логического уровня периода тактового сигнала в поле редактирования *Clock high time* и *Clock low time* соответственно. Кроме того, в поле редактирования *Input setup time* следует указать минимальное значение времени установки входных сигналов, а в поле *Output valid delay* — максимальное значение задержки выходных сигналов. Программа *HDL Bench* автоматически определяет, чем тактируется устройство — фронтом (*Rising edge*) или спадом (*Falling edge*) тактового сигнала, но разработчик может выбрать это вручную, используя соответствующие кнопки диалоговой панели (рис. 5). При создании тестов для комбинационных устройств нужно указать только значения временных интервалов между моментами подачи входных воздействий и контроля выходных сигналов *Check outputs* и *Assign Inputs*. Эти параметры позволяют исключить возможные конфликты в моменты изменения входных и выходных сигналов.

Процесс установки требуемых значений временных параметров завершается нажатием кнопки ОК, расположенной в нижней части диалоговой панели (рис. 5), после чего в рабочей области окна программы *HDL Bench* отображаются заготовки временных диаграмм (рис. 6).

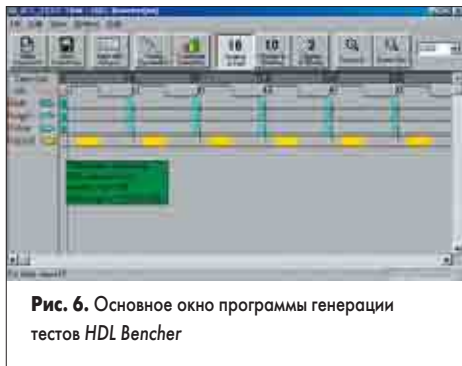


Рис. 6. Основное окно программы генерации тестов *HDL Bench*

В верхней строке рабочей области окна программы *HDL Bench* расположена временная шкала, деления которой соответствуют периодам тактового сигнала. В левом столбце отображается список названий интерфейсных сигналов устройства. Справа от названия сигнала приведена пиктограмма, обозначающая его тип:

- тактовый;
- входной;
- выходной;
- двунаправленный.

Сигналы векторного типа отображаются в виде шин соответствующей разрядности.

Изменить значение сигнала в какой-либо момент времени можно несколькими способами. Чтобы переключить значение сигнала на противоположное — из состояния низкого логического уровня в высокое или наоборот, — достаточно поместить курсор в требуемое место временной диаграммы и щелкнуть левой кнопкой мыши. Чтобы выбрать значение сигнала из списка возможных состояний, следует щелчком правой кнопки активизировать контекстно-зависимое всплывающее меню, в котором выбрать строку *Set Value*. На экране появится диалоговая панель, содержащая поле выбора значения сигнала (рис. 7). Для установки требуемого значения сигнала следует нажать кнопку управления выпадающим списком возможных состояний, в котором выбрать затем соответствующую строку, после чего подтвердить выбор нажатием кнопки ОК (рис. 7).

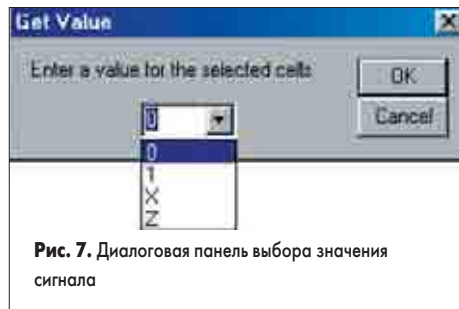


Рис. 7. Диалоговая панель выбора значения сигнала

Чтобы сформировать последовательность значений для выбранного сигнала, можно воспользоваться соответствующим «мастером» *Pattern Wizard*. Для его активизации необходимо поместить курсор мыши в исходную точку временной диаграммы и дважды щелкнуть левой кнопкой, после чего воспользоваться появившейся кнопкой *Pattern*. В *Pattern Wizard* (рис. 8), следует выбрать алгоритм переключений сигнала в поле *Choose Pattern* и указать: начальное значение в поле редактирования *Initial Value*, альтернативное значение в поле *Other Value*, интервал переключения в поле *Toggle Every* и количество формируемых периодов сигнала в поле *Repeat pattern*. После нажатия кнопки ОК, расположенной в нижней части диалоговой панели (рис. 8), сформированный фрагмент

