

Масштабируемая система эмуляции сложных вычислительных систем

С.В. Юрлин^{1,2}, А.С. Воробьев^{1,2}

¹ АО «МЦСТ», ² ПАО «ИНЭУМ им И. С. Брука»

Аннотация — специализированное прототипирование является комплексной многоуровневой задачей с критически малым, как правило, временем на ее решение. Применение стандартных конструктивных решений для реализации прототипов многоядерных микропроцессоров не позволяет создавать эффективное оборудование. В статье рассматриваются рассчитанные на стандартизацию принципы и детали конструктивного исполнения основных функциональных компонентов, входящих в состав масштабируемых систем эмуляции (прототипов) нового поколения микропроцессоров семейства «Эльбрус». Заложенные в ней возможности должны обеспечить повторное использование ранее созданной аппаратуры и автоматизацию процессов подготовки прототипов к эксплуатации. Это может существенно увеличить время, непосредственно затрачиваемое на верификацию. По мнению автора, ряд решений этого проекта может быть взят в качестве стандарта при построении матричных масштабируемых систем эмуляции.

Ключевые слова — СнК, аппаратура, ПЛИС, прототип, «Эльбрус», эмуляция, КУБ-ПРО.

Scalable emulation system for complex computing systems

S.V. Yurlin^{1,2}, A.S. Vorobev^{1,2}

¹ JSC MCST, ² PJSC Brook INEUM

Abstract — Specialized prototyping is a complex multilevel task with critically small, as a rule, time for its solution. The use of standard constructive solutions for the implementation of prototypes of multi-core MPs does not allow creating efficient equipment. In this article, the principles and details of the design of the main functional components included in the scalable emulation systems (prototypes) of a new generation of microprocessors of the Elbrus family are considered. The opportunities inherent in it should ensure the reuse of previously created equipment and automate the processes of preparing prototypes for operation. This can substantially increase the time directly spent on verification. In the author's opinion, a number of solutions of this project can be taken as a standard in the construction of matrix scalable emulation systems.

Keywords — SoC, hardware, FPGA, prototype, Elbrus, emulation, KUB-PRO.

I. ВВЕДЕНИЕ

При упоминании прототипов обычно имеется в виду средство верификации RTL-модели микропроцессора (МП). Как следствие, во главу угла ставится эффективность верификации, но не аспекты сложности создания самих прототипов, как инструментов поиска ошибок. Тем не менее, в процессе их создания приходится решать большое количество технических задач. Особенно это актуально для специализированных прототипов, проектируемых как представление полноценной вычислительной системы. Это позволяет проверять функционирование нового микропроцессора в естественном окружении (оперативная память, карты расширения, контроллеры периферийных интерфейсов (КПИ)) и под управлением реального программного обеспечения (операционная система, программа начального старта, приложения).

Опыт АО «МЦСТ» и других производителей в проектировании, реализации и сопровождении специализированных прототипов [1, 2, 3, 4, 5], с одной стороны, убедительно подтвердил несомненное значение этой работы в общем процессе создания высокопроизводительных вычислительных средств на базе многоядерных микропроцессоров, с другой, позволил выявить свойственные ей проблемы. Реализация таких систем является комплексной многоуровневой задачей с критически малым, как правило, временем на её решение [6]. В этих обстоятельствах для своевременного выполнения поставленной задачи важно для каждого уровня сформулировать представление с базовыми решениями, которые в комплексе обеспечивают построение единой системы. В данной статье приведено описание представлений, связанных с реализацией аппаратуры специализированных прототипов.

II. ПРОТОТИП КАК ВЫЧИСЛИТЕЛЬНАЯ СИСТЕМА

При проектировании специализированных прототипов необходимо учитывать несколько уровней абстракции создаваемой вычислительной системы. Первый уровень - это микроархитектура эмулируемого микропроцессора с учётом его периферийных интерфейсов (рис. 1). Объём RTL-модели современных серверных МП значительно превышает логическую ёмкость ПЛИС. Возникает необходимость реализации распределённой системы эмуляции, связи между блоками которой определены микроархитектурой МП. Так для организации прототипа «Эльбрус-4С» понадобилось 10 самых ёмких на тот момент ПЛИС Intel (ранее Altera) серии Stratix IV EP4SE820, а для прототипа «Эльбрус-8С» уже 21. При этом их взаимосвязь задана определённым образом [7].

Второй уровень – многопроцессорная вычислительная система с набором периферийных устройств [8]. Её структура определяется целевым применением МП (рис. 2). Это позволяет не только автономно проверять непосредственную работу его RTL-модели, но и осуществлять системное тестирование одного МП с его окружением или комбинации таких систем. Следовательно, появляется возможность до изготовления кристалла выявлять сложнофункциональные ошибки, которые в силу объективных причин не могут быть выявлены моделированием или автономной эмуляцией. Например, межпроцессорное взаимодействие, интенсивные DMA запросы в память другого МП или особенности работы NUMA.

