

ИНТЕРФЕЙСЫ КОНТРОЛЛЕРА БАЗОВЫХ МОДУЛЕЙ ДЛЯ РЕКОНФИГУРИРУЕМЫХ МНОГОПРОЦЕССОРНЫХ СИСТЕМ

*НИИ многопроцессорных вычислительных систем имени академика
А.В. Каляева Южного федерального университета, г. Таганрог
maxim_work2003@mail.ru*

Контроллер базового модуля (КБМ) реконфигурируемых многопроцессорных систем предназначен для информационного обмена базового модуля (БМ) с управляющей ЭВМ, загрузки «прошивок» (файлов конфигурации) в ПЛИС, контроля температуры ПЛИС, индикации различных режимов работы БМ.

Основная функция КБМ - это реализация информационного обмена между вычислительными ПЛИС БМ и управляющей ЭВМ. Для этого разработаны два интерфейса: регистровый и потоковый.

Регистровый интерфейс предназначен для программной и числовой информации с вычислительными ПЛИС БМ, в которые загружены пользовательские «прошивки», реализующие специальные функционально законченные процедуры пользователя. Основные сигналы регистрового интерфейса:

- Command(31:0) – шина команд;
- Str_com – сигнал, подтверждающий валидность команды;
- Data(31:0) – данные, поступающие в ПЛИС и принимаемые из ПЛИС;
- RDWR – сигнал, подтверждающий валидность данных при записи, запрос данных при чтении;
- Ack – сигнал-подтверждение принятия данных ПЛИС при записи, сигнал, подтверждающий валидность данных при чтении;
- Irq - сигнал прерывания от ПЛИС.

Временные диаграммы обмена данными по локальной шине для регистрового интерфейса приведены на рисунках 1, 2 и 3.

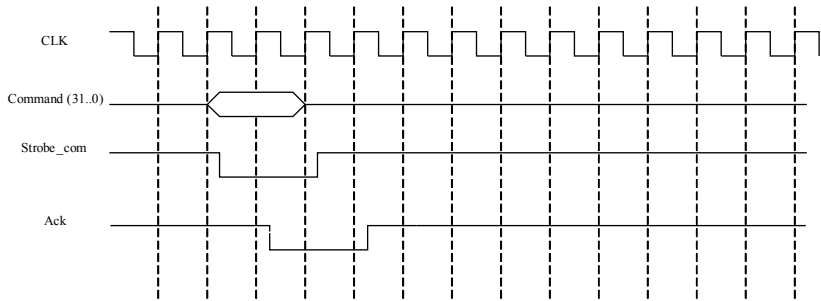


Рисунок 1 - Операция записи команд в ПЛИС

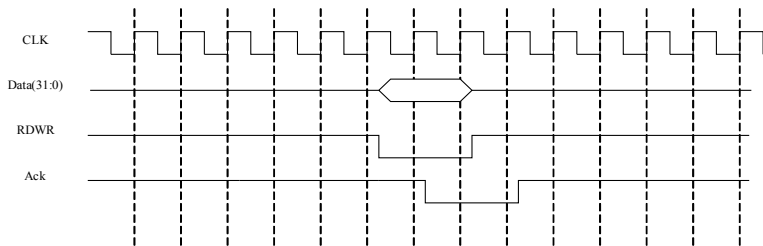


Рисунок 2 - Операция записи данных в ПЛИС

При операции записи команд в ПЛИС на шину команд Command(31:0) КБМ выставляет данные и сигналом Str_com подтверждает валидность этой команды. ПЛИС, получив сигнал Str_com, записывает команду во внутренний регистр и выставляет сигнал подтверждения записи - Ack. КБМ, получив сигнал Ack, переводит сигнал Str_com в неактивное состояние.

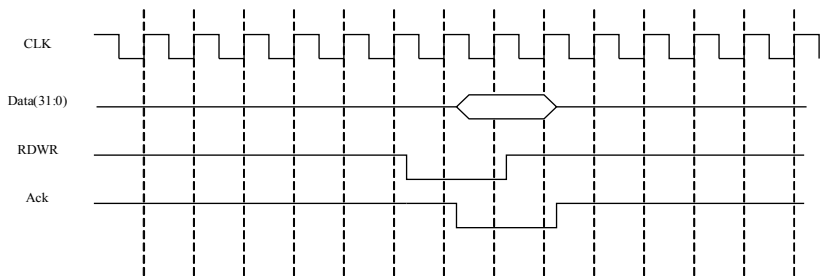


Рисунок 3 - Операция чтения данных из ПЛИС

При операции записи данных в ПЛИС на шину данных Data(31:0) КБМ выставляет данные и сигналом RDWR, подтверждает валидность этих данных. ПЛИС, получив сигнал RDWR, записывает данные во внутренний регистр и выставляет сигнал подтверждения записи - Ack. КБМ, получив сигнал Ack, переводит сигнал RDWR в неактивное состояние.

