

УДК 004.31

## МОДУЛЬ АССОЦИАТИВНОГО СОПРОЦЕССОРА НА БАЗЕ ПЛИС ДЛЯ СПЕЦИАЛИЗИРОВАННЫХ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ

А. И. Мартышкин, доцент кафедры ВМиС ПензГТУ, к.т.н.; alexey314@yandex.ru

Рассматривается возможность реализации модуля ассоциативного сопроцессора на современной элементной базе для специализированных вычислительных систем. **Целью статьи** является разработка и исследование модуля ассоциативного сопроцессора на базе ПЛИС для специализированных вычислительных, например, многопроцессорных систем, выполняющих ассоциативные функции и функции хранения данных. **Объектом разработки и исследования** данной статьи является ассоциативный сопроцессор на базе ПЛИС. Операция поиска широко используется в программах разного назначения как пользовательских, так и системных. Однако эта операция является одной из наиболее трудоемких и требует больших затрат времени при реализации традиционным способом, когда искомые данные последовательно читаются из оперативной памяти машины в процессор, где над ними выполняются соответствующую операцию. В статье предлагается ассоциативный сопроцессор, подключаемый к шине PCI вычислительной системы, обеспечивающий операции поиска и сравнения на «большие – меньше» одновременно по 32-м словам, загруженным предварительно в ассоциативную память. В заключении сделаны соответствующие выводы по работе. Применение языка VHDL как универсального средства описания аппаратуры интегральных схем обеспечивает гибкость проекта и легкость при отладке функционирования устройства. Моделирование работы ассоциативного сопроцессора проводилось в САПР Web pack ISE фирмы Xilinx. Это позволяет производить оценку правильности работы аппаратного сопроцессора в составе вычислительных систем без построения реальных макетов. Эффективность внедрения разработанного модуля обеспечивается за счет того, что сопроцессор выполняет трудоемкие операции поиска и сравнения данных, тем самым разгружает центральный процессор и увеличивает производительность вычислительной системы в целом. Результаты, полученные в статье, могут найти применение в поисковых системах различного назначения: серверах баз данных, в поисковых машинах, применяемых на вокзалах, аэропортах, а также для быстрой реализации задач поиска операционных систем.

**Ключевые слова:** модуль, ассоциативная память, адресация, сопроцессор, вычислительная система, ячейка памяти, фиксация реакций, анализатор многократного совпадения, адресация памяти, шинный интерфейс, аппаратная реализация, приоритетный анализатор, цикл записи, цикл чтения.

DOI: 10.21667/1995-4565-2016-58-4-75-82

### Введение

Основная область применения компьютеров сегодня – работа с большими объемами данных, при которой самыми трудоемкими операциями являются всевозможные поиски и сортировки данных. Существующие вычислительные системы (ВС) используют адресную архитектуру памяти, т.е. для того чтобы осуществить поиск данных в памяти, необходимо произвести чтение по каждому адресу модуля памяти и сравнить его с аргументом поиска. В итоге на поиск информации в памяти уходит много машинного времени. Это обстоятельство весьма негативно сказывается на быстродействии ВС в целом. Намного быстрее получать доступ к данным по ассоциации (содержанию). Сущность принципа адресации по содержанию заключается в следующем (рисунок 1). Имеется массив данных емкостью N слов, в нем требуется найти все слова, начинающиеся с символа «А» и заканчивающиеся символом «Н».

Здесь аргументом поиска является слово А\*\*\*Н, где значком \* отмечены разряды, не оказывающие влияние на результат поиска. Запоминающий массив на аппаратном уровне строится так, что на выходе ячеек памяти (ЯП), содержимое которых совпадает со значением поступившего аргумента поиска, появляется сигнал – указатель совпадения. Далее по выработанным сигналам совпадения выполняется выборка ЯП.

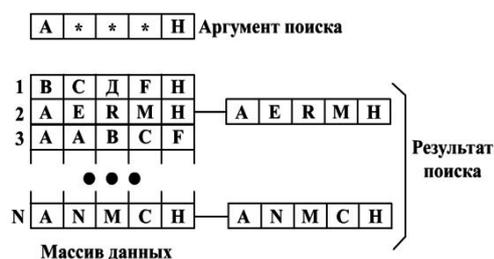


Рисунок 1 – Сущность принципа адресации по содержанию

### Теоретическая часть

Настоящая статья в целом носит исследовательский характер. В ходе изучения предметной области была проанализирована литература [1] с целью поиска незатронутых и нерешенных проблем. Ряд вопросов, связанных с возможностью аппаратной реализации ассоциативного сопроцессора для быстрого поиска данных, не нашел должного отражения в публикациях, однако частично проблемные вопросы были рассмотрены в работах [2 – 10].

Целью настоящей статьи является разработка и исследование модуля ассоциативного сопроцессора на базе ПЛИС для специализированных вычислительных, например, многопроцессорных систем. Данный вопрос сегодня является актуальным ввиду глобальной информатизации и практически повсеместном оперировании колоссальными объемами различных данных.

