**Специализированный стенд для тестирования поведенческих моделей операционных устройств TEST\_ALG**

1. **Структура стенда TEST\_ALG**

Стенд представляет собой проект с именем TEST\_ALG, выполненный в САПР QUARTUS II с использованием языка VHDL. Все файлы проекта находятся в одноименной папке. Модуль верхнего уровня стенда содержится в одноименном файле с расширением bdf. Он представляет собой схемный проект, приведенный на рис. 1.

 

Рис. 1 Структура специализированного стенда TEST\_ALG

Модуль **multiply**, синтезированный в QUARTUS по поведенческому описанию операционного устройства, созданного студентом, входит в состав стенда в качестве испытуемого модуля. Для задания его разрядности используется настраиваемый параметр n. Помимо испытуемого модуля в состав стенда входят следующие блоки: генератор тестовых наборов **gen\_test**, блок формирования эталонного результата **actual\_result** и блок анализа результата **analise\_unit**. Описания этих блоков на языке VHDL содержатся в одноименных файлах с расширением vhd. Причем в качестве настраиваемого параметра в них используется вышеупомянутый параметр n.

Генератор тестовых наборов представляет собой двоичный счетчик, который формирует последовательные двоичные наборы, разрядностью 2n, начиная с начального и до конечного. Начальное и конечное его значения задаются в виде констант start\_patern и stop\_patern, которые легко можно изменить. По умолчанию начальное значение представляет комбинацию из всех нулей. Конечное значение – комбинацию из всех единиц. То есть, по умолчанию генератор перебирает все возможные значения, обеспечивая тем самым исчерпывающее тестирование операционного устройства. Младшие n разрядов счетчика используются для задания множимого, а старшие n разрядов – для задания множителя. После того, как генератор сформирует последний тестовый набор, он выработает сигнал finish, который будет сигнализировать о завершении тестирования.

Блок  **actual\_result** формирует эталонный результат true\_rez, который зависит не только от значений операндов - множимого и множителя, но и от формата и кода, в котором они представлены. Для этого используется параметр mode. Варианты задания параметра mode приведены в файле с описанием этого модуля.

Блок **analise\_unit** сравнивает результат, формируемый испытуемым устройством real\_rez, с эталонным результатом true\_rez, вырабатываемым блоком **actual\_result**. Анализ выполняется по положительному фронту синхросигнала clk. Если хотя бы один раз результат не совпадет с эталонным, то в этом случае будет сформирован единичный сигнал defect, который не изменит своего значения до конца тестирования. Сигнал okey, формируется каждый раз после анализа результата выполнения испытуемым модулем **multiply** операции для очередного тестового набора. Его единичный уровень свидетельствует о совпадении результата с эталонным.

Все файлы, связанные с тестируемым модулем, находятся в папке MY\_OY. Там же содержится файл test\_for\_stand с временными диаграммами внешних сигналов, подаваемых на стенд.

1. **Принцип работы стенда TEST\_ALG**

Стенд TEST\_ALG работает следующим образом. По единичному внешнему сигналу reset стенд установится в начальное состояние. В этом случае генератор установится в соответствии с параметром start\_patern в состояние, соответствующее первому тестовому набору.

После снятия активного сигнала reset стенд начнет тестирование испытуемого модуля. Каждый раз по положительному фронту сигнала clk блок анализа результата будет сравнивать формируемый испытуемым модулем результат с эталонным, вырабатываемым модулем **actual\_result**. Генератор тестов будет формировать следующий тестовый набор до тех пор, пока он не выработает конечный набор, задаваемый константой stop\_patern. В этом случае генератор сформирует сигнал finish, свидетельствующий о завершении тестирования поведенческой модели операционного устройства.

Если хотя бы раз будет обнаружено несоответствие формируемого испытуемым модулем результата с эталонным, установится единичный сигнал defect, который не изменит своего значения до завершения текущего сеанса тестирования поведенческой модели операционного устройства. Сигнал okey формируется каждый раз по положительному фронту сигнала clk. Его единичное значение является дополнительным подтверждением корректности формируемого испытуемым модулем результата при сравнении его с эталонным результатом.

**3. Подключение испытуемой поведенческой модели операционного устройства к стенду**

Чтобы подключить испытуемую модель операционного устройства к стенду рекомендуется в модуле верхнего уровня проекта стенда удалить символ испытуемого модуля **multiply** и вместо него вставить символ модуля, синтезированного QUARTUS по поведенческому описанию студента. Эти действия выполняются аналогично тому, как выполнялось подключение структурных автоматов, разработанных студентами к специализированному стенду TSTAND.

 Для корректной работы блока формирования эталонного результата, перед выполнением компиляции проекта необходимо задать параметр mode, соответствующий Вашему варианту. Этот параметр зависит от кода и формата, в котором представлены операнды.

**4. Тестирование поведенческой модели операционного устройства с помощью специализированного стенда TEST\_ALG**

Чтобы выполнить тестирование поведенческой модели операционного устройства с помощью стенда TEST\_ALG, необходимо промоделировать поведение стенда с включенным в его состав испытуемой моделью. Для этого необходимо подать на стенд внешние сигналы reset и clk. Первый сигнал устанавливает стенд в исходное состояние. Второй сигнал используется для тактирования работы стенда. В папке MY\_AVT имеется файл с подготовленными временными диаграммами test\_for\_stand. После открытия этого файла, необходимо запустить функциональное моделирование.

**5. Анализ полученных результатов**

После завершения моделирования необходимо проанализировать полученные результаты, представленные на временной диаграмме. Во-первых, необходимо убедиться, что происходит изменение подаваемых на испытуемую модель операционного устройства тестовых наборов до тех пор, пока не будет сформирован единичный сигнал finish. Во-вторых, сигнал defect, свидетельствующий об обнаружении несоответствия полученного с испытуемой модели результата с эталонным, должен быть равен нулю на всем протяжении временной диаграммы. В противном случае его единичное значение будет указывать о неверном результате выполнения операции. Проанализировав момент времени формирования этого сигнала, студент сможет выявить причину и устранить её. В-третьих, сигнал okey, подтверждающий правильную работу модели ОУ, также на всем протяжении временной диаграммы должен быть равен единице.