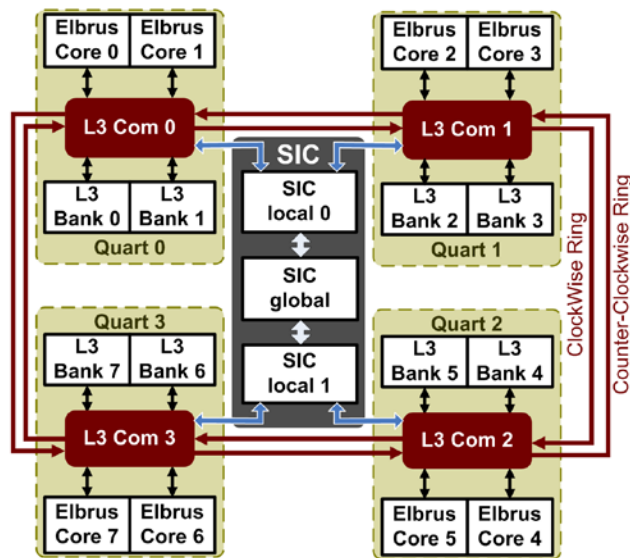


Рисунок 1 – Схема микроархитектурных связей в МП «Эльбрус-8С».

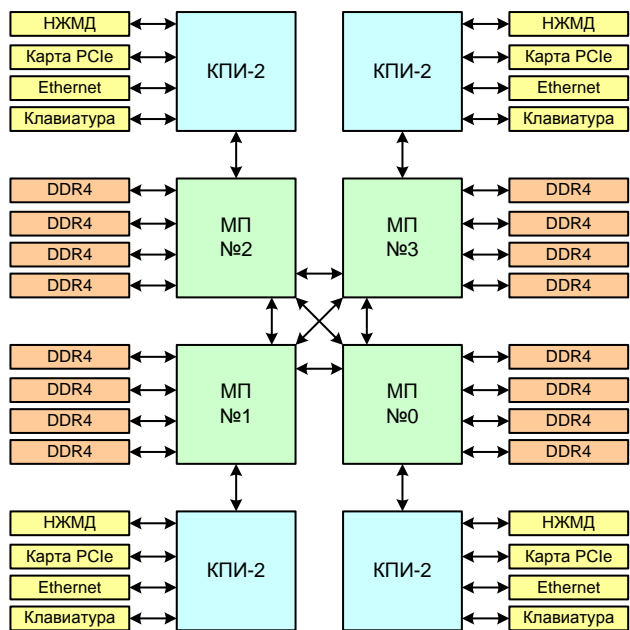


Рисунок 2 – Схема вычислительной системы на основе МП «Эльбрус-8С».

Третий уровень – проекты ПЛИС, которые определяют особенности реализации аппаратуры прототипа. Сюда стоит относить как FPGA, так и CPLD микросхемы. Данные проекты индивидуальны не только для конкретных моделей ПЛИС, но и для различных блоков RTL-описания загружаемых в одинаковые микросхемы. На низком уровне имеется привязка к типу корпуса, расположению и назначению контактов, используемым физическим уровням, а также к эмулируемой логике (рис. 3).

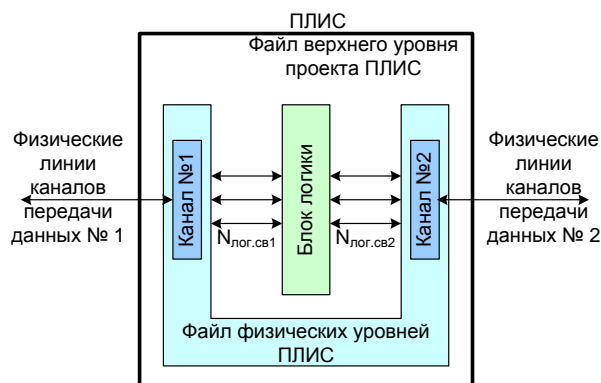


Рисунок 3 – Схема микроархитектурных связей в МП «Эльбрус-8С».

Четвёртый уровень – аппаратная реализация прототипа как механико-электрической системы. Данное представление является результирующей реализацией, создаваемой с учётом первых трёх уровней, физических ограничений, а также выбранных типов модулей, интерфейсов и конструкций.

Пятый уровень – система управления, настройки и сборки, которая определяется комплексом логических правил и программ, предназначенных для работы с аппаратной реализацией прототипа, контроля и управления его инженерными системами, разделения RTL-модели на блоки и их распределение по ПЛИС.

III. НОМЕНКЛАТУРА СИСТЕМЫ ЭМУЛЯЦИИ

При реализации четвертого уровня абстракции прототипа особенности первого и второго уровней обеспечиваются за счёт применяемых электронных модулей и средств их взаимодействия. Требуемый аппаратный состав масштабируемой системы эмуляции должен состоять из модулей эмуляции, модулей расширения, комплектов кабелей и жгутов, средств крепления и периферийного оборудования. Схема аппаратного состава масштабируемой системы эмуляции приведена на рисунке 4.



Рисунок 4. Схема аппаратного состава масштабируемой системы эмуляции.

IV. Модули эмуляции

Модуль эмуляции – вычислительный модуль, спроектированный на базе ПЛИС FPGA большой ёмкости, предназначенный для эмуляции блоков микропроцессора, который может содержать как интерфейсы взаимодействия с другими модулями эмуляции, так и ограниченный набор интерфейсов конечной вычислительной системы. Они нужны для обеспечения работы представления RTL-модели МП. Из [9] следует, что при построении прототипов больших систем имеет смысл говорить о матричных масштабируемых системах эмуляции, каждый модуль которой предлагается реализовывать в форм-факторе «КУБ-ПРО». Это позволяет уменьшить влияние

зависимости от третьего уровня абстракции за счёт минимизации физических реализаций одной и той же микросхемы.