Для достижения поставленной цели в статье решаются задачи по определению структуры устройства и принципов его функционирования.

Разрабатываемый аппаратный сопроцессор обладает возможностью адресного и ассоциативного доступа к хранящимся в памяти данным. Адресный доступ необходим для работы с конкретной записью и для использования библиотек тестов, разработанных для адресной памяти.

Устройство состоит из двух частей: основной, реализующей функции ассоциативного сопроцессора, а именно: обычная (адресная) запись в ассоциативное запоминающее устройство (АЗУ); ассоциативная запись в АЗУ; обычное чтение из АЗУ; ассоциативное чтение из АЗУ; поиск совпадений и части сопряжения с ВС, на которую возлагается функция преобразования сигналов, идущих от центрального процессора (ЦП), в сигналы, с которыми будет работать сопроцессор, т.е. эта часть устройства организует интерфейс с ЦП.

Сегодня ассоциативный доступ к данным реализуется двумя способами: программным, основанным на распределении памяти в зависимости от содержания данных и реализуемым с помощью программных средств, и аппаратным, основанным на применении специальных аппаратных средств, предназначенных для хранения и ассоциативного поиска элементов данных. Он может быть реализован в виде параллельного АЗУ, где аргумент поиска поступает во все ЯП параллельно. В результате выполняется массовая операция сравнения, как итог – поиск выполняется за один такт. Другим вариантом реализации аппаратного способа ассоциативного доступа к данным является последовательно–поразрядное АЗУ, где поиск происходит поразрядно. В этом случае время поиска напрямую зависит от количества разрядов шины данных.

Процессы, аналогичные биологическим механизмам запоминания и обработки информации, можно представить с помощью различных моделей ассоциативной памяти (АП), позволяющих отобразить отношения (ассоциации) произвольной сложности между информационными объектами. Однако все эти отношения могут быть реализованы в виде простых конструкций – троек компонентов: упорядоченной пары информационных объектов  $O$  и  $V$  и типа отношения  $A$ :  $O \xleftarrow{A} V$  [11]. Одна из простейших моделей АП для отображения таких отношений показана на рисунке 2, а. Модель состоит из ассоциативной запоминающей среды, связанной с двумя каналами ввода и одним каналом вывода информации.

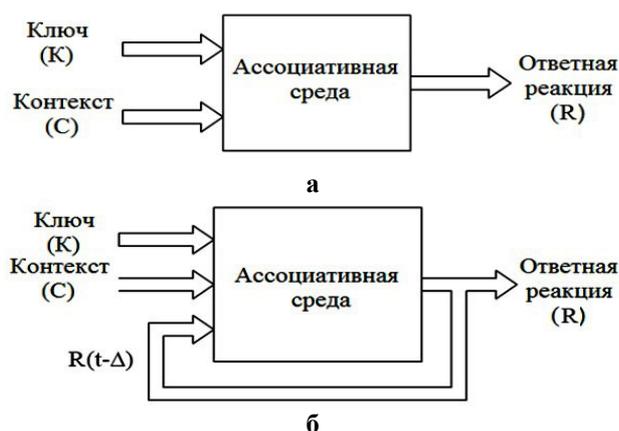
На этапе записи с первого канала ввода на вход  $K$  подается входная информация, а по второму каналу – признаковая информация  $C$ , представляющая контекст, в котором входная информация записывается в память.

На этапе ассоциативной выборки при появлении ключа  $K$  на выходе памяти формируется ответная реакция  $R$ , связанная с ключом  $K$ . Таким образом, записанная в память информация может быть выбрана с применением любых ее фрагментов, используемых в качестве поисковых. Задавая различный контекст  $C$ , можно точнее конкретизировать информацию, подлежащую выборке.

В контексте введенного определения АП стоит вопрос об организации накопления и поиска структурированных данных таким образом, чтобы доступ к ним был возможен на основе ассоциативной выборки. На рисунке 2, б представлена модель АП, позволяющая ответить на вопросы: как можно организовать запись и выборку элементов структурированной информации, а также осуществить цикличность процесса поиска, при котором выбранный элемент информации становится ключом для поиска новой информации [11].

По трем входным каналам одновременно могут вводиться наборы значений. Выходной канал служит для выборки информации. По первому каналу в момент времени  $t$  подается адресная информация  $K(t)$ , а по второму – признак  $C(t)$ . Отклик  $R(t)$  по каналу обратной связи подается также на вход ассоциативной среды. При функционировании такой АП ключи  $K(t)$  и признаки  $C(t)$  подаются через интервалы времени, соответствующие задержке канала обратной связи.

Процесс работы памяти будем рассматривать в предположении, что тройка  $[K(t), C(t), R(t-\Delta)]$  представляет собой единый статический образ, заданный в момент времени  $t$ , причем возможна его одновременная запись в память за одну операцию. Допустим также, что на этапе записи  $R(t)$  и  $K(t)$  одинаковы.



