БЛОК УПРАВЛЕНИЯ КОММУТАЦИЕЙ СМПО С ФУНКЦИЯМИ КОЛЛЕКТИВНЫХ И СЛУЖЕБНЫХ ОПЕРАЦИЙ

М. О. Костина, П. О. Костин

ФГУП «РФЯЦ-ВНИИЭФ», г. Саров Нижегородской обл.

В настоящее время разработана отечественная коммуникационная система межпроцессорных обменов (далее СМПО), предназначенная для построения высокопроизводительных вычислительных систем.

СМПО является программно-аппаратным комплексом в состав, которого входит аппаратный модуль СМПО-10СА и стек программного обеспечения.

В состав аппаратного модуля СМПО-10СА входит два блока (рис. 1):

– адаптерный блок СМПО-10СА-AD;

– коммутаторный блок СМПО-10СА-SW.

СБИС-СМПО объединяет в себе функционал необходимый для реализации на базе одной микросхемы, как адаптерного блока, так и коммутаторного блока. Выбор режима работы задается уровнем напряжения на соответствующем выводе микросхемы. Структурная схема СМПО представлена на рис. 2.

Блок коммуникационного управления выполняет функции транспортного уровня, и частично функции сетевого уровня. Доступ к памяти пользователя осуществляется через интерфейс PCI Express. Доступ к коммуникационной среде осуществляется через блок обработки пакетов сетевого уровня.

В СМПО для управления процессами приема и передачи данных, а так же для управления сервисными функциями используются процессоры, разработанные на основе открытого кода процессора Plasma.

Канальный блок системы межпроцессорного обмена – это блок высокоскоростного последовательного ввода/вывода, который позволяет работать с приемопередатчиками, расположенными на кристалле и использует медные или оптические кабели в качестве среды передачи данных.

Блок управления коммутацией имеет одиннадцать двунаправленных интерфейсов. Десять из этих интерфейсов взаимодействуют с канальными блоками, каждый из которых содержит по 2 виртуальных каналов. Соответственно одиннадцатый интерфейс, взаимодействует с блоком коммуникационного управления посредствам блока обработки и формирования пакетов сетевого уровня. Для блока управления коммутацией предусмотрено два режима работы, которые определяют типы каналов для взаимодействия, что делает его универсальным и позволяет создавать вычислительные сети с различной топологией.

Блок управления коммутацией может работать в одном из двух режимов:

– режим адаптера – в этом режиме блок управления коммутацией взаимодействует с 4-мя каналами и с блоком коммуникационного управления;

– режим коммутатора – в этом режиме блок управления коммутацией взаимодействует с 10-ю каналами;

Переключение режимов работы блока управления коммутацией происходит с помощью внешних сигналов. Смена типа маршрутизации выполняется с помощью внутренних сигналов, в режиме адаптера смена производится через PCIExpress, в режиме коммутатора через сервисный процессор, что позволяет гибко перенастраивать систему без изменения аппаратного обеспечения.

а б

Рис. 1. Аппаратные модули СМПО-10СА: а – адаптерный и коммутаторный блоки СМПО-10СА, б – блок коммутаторов СМПО-10СА

