

Метод обеспечения отказоустойчивости коммуникационной сети с топологией многомерный тор

Симонов А.С.*, Жабин И.А.**

*Научно-исследовательский центр электронной вычислительной техники, НИЦЭВТ,
Варшавское шоссе, 125, Москва, 117587, Россия*

**e-mail: simonov@nicevt.ru*

***e-mail: zhabin@nicevt.ru*

Статья поступила 21.09.2020

Аннотация

В статье описан метод обеспечения отказоустойчивости коммуникационной сети с топологией многомерный тор, апробированный при создании высокоскоростной сети Ангара.

Работа по обеспечению отказоустойчивости велась по следующим направлениям: повышение качества передаваемого сигнала за счет тонкой настройки параметров высокоскоростных приемопередатчиков, разработка отказоустойчивого алгоритма контроля целостности линка с возможностью повторной передачи поврежденных данных, разработка алгоритмов сетевого уровня для мониторинга состояния сети, и оперативного перестроения таблиц маршрутизации.

Описываемые в статье методы позволяют для вычислительных систем большого размера получить коэффициент готовности более 0,99.

Ключевые слова: отказоустойчивость, коммуникационная сеть.

Современные коммуникационные сети [1] вычислительных кластеров и суперкомпьютеров позволяют объединять от нескольких сотен до десятков тысяч вычислительных узлов. Очевидно, что избежать возникновения отказов в таких больших системах практически невозможно, поэтому так важно обеспечить их полноценную обработку и купирование возможных последствий для приложений.

Простой расчёт показывает, что без применения специальных мер по защите от сбоев и отказов коэффициент готовности МВС будет в значительной степени зависеть от её размера и уже для нескольких десятков узлов будет достигать неприемлемо низких значений.

Для решения данной проблемы в рамках проекта по созданию высокоскоростной коммуникационной сети (далее ВКС) Ангара [2-5] был разработан и апробирован метод обеспечения отказоустойчивости ВКС с топологией многомерный тор, включающий:

- методику отказоустойчивой передачи данных на уровне линка;
- алгоритмы отказоустойчивой передачи данных на уровне узла;
- алгоритмы обеспечения отказоустойчивости на уровне сети.

Методика отказоустойчивой передачи данных [6-8] на уровне линка позволяет обеспечить толерантность ВКС к ошибкам, возникающим в процессе передачи данных по линиям связи. Для этого в рамках методики предложены:

- алгоритм инициализации связи по линку;
- механизмы контроля целостности данных и управления потоком данных [9-12].

Алгоритм инициализации связи по линку обеспечивает минимизацию интенсивности возникновения ошибок на линии связи (BER [13, 14]) за счёт автоматической оптимизации параметров формирователей предскажений трансиверов под характеристики линии связи, снижения пропускной способности трансиверов для линий связи с высоким уровнем ошибок, а также отключение неисправных лейнов.

Современные высокоскоростные трансиверы (SerDes), используемые для передачи данных по дифференциальным каналам связи, включают в свой состав средства, позволяющие за счёт коррекции сигналов обеспечить приемлемые значения интенсивности ошибок в линии передачи данных (Bit Error rate, BER). Это специальные блоки, обеспечивающие внесение предскажений в исходный сигнал таким образом, чтобы компенсировать искажения, вносимые линией связи.

Для контроля качества сигнала, поступающего из линии связи, с использованием высокоскоростных цифровых осциллографов можно осуществить контроль глазковой диаграммы [15, 16]. Этот метод позволяет оценить значения запасов (margin) по амплитуде и времени, и, таким образом, сделать вывод о соответствии или несоответствии требованиям стандартов. На рис. 1 представлен пример такой диаграммы с шаблоном запасов (margin), определяемых требованиями стандартов.

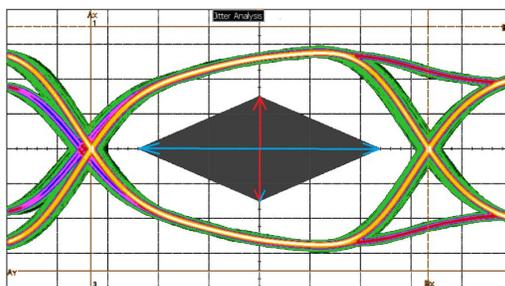


Рис. 1. Пример глазковой диаграммы с шаблоном (серый ромб внутри глазка) минимально допустимых значений запасов (margin) по амплитуде (красная стрелка) и времени (голубая стрелка)

Если шаблон помещается в глазок диаграммы, то трансиверы обеспечивают приемлемый уровень BER, как правило, в пределах 10^{-12} , если не помещается – уровень BER существенно возрастает.