Согласно базовому принципу этого подхода каждый модуль содержит только одну ПЛИС FPGA, являющуюся его центральным элементом. Для её связи с внешним миром используются описанные ниже интерфейсы прототипа типа А и Б. Соединители интерфейса типа А располагаются в центре каждой из сторон. Рядом с ними размещаются 6-контактные угловые соединители питания Mini-Fit для подачи +12 В. Такое расположение обеспечивает короткое прямое подключение кабелями соседних, граничащих с одной из сторон, модулей. Таким образом, реализовано двумерное масштабирование, которое определяет матричную конфигурацию модуля. Для увеличения возможностей масштабирования используется интерфейс прототипа типа Б, соединители которого равномерно расположены по периметру модуля.

Оптимальное количество каналов каждого интерфейса в такой компоновке равно четырём. Уменьшение приводит к снижению масштабируемости системы, а увеличение бесполезно ввиду геометрических ограничений, обусловленных самими модулями.

Принцип КУБ-ПРО, предполагающий размещение не более одной ПЛИС FPGA на модуле, упрощает отладку и уменьшает время ввода прототипа в эксплуатацию. Этому способствует и облегчение доступа к служебным сигналам ПЛИС и прототипа, достигаемое за счет расположения ПЛИС и теплоотводящего радиатора на нижней стороне панели. В этом случае доступны все переходные отверстия больших BGA корпусов, и увеличивается площадь на верхнем слое, которую можно использовать для дополнительных соединителей.

Макетная реализация модуля в форм-факторе «КУБ-ПРО», предназначенного для организации матричных масштабируемых систем эмуляции, в различных проекциях приведена на рисунке 5.

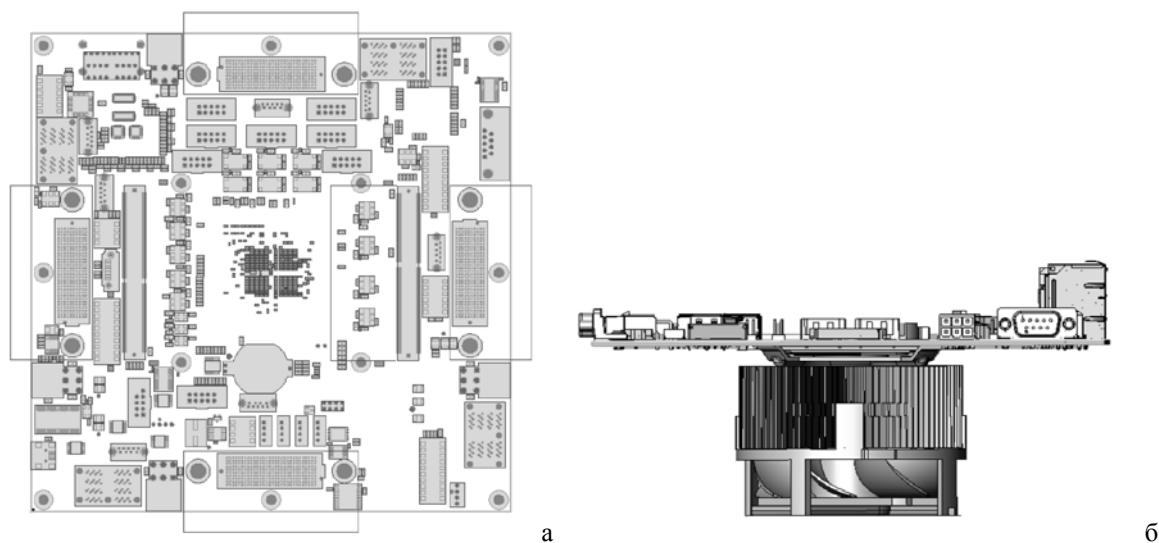


Рисунок 5 – Форм-фактор «КУБ-ПРО»:

а) пример положения элементов модуля,

б) 3D модель модуля (вид сбоку)

V. МОДУЛИ РАСШИРЕНИЯ

Модули расширения предназначены для реализации переменного окружения МП в различных конфигурациях вычислительной системы, а также для изменения способов тестирования RTL-модели. Такие модули могут быть созданы без использования больших и дорогих ПЛИС. Следует разделять модули расширения вычислительной системы (второго уровня абстракции) и системы эмуляции (первого уровня абстракции).

В первом случае речь идёт о стандартизованных аппаратных средствах, получивших широкое распространение. Например, видео или сетевые карты, предназначенные для подключения в слот PCIe. Их механическая и электрическая реализация определены.

Во втором – о специализированных аппаратных решениях, предназначенных для расширения интерфейсного поля модулей эмуляции, используя стандартизованные интерфейсы прототипа. В этом случае осуществляется их видоизменение. Оно может быть либо механическим – преобразование соединителей с сохранением физических уровней и протоколов работы, либо наоборот электрическим – изменение протоколов с

использованием соединителей интерфейсов прототипа. Последнее осуществимо ввиду возможностей переконфигурации ПЛИС FPGA. Примерами механического видоизменения является:

1) Панель PCIe BP-0INT-KUBPRO, выполненная в форм-факторе «КУБ-ПРО» и осуществляющая роль переходника с интерфейса прототипа типа Б модуля эмуляции на сокет PCIe x16 (рис. 6), что позволяет осуществлять подключение модулей расширения вычислительной системы к модулям эмуляции.