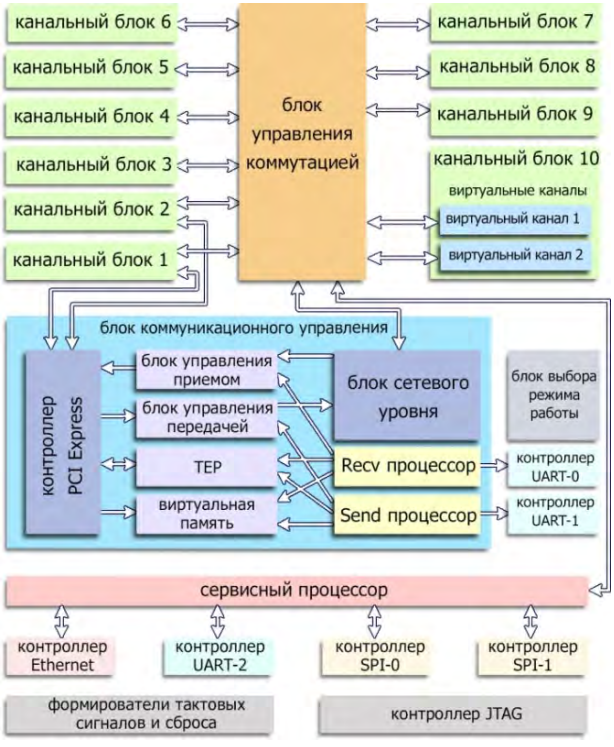


Рис. 2. Структурная схема СМПО

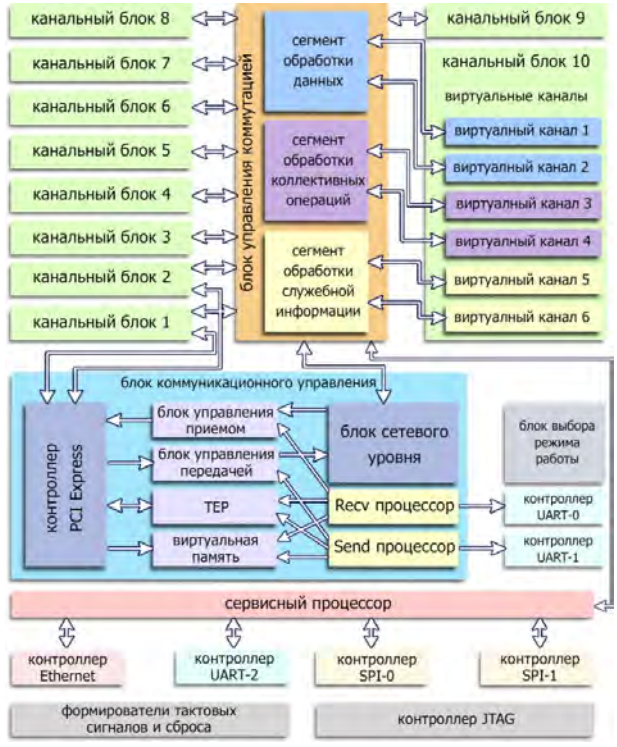


Рис. 3. Структурная схема СМПО с новым блоком управления коммутацией

Основная функция СМПО – это обеспечение взаимодействия прикладных программ путём обмена сообщениями. Одной из наиболее важных операций в параллельных приложениях являются коллективные обмены. От эффективности реализации этих операций зависит и эффективность исполняемой MPI-задачи. В данном докладе рассматривается реализация 3-х типов операций:

1. Unicast – однонаправленная (односторонняя) передача данных подразумевает под собой передачу пакетов единственному адресату;

2. Multicast – форма широковещания, при которой адресом назначения сетевого пакета является группа узлов;

3. Broadcast – метод передачи данных в сетях, при котором передаваемый пакет предназначен для приёма всеми участниками сети.

Ранее в СМПО была реализована только поддержка операций unicast. Такая реализация имеет ряд недостатков, таких как большая доля дублирующего трафика и плохая масштабируемость. В свою очередь аппаратная поддержка операций multicast и broadcast способствует повышению эффективности выполнения и масштабируемости параллельных программ. Поэтому для расширения функциональных возможностей коммуникационной среды, а именно реализации функций коллективных операций и передачи служебных пакетов, был разработан блок управления коммутацией СМПО с поддержкой функций коллективных и служебных операций. Структурная схема СМПО с новым блоком управления коммутацией представлена на рис. 3.

Разработанный блок управления коммутацией имеет одиннадцать двунаправленных интерфейсов. Для разделения потоков данных по сети и выполнения параллельной обработки пакетов разных типов блок управления коммутацией разделен на 3 сегмента: 1-ый сегмент отвечает за обработку пакетов с данными, 2-ой сегмент выполняет обработку коллективных операций, 3-й сегмент выполняет обработку служебных пакетов. Для каждого сегмента реализована своя виртуальная подсеть, состоящая из 2-х виртуальных каналов.

К основным функциям блока управления коммутацией относятся: коммутация пакетов данных между виртуальными каналами, в соответствии с адресом назначения, установленным в пакете, вычисление выходного порта для передачи каждого пакета, маршрутизация пакетов по сети согласно выбранному алгоритму.

Структурная схема блока управления коммутацией подставлена на рис. 4.

Как видно из рис. 4 все 3 сегмента схожи по структуре, но имеют различный функционал. Далее рассмотрим более подробно принципы работы каждого из 3-х сегментов.

Сегмент, выполняющий обработку пакетов с данными, взаимодействует с виртуальной подсетью, которая состоит из двух виртуальных каналов с отдельными буферами и специальными правилами маршрутизации (рис. 5). Виртуальная подсеть может быть построена по топологиям 2d, 3d , multi тор.