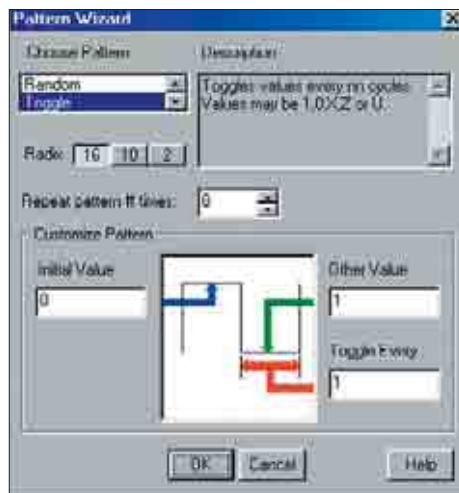


Рис. 8. Диалоговая панель «мастера» *Pattern Wizard*

временной диаграммы отображается в рабочей области. Команды меню *Edit* и контекстно-зависимого всплывающего меню позволяют копировать в буфер и затем вставлять в нужное место выбранные фрагменты временных диаграмм.

На изображении временных диаграмм присутствует указатель конечной точки тестовой последовательности, положение которого устанавливается автоматически или вручную. Выбор режима определения окончания тестового вектора осуществляется с помощью команды *Configuration* из всплывающего меню *Options*, в результате выполнения которой на экран выводится диалоговая панель, показанная на рис. 9. Параметр *Automatically determine end of test bench* управляет выбором режима, в котором выполняется установка маркера конечной точки тестового вектора. По умолчанию используется значение «включено», соответствующее автоматическому режиму, при котором указатель располагается в начале тактового интервала, следующего за моментом последнего изменения входных сигналов. При перемещении маркера с помощью мыши происходит переключение из автоматического режима в ручной.

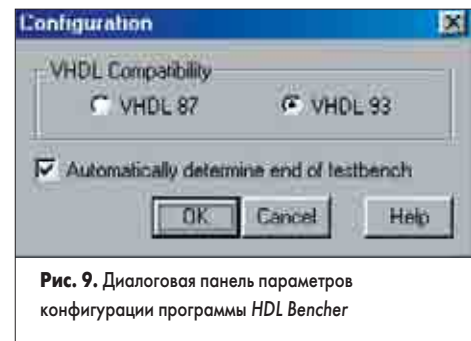


Рис. 9. Диалоговая панель параметров конфигурации программы *HDL Bench*

Параметр *VHDL Compatibility Mode*, представленный в этой же диалоговой панели в виде двух кнопок, позволяет выбрать стандарт языка VHDL, в соответствии с которым формируется описание тестовой системы. По умолчанию установлен режим совместимости со стандартом IEEE VHDL Std 1076-1993.

Завершив редактирование временных диаграмм тестовых сигналов, следует сохранить их, используя команду *Save Waveform* из меню *File* или кнопку на оперативной панели окна *HDL Bench*. При этом, если указатель окончания тестовой последовательности был установлен до момента последнего изменения входных или контроля формирования выходных сигналов, выводится панель предупреждения, в которой предлагается перенести маркер в соответствующую позицию.

По окончании формирования тестового файла следует выделить строку с его названием в окне исходных модулей *Навигатора проекта* (рис. 10), поместив на нее курсор мыши и щелкнув левой кнопкой. В результате в окне процессов отображается интерактивный список этапов моделирования проектируемого устройства. Содержание окна процедур в этом режиме определяется видом семейства ПЛИС, выбранного для реализации проекта, но независимо от типа используемого кристалла первым в списке является этап функционального моделирования.