2) Пассивная карта расширения ПК в форм-факторе PCIe, предназначенная для подключения модулей эмуляции прототипа через механическую часть интерфейса прототипа типа Б для работы с прототипируемым изделием как с End Point (EP) PCIe устройством, либо для гибридного прототипирования.

Примером электрического видоизменения является соединение плата-плата, аналогично мезонинному, когда линии интерфейса прототипа используются для подключения DVI трансмиттера.

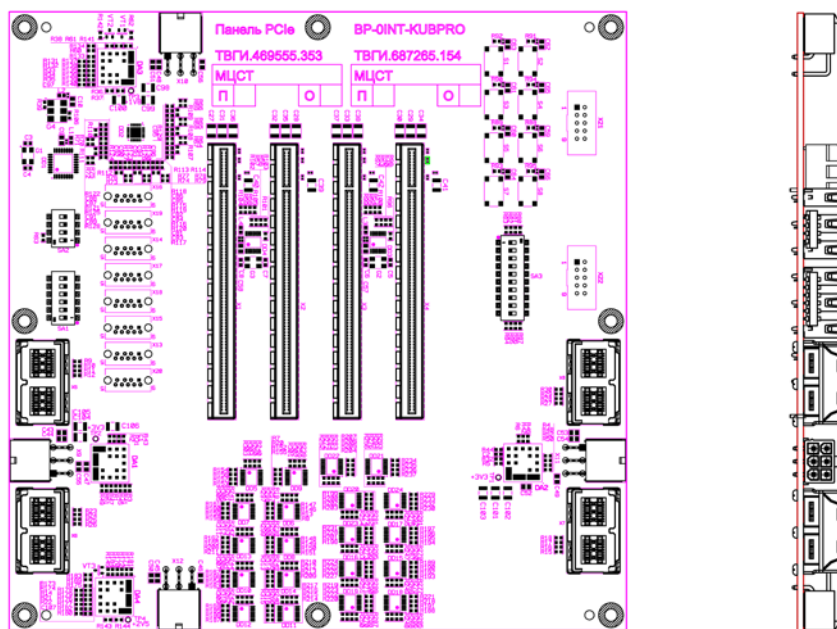


Рисунок 6 – Панель PCIe BP-0INT-KUBPRO.

Такой подход позволяет в широком диапазоне видоизменять и расширять конфигурацию интерфейсов прототипа. В том числе за счёт возможности их комбинации. Например, карта расширения внешнего КПИ-2 EC-OPER-PCIe (рис. 7) для проверки работы вычислительной системы с реальным контроллером периферийных интерфейсов и в режиме множественности таких контроллеров и карта эмуляции с ПЛИС FPGA для потокового USB тестирования с инженерного ПК. Они обе выполнены в форм-факторе PCIe, но используют интерфейс IOWLink, имеющий аналогичный физический уровень. Такие карты могут быть одинаково использованы как в составе прототипа, так и в составе стенда тестирования и разбраковки МП. А идентичность интерфейса их подключения позволяет взаимозаменять их в процессе тестирования.

В результате достигается высокая степень унификации создаваемых базовых решений, которые находят применение на всех стадиях жизни проектируемого МП.

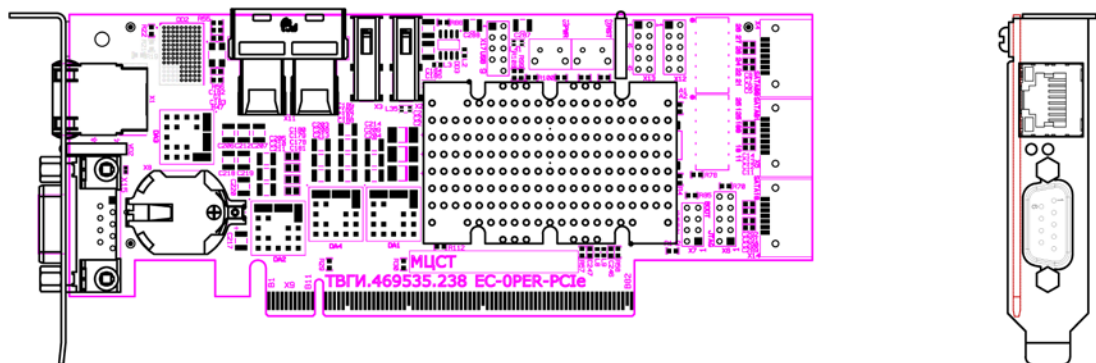


Рисунок 7 – Карта расширения внешнего КПИ-2.

VI. ИНТЕРФЕЙСЫ ПРОТОТИПА

Для обеспечения масштабируемости и повторного использования аппаратуры прототипа необходимо применение единых принципов построения, позволяющих создавать совместимые друг с другом устройства. Это актуально и для средств взаимодействия модулей. Однако для разных уровней абстракции прототипа специфика решаемых задач имеет различные особенности, которые в равной степени должны быть учтены при выборе интерфейса прототипа.

