

Trial cross-bar controller Device guide

Вячеслав Тарасов

8 декабря 2021 г.

Содержание

1	Рабочее окружение	3
1.1	doc	3
1.2	rtl	3
1.3	sim	3
1.4	syn	3
1.5	tb	3
2	Архитектура	4
2.1	Описание портов	4
2.2	Тактирование и сброс	4
2.3	Описание модулей	4
2.3.1	cross_bar_top	4
2.3.2	cross_bar_pkg	4
2.3.3	cross_bar_master	5
2.3.4	cross_bar_slave	5
2.3.5	cross_bar_rr_arbiter	5
3	Функционирование	6

1 Рабочее окружение

Описание всех основных директорий рабочего окружения.

1.1 doc

Директория с исходниками документации.

- **dev_guide.tex** - руководство разработчика в L^AT_EX версии;
- **preamble.tex** - преамбула для L^AT_EX окружения;
- **dev_guide.pdf** - руководство разработчика, сохраненное в PDF формате.

1.2 rtl

Директория с исходниками cross-bar контроллера.

- **cross_bar_pkg.sv** - пакет со всеми ключевыми параметрами дизайна;
- **cross_bar_top.sv** - модуль верхнего уровня контроллера;
- **cross_bar_master.sv** - модуль мультиплексоров для сигналов из slave в master;
- **cross_bar_slave.sv** - модуль мультиплексоров для сигналов из master в slave с арбитражной логикой;
- **cross_bar_arbiter.sv** - модуль арбитра.

1.3 sim

Содержит скрипты для запуска симуляции в Xcelium.

- **src.files** - файл со списком всех исходников для симуляции.

1.4 syn

Директория со скриптами запуска синтеза в Genus. Используется для оценки временных/пространственных параметров блоков.

- **lib** - директория с библиотеками.

Директория **cross_bar** построена по принципу:

- **syn_genus/run_genus** - скрипт для запуска Genus, в который передаются Tcl скрипты, описанные ниже;
- **syn_genus/elab.tcl** - скрипт с командами подключения библиотек, исходников и сборки;
- **syn_genus/syn.tcl** - скрипт с командами ограничений для тактового сигнала и синтеза;
- **syn_genus/clean** - скрипт для очистки всех артефактов синтеза;
- **syn_wrp.sv** - файл обертка для синтезируемого модуля;

1.5 tb

Директория с тестбенчами.

- **dev** - рабочий тестбенч; используется при разработке контроллера;
- **rr_arbiter** - тестбенч для тестирования арбитра;
- **tb_vip_slave.sv** - отдельный файл, в который вынесен модуль Verification IP инфраструктуры.

2 Архитектура

Cross-bar controller позволяет осуществлять коммутацию нескольких master и нескольких slave устройств. Параметры контроллера:

- Количество master устройств (parameter **MASTER_N**), не менее 2-х;
- Количество slave устройств (parameter **SLAVE_N**);
- Ширина шины адреса (parameter **ADDR_W**);
- Ширина шины данных (parameter **DATA_W**);
- Выбор slave устройства определяется старшими битами адреса;
- Арбитраж между несколькими master запросами в одно slave устройство осуществляется по дисциплине "round-robin".

2.1 Описание портов

Таблица 1: Тактирование и сброс

Сигнал	Направление	Примечание
clk	вход	тактовый сигнал
aresetn	вход	асинхронный сброс с активным нулем

Таблица 2: Интерфейс к master устройствам

Сигнал	Направление	Примечание
master_req	вход	запрос на выполнение транзакции
master_addr[ADDR_W - 1: 0]	вход	содержит адрес запроса
master_cmd	вход	признак операции: 0 - read, 1 - write
master_wdata[DATA_W - 1: 0]	вход	содержит записываемые данные
master_ack	выход	сигнал-подтверждение от slave устройства
master_rdata[DATA_W - 1: 0]	выход	содержит считываемые данные

Таблица 3: Интерфейс к slave устройствам

Сигнал	Направление	Примечание
slave_req	выход	запрос на выполнение транзакции
slave_addr[ADDR_W - 1: 0]	выход	содержит адрес запроса
slave_cmd	выход	признак операции: 0 - read, 1 - write
slave_wdata[DATA_W - 1: 0]	выход	содержит записываемые данные
slave_ack	вход	сигнал-подтверждение от slave устройства
slave_rdata[DATA_W - 1: 0]	вход	содержит считываемые данные

2.2 Тактирование и сброс

- Разделения на тактовые домены внутри нет;
- Для всех триггеров внутри контроллера используется асинхронный сброс с активным нулём.

2.3 Описание модулей

2.3.1 cross_bar_top

Модуль верхнего уровня контроллера. Здесь происходит подключение всех основных модулей. Все параметры задаются через пакет cross_bar_pkg.

2.3.2 cross_bar_pkg

Пакет со всеми глобальными для дизайна параметрами, типами и пр. Этот пакет импортируется во все модули контроллера.

2.3.3 cross_bar_master

Чистая комбинационная логика мультиплексоров сигналов ack и rdata от slave к master.

2.3.4 cross_bar_slave

Модуль содержит в себе комбинационную логику мультиплексоров сигналов req, addr, cmd и wdata от master к slave, фильтр запросов req и модуль арбитража.

2.3.5 cross_bar_rr_arbiter

Модуль арбитра, переписанный из открытых источников, требует дополнительной проверки. Также присутствует логика синхронизации сигналов req.

3 Функционирование

Полностью параметризованный модуль cross-bar, позволяющий производить автоматическую коммутацию настраиваемого количества master устройств и slave устройств.

Выбор slave устройства определяется старшими битами сигналов addr.

В случае одновременного обращения нескольких master устройств к одному slave устройству осуществляется арбитраж по дисциплине “round-robin”.