

РАЗРАБОТКА ФУНКЦИОНАЛЬНОЙ СХЕМЫ МЕЖИНТЕРФЕЙСНОГО АДАПТЕРА AXI-TO-SPI НА ПЛИС

Старшинов В.С.

Научный руководитель: Мальчуков А.Н., доцент каф. ИСТ ТПУ

Томский политехнический университет

E-mail: vss21@tpu.ru

Введение

Одной из самых острых и актуальных проблем в области систем связи на сегодняшний день является проблема совместимости различных видов устройств. Для решения задачи совместимости интерфейсов используются межинтерфейсные адаптеры. В данном случае стояла задача совмещения персонального компьютера и цифро-аналогового преобразователя через программируемые логические интегральные схемы (ПЛИС).

Данный преобразователь можно использовать в качестве радиопередатчика сигналов, для воспроизведения звука и видеопроигрывания [1].

Описания блоков адаптера

Блок AXI_SLAVE является устройством, обеспечивающим прием данных с ПК (рис. 1).

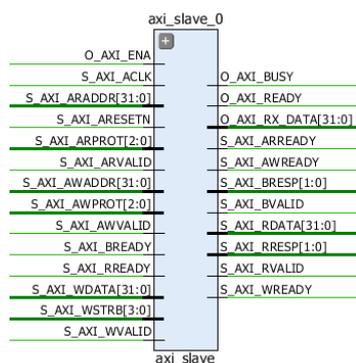


Рис. 1. Блок AXI_SLAVE

Входы:

S_AXI_ACLK – глобальный сигнал тактирования частотой 100MHz, S_AXI_ARESETN – глобальный сигнал сброса, активный уровень – низкий, S_AXI_ARADDR [31..0] – входной синхросигнал 250МГц, S_AXI_AWPROT [2..0] – адрес, по которому будут записываться данные, S_AXI_AWVALID – сигнал подтверждает, что на AWADDR выставлен актуальный адрес, S_AXI_WDATA [31..0] – данные для записи, S_AXI_WSTRB [2..0] – сигнал показывает, какие из байтов на WDATA будут записаны, S_AXI_WVALID – сигнал подтверждает, что на WDATA находятся актуальные данные, S_AXI_BREADY – сигнал отражает готовность устройства принять статус записи, S_AXI_ARADDR [31..0] – адрес, по которому будут прочитаны данные, S_AXI_ARPROT [2..0] – сигнал устанавливает тип привилегий и уровень безопасности для транзакций, S_AXI_ARVALID – сигнал подтверждает, что на ARADDR выставлен актуальный адрес, S_AXI_RREADY – сигнал отражает готовность устройства принять статус

записи, O_AXI_ENA [31..0] – сигнал, который дает разрешение на передачу данных в следующий блок.

Выходы:

S_AXI_AWREADY – сигнал отражает готовность устройства принять адрес записи и соответствующие сигналы управления, S_AXI_WREADY – сигнал отражает готовность устройства принять данные и соответствующие сигналы управления, S_AXI_BRESP [1..0] – сигнал отражает готовность устройства, S_AXI_BVALID – сигнал подтверждает, что на BRESP находятся актуальные данные, S_AXI_ARREADY – сигнал отражает готовность устройства принять адрес чтения и соответствующие сигналы управления, S_AXI_RDATA [31..0] – прочитанные данные, S_AXI_RRESP [1..0] – статус произведенной операции чтения, S_AXI_RVALID – сигнал актуальности данных на RRESP, O_AXI_BUSY – сигнал, который показывает, идет ли передача данных, O_AXI_RX_DATA [31..0] – данные, передаваемые на блок AXI_TO_SPI, O_AXI_READY – сигнал, который сообщает о завершении передачи данных.

Принцип работы блока AXI_SLAVE:

Блок AXI_SLAVE принимает команды и адрес и с помощью многочисленных настроек осуществляет взаимодействие с ПК. Данный блок является ведомым устройством, который принимает данные от ведущего устройства, которым выступает в данной ситуации ПК Частота входящих команд на данный блок равно 250 МГц [2].

Блок DIVIDER_FREQ является делителем входной частоты (уменьшает входную частоту до 25 МГц) (рис. 2).

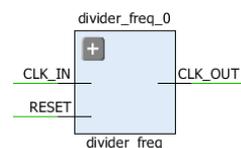


Рис. 2. Блок SPI_MASTER

Входы:

CLK_IN – входной синхросигнал 250МГц, RESET – глобальный сброс.

Выход:

CLK_OUT – выходной синхросигнал 25МГц.

Блок AXI_TO_SPI является устройством, обеспечивающим передачу сигнала с блока AXI_SLAVE в блок SPI_MASTER, то есть фактически является межинтерфейсным адаптером (рис. 3) [2, 3].