Распределённая система эмуляции подразумевает деление синхронной RTL-модели микропроцессора на блоки для их загрузки в различные ПЛИС, что приводит к необходимости передачи состояний логических связей между ними. В то же время, несмотря на проведённое разделение, необходимо сохранять потактовую целостность RTL-модели на уровне системы. Поэтому для первого уровня абстракции прототипа стоит выделить следующие особенности [9]:

- 1) Частота эмуляции зависит от количества логических связей между блоками СнК, эмулируемыми в ПЛИС, и характеристиками приёмо-передатчиков, используемых ПЛИС для передачи состояний логических связей.
- 2) Частота эмуляции в разных ПЛИС должна быть синфазной и не разбегаться с течением времени.
- 3) Все ПЛИС прототипа одной СнК должны объединяться в единую JTAG цепь, имеющую возможность переконфигурации для обхода неиспользуемых или неисправных ПЛИС или модулей.
- 4) Для фиксации событий, затрагивающих блоки RTL-модели, находящиеся в разных ПЛИС, необходимы сигналы прерываний работы логических анализаторов, передаваемые из одного ПЛИС во все остальные.
- 5) Сброс прототипа и вычислительной системы в целом должен осуществляться от любого модуля.
- 6) Высокоуровневые интерфейсы не лучшим образом подходят для прототипирования из-за их объёмной логической части и сложности алгоритмов установки соединения и маршрутизации. Приемлемых характеристик прототипа можно достичь только непосредственным применением схем физического уровня определенной ПЛИС и связанной с ними упрощенной логики.

Многопроцессорная вычислительная система подразумевает использование стандартных типов интерфейсов. Часть из них могут быть размещены на модуле эмуляции, так как они предназначены для подключения конечных устройств (например, RS-232, USB, DDR4, SATA). Другие же используются для реализации переменного окружения МП в различных конфигурациях вычислительной системы, а также для изменения способов тестирования RTL-модели (например, PCIe, SAS, IPLink, IOWLink). Причём их реализация и особенности применения определены в соответствующих стандартах, что не позволяет объединять их в единой конструкции.

ПЛИС, используемые при создании прототипа, определяют доступные физические уровни и контроллеры. На их основе реализуются периферийные интерфейсы вычислительной системы и интерфейсы распределённой системы эмуляции. Поэтому необходимо найти баланс между количеством доступных для конкретных физических уровней контактов, скоростью передачи данных и объёмом логической части используемых контроллеров.

Аппаратная реализация прототипа собирается на основе модулей эмуляции и модулей расширения с учётом схемы соединений распределённой системы эмуляции и оборудования, используемого в многопроцессорной вычислительной системе. При сборке подобных систем в индустриальном конструктиве возникает множество вопросов, связанных с отводом тепла, сложностью использования и отладки модулей, а также ограниченностью масштабирования посредством соединения плата-плата. Кроме того, увеличение масштабов системы эмуляции на основе индустриального конструктива приводит к запутанной схеме связей на соединительной панели. Если же осуществлять прямое соединение одинаковых модулей эмуляции друг с другом, без модулей посредников, возникает необходимость перекрещивания передатчиков и приёмников. Это приводит к появлению множества различных проектов ПЛИС, содержащих идентичные блоки RTL-описания, загружаемые в идентичные модули с ПЛИС, но отличающиеся физической конфигурацией микросхемы. Поэтому в качестве средства взаимодействия модулей эмуляции разумно использовать кросс-кабельные соединения.

Сложность аппаратной реализации прототипа существенно повышает трудоёмкость ручного контроля за работоспособностью системы и локализацию проблемных мест. Для данных процессов необходим модуль контроля и управления инженерными системами прототипа. В качестве средства взаимодействия для этих целей в серверных вычислительных системах применяется интерфейс I2C. На сегодняшний день этот интерфейс доступен практически в любом оборудовании электронных модулей.

Объединение требований различных уровней абстракции было окончательно реализовано в проекте КУБ-ПРО путем распределения коммуникационных и управляющих цепей интерфейса между ПЛИС разных типов, более соответствующих этим группам, и точной спецификации цепей каждой группы. Принципиальная схема соединения двух модулей прототипа через полнодуплексный интерфейс приведена на рисунке 7.

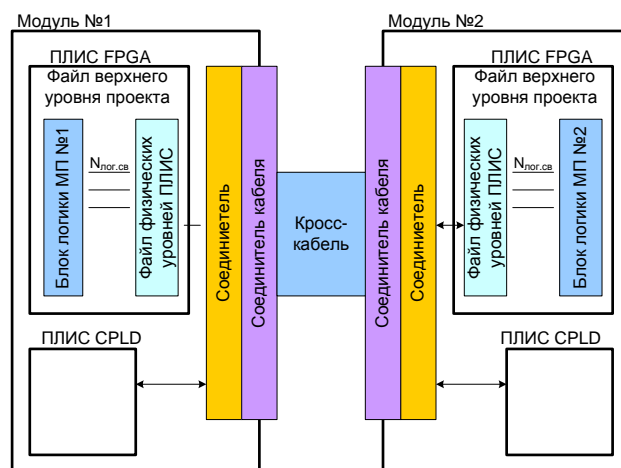


Рисунок 7 – Схема интерфейса прототипа.

Непосредственная передача сериализованных данных выполняется через шину, реализованную на физическом уровне ПЛИС FPGA, к которому отнесены и некоторые сигналы активации/деактивации аппаратуры прототипа. На базе ПЛИС CPLD реализованы сигналы из состава JTAG Altera для конфигурации эмулирующих ПЛИС, а также сигналы прерываний и управления статусом и рабочим режимом модулей.