Для осуществления настройки блоков предискажений разъёмы кабелей сетевого оборудования обычно включают в свой состав специальную микросхему, в которой хранится идентификатор кабеля. Учитывая то, что характеристики кабелей за счёт устойчивой технологии изготовления, имеют высокую повторяемость, идентификатор позволяет осуществить корректную настройку блоков предискажений трансиверов и компенсировать таким образом искажения, вносимые кабелем связи.

Однако, данный путь имеет существенный недостаток: каждый новый кабель или канал связи требует идентификации, проведения соответствующих измерений и внесения корректировок в сервисное программное обеспечение, осуществляющее настройку трансиверов, а это не всегда удобно, так как требует обращения к предприятию – разработчику сетевого оборудования. Кроме того, указанный метод не работает в случае необходимости стыковки кабелей, применяемой, например, при создании МВС с иммерсионной системой охлаждения.

Решение данного вопроса заключается в разработке и реализации алгоритма инициализации связи по линку, предусматривающего автоматическую настройку параметров блоков предискажений трансиверов под характеристики линии связи каждый раз при включении сетевого оборудования. Для реализации данного варианта трансивер должен иметь в своём составе специальный блок контроля глазковой диаграммы.

В данной работе такие алгоритмы были разработаны (см. рис. 2).

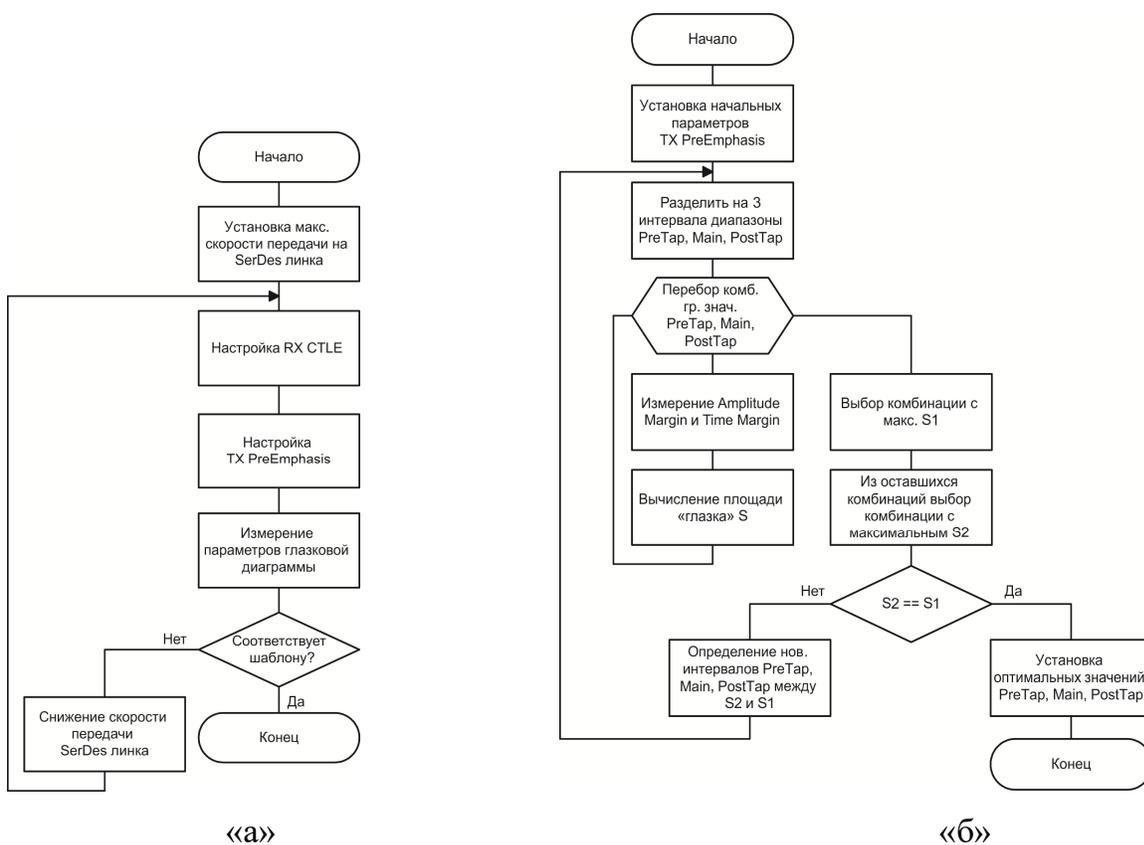


Рис. 2. Алгоритмы инициализации связи по линку с автоматической настройкой параметров блоков предискажений трансиверов под характеристики линии связи («а» - обобщённый алгоритм; «б» - алгоритм настройки TX PreEmphasis)

Тем не менее, применение настроек трансиверов не позволяет обеспечить безошибочную передачу данных по линиям связи. При современных скоростях

передачи данных нормальное значение BER на уровне 10^{-12} и ниже позволяет понизить интенсивность возникновения ошибок в линии связи до нескольких ошибок в час, что неприемлемо для большинства приложений.