При операции чтения данных из ПЛИС КБМ перестает управлять шиной данных Data(31:0) и через такт выставляет сигнал запроса данных RDWR. ПЛИС, получив сигнал RDWR, выставляет данные на шину данных Data(31:0) и подтверждает их валидность сигналом Ack. КБМ, получив сигнал Ack, записывает данные во внутренний регистр и переводит сигнал RDWR в неактивное состояние.

Сигнал прерывания от ПЛИС Irq непрерывно считывается КБМ из ПЛИС и записывается в 28-й бит регистра команд КБМ.

Команды подразделяются на три основных группы: команды управления, команды чтения и команды записи. Формат команд для регистрового интерфейса приведен на рисунке 4.

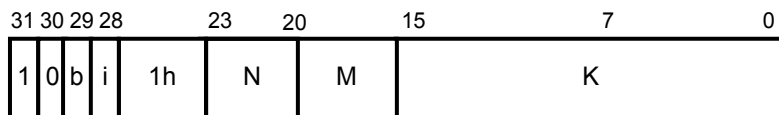


Рисунок 4 – Формат команд регистрового интерфейса

На рисунке 4:

- биты 31..30 определяют тип команды: 00 - для команд управления, 01 - для команд записи; 10 - для команд чтения,
- бит 29 указывает на режим широкого вещания команды (broadcast mode),
- бит 28 указывает на поступление сигнала прерывания Irq от ПЛИС,
- биты 27..23 определяют номер команды,
- поле N - номер ПЛИС,
- поле M - номер записываемого блока для команд записи, номер читаемого блока для команд чтения,
- поле K - количество принимаемых данных для команд записи или чтения, задержка (число тактов) выполнения команды для команд управления.

Для обеспечения взаимодействия с контроллером распределяемой памяти (КРП) создан потоковый интерфейс. КРП – это унифицированная «прошивка» ПЛИС, поставляемая НИИ МВС ЮФУ

для обеспечения доступа к большому объему распределяемой памяти БМ. Данный интерфейс позволяет осуществлять более быстрый обмен данными за счет использования пакетного режима по сравнению с регистровым интерфейсом.

Основные сигналы потокового интерфейса:

- Data(63:0) – данные, поступающие в ПЛИС и принимаемые из ПЛИС;
- RDWR – сигнал, подтверждающий валидность данных при записи, запрос данных при чтении;
- Strobe_V – сигнал, информирующий о том, что текущая операция относится к регистру команд КПП;
- Ack – сигнал-подтверждение принятия данных ПЛИС при записи, сигнал, подтверждающий валидность данных при чтении;
- Stop – останов КПП;
- Fail – отказ КПП;
- CMD – режим работы КПП;
- wr_en_fifo – сигнал, подтверждающий валидность данных при записи в FIFO КБМ;
- prog_full_fifo – сигнал, информирующий о том, что FIFO КБМ заполнено на 2/3.

Временные диаграммы обмена данными по локальной шине, для потокового интерфейса аналогичны приведенным на рисунках 1, 2 и 3.

При операции записи команд в КПП на шину данных Data(63..0) КБМ выставляет данные и сигналом RDWR подтверждает валидность этих данных. КПП, получив сигнал RDWR, записывает команду во внутренний регистр и выставляет сигнал подтверждения записи - Ack. КБМ, получив сигнал Ack, переводит сигнал RDWR в неактивное состояние.

Аналогичным образом происходит запись данных в КПП за исключением сигнала Strobe_V.

При операции чтения данных из КПП КБМ перестает управлять шиной данных Data(63..0) и через такт выставляет сигнал запроса данных RDWR. КПП, получив сигнал RDWR, выставляет данные на шину данных Data(63..0) и подтверждает их валидность сигналом Ack. КБМ, получив сигнал Ack, записывает данные во внутренний регистр и переводит сигнал RDWR в неактивное состояние.

Аналогичным образом происходит чтение данных из КПП за исключением сигнала Strobe_V.

базового модуля, позволяя решать задачи связанные как с использованием большого объема памяти, с применением унифицированных «прошивок» ПЛИС, содержащих КРП, так и специальных процедур с использованием «прошивок» пользователя реконфигурируемой многопроцессорной системы. Имеется возможность динамического переключения интерфейсов БМ по команде ЭВМ, в том числе в процессе решения задачи. Регистровый и потоковый интерфейсы полностью поддерживаются соответствующим программным обеспечением ЭВМ.

Использование пользователем данных интерфейсов значительно сокращает сроки разработки как прикладных программ для ЭВМ, так и вычислительных структур для многопроцессорных вычислительных систем.