Интерфейс прототипа реализован в двух типах: А и Б, отличающихся физическими уровнями каналов передачи данных и, соответственно, используемыми конструкциями разъёмов и кабелями. При использовании интерфейса первого типа все состояния логических связей между блоками RTL-описания должны быть переданы в течении одного такта частоты эмуляции. Связь двух ПЛИС выполняется путем LVDS-сигнализации шириной x16 с сопровождающим синхросигналом или без него с применением схем DPA. Для реализации выбран соединители серии SEARAY фирмы Samtec с матрицей 30x10 и кабель Samtec, разработанный с участием той же фирмы.

Интерфейс прототипа типа Б предназначен для работы с двухтактными передачами между блоками RTL-модели или в местах, где задержка распространения передачи не существенна, например при пакетных передачах в интерфейсе. Двухтактные передачи допускают пересылку состояний логических связей между блоками RTL-описания в течение двух тактов частоты эмуляции. В отличие от интерфейса первого типа интерфейс типа Б не поддерживает требований по максимизации частоты эмуляции из-за больших задержек в аналоговых схемах приёмо-передатчиков, но позволяет реализовывать через него высокоскоростные интерфейсы (IOWLink, IPLink, RDMA, PCIe, SATA, SAS). На физическом уровне интерфейса между двумя ПЛИС используются высокоскоростные приёмо-передатчики (BCSP, High speed transceivers) шириной x4. Для интерфейса типа Б выбрано два типа кабеля, один из которых предназначен для внешнего соединения длиной более 1м, например, при организации связи между прототипами в случае эмуляции многопроцессорных систем, другой – для внутренней передачи служебных сигналов через встроженный плоский кабель.

VII. КОНСТРУКТИВ ПРОТОТИПА

Для построения масштабируемой матричной системы эмуляции необходимо основание или каркас (корпус, chassis), выполняющее функции крепления модулей, отвода тепла от радиаторов модуля и подачу на них холодного притока воздуха, а также позволяющего реализовать масштабирование.

Из описания форм-фактора следует, что при креплении модулей доступ к верхнему слою МПП должен быть свободным для обеспечения коммутации соединителей и контакта с переходными отверстиями BGA корпусов. Соответственно, несущее основание должно располагаться снизу модуля, причём, (с учётом расположения монтажных отверстий) по его периметру. Из-за присутствия на нижней стороне модуля массивного теплоотводящего радиатора микросхемы FPGA основание должно быть полым, причём внутри него должен быть предусмотрен канал для воздушных потоков, отводящий тепло.

Форм-фактором предусмотрено масштабирование через интерфейсы прототипа. В частности, интерфейс типа А предназначен для связи соседних модулей, что обеспечивает двумерное масштабирование. Размещение системы модулей, объединённых таким образом, в одной плоскости требует существенного пространства. С целью более компактного размещения плоскость можно сворачивать в кольцо или тор. Для крепления такой конструкции требуется трёхмерная конструкция основания. С учётом этого основание можно представить в виде полого каркаса куба, состоящего из рёбер. Рёбра позволяют размещать до четырёх модулей форм-фактора КУБ-ПРО на сторонах, замыкающихся в кольцо, а незадействованные для крепления модулей стороны использовать для активно-пассивной системы охлаждения.

Доступ к соединителям модулей и необходимость пространства для забора воздуха требуют обеспечить технические области вокруг куба. Для этого на вершинах куба каркас должен иметь выступающие элементы, которые являются опорами и позволяют размещать его любой стороной на горизонтальных поверхностях, либо механически крепить к аналогичным каркасам. Их наличие позволяет масштабировать системы по трём осям путём механического соединения с другими аналогичными каркасами. Каркас также может быть использован для размещения инженерного и периферийного оборудования небольшого размера, в том числе блока питания или корзины для дисков.

3D модель каркаса приведена на рисунке 8.