Для устранения ошибок были разработаны адаптированные для применения совместно с дисциплиной передачи данных VCT [17, 18, 19] механизмы контроля целостности данных и управления потоком данных, обеспечивающие повторную передачу искажённых данных и защиту от переполнения буферов [11, 12].

Суть механизма контроля целостности данных состоит в том, что при отправке пакетов данных в линию связи на стороне передатчика также осуществляется и их сохранение в специальный RETRY-буфер. Если на стороне приёмника пакеты приняты корректно и это подтверждается в служебных полях ответных пакетов, то они удаляются из RETRY-буфера, если же пакеты приняты с ошибкой, осуществляется их повторная передача.

Функционирование механизма контроля целостности данных целесообразно рассмотреть на примере взаимодействия двух линков «А» и «В» (см. рис. 3).

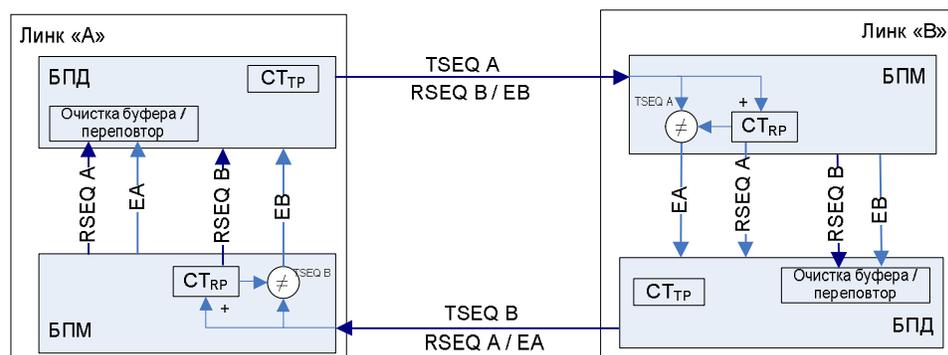


Рис. 3. Функционирование механизма контроля целостности данных

В выходной части линка «А» расположен счётчик СТТР, осуществляющий подсчёт числа переданных данным линком пакетов. Текущее значение данного

счётчика (порядковый номер пакета, SEQ) используется для заполнения поля {TSEQ} заголовочного флига каждого передаваемого линком «А» пакета (значение TSEQ А).

Во входной части линка «В» находится счётчик CTRP, необходимый для подсчёта принятых пакетов. Значение на выходе данного счётчика показывает номер следующего (ожидаемого) линком пакета. При приёме заголовка пакета выполняется сравнение значения поля {TSEQ}, сформированного линком «А» (TSEQ А), и значения на выходе данного счётчика. Если сравнение показало их равенство, линк «В» выполняет приём тела пакета.

По окончании приёма пакета, при отсутствии ошибок, значение счётчика CTRP увеличивается на единицу и передаётся на блок БПД линка «В» (значение RSEQ А) для последующей обратной передачи на линк «А» в составе заголовочного флига пакета любого типа в поле {RSEQ}.