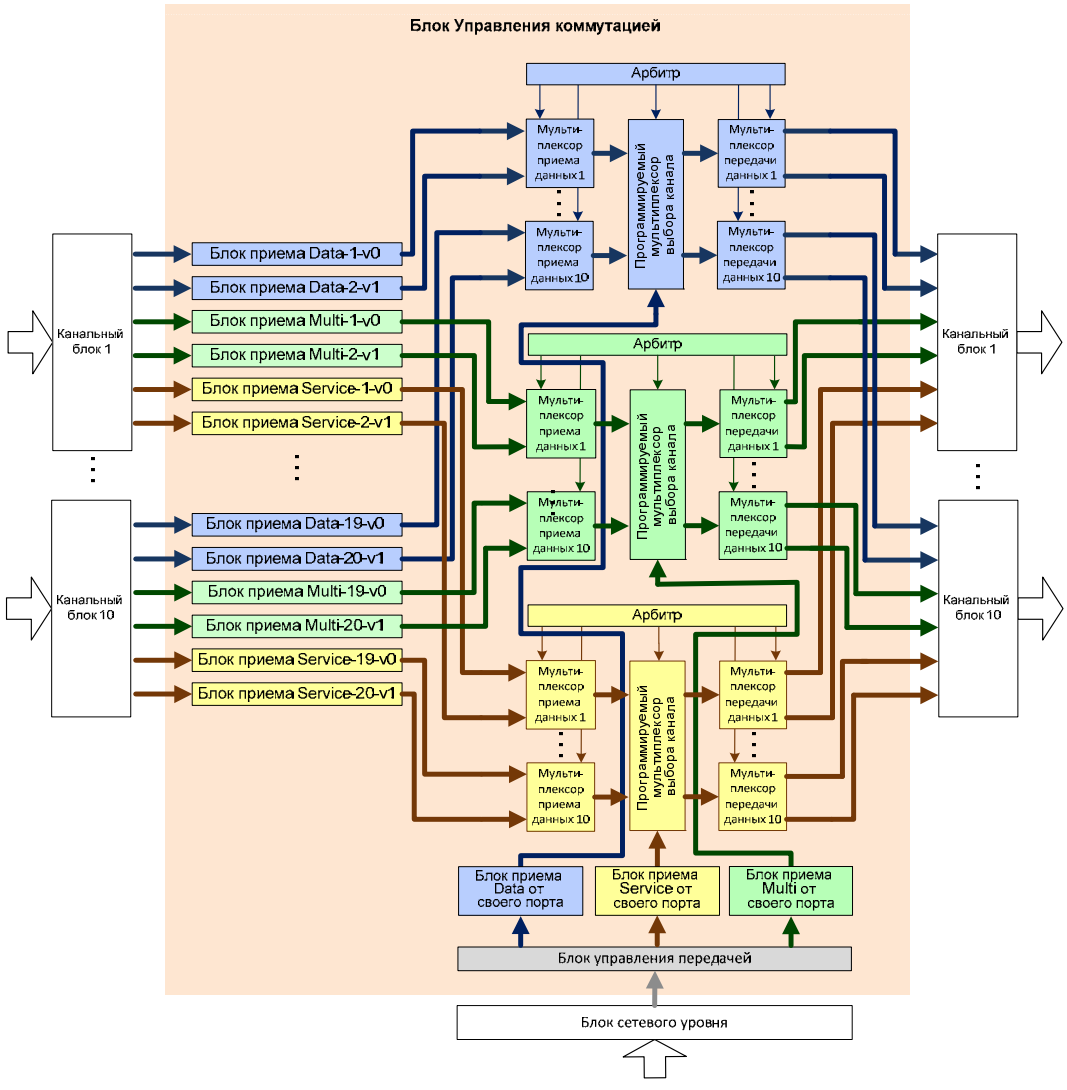


Рис. 4. Структурная схема блока управления коммутацией

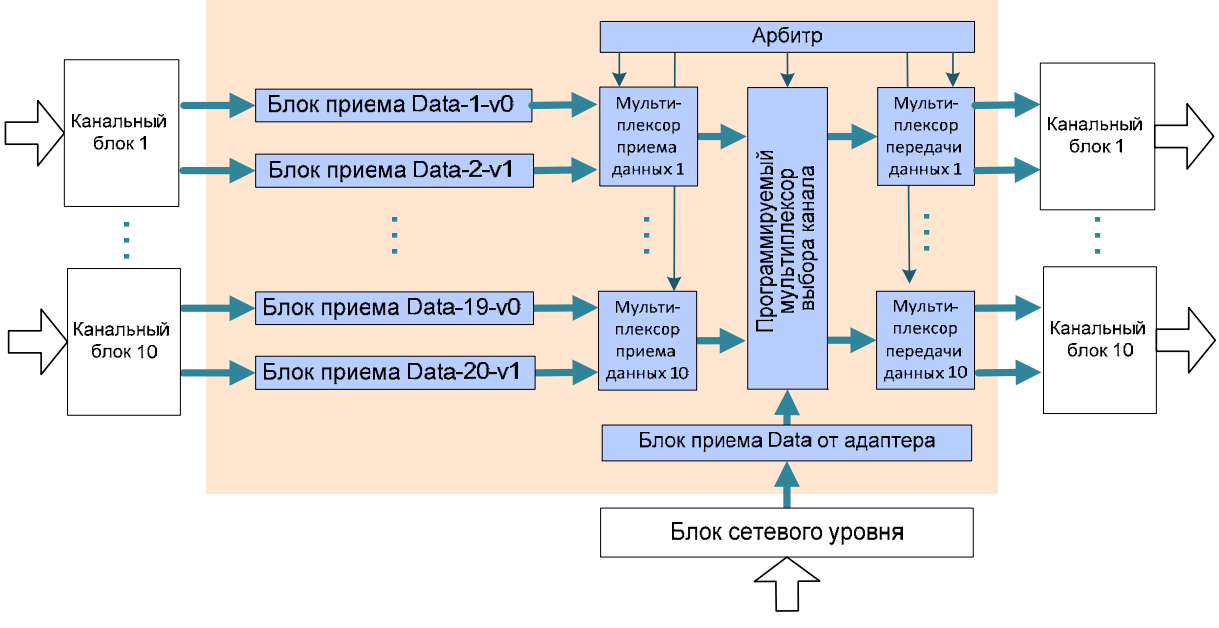


Рис. 5. Структурная схема сегмента обработки данных

В зависимости от выбранного режима работы (адаптер или коммутатор) и топологии в данном сегменте используется один из следующих типов маршрутизации пакетов по сети:

– адаптивный маршрутный алгоритм с топологией 2D тор (в режиме адаптера);

– адаптивный маршрутный алгоритм с топологией 3d или MultiTor (в режиме адаптера);

– адаптивный маршрутный алгоритм с топологией 3d или MultiTor (в режиме коммутатора);

– табличная маршрутизация (в режиме адаптера или коммутатора).

Сегмент, выполняющий обработку коллективных операций, взаимодействует с виртуальной подсетью, которая состоит из двух виртуальных каналов (рис. 6). Виртуальная подсеть имеет топологию дерева, наложенного на мульти-тор. Каждому направлению соответствует свой виртуальный канал. Маршрутизация пакетов осуществляется по данным из коммутационной таблицы, которая рассчитываться