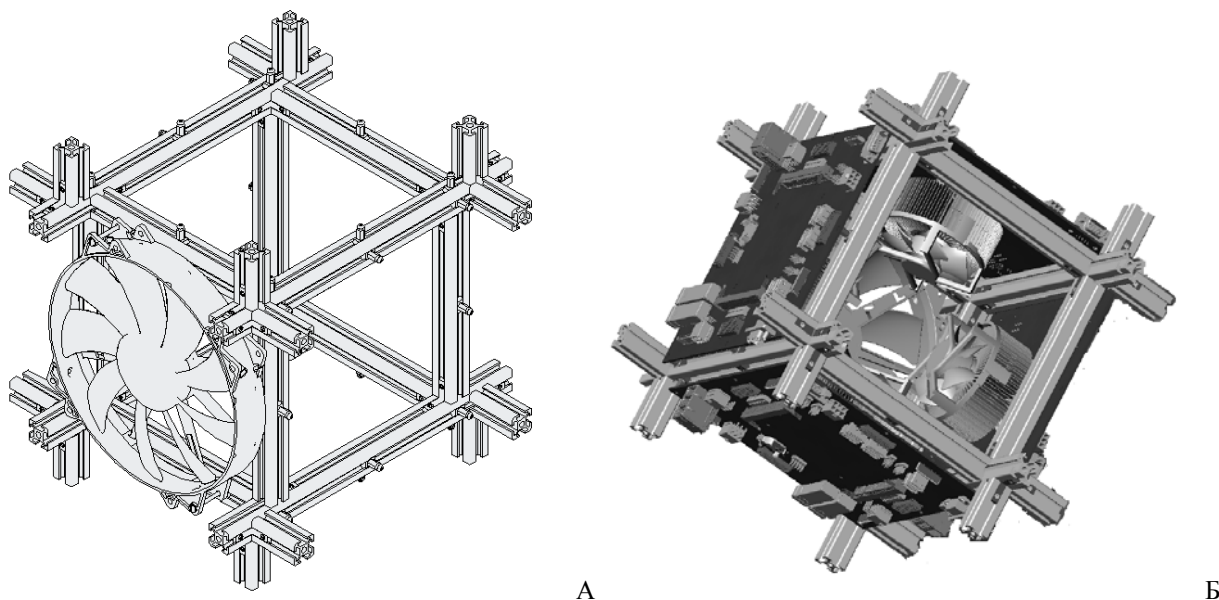


Рисунок 8 – 3D модель каркаса.

А – внешний вид каркаса.

Б –каркас с 4 модулями в форм-факторе «КУБ-ПРО».

VIII. ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ПРОТОТИПА

Сопровождение распределённых электронных систем с большим количеством уникальных блоков является трудоёмкой задачей. Она сопоставима с поддержкой целого вычислительного кластера. Однако, в классических модульных системах чувствительность к выходу из строя одного или нескольких узлов сведена к минимуму за счёт резервирования и распараллеливания. Для прототипов, как распределённой системы эмуляции, такой подход неприменим. Более того, программы, загружаемые в каждый конкретный узел, являются уникальными. Поэтому на всех этапах работы, необходимо не только обеспечивать надёжность работы системы на уровне аппаратуры, но и следить за правильностью разделения и сборки RTL-описания МП, чтобы функционирование результирующей системы прототипа функционально совпадало с исходной моделью.

Контроль за аппаратным уровнем прототипа включает в себя удалённый сбор статистики по уровням напряжения и температуры, управление напряжением и вентиляторами, выставление сбросов прототипа и СнК, а также оповещение об аварийных ситуациях с фиксацией событий и указания места поломки. Для этого создаётся система удаленной диагностики и управления (СУДУ), которая в качестве программного продукта разворачивается на внешнем устройстве (менеджере) функционирование которого не связано с работой аппаратуры прототипа. Их взаимодействие осуществляется через шину I2C путём кабельного соединения.

Аппаратура прототипа становится распределённой системой эмуляции только после деления RTL-модели МП на блоки и их загрузки в ПЛИС. Эта задача имеет высокую трудоёмкость, сопряжённую с высоким риском ошибки. Она учитывает особенности блоков RTL-модели МП, соединённых фиксированным образом, и модулей эмуляции, характеризующиеся электрическими связями и проектом ПЛИС (рис. 9). По сути, происходит создание описания на языке Verilog модели всего специализированного прототипа. Минимизации рисков в данной работе возможна путём автоматизации процессов деления RTL-модели МП, ассоциации портов полученных блоков с портами в проектах ПЛИС, а затем соединение полученных файлов в единую систему в соответствии с

микроархитектурой МП для последующего моделирования. Для этого применяется система скриптовых программ. Их алгоритм работы корректируется изменением конфигурации файлов настроек.

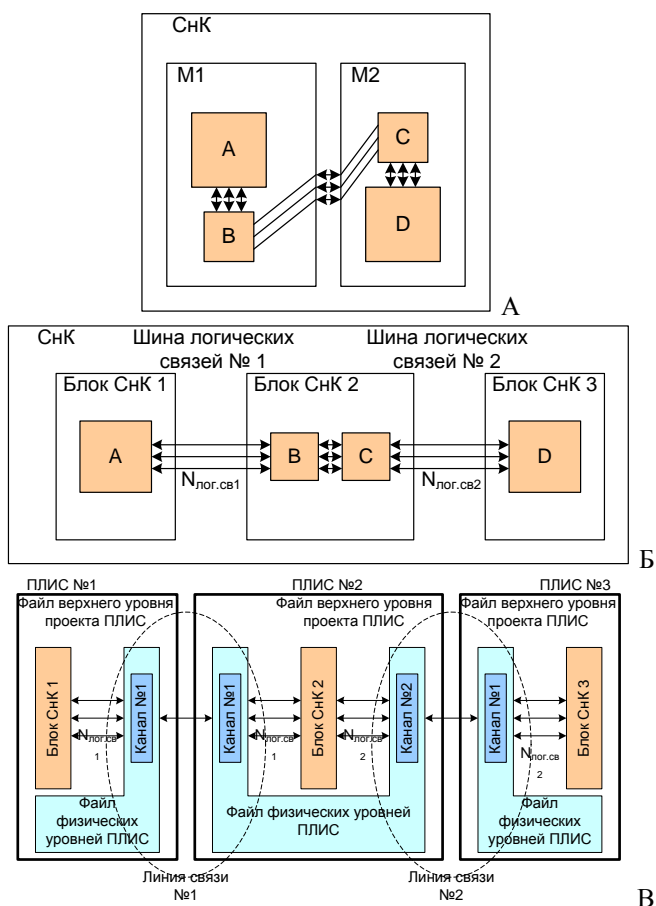


Рисунок 9 – Схема распределения единой RTL-модели МП в системе эмуляции из ПЛИС.

А – Схема RTL-модели.