Блок БПД линка «А», приняв от своего блока БПМ значение RSEQ А, находит в RETRY-буфере все пакеты, имеющие значение TSEQ меньше, чем принятое, и удаляет их из буфера.

Если сравнение в блоке БПМ линка «В» показало несоответствие принятого значения TSEQ А и значения внутреннего счётчика CTRP, либо проверка контрольной суммы [20, 21] {ECRC} показала наличие ошибки и бит правильности пакета {VB}=0, блок БПМ линка «В» передаёт на блок БПД линка «В» и далее в заголовочном флиге пакета бит {E}=1, означающий ошибку приёма.

При этом, если пакет ещё находится во входном буфере линка, он удаляется, иначе полю {VB} окончного флига присваивается значение 0xEА и ошибочный

пакет передаётся дальше. Если при приёме пакета поле $\{VB\}$ окончного флита уже равно $0xEA$, то если пакет ещё во входном буфере, он удаляется, иначе передаётся без передачи на блок БПД линка «В» бита $\{E\}=1$, означающего ошибку приёма.

Блок БПМ линка «В», выдав бит $\{E\}=1$, игнорирует все принимаемые флиты вплоть до прихода заголовочного флита с верным значением TSEQ или флита инициализации, содержащего новое значение порядкового номера пакета (SEQ) для счётчика STRP.

Блок БПД линка «А», приняв установленный бит $\{E\}=1$, находит во внутреннем буфере пакет со значением TSEQ, равным принятому в поле $\{RSEQ\}$, и осуществляет повторную передачу пакетов, начиная с найденного.

Основная задача, решаемая механизмом управления потоком данных, состоит в недопущении переполнения входных буферов линка и, как следствие, потери пакетов. Особенностью данного механизма является то, что в физическом канале может одновременно находиться сразу несколько пакетов данных, что затрудняет предсказание возможности передачи каждого следующего пакета на передающей стороне на основании получаемой информации о количестве свободного места на приёмной стороне, так как последняя к моменту начала анализа уже является неактуальной.

Механизм управления потоком данных основан на том, что на передающую сторону возвращается информация не о суммарном объёме свободного места в принимающем буфере, а лишь об объёме освободившегося места с момента предыдущей передачи указанной информации. Таким образом, на передающей

стороне появляются данные о количестве флитов, которые можно передать в физический канал без потерь из-за переполнения. Схожее решение описано в [22].

Учитывая, что входные буфера принадлежат виртуальным каналам, в каждом из них имеется свой механизм управления потоком данных. Работает он следующим образом (см. рис. 4).

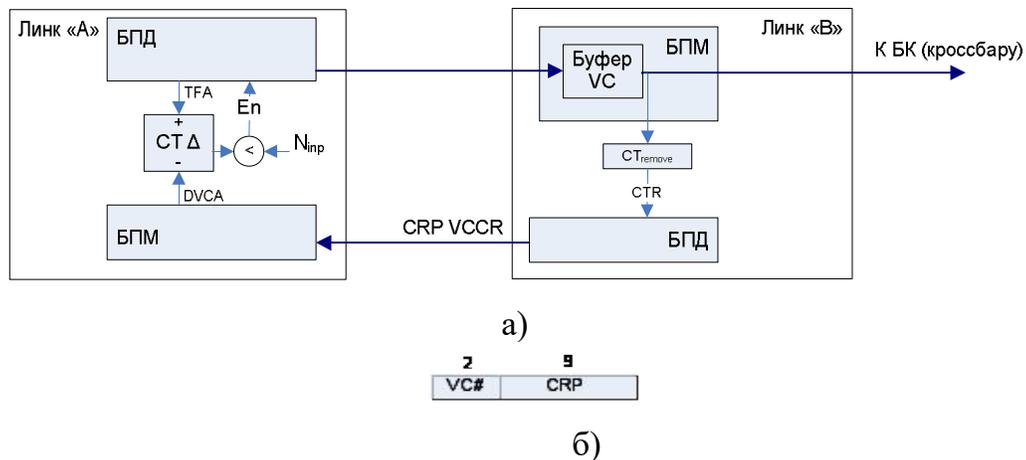


Рис. 4. Порядок функционирования механизма управления потоком данных одного виртуального канала (а) и формат поля {VCCR} заголовочного флита (б)