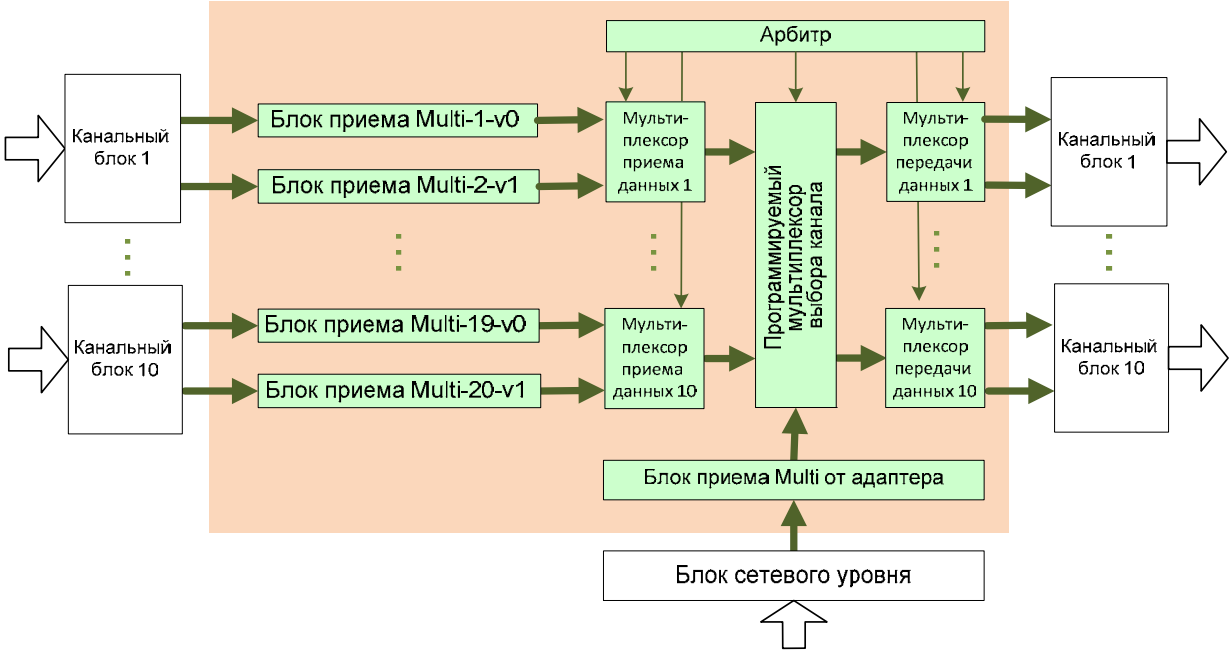


Рис. 6. Структурная схема сегмента обработки коллективных операций

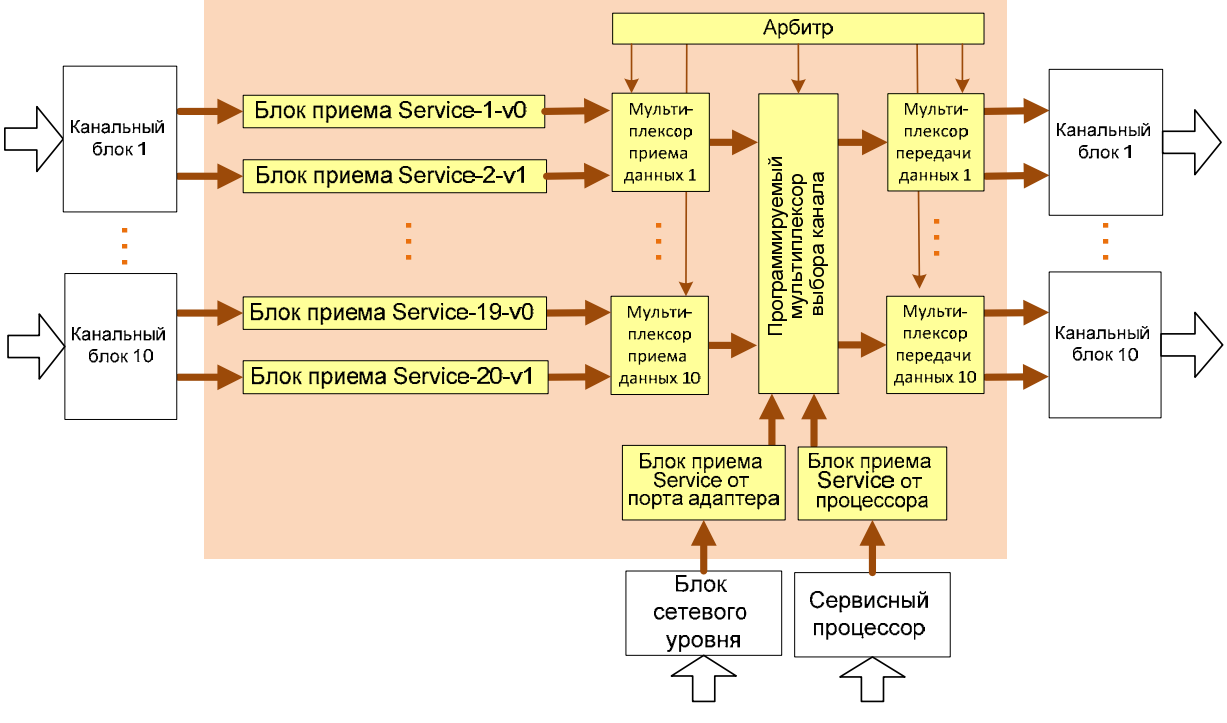


Рис. 7. Структурная схема сегмента обработки служебной информации

для каждого виртуального канала. В этом сегменте реализован механизм выдачи копий пакета и поддержка операций multicast и broadcast.