**Рисунок 2 – Модель ассоциативной памяти без обратной связи (а) и с обратной связью (б)**

На этапе записи на входы АП поступают  $K(t)$  и  $C(t)$ , при этом на выходе формируется  $R(t)$ , идентичный  $K(t)$ . После этого с задержкой  $\Delta$  на входе формируется  $R(t-\Delta)$ . Каждая новая тройка, появляющаяся на входах, записывается в память.

На этапе выборки из АП на вход подается ключ  $K$ , связанный с контекстной информацией  $C$ , после чего  $K$  можно снять с входа. В результате на выходе в качестве отклика появляется копия  $K$ . Когда на входе памяти появится задержанный сигнал  $R(t-\Delta)$ , новым ключевым признаком становится пара  $(C, R)$ , приводящая к ассоциативной выборке следующего образа  $R(t)$  и т. д. Таким образом, выбирается вся записанная последовательность образов вместе с контекстной информацией. Рассмотренная модель реализует память, пригодную для записи и выборки структурированных знаний.

Подключение описываемого устройства к ВС возможно несколькими способами [12]: прямое подключение к шине процессора, при этом сопроцессор включают в состав системной платы; подключение к последовательному интерфейсу (USB), при таком способе подключения сопроцессор выполняется в виде отдельного корпуса и снабжается отдельным блоком питания; подключение к шине расширения компьютера (PCI), в этом случае сопроцессор выполняется в виде платы расширения.

На основе упомянутых возможных способов подключения разрабатываемого устройства к ВС определим архитектурно-структурные особенности ассоциативного сопроцессора, предварительно сравнив их с существующими аналогами.

Как было указано выше, при прямом подключении к шине процессора устройство придется включать в состав системной платы, что приведет к увеличению стоимости сопроцессора и его универсальности. Быстродействие такой системы оказывается высоким.

При подключении с помощью интерфейса USB сопроцессор будет иметь вид отдельного внешнего модуля, но при этом работа с ним будет осуществляться последовательно, что приводит к снижению быстродействия. Зато такой блок сравнимо дешевле предыдущего.

Подключение сопроцессора к шине PCI позволит реализовать его в виде платы расширения, сравнительно дешевой в реализации в отличие от упомянутых аналогов. При этом работа с модулем будет осуществляться по параллельному интерфейсу, что позволит достичь максимальной производительности по сравнению с аналогами.

Проанализировав достоинства и недостатки перечисленных известных и применяемых на практике способов организации основной части устройства и части сопряжения с ВС, было решено основную часть выполнить в виде параллельного АЗУ, так как данный способ обладает максимальной производительностью. Подключение к специализированной ВС реализуется по шине PCI, т.к. она обладает достаточно высокой пропускной способностью. Более того, ассоциативный сопроцессор будет размещаться в адресном пространстве ввода/вывода ВС.

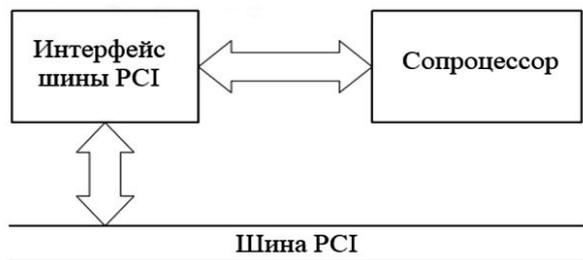
### **Определение структуры модуля ассоциативного сопроцессора**

Упрощенную структурную схему сопроцессора можно представить в виде двух блоков (рисунок 3, а): сопроцессор (основная часть) и интерфейс шины PCI (часть сопряжения с системой).

Проанализировав выбранный способ реализации основной части в составе сопроцессора, решено, что главным блоком является модуль памяти АЗУ, представляющий собой массив ЯП и выполняющий функции хранения данных и поиска ассоциаций с аргументом. ЯП состоит из запоминающего элемента, выполняющего функции хранения данных, и схемы сравнения, осуществляющей поиск, т.е. вырабатывающей сигналы, указывающие на равенство или неравенство содержимого ячейки аргументу (рисунок 3, б).

Для реализации функций записи и чтения данных из ЯП в состав сопроцессора необходимо добавить блок мультиплексора и дешифратора адреса для выработки сигнала «Выбор ячейки» конкретной ЯП. Также нужно включить в состав модуля сопроцессора блок памяти фиксации реакций для сохранения значений откликнувшихся ячеек. Для подсчета количества откликнувшихся ячеек в состав сопроцессора введен блок подсчета совпадений. Блок анализатора многократных совпадений (АМС) необходим для приоритетного выбора, по результатам рабо-