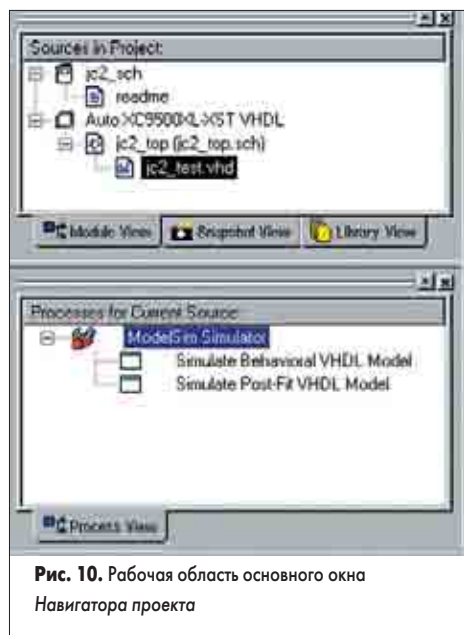



Рис. 10. Рабочая область основного окна Навигатора проекта

Установка значений параметров функционального моделирования проекта

Перед запуском средств моделирования следует проконтролировать и при необходимости установить требуемые значения параметров инициализации программы *ModelSim* и соответствующей модели. Для этого нужно в окне процессов (рис. 10) щелчком левой кнопки мыши выделить строку *Simulate Behavioral VHDL Model*, после чего нажать кнопку , расположенную на оперативной панели *Навигатора проекта*, или воспользоваться командой *Properties* контекстно-зависимого всплывающего меню, которое выводится при щелчке правой кнопки мыши. В результате выполненных действий на экране монитора отображается диалоговая панель параметров процесса моделирования (рис. 11).

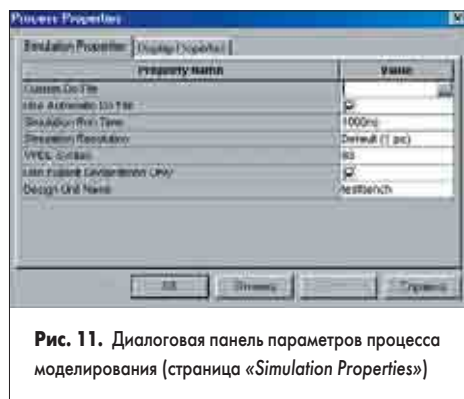


Рис. 11. Диалоговая панель параметров процесса моделирования (страница «Simulation Properties»)

Эта диалоговая панель в случае функционального моделирования содержит две страницы, снабженные закладками с их названиями: «*Simulation Properties*», «*Display Properties*». Каждая из этих страниц содержит соответствующую группу параметров, представленных в виде таблицы, структура которой подробно рассмотрена в первых статьях [1, 2]. Чтобы изменить значение параметра, следует щелчком левой кнопки мыши активизировать соответствующее поле таблицы, а затем воспользоваться кнопкой управления выпадающим списком значений или, если параметр имеет два состояния («включено»/«выключено»), щелк-

нуть левой кнопкой мыши на поле индикатора состояния. Для некоторых параметров требуемое значение может быть введено непосредственно с клавиатуры. Содержание страниц в диалоговой панели параметров моделирования зависит от выбранного семейства ПЛИС и языка описания HDL. Рассмотрим подробнее эти параметры для случая использования семейств ПЛИС CPLD и средств синтеза XST VHDL.

На странице «*Simulation Properties*» представлены общие параметры процесса моделирования. Параметр *Custom Do File* позволяет указать название командного файла, который будет выполняться при активизации средств моделирования в дополнение к файлу, формируемому автоматически пакетом WebPACK ISE. Название командного файла может быть введено непосредственно с клавиатуры после активизации поля редактирования значения этого параметра или выбрано при использовании стандартной диалоговой панели открытия файла, которая выводится при нажатии кнопки с пиктограммой в виде многоточия («...»).

Значение параметра *Use Automatic Do File* разрешает или запрещает автоматическое создание и исполнение командного файла, который содержит директивы компиляции всех VHDL-файлов проекта, активизации процесса моделирования и отображения его результатов. По умолчанию установлено значение «включено», разрешающее генерацию и выполнение этого пакетного файла.

Simulation Run Time устанавливает длительность интервала моделирования, которое выполняется автоматически в пакетном режиме. Значение этого параметра указывается с помощью клавиатуры после активизации соответствующего поля редактирования. По умолчанию длительность интервала моделирования принимается равной 1000 нс.