Сегмент взаимодействует с виртуальной подсетью, состоящей из двух виртуальных каналов (рис. 7). Виртуальная подсеть может быть построена по топологиям 2d, 3d , multi tor. В отличии от сегмента обработки данных, в этом сегменте добавился блок приема пакетов от сервисного процессора, который в режиме коммутатора позволяет: производить назначение адресов, выполнять опрос сети, считывать диагностическую информацию состояния внутренних блоков СМПО, выполнять загрузку таблиц маршрутизации. Ранее все эти операции в коммутаторном модуле СМПО-10CA-SW проводились с через Ethernet, что не обеспечивало необходимый уровень быстродействия и требовало прокладки дополнительных линий связи. В зависимости от выбранного режима работы (адаптер или коммутатор) и топологии в данном сегменте используется один из следующих типов маршрутизации пакетов по сети:

– адаптивный маршрутный алгоритм с топологией 2D tor (в режиме адаптера);

– адаптивный маршрутный алгоритм с топологией MultiTor (в режиме адаптера);

– адаптивный маршрутный алгоритм с топологией MultiTor (в режиме коммутатора).

Все сегменты состоят из следующих основных блоков:

– блок приема данных – выполняет обработку принимаемого пакета, формирует заявку на передачу и запрашивает разрешение на передачу данных в выходной канал.

– арбитр – отвечает за синхронизацию внутренних блоков, обрабатывает заявки на коммутацию, отслеживает состояние и управляет работой все блоков входящих в состав коммутатора.

– программируемый мультиплексор выбора канала, мультиплексор приема данных и мультиплексор передачи данных – отвечают за коммутацию пакетов с данными.

Блок приема данных выполняет обработку принимаемого пакета, запрашивает разрешение на передачу данных в выходной канал. Структурная схема блока приема данных представлена на рис. 8.

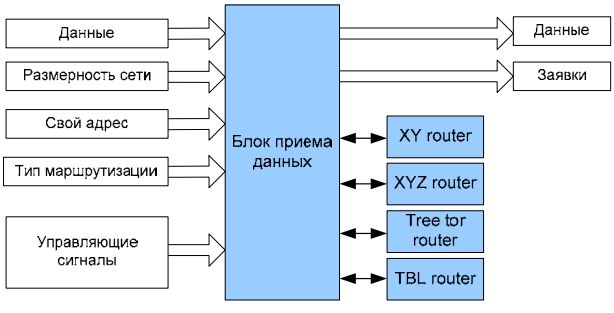


Рис. 8. Структурная схема блока приема данных

На вход блоку приема данных идут такие сигналы как: размерность сети, адрес узла, тип маршрутизации, данные. Блок приема данных взаимодействует с блоками маршрутизаций: XY router, XYZ router, Treetorrouter, TBL router.

Блок приема данных ожидает поступления данных в канал. Как только данные поступают в канал, блок приема данных считывает заголовок пакета данных. После того, как заголовок пакета был получен, производится отработка полученных данных. Проверяется TTL пакета, оно не должно превышать максимально разрешенного значения иначе пакет автоматически отбрасывается. Производится проверка сигнала отвечающего за выбор типа маршрутизации и запрашивается выходной порт. Далее формируется заявка на коммутацию и запрашивает разрешение на передачу данных у арбитра. Формат заявки на коммутацию представлен на рис. 9.

Рис. 9. Формат заявки на коммутацию

Структурная схема арбитра приведена на рис. 10.

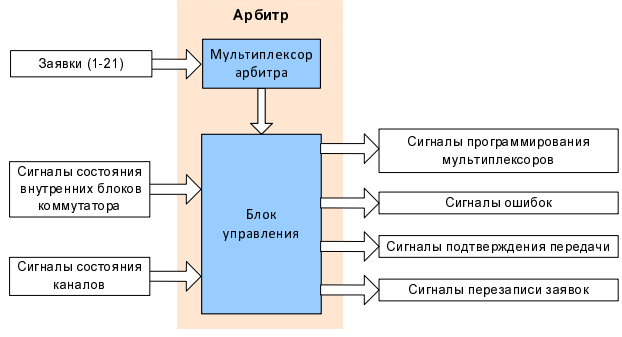


Рис. 10. Структурная схема арбитра

Арбитр состоит из двух модулей: программируемый мультиплексор и блок управления, напрямую соединенных между собой.