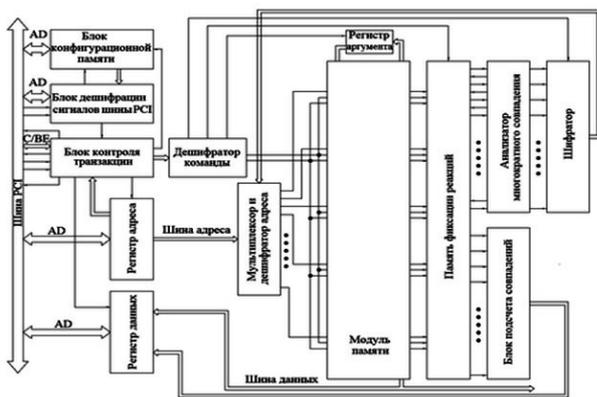
ты которого вырабатываются сигналы для блока шифратора, необходимого для преобразования двоичной последовательности в адрес, при ассоциативном чтении или записи поступающий на входы селектора адреса. Для хранения аргумента поиска в модуле имеется регистр аргумента. Управляет работой сопроцессора блок дешифрации команд, вырабатывающий управляющие сигналы. Из состава сопроцессора исключен регистр маски, т. к. схема сравнения вырабатывает три сигнала («РАВНО», «БОЛЬШЕ» и «МЕНЬШЕ») и надобность в маскировании бит аргумента поиска отпадает (рисунок 3, в).



а



б



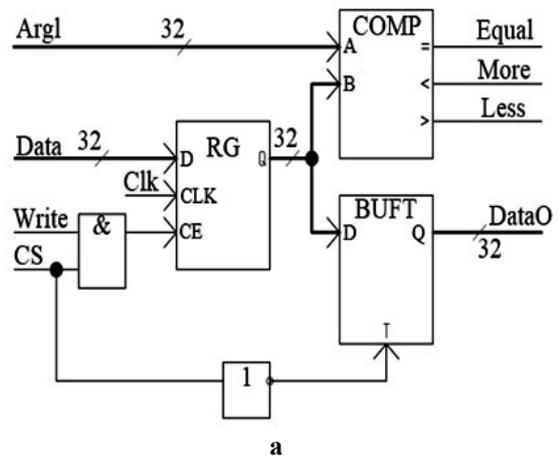
в

Рисунок 3 – Упрощенная структурная схема модуля ассоциативного сопроцессора (а); структурная схема ячейки памяти (б); подробная структурная схема модуля ассоциативного сопроцессора (в)

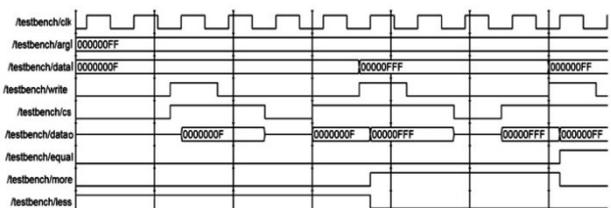
**Описание модуля ассоциативного сопроцессора на функциональном уровне**

Основой ассоциативного сопроцессора является модуль памяти, представляющий собой массив ЯП. Запоминающий элемент ЯП реализо-

ван на параллельном регистре, являющемся массивом D–триггеров, обеспечивающих максимальную производительность и минимум логики, необходимой для обеспечения запоминания информации. Основными сигналами для регистра являются: 32-х разрядный сигнал D, по которому поступают данные, сигнал CE и 32-х разрядный сигнал Q, с которого данные, хранящиеся в регистре, считываются. Схема сравнения реализована на базе компаратора. Основными сигналами для него являются: 32-х разрядные сигналы A и B, на которые подаются аргументы для сравнения, сигналы =, < и >, с которых считываются результаты сравнения. В состав ячейки памяти включен буферный элемент для отключения выходной шины данных ЯП от общей выходной шины. Основными сигналами буферного элемента являются: 32-х разрядный сигнал D, 32-х разрядный сигнал Q и сигнал T. Функциональная схема ЯП приведена на рисунке 4, а. Принцип работы ЯП следующий. На вход ArgI подается аргумент поиска. Через вход DataI данные попадают в ЯП. С выхода DataO данные считываются из ЯП. Входы Write и CS служат для управления работой ЯП. Сигналом CS ЯП выбирается, т.е. буферный элемент BUFT пропускает сигналы с выхода регистра RG на выходную шину DataO. По единичному сигналу Write данные с входа DataI записываются в RG. На выходах Equal, More и Less формируются сигналы «Равно», «Больше» и «Меньше» соответственно.