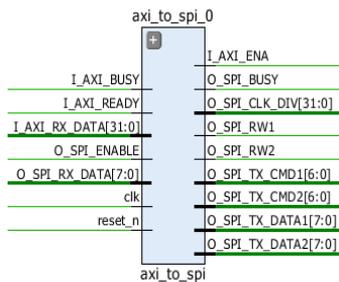


Рис. 3. Блок AXI_TO_SPI

Входы:

clk – входной синхросигнал 250МГц, reset_n – глобальный сигнал сброса, I_AXI_BUSY – сигнал, передачи данных, I_AXI_READY – сигнал, который сообщает о завершении передачи данных, I_AXI_RX_DATA [31..0] – данные, получаемые с блока AXI_SLAVE, O_SPI_ENABLE – настройка адреса для транзакции, O_SPI_RX_DATA [7..0] – данные, получаемые блоком SPI_MASTER.

Выходы:

I_AXI_ENA – сигнал, который дает разрешение на передачу данных в следующий блок, O_SPI_BUSY – показывает, передаются ли данные в данный момент, O_SPI_CLK_DIV [31..0] – установка скорости, O_SPI_RW1 – чтение/запись 1 команды, O_SPI_RW2 – чтение/запись 2 команды, O_SPI_TX_CMD1 [6..0] – 1 порция передаваемых команд, O_SPI_TX_CMD2 [6..0] – 2 порция передаваемых команд, O_SPI_TX_DATA1 [7..0] – 1 порция передаваемых данных, O_SPI_TX_DATA2 [7..0] – 2 порция передаваемых данных.

Принцип работы блока AXI_TO_SPI:

Блок AXI_TO_SPI выполняет роль преобразователя данных из параллельного вида в последовательный. Данный блок разделяет подаваемое с блока AXI_SLAVE 32-разрядное слово на 4 порции данных по 8 бит каждая, из которых 2 порции являются командами, выполняющие роль адресов необходимых для использования регистров, остальные 2 порции являются названиями этих регистров (рис. 4).

Bit Description:	MSB				LSB	
	RW	Commands	Data	RW	Commands	Data
Bits:	31	30-24	23-16	15	14-8	7-0

Рис. 4. Формат транзакций блока AXI_TO_SPI

Блок SPI_MASTER выводит данные побитово на ЦАП (рис. 5).

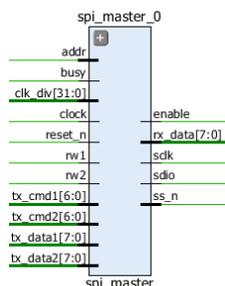


Рис. 5. Блок SPI_MASTER

Входы:

clock – входной синхросигнал 25МГц, reset_n – глобальный сигнал сброса, addr – установка адреса, busy – занятость сигнала, clk_div [31..0] – установка скорости, rw1 – 1 команда чтения/записи, rw2 – 2 команда чтения/записи, tx_cmd1 [6..0] – 1 порция передаваемых команд, tx_cmd2 [6..0] – 2 порция передаваемых команд, tx_data1 [7..0] – 1 порция передаваемых данных, tx_data2 [7..0] – 2 порция передаваемых данных.

Выходы:

enable – разрешение на передачу данных, rx_data [7..0] – считываемые данные, sclk – тактовая частота передаваемых сигналов 25 МГц, sdio – последовательная выдача инструкций и данных, ss_n – выбор ведомого устройства.

Принцип работы SPI_MASTER:

После передачи данных на блок SPI_MASTER они записываются в сдвиговый регистр, после чего выдача данных побитово. В качестве последовательности выдаваемых данных выступает бит инструкции и бит данных. Эта последовательность данных передается на блок передатчика ЦАП [3].

Заключение

В работе представлена реализация блоков, а также описан принцип работы блоков AXI_SLAVE и DIVIDER_FREQ. Совместная работа данных блоков позволяет принять данные и адреса регистров с ПК с частотой передачи 250МГц для передачи данных в остальные блоки.

Список использованных источников

1. Архитектура ПЛИС (FPGA) // Марсход – open source hardware project [Электронный ресурс]. – URL: <http://micro-proger.ru/2016/03/17/plis-fpga-i-mikrokontroller-v-chem-raznica/> (дата обращения 12.04.2017).
2. AXI4-Lite IP Interface (IPIF) // Xilinx All Programmable [Электронный ресурс]. – URL: https://www.xilinx.com/products/intellectual-property/axi_lite_ipif.html (дата обращения 12.04.2017).
3. Ключев А.О., Ковязина Д.Р., Петров Е.В., Платунов А.Е. Интерфейсы периферийных устройств. – СПб.: СПбГУ ИТМО, 2010. – 123 с.