Б – Представление блоков RTL-модели МП после деления.

В – Представление блоков RTL-модели МП в ПЛИС распределённой системы эмуляции.

IX. ЗАКЛЮЧЕНИЕ

В статье рассмотрены базовые решения, используемые для реализации прототипа на разных уровнях его абстракции. Заложенные в них возможности должны обеспечить повторное использование ранее созданной аппаратуры и автоматизацию процессов подготовки прототипов к эксплуатации. Это может существенно увеличить время, непосредственно затрачиваемое на верификацию. По мнению автора, ряд решений этого проекта может быть взят в качестве стандарта при построении матричных масштабируемых систем эмуляции.

ЛИТЕРАТУРА

- [1] Слесарев М.В., Юрлин С.В. Определение расчётной частоты эмуляции микропроцессора в прототипе на основе ПЛИС // Вопросы радиоэлектроники, серия ЭВТ, выпуск 3, 2014, с.119-130.
- [2] Будьлин Ф.К., Полищук И.А., Слесарев М.В., Юрлин С.В. Опыт прототипирования микропроцессоров компании ЗАО «МЦСТ» // Вопросы радиоэлектроники, серия ЭВТ, выпуск 3, 2012, С. 132-142.
- [3] Schelle G., Collins J., Schuchman E. et al. Intel Nehalem processor core made FPGA synthesizable. Proceedings of the 18th ACM/SIGDA international symposium on Field programmable gate arrays, 2010, pp. 3–12.
- [4] Synopsys Inc. – Synopsys’ HAPS® 80 brochure. Synopsys’ HAPS® Physical Prototyping Solution, 2016.
- [5] PRO DESIGN Electric GmbH – proFPGA overview brochure. FPGA Based Prototyping Solution. Modular, Flexible and Scalable - offering Highest System Performance, 2017.
- [6] Бычков, И. Н., Юрлин, С. В. Прототипирование на основе ПЛИС для верификации многоядерных микропроцессоров. // Проблемы разработки перспективных микро- и нанoeлектронных систем—2014: Сборник трудов / под общ. ред. академика РАН А.Л. Стемпковского. — М.: ИПИМ РАН, 2014. — Часть IV. — С. 45—50.

- [7] Альфонсо Д.М., Деменко Р.В., Кожин А.С., Кожин Е.С., Колычев Р.Е., Костенко В.О., Поляков Н.Ю., Смирнова Е.В., Смирнов Д.А., Смольянов П.А., Тихорский В.В. Микроархитектура восьмиядерного универсального микропроцессора Эльбрус-8С // Вопросы радиоэлектроники / № 3. Сер. ЭВТ. М., 2016. С. 6-14.
- [8] Бычков И.Н., Воробьев А.С., Рябцев Ю.С. Стенд тестирования и разбраковки многоядерных процессоров // Приборы / №2(176) 2015, с.16-22.
- [9] С.В. Юрлин. Универсальный подход к построению масштабируемых прототипов многоядерных микропроцессоров (КУБ-ПРО). // Вопросы радиоэлектроники. –2018. – №3. – Сер. ЭВТ. – Выпуск 2. – С.93-98.

REFERENCES

- [1] Slesarev M.V., Yurlin S.V. Determination of the estimated emulation frequency of the microprocessor in FPGA-based prototype // Voprosy radioelektroniki / № 3, ser. EVT, M., 2014, pp. 119-130. (in Russian).
- [2] F.K. Budylin, I.A. Polishyk, M.V. Slesarev, S.V. Yurlin. The experience of prototyping MCST CJSC' microprocessors // Voprosy radioelektroniki / № 3, ser. EVT, M., 2012, pp. 132-142. (in Russian).
- [3] Schelle G., Collins J., Schuchman E. et al. Intel Nehalem processor core made FPGA synthesizable. Proceedings of the 18th ACM/SIGDA international symposium on Field programmable gate arrays, 2010, pp. 3–12.
- [4] Synopsys Inc. – Synopsys' HAPS® 80 brochure. Synopsys' HAPS® Physical Prototyping Solution, 2016.
- [5] PRO DESIGN Electric GmbH – proFPGA overview brochure. FPGA Based Prototyping Solution. Modular, Flexible and Scalable - offering Highest System Performance, 2017.
- [6] Yurlin S.V., Bychkov I.N. FPGA prototyping for functional verification of multi-core processors // Problems of Perspective Micro- and Nanoelectronic Systems Development - 2014. Proceedings / edited by A. Stempkovsky, Moscow, IPPM RAS, 2014. Part IV. P. 45-50.
- [7] Alfonso D., Demenko R., Kozhin A., Kozhin E., Kolychev R., Kostenko V., Polyakov N., Smirnova E., Smirnov D., Smolyanov P., Tikhorskiy V. Eight-core Elbrus-8C Processor Microarchitecture // Voprosy radioelektroniki / No. 3, ser. EVT, Moscow, 2016, pp. 6-14. (in Russian).
- [8] Bychkov I.N., Ryabtsev Y.S., Vorobyev .A.S. Test bench for examination and rating of multi-core processors // Pribory / No. 2(176) 2015, p.16-22
- [9] S. V. Yurlin. A universal approach to building scalable prototypes of multi-core microprocessors (KUB-PRO). // Voprosy radioelektroniki / № 3, ser. EVT, M., 2018, pp. 93-98. (in Russian).