а



б

Рисунок 4 – Функциональная схема ячейки памяти (а); временные диаграммы работы ячейки памяти (б)

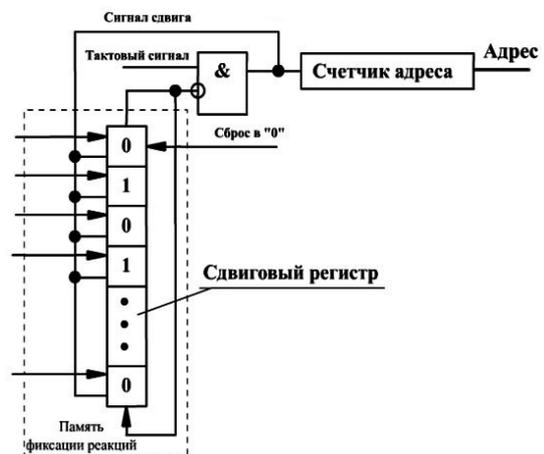
Временные диаграммы работы ЯП, подтверждающие ее работоспособность, приведены на рисунке 4, б. Здесь показана запись ряда значений: «F», «FFF» и «FF». Аргументом поиска является «F». Из полученных временных диаграмм видно, пока сигнал CS равен логическому нулю, выходная шина (DataO) находится в третьем состоянии, т.е. отключена. При подаче на вход CS логической единицы с выходной шины данных (DataO) можно считывать значения, хранящиеся в RG. Запись в ячейку осуществляется подачей логической единицы на вход Write. Из рисунка 4, б видно, что как только в ЯП записывается новое значение, на выходах Equal, More и Less устанавливаются результаты поиска.

АМС можно реализовать на базе регистра сдвига (рисунок 5, а) и на базе приоритетного анализатора (рисунок 5, б). Основными элементами АМС на базе регистра сдвига являются закольцованный регистр сдвига и счетчик адреса. Результат поиска всех ЯП записывается в закольцованный регистр сдвига (память фиксации реакций). Затем на регистр и на счетчик подается последовательность тактовых импульсов. Содержимое регистра сдвигается в сторону верхних разрядов до тех пор, пока в первом из них не окажется логическая «1». В этот момент тактовый сигнал автоматически блокируется. Если в исходном состоянии в счетчике были записаны одни нули, то по окончании счета его содержимое непосредственно указывает адрес первого совпавшего слова. Этот код заносится в адресный регистр, после чего осуществляется считывание слова. Далее единица в первом разряде регистра сбрасывается и автоматически возобновляется подача тактовых импульсов. Вновь содержимое регистра сдвигается вверх до появления в его первом разряде очередной единицы. После этого считывается следующее совпавшее слово и т. д. до тех пор, пока не будет обслужена вся очередь. Следует отметить, что при такой организации выборки отпадает необходимость в шифраторе адреса.

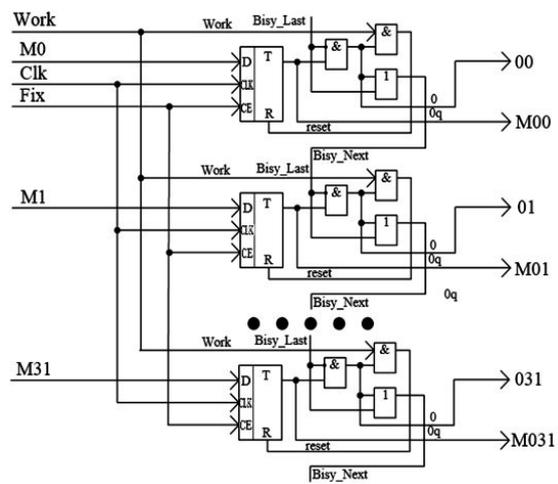
АМС на базе приоритетного анализатора состоит из D-триггеров, выполняющих функции памяти фиксации реакций и комбинационной логики. Данная схема работает по фронту сигнала Clk. Результаты поиска в ЯП поступают на входы M0...M31, по сигналу Fix они фиксируются на D-триггерах. Сигнал Work разрешает работу АМС. Сам приоритетный анализатор представляет собой логическую цепь, позволяющую выделять среди своих входов, установленных в «1», линию с наименьшим номером. Он построен по принципу последовательного соединения разрядов. Каждый единичный вход

этой цепи блокирует действие линий с большими номерами, в результате чего в единицу устанавливается только выход, соответствующий первой активной линии. Единичный сигнал, соответствующий первой активной линии, автоматически идет на выход, и функцией сигнала reset является сброс первого из «ответивших» триггеров.

Память фиксации реакций и АМС реализованы на базе схемы приоритетного анализатора, т.к. эта схема обладает большим быстродействием, чем схема на базе регистра сдвига. Полученные временные диаграммы работы схемы представлены на рисунке 5, в.



а



б

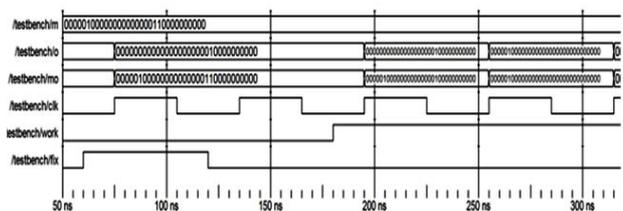
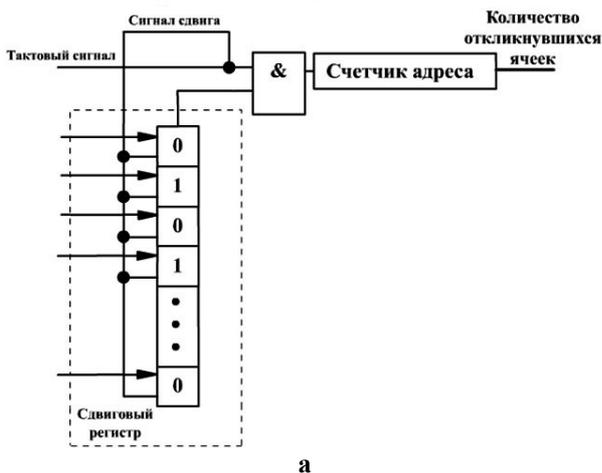