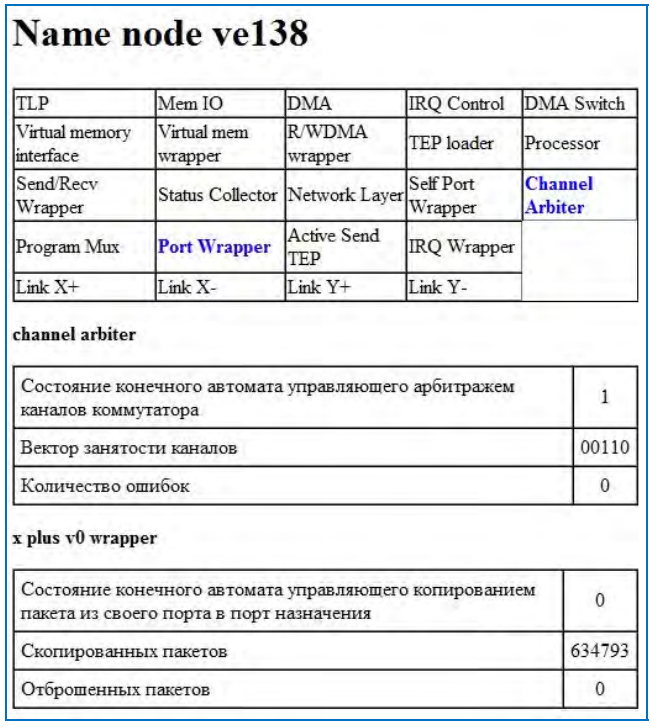
Вначале программируемый мультиплексор выполняет проверку очереди поступивших заявок. Если есть новые заявки, то очередь сохраняется, а заявки в ней последовательно обрабатываются и передаются арбитру. Когда сохраненная очередь полностью обработана, выполняется проверка очереди повторных заявок на передачу. Если в этой очереди появились заявки, то очередь сохраняется, а заявки в ней последовательно обрабатываются и передаются арбитру. После этого алгоритм выполняется заново.

Управляет процессом коммутации блок управления. Он обрабатывает заявки на коммутацию, отслеживает состояние и управляет работой все блоков входящих в состав коммутатора.

Блок управления ожидает поступления новых заявок от программируемого мультиплексора. Если заявка поступила, начинается проверка условий.

Сначала проверяется наличие линка на стороне приемника, затем выполняется проверка статусов мультиплексоров приема и передачи данных и проверка сигнала переполнения буфера на стороне приемника. Если все условия выполнены, выставляется разрешение на передачу данных и производится настройка необходимых мультиплексоров. Иначе выставляется сигнал повтора заявки. Затем алгоритм повторяется.

Для визуального мониторинга состояния блока управления коммутацией в нем реализована система сбора статистики. Интерфейс системы сбора статистики представлен на рис. 11.



Name node ve138

TLP	Mem IO	DMA	IRQ Control	DMA Switch
Virtual memory interface	Virtual mem wrapper	R/WDMA wrapper	TEP loader	Processor
Send/Recv Wrapper	Status Collector	Network Layer	Self Port Wrapper	**Channel Arbiter**
Program Mux	**Port Wrapper**	Active Send TEP	IRQ Wrapper	
Link X+	Link X-	Link Y+	Link Y-	

channel arbiter

Состояние конечного автомата управляющего арбитражем каналов коммутатора	1
Вектор занятости каналов	00110
Количество ошибок	0

x plus v0 wrapper

Состояние конечного автомата управляющего копированием пакета из своего порта в порт назначения	0
Скопированных пакетов	634793
Отброшенных пакетов	0