Значение параметра *Simulation Resolution* определяет разрешающую способность процесса моделирования, то есть величину временной дискретизации, с которой вычисляется состояние проектируемого устройства. Требуемое разрешение выбирается из выпадающего списка возможных значений, который содержит 18 позиций, отличающихся на порядок по отношению к соседним. По умолчанию величина временной дискретизации моделирования установлена равной 1 пс.

С помощью параметра *VHDL Syntax* выбирается стандарт синтаксиса (версия) языка VHDL, в соответствии с которым формируется описание моделируемой системы. Выпадающий список возможных значений этого параметра содержит две строки: 93 (установлено по умолчанию) и 87, которые соответствуют стандартам IEEE VHDL Std 1076-1987 и IEEE VHDL Std 1076-1993.

Параметр *Use Explicit Declarations Only* используется для предотвращения конфликтов из-за неоднозначности, возникающей при использовании перегрузки функций. Значение «включено», принятое по умолчанию для данного параметра, устанавливает более высокий приоритет явного определения функций, чем для случая их неявного использования.

Значение параметра *Design Unit Name* определяет название моделируемого устройства.

По умолчанию используется значение «*testbench*», которое устанавливается в качестве названия объекта модели испытательного стенда при автоматическом формировании тестового модуля средствами пакета WebPACK ISE.

На странице «*Display Properties*» представлены параметры, управляющие отображением окон программы моделирования *ModelSim* при ее активизации (рис. 12). При запуске средств моделирования непосредственно из управляющей оболочки пакета WebPACK ISE автоматически открываются только те окна *ModelSim*, для которых соответствующий параметр установлен в состояние «включено». Параметр *Signal window* определяет режим отображения окна сигналов, *Wave window* — окна временных диаграмм, *Structure window* — окна структуры проекта, *Source window* — окна исходного кода, *List window* — окна табличной формы результатов моделирования, *Variables window* — окна переменных, *Process window* — окна процессов, *Data Flow window* — окна трассировки сигналов. По умолчанию в начале сеанса моделирования автоматически открываются окна сигналов, временных диаграмм и структуры проекта.

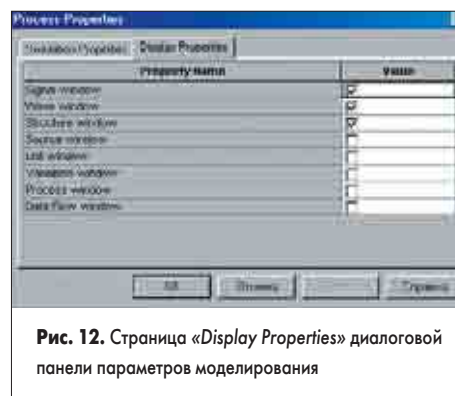


Рис. 12. Страница «Display Properties» диалоговой панели параметров моделирования

Выполнив необходимые изменения параметров процесса моделирования, следует подтвердить их нажатием клавиши OK в нижней части диалоговой панели (рис. 11, 12).

Функциональное моделирование проекта с помощью программы ModelSim

Процесс функционального моделирования проекта активизируется двойным щелчком левой кнопки мыши на строке «*Simulate Behavioral VHDL Model*» в окне процессов *Навигатора проекта*. При этом средствами пакета WebPACK ISE автоматически создается пакетный файл, который содержит последовательность команд управления программой *ModelSim*, необходимых для проведения сеанса функционального моделирования проектируемого устройства. Далее автоматически производится запуск средств моделирования, сразу после инициализации которых выполняется этот командный файл (если только параметр *Use Automatic Do File* не был установлен в состояние «выключено»). Информация о ходе выполнения пакетного файла отображается в консольной области основного окна программы *ModelSim*. При этом последовательно открываются дочерние окна этой программы, которые были указаны в параметрах инициализации.

После завершения выполнения командного файла при необходимости моделирование может быть продолжено в интерактивном режиме. На рис. 13 приведены в качестве примера результаты функционального моделирования счетчика Джонсона, разработка которого рассмотрена в предыдущих публикациях цикла.