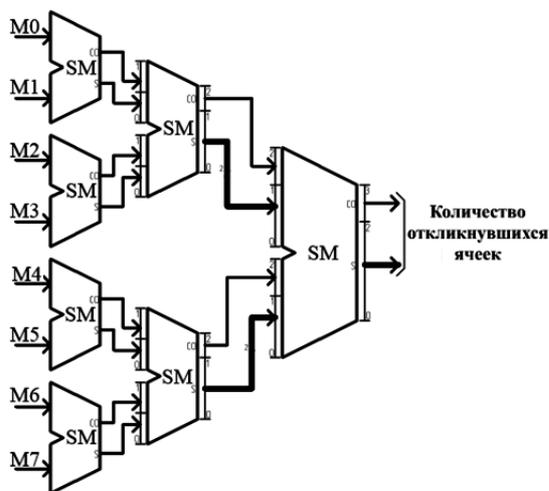
Рисунок 5 – Анализатор многократного совпадения на базе регистра сдвига (а) и на базе приоритетного анализатора (б); временные диаграммы работы памяти фиксации реакций и анализатора многократного совпадения (в)

Блок подсчета совпадений можно реализовать несколькими способами: на базе регистра сдвига и счетчика и на базе каскадов сумматоров.

Схема блока подсчета совпадений на базе регистра сдвига и счетчика (рисунок 6, а) состоит из сдвигового регистра, в который загружается последовательность единиц и нулей, соответствующая откликнувшимся и не откликнувшимся ячейкам, счетчика, который подсчитывает количество, и элемента «логическое И», управляющего процессом подсчета. На схему подается тактовый сигнал. Содержимое сдвигового регистра сдвигается на один разряд вверх и в зависимости от значения на выходе сдвигового регистра значение счетчика увеличивается или не увеличивается, т.е. происходит подсчет всех разрядов регистра, равных единице. Количество тактовых импульсов, необходимых для подсчета всех ячеек, равно количеству всех ячеек памяти (в нашем случае это 32 такта). К тому же данная схема требует дополнительного такта для загрузки сдвигового регистра.



а



б

Рисунок 6 – Блок подсчета совпадений на базе регистра сдвига и счетчика (а) и на базе каскадов сумматоров (б)

Схема на базе каскадов сумматоров (рисунок 6, б) представляет собой каскады сумматоров разной разрядности, т.е. входную двоичную последовательность разбиваем на группы по два и подаем эти группы на входы одноразрядных сумматоров. Промежуточные результаты, полученные на одноразрядных сумматорах, так же объединяем в группы по два и подключаем вместе с сигналом переноса на вход двух разрядных сумматоров и так далее. По сравнению с предыдущей схемой эта является более быстродействующей и в принципе может подсчитывать количество ячеек за один такт, поэтому блок подсчета реализован на базе каскадов сумматоров.

Правильность функционирования модуля ассоциативного сопроцессора проверялась в ходе выполнения ряда вычислительных экспериментов. Опытным путем было получено, что применение аппаратного сопроцессора (при выполнении операций поиска и сравнения) позволяет повысить производительность вычислительной системы в среднем примерно на 25 % по сравнению с системами, включающими только лишь традиционные процессоры. Иными словами, система с ассоциативным сопроцессором показывает в 1,25 раза большую производительность, чем аналоги. Отсюда следует целесообразность применения данной разработки на практике. Плата расширения с модулем ассоциативного сопроцессора может быть использована для повышения производительности серверов (за счет обработки информации на этапе считывания из памяти, а также уменьшения нагрузки на шины) баз данных типа Oracle и SQL Server, расположенных на машинах x86, а также для обработки графики, при этом применение платы расширения должно быть экономически оправдано по критерию «стоимость – эффективность».

В статье разработана структура модуля ассоциативного сопроцессора на базе ПЛИС, отличающаяся от аналогов тем, что рассматриваемое устройство выполнено в виде функционально независимого блока, в котором реализованы алгоритмы для выполнения трудоемких операций поиска и сравнения данных. В традиционной организации подобных устройств эти операции выполнялись непосредственно процессором. Также необходимо отметить еще одно преимущество данной разработки, которое связано с реализацией рассматриваемого модуля на современной элементной базе (ПЛИС).