При передаче каждого флита счётчик CT_{Δ} линка «А» увеличивается на единицу. При этом выходное значение данного счётчика сравнивается со значением N_{inp} , объёма свободного места в буфере данного виртуального канала на приёмной стороне. Передача следующего флита возможна, если увеличенное на единицу значение на выходе CT_{Δ} меньше, чем N_{inp} . В свою очередь, линк «В» принимает флиты, переданные линком «А», которые в дальнейшем поступают на кроссбар для осуществления маршрутизации. При этом, счётчик CT_{remove} линка «В» осуществляет подсчёт количества флитов, ушедших из входного буфера данного VC, а блок БПД периодически посылает на линк «А» значение данного счётчика в поле {CRP VCCR} заголовочного флита, устанавливая при этом в поле {VC#} номер виртуального

канала, которому соответствует информация. Приняв заголовочный флит линк «А» уменьшает значение счётчика СТ Δ на полученное число освободившегося места из поля {CRP VCCR}. Очевидно, что для обеспечения толерантности системы к задержке передачи флитов входные буфера линка должны иметь достаточный объём.

Предложенные алгоритм и механизмы позволили обеспечить непрерывность потока передаваемых данных и их гарантированную доставку по линиям связи между узлами, исключение потерь, дублирований и искажений пакетов в процессе передачи.

При полном отказе линии связи соответствующие линки соседних маршрутизаторов переходят в состояние «неисправен» с периодическим контролем восстановления связи. При этом транзитным пакетам данных, попавшим в узел с неисправным линком, необходимо обеспечить возможность обхода отказавшего участка. Для этого был разработан алгоритм отказоустойчивой передачи на уровне узла, обеспечивающий корректную обработку отказа линка (см. рис. 5).

