

Вопрос 1.

С использованием типов данных и конструкций расширения System Verilog создайте описание устройства

Алгоритм работы:

- устройство принимает **адресуемые** данные **с двух** N-х разрядных входов А и В;
- осуществляет: выполнение функции  $R = 3 \cdot A - 9 \cdot B$  (без операции умножения)
- формирует выходные данные R

Выводы устройства (имена выводов модуля м.б. выбраны любыми, рекомендуется выбрать имена, облегчающие интеграцию с Platform Designer (PD)):

- На входах д.б использованы регистры.
  - Входы должны быть ориентированы на использование Avalon MM интерфейса в Platform Designer (PD)
- На выходе д.б использован регистр
  - Выход должен быть ориентирован на использование Conduit интерфейса в Platform Designer (PD).
- Вход: тактового сигнала – clk.
- Вход: сигнала синхронного сброса – srst.

*На этом месте в файле с ответами приведите созданное текстовое описание.*

Вопрос 2.

С использованием типов данных и конструкций расширения System Verilog для устройства, созданного в вопросе 1, разработайте тест класса 1.

Тест должен обеспечивать проверку всех режимов работы устройства (включая сброс).

*На этом месте в файле с ответами приведите созданное текстовое описание теста.*

Вопрос 3.

В пакете ModelSim, используя созданный в вопросе 2 тест, проведите моделирование созданного в вопросе 1 устройства.

*На этом месте в файле с ответами приведите временные диаграммы результатов моделирования.*

Вопрос 4.

Интегрируйте устройство, созданное в вопросе 1, как библиотечный компонент в PD (библиотечная папка для модуля – eham).

*На этом месте в файле с ответами приведите снимки экрана:*

- *с библиотекой PD в которой есть папка eham с созданным компонентом*
- *настройки интерфейсной части компонента.*

Вопрос 5.

- В PD создайте описание системы, включающей модуль тактового сигнала и компонент, созданный в вопросе 4.
- Экспортируйте выводы данных.
- Создайте HDL описание в приложении PD.
- С использованием типов данных и конструкций расширения System Verilog создайте описание верхнего уровня, в котором созданная система используется как компонент.
- Вход сброса, в файле верхнего уровня, должен быть подключен через два триггера.

- Осуществите компиляцию и получите структуру системы, используя RTL Viewer в пакете Quartus.

*На этом месте в файле с ответами приведите снимки экрана:*

- *структуры системы в PD*
- *созданного описания верхнего уровня*
- *структуры, полученной в RTL Viwer.*