Рассматриваемый аппаратный модуль ассоциативного сопроцессора реализован на ПЛИС фирмы Xilinx. Разработаны VHDL-коды устройства, состоящего из четырех модулей, включающих основные блоки рассматриваемого уст-

ройства: ассоциативный сопроцессор, память фиксации реакций и анализатора многократного совпадения, описание PCI – интерфейса, сопряжение ассоциативного сопроцессора и PCI – интерфейса.

### Выводы

В работе описана структура модуля ассоциативного сопроцессора на базе ПЛИС. На основании приведенного описания блоков устройства на функциональном уровне был разработан и отлажен VHDL-код ассоциативного сопроцессора.

Упомянутое в работе устройство физически можно реализовать в виде платы расширения ВС, подключаемой по интерфейсу PCI. Ассоциативный сопроцессор реализован аппаратно, что позволяет выполнять трудоемкие операции поиска и сравнения, тем самым разгружая центральный процессор и увеличивая производительность ВС в целом.

Описанный в статье ассоциативный сопроцессор выполняет следующие функции:

- 1) адресная запись в АЗУ;
- 2) адресное чтение из АЗУ;
- 3) поиск данных равных, больших и меньших аргумента поиска;
- 4) ассоциативное чтение (чтение элементов больше, меньше или равных аргументу);
- 5) ассоциативная запись (запись в ячейки памяти, содержимое которых больше, меньше или равно аргументу).

Работоспособность устройства и отдельных блоков проверена тестированием и отладкой разработанных кодов VHDL.

*Работа выполнена при финансовой поддержке РФФИ (грант № 16-07-00012).*

### Библиографический список

1. **Кохонен Т.** Ассоциативные запоминающие устройства: пер. с англ. М.: Мир, 1982. 384 с.
2. **Таненбаум Э., Бос Х.** Современные операционные системы. СПб.: Питер, 2015. 1120 с.
3. **Мартышкин А. И.** Исследование подсистем памяти с буферизацией транзакций на моделях массового обслуживания // XXI век: итоги прошлого и проблемы настоящего плюс. 2011. № 3. С. 124-131.
4. **Мартышкин А. И.** Разработка аппаратного буферного устройства памяти многопроцессорной системы // Фундаментальные исследования. 2015. № 12-3. С. 485-489.
5. **Мартышкин А. И.** Реализация аппаратного буфера памяти многопроцессорной системы // Труды XII Международной научно-технической конференции «Новые информационные технологии и системы» (г. Пенза, 17-19 ноября 2015 г.). Пенза: ПГУ, 2015. С. 96-99.
6. **Мартышкин А. И.** Математическое моделирование аппаратного буфера памяти многопроцессорной системы // Сборник материалов XII Международной научно-технической конференции «Опτικο-электронные приборы и устройства в системах распознавания образов, обработки изображений и символьной информации. Распознавание-2015» (г. Курск, 12-16 мая 2015 г.). Курск: Юго-Зап. гос. у-нт, 2015. С. 247-249.
7. **Мартенс-Атюшев Д. С., Мартышкин А. И.** Реконфигурируемый вычислительный кластер для цифровой обработки сигнала // Сборник статей XIII Всероссийской научно-технической конференции «Современные методы и средства обработки пространственно-временных сигналов» (г. Пенза, 3-4 июня 2015 г.). Пенза: ПДЗ, 2015. С. 112-117.
8. **Мартенс-Атюшев Д. С., Мартышкин А. И.** Разработка реконфигурируемой вычислительной системы для цифровой обработки сигнала // Технические науки – от теории к практике. 2015. № 52. С. 50-58.
9. **Мартышкин А. И.** Разработка и исследование разомкнутых моделей подсистемы «процессор-память» многопроцессорных вычислительных систем архитектур UMA и NUMA // Вестник Рязанского государственного радиотехнического университета. 2015. № 54-1. С. 121-126.
10. **Мартышкин А. И.** Разработка и исследование математических моделей подсистемы «процессор-память» многопроцессорных вычислительных систем архитектур NUMA и SUMA на разомкнутых сетях массового обслуживания // Вестник Рязанского государственного радиотехнического университета. 2016. № 55. С. 55-63.
11. **Огнев И. В., Борисов В. В.** Ассоциативные среды. М.: Радио и связь, 2000. 312 с.
12. **Цилькер Б. Я., Орлов С. А.** Организация ЭВМ и систем (2-е изд.), СПб: Питер, 2011. 688 с.