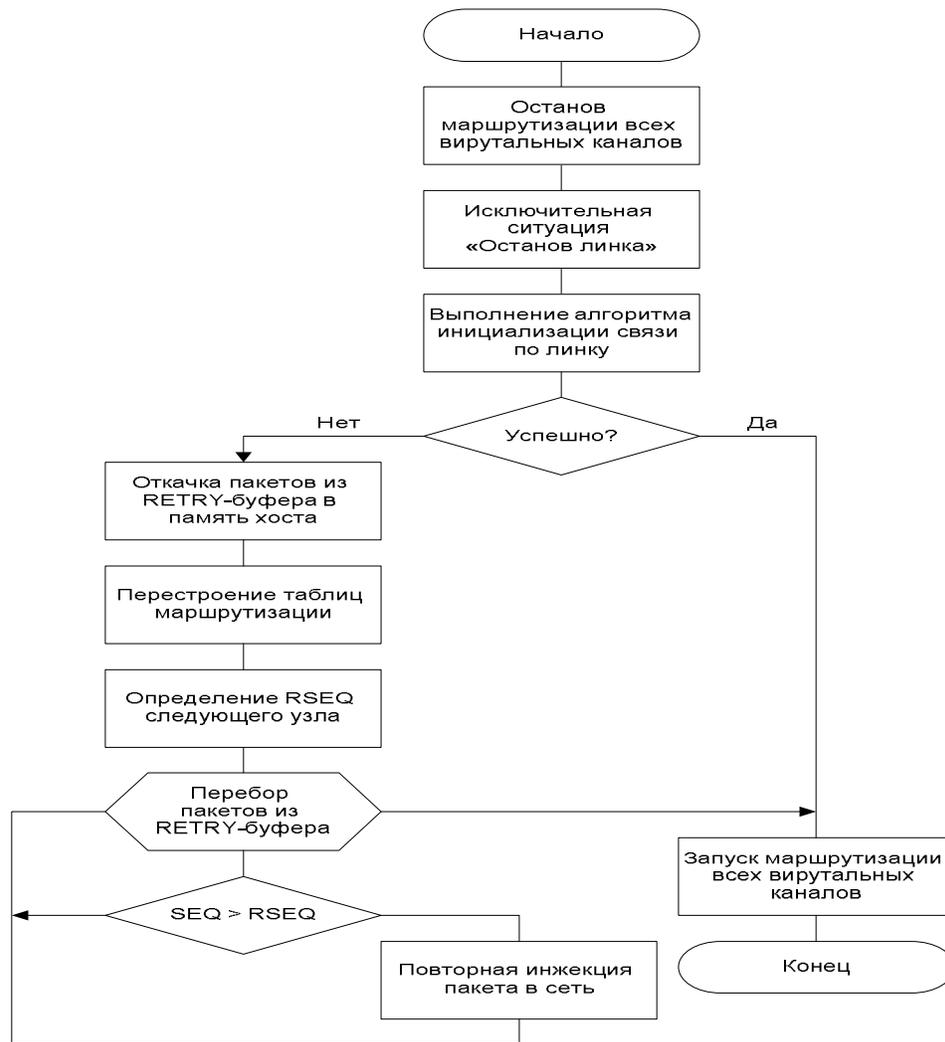


Рис. 5. Обобщённый алгоритм обработки отказа линка

Учитывая, что в каждой из трёх подсетей пакеты маршрутизируются по разным алгоритмам, для каждой из них предлагается свой алгоритм обработки отказа линка.

Так, для подсетей детерминированной передачи данных, образованных виртуальными каналами VCD0 и VCD1, требуется обеспечить не только обход отказавшего участка, но и сохранение порядка следования пакетов. Учитывая то, что решение этих вопросов аппаратными средствами было бы крайне затратным, предлагаемый алгоритм предполагает откачку всех поступивших пакетов в

служебный сегмент памяти хоста и их перемаршрутизацию после выполнения перестроения таблиц маршрутизации на всех узлах МВС.

При отказе одного или нескольких узлов МВС необходимо обеспечить:

- для транзитного трафика возможность обхода отказавших узлов путём перестроения таблиц маршрутизации;

- для задач, использовавших отказавшие узлы, перезапуск с контрольной точки на узлах, выделенных взамен отказавших;

- возможность «горячей замены» отказавших элементов (узлов, кабелей связи) с целью снижения времени на восстановление и обеспечения приемлемого значения коэффициента готовности МВС.

Перечисленные функции реализованы в алгоритме обеспечения отказоустойчивости на уровне сети.

Обход отказавших узлов МВС для транзитного трафика обеспечивается путём внесения соответствующей информации в таблицу запретов и перестроения таблиц маршрутизации [22, 23]. Узел, на котором изменилось состояние линка, с использованием сервисной сети направляет на корневой узел пакет данных, содержащий строку запретов, включающую информацию о состоянии для каждого линка. Полученная строка вносится в таблицу запретов, которая рассылается всем узлам. Узлы, в свою очередь, получив эту таблицу, осуществляют перестроение таблиц маршрутизации в соответствии с обобщённым алгоритмом, приведённым на рис. 6.