Рис. 11. Интерфейс системы сбора

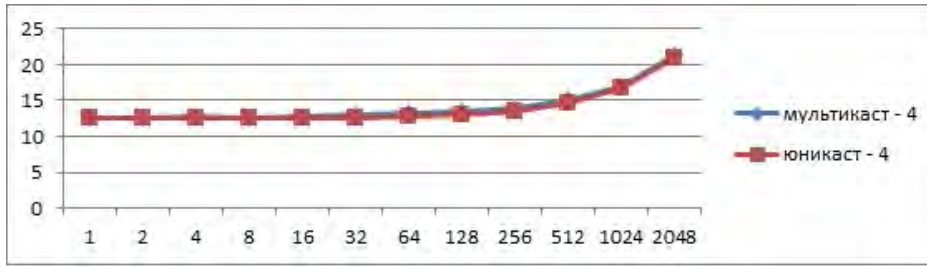


Рис. 12. Результаты тестирования на 4-х узлах

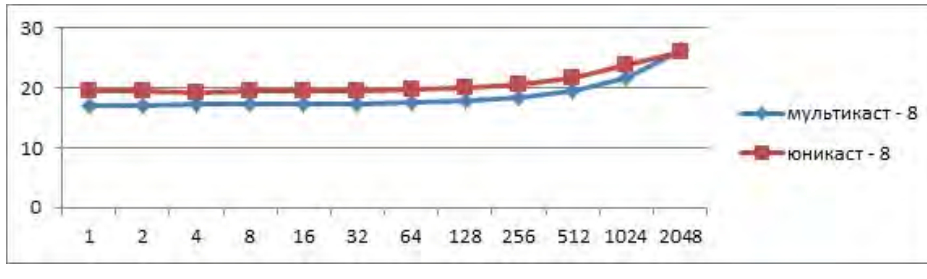


Рис. 13. Результаты тестирования на 8-ми узлах

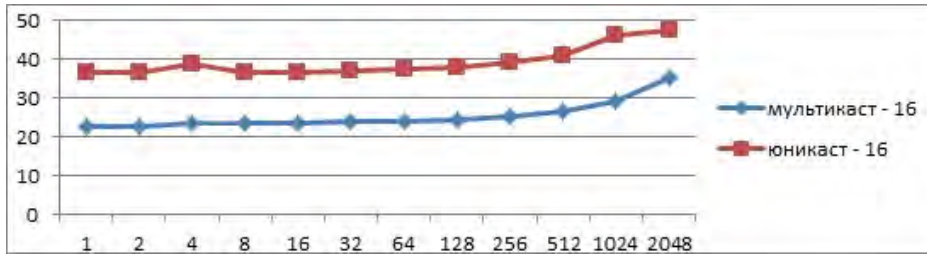


Рис. 14. Результаты тестирования на 16-ти узлах

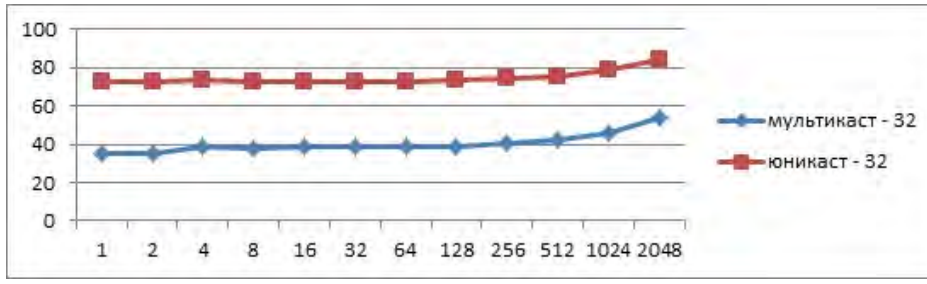


Рис. 15. Результаты тестирования на 32-х узлах

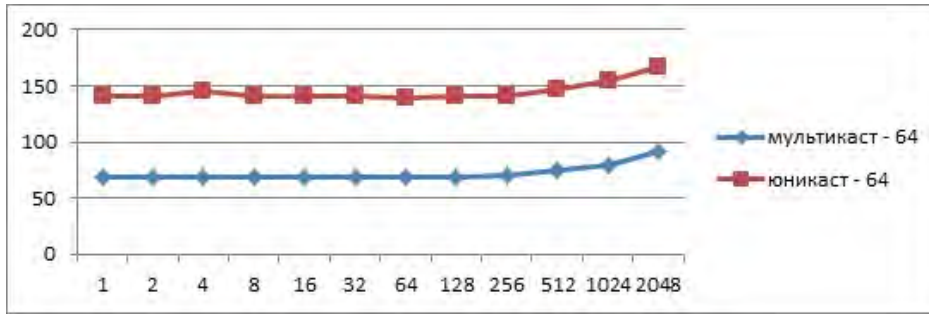


Рис. 16. Результаты тестирования на 64-х узлах

Система сбора статистики предоставляет доступ к следующим данным:

- количество переданных пакетов;
- количество отброшенных пакетов;
- заявка на коммутацию каналов;
- состояния конечных автоматов управляющих передачей пакетов;
- регистры управления каждым из каналов.

Для оценки производительности коммуникационной среды с использованием разработанного блока управления коммутацией выполнялось тестирование выполнения операций коллективных обменов на вычислительной системе соединенной по топологии мульти-тор. Тестирование проводилось на 4, 8, 16, 32 и 64 узлах с использованием функций multicast и unicast.

Результаты, полученные при тестировании, представлены на рис. 12–16.

Проведенное тестирование показало, что с использованием функций широковещания multicast при увеличении числа узлов в сети в разы уменьшается время, затрачиваемое на выполнение операций коллективных обменов, что способствует повышению эффективности выполнения параллельных программ и росту производительности.