UDC 004.31

## ASSOCIATIVE COPROCESSOR MODULE BASED ON FPGA FOR SPECIALIZED COMPUTER SYSTEMS

**A. I. Martyshkin**, PhD (technical sciences), associate Professor at the Department of computers and systems, PSTU, Penza; alexey314@yandex.ru

*The paper considers the possibility of implementing a module of associative coprocessor by modern element base for specialized computer systems. **The aim of the article** is the development and study of associative co-processor module on FPGA, for specialized computing, for example, multi-processor, systems, performing associative function and data storage functions. **The object of research and development** of this article is an associative coprocessor based on FPGA. The search operation is widely used by users and systems for different purposes. However, this operation is one of the most labor-intensive and time-consuming when implemented in traditional way, when search data is sequentially read from RAM to CPU, where they perform appropriate operations. The paper proposes associative co-processor connected to PCI bus of a computer system, providing search and comparison of «more – less» simultaneously in 32 words pre-loaded in associative memory. Finally, conclusions have been made. The use of VHDL language as a universal means of hardware description of integrated circuits provides the flexibility of the project and the ease of debugging the operation of the device. Modeling the associative co-processor was carried out in CAD Web pack ISE of Xilinx company. This allows to evaluate the correct operation of hardware coprocessor in the composition of computing systems without building actual layout. Introduction efficiency of the module is assured by the fact that the coprocessor performs time-consuming operation to search for and compare data and thus relieves CPU and increases the performance of computing system as a whole. The results obtained in the article can find application in search engines for different purposes: servers, databases, search machines used at stations, airports and for fast implementation of operating systems search tasks.*

**Key words:** module, associative memory, addressing, coprocessor, computer system, memory cell, fixation reactions, analyzer of multiple coincidence, memory addressing, bus interface, hardware implementation, priority analyzer, write cycle, read cycle.

**DOI:** 10.21667/1995-4565-2016-58-4-75-82

### References

1. **Kohonen T.** Associativnye zapominajushhie ustrojstva (Associative storage device): per. s angl., M.: Mir, 1982, 384 p. (in Russian).
2. **Tanenbaum A., Bos H.** Sovremennye operacionnye sistemy (Modern operating systems). SPb, Piter, 2015, 1120 p. (in Russian).
3. **Martyshkin A. I.** Issledovanie podsystem pamjati s buferizaciej tranzakcij na modeljah massovogo obsluzhivaniya. XXI vek: itogi proshlogo i problemy nastojashhego pljus. 2011, no. 3. pp. 124-131. (in Russian).
4. **Martyshkin A. I.** Razrabotka apparatnogo bufernogo ustrojstva pamjati mnogoprocessornoj sistemy. Fundamental'nye issledovanija. 2015, no. 12-3, pp. 485-489. (in Russian).
5. **Martyshkin A. I.** Realizacija apparatnogo bufera pamjati mnogoprocessornoj sistemy. Trudy XII Mezhdunarodnoj nauchno-tehnicheskoy konferencii Novye informacionnye tehnologii i sistemy (Penza, 17-19 nojabrja 2015). Penza:PSU, 2015, pp. 96-99. (in Russian).
6. **Martyshkin A. I.** Matematicheskoe modelirovanie apparatnogo bufera pamjati mnogoprocessornoj sistemy. Sbornik materialov XII Mezhdunarodnoj nauchno-tehnicheskoy konferencii Optiko-jelektronnye pribory i ustrojstva v sistemah raspoznavaniya obrazov, obrabotki izobrazhenij i simvol'noj informacii. Raspoznavanie-2015 (Kursk, 12-16 maja 2015). Kursk:SWSU, 2015, pp. 247-249. (in Russian).
7. **Martens-Atjushev D. S., Martyshkin A. I.** Rekonfiguriruemij vychislitel'nyj klaster dlja cifrovoj obrabotki signala. Sbornik statej XIII Vserossijskoj nauchno-tehnicheskoy konferencii Sovremennye metody i sredstva obrabotki prostranstvenno-vremennyh signalov (Penza, 3-4 ijunja 2015). Penza:PDZ, 2015, pp.112-117. (in Russian).
8. **Martens-Atjushev D. S., Martyshkin A. I.** Razrabotka rekonfiguriruemoj vychislitel'noj sistemy dlja cifrovoj obrabotki signala. Tehnicheskie nauki – ot teorii k praktike. 2015, no. 52. pp. 50-58. (in Russian).
9. **Martyshkin A. I.** Razrabotka i issledovanie razomknutyh modelej podsystemy «processor-pamjat'» mnogoprocessornyh vychislitel'nyh sistem arhitektur UMA i NUMA. Vestnik Rjazanskogo gosudarstvennogo radiotehnicheskogo universiteta. 2015. no. 54, pp. 121-126. (in Russian).
10. **Martyshkin A. I.** Razrabotka i issledovanie matematiceskikh modelej podsystemy «processor-pamjat'» mnogo-processornyh vychislitel'nyh sistem arhitektur NUMA i SUMA na razomknutyh setjah massovogo obsluzhivaniya. Vestnik Rjazanskogo gosudarstvennogo radiotehnicheskogo universiteta. 2016, no 55, pp. 55-63. (in Russian).
11. **Ognev I. V., Borisov V. V.** Associativnye sredy, M.: Radio i svjaz', 2000, 312 p. (in Russian).
12. **Cil'ker B. Ja., Orlov S. A.** Organizacija JeVM i sistem (2-e izd.), SPb, Piter, 2011, 688 p. (in Russian).