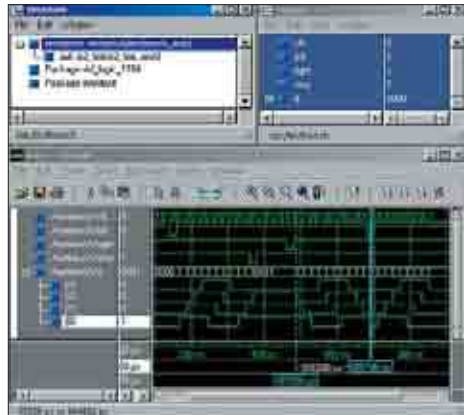









Рис. 13. Результаты функционального моделирования проекта счетчика Джонсона

Как правило, наибольший интерес для разработчика представляют результаты моделирования, которые отображаются в графическом виде в окне временных диаграмм программы ModelSim. Для их просмотра, прежде всего, необходимо выбрать требуемый масштаб изображения. Увеличение масштаба осуществляется с помощью кнопки , расположенной на оперативной панели управления окна временных диаграмм, или команды *Zoom In* из всплывающего меню *Zoom*. Для уменьшения масштаба изображения используется кнопка быстрого доступа  на оперативной панели или команда *Zoom Out*. Однократное применение этих инструментов приводит к изменению масштаба в два раза. Для детального просмотра выбранного фрагмента временных диаграмм следует воспользоваться кнопкой  на оперативной панели или командой *Zoom Area with Mouse Button 1*. При этом, чтобы указать интересующую область изображения результатов моделирования, следует поместить курсор мыши на одну из ее границ, нажать левую кнопку мыши и, не отпуская ее, растянуть появившийся прямоугольный контур до требуемых размеров, после чего освободить нажатую кнопку.

Перемещение по временным диаграммам осуществляется с помощью элементов прокрутки, расположенных вдоль нижней и правой границ панели результатов моделирования в окне временных диаграмм. Кроме того, на оперативной панели имеются кнопки быстрого доступа  и , обеспечивающие перемещение курсора в предыдущую или следующую точку переключения выбранного сигнала. При открытии окна диаграмм сигналов курсор находится в начальной точке временной шкалы, которая соответствует нулевому значению времени моделирования. Чтобы переместить его в требуемую точку, следует расположить указатель мыши в этой позиции и щелкнуть левой кнопкой, после чего изображение курсора будет зафиксировано в новом положении. Допускается также перетаскивание курсора

в новую точку с помощью мыши, которое выполняется при нажатой левой кнопке.

Для измерения временных интервалов на диаграммах сигналов, полученных в результате моделирования, следует использовать дополнительные курсоры. Чтобы добавить еще один курсор, необходимо нажать кнопку  на оперативной панели управления или выбрать команду *Add Cursor* из всплывающего меню *Cursor* окна временных диаграмм. При наличии нескольких указателей активный курсор изображается сплошной линией, а остальные — штриховой. Поместив курсоры на границы измеряемого временного интервала, можно определить его длительность, которая отображается в нижней части окна диаграмм в виде разности координат указателей. При необходимости можно использовать несколько дополнительных курсоров. В последующем для удаления неиспользуемых указателей можно воспользоваться кнопкой  на оперативной панели или командой *Delete Cursor* из всплывающего меню *Cursor*.

В ряде случаев для большей наглядности результатов моделирования можно представить совокупность выбранных сигналов в виде шины. Для этого необходимо выделить названия соответствующих сигналов в окне временных диаграмм и выбрать команду *Combine* из всплывающего меню *Edit*, после чего выводится диалоговая панель, показанная на рис. 14. В поле редактирования *Name* с помощью клавиатуры необходимо указать название формируемого виртуального сигнала (шины). Кнопки *Ascending* и *Descending* позволяют выбрать порядок следования сигналов в шине. Параметр *Remove selected signals after combining* предоставляет возможность удаления сигналов после их объединения в виртуальную шину. По умолчанию для этого параметра установлено значение «выключено», запрещающее удаление из окна временных диаграмм сигналов, вошедших в состав виртуальной шины. Сформированный виртуальный сигнал отображается в списке пиктограммой в форме ромба красного цвета, в отличие от реальных сигналов, обозначаемых пиктограммой в виде квадрата синего цвета.

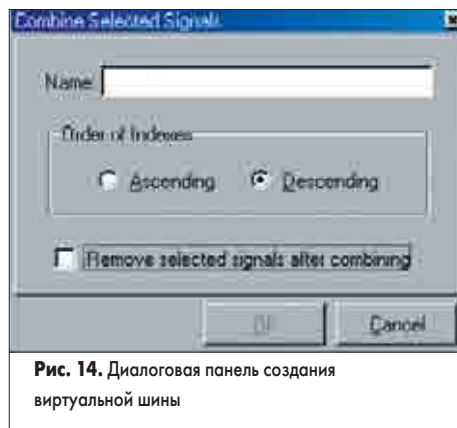




Рис. 14. Диалоговая панель создания виртуальной шины

При автоматическом создании командного файла для сеанса моделирования средства пакета WebPACK ISE включают в список контролируемых сигналов только интерфейсные сигналы моделируемого объекта. Но в процессе отладки часто возникает необходимость отображения поведения внутренних сигналов

в различных модулях проекта. Для этого следует в иерархической структуре проекта, представленной в окне *Structure*, выбрать интересующий объект, поместив курсор мыши на строку с его названием и щелкнув левой кнопкой мыши. При этом в окне *Signals* отобразится соответствующий список сигналов, в котором следует захватить требуемый элемент, поместив курсор мыши на строку с его названием, нажав левую кнопку, и, не отпуская ее, перетащить его в область названий сигналов окна временных диаграмм. Далее можно продолжить моделирование с текущего момента времени или повторить выполненные процедуры с начала. В последнем случае следует сбросить результаты моделирования и вернуть систему в начальное состояние, выполнив команду *Restart* из всплывающего меню *Run* основного окна программы ModelSim, или нажав кнопку  на оперативной панели. Повторное моделирование инициируется командой *Run-All* или кнопкой быстрого доступа .

Результаты моделирования, полученные в текущем сеансе, автоматически сохраняются в виде набора данных (*dataset*) с именем *vsim* в формате WLF (Wave log format).

Для завершения этапа функционального моделирования следует выполнить команду *End Simulation* из меню *Design* и закрыть основное окно программы ModelSim.

Другие виды моделирования будут рассмотрены в следующей публикации цикла. ■

Литература

1. Зотов В. WebPACK ISE — свободно распространяемый пакет проектирования цифровых устройств на базе ПЛИС фирмы Xilinx // Компоненты и технологии. 2001. № 6.
2. Зотов В. WebPACK ISE: Интегрированная среда разработки конфигурации и программирования ПЛИС фирмы Xilinx. Создание нового проекта // Компоненты и технологии. 2001. № 7.
3. Зотов В. Схемотехнический редактор пакета WebPACK ISE. Создание принципиальных схем и символов // Компоненты и технологии. 2001. № 8.
4. Зотов В. Синтез, размещение и трассировка проектов, реализуемых на базе ПЛИС CPLD фирмы Xilinx, в САПР WebPACK ISE // Компоненты и технологии. 2002. № 1.
5. Зотов В. Программирование ПЛИС-семейств CPLD фирмы Xilinx в САПР WebPACK ISE. // Компоненты и технологии. 2002. № 2.
6. Зотов В. Синтез проектов, реализуемых на базе ПЛИС FPGA фирмы Xilinx, в САПР WebPACK ISE // Компоненты и технологии. 2002. № 3.
7. Зотов В. Реализация проектов на базе ПЛИС-семейств FPGA фирмы Xilinx в САПР WebPACK ISE // Компоненты и технологии. 2002. № 4.
8. Зотов В. Конфигурирование ПЛИС-семейств FPGA фирмы Xilinx в САПР WebPACK ISE // Компоненты и технологии. 2002. № 5.
9. Зотов В. ModelSim — система HDL-моделирования цифровых устройств // Компоненты и технологии. 2002. № 6.