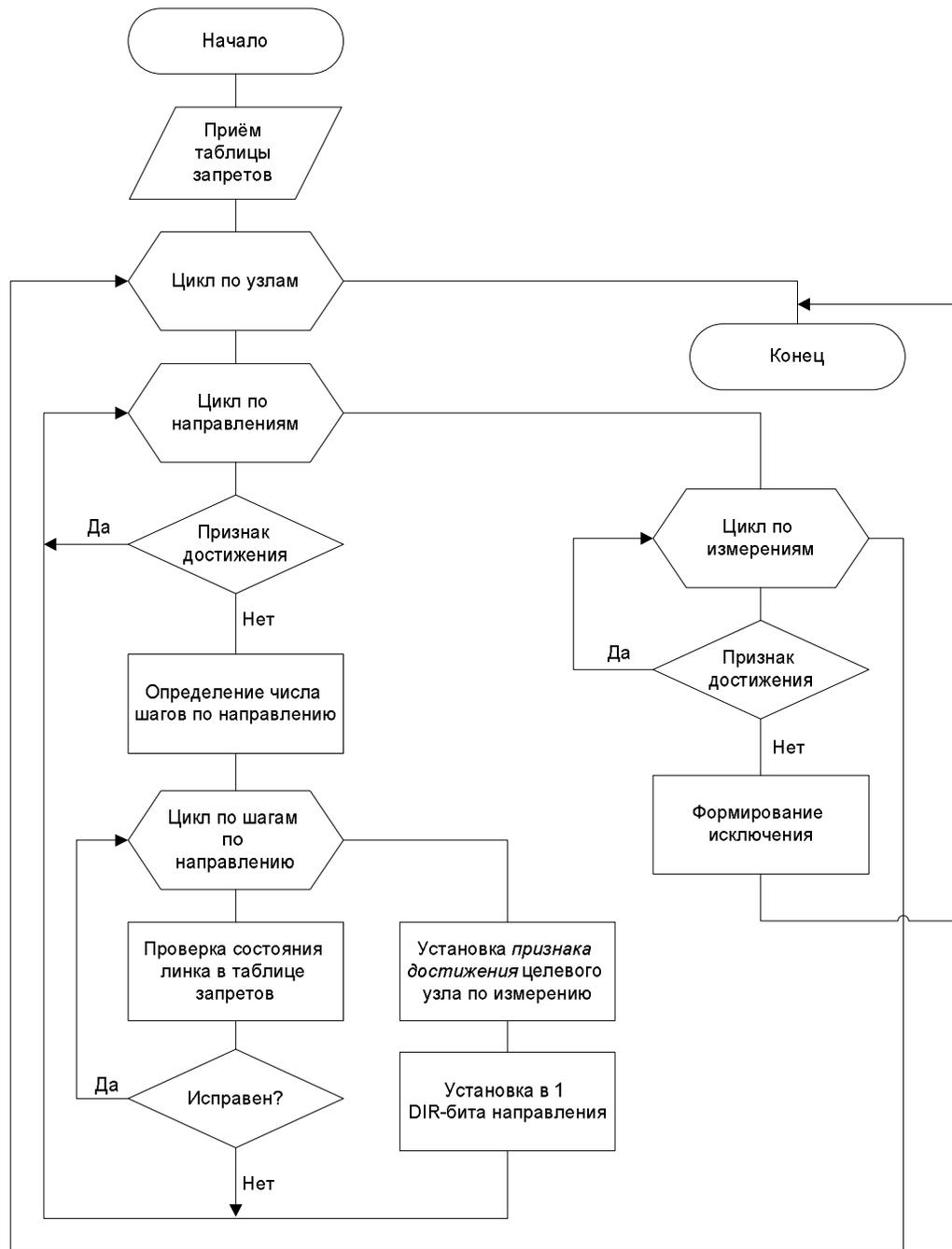


Рис. 6. Обобщённый алгоритм перестроения таблиц маршрутизации

Приведённый обобщённый алгоритм работает следующим образом. После приёма от корневого узла таблицы запретов запускается процесс пересчёта таблицы маршрутизации. В цикле для каждого целевого узла осуществляется проход по каждому направлению в последовательности $+X$, $+Y$, $-X$, $-Y$ (для 2D-тора) с

проверкой состояния линков транзитных узлов по таблице запретов. Если какой-то линк на пути неисправен, осуществляется переход к следующему направлению. Если все линки на пути по направлению исправны – осуществляется установка соответствующего DIR-бита в состояние 1 и установка признака достижения целевого узла по измерению.

По завершению цикла по направлениям осуществляется проверка признаков достижения целевого узла по всем измерениям. Если все признаки установлены, строка таблицы маршрутизации считается заполненной, после чего аналогичным образом обрабатывается следующий узел.

В отличие от приведённого обобщённого алгоритма, предназначенного для отражения общих идей, положенных в основу работы соответствующего программного обеспечения, реальный алгоритм перестроения таблиц маршрутизации значительно сложнее. Он обеспечивает заполнение полей DIR-битов отдельно для адаптивной подсети по маршрутам минимальной длины, установку признака возможной адаптивной передачи с учётом отказов линков на пути следования и т.д.

Возможность перезапуска задач с контрольной точки обеспечивается использованием соответствующих версий библиотек, обеспечивающих необходимую поддержку, например, основанные на BLCR контрольные точки системного уровня, реализованные в библиотеках MPICH, MVARICH, OpenMPI.

Возможность «горячей замены» отказавших элементов (узлов, кабелей связи) обеспечивается с использованием специального алгоритма инициализации,

обеспечивающего автоматическое «знакомство» и назначение физических адресов узлов в структуре многомерного тора.

Данный алгоритм, применительно к коммуникационной сети с топологией 2D-тор, приведён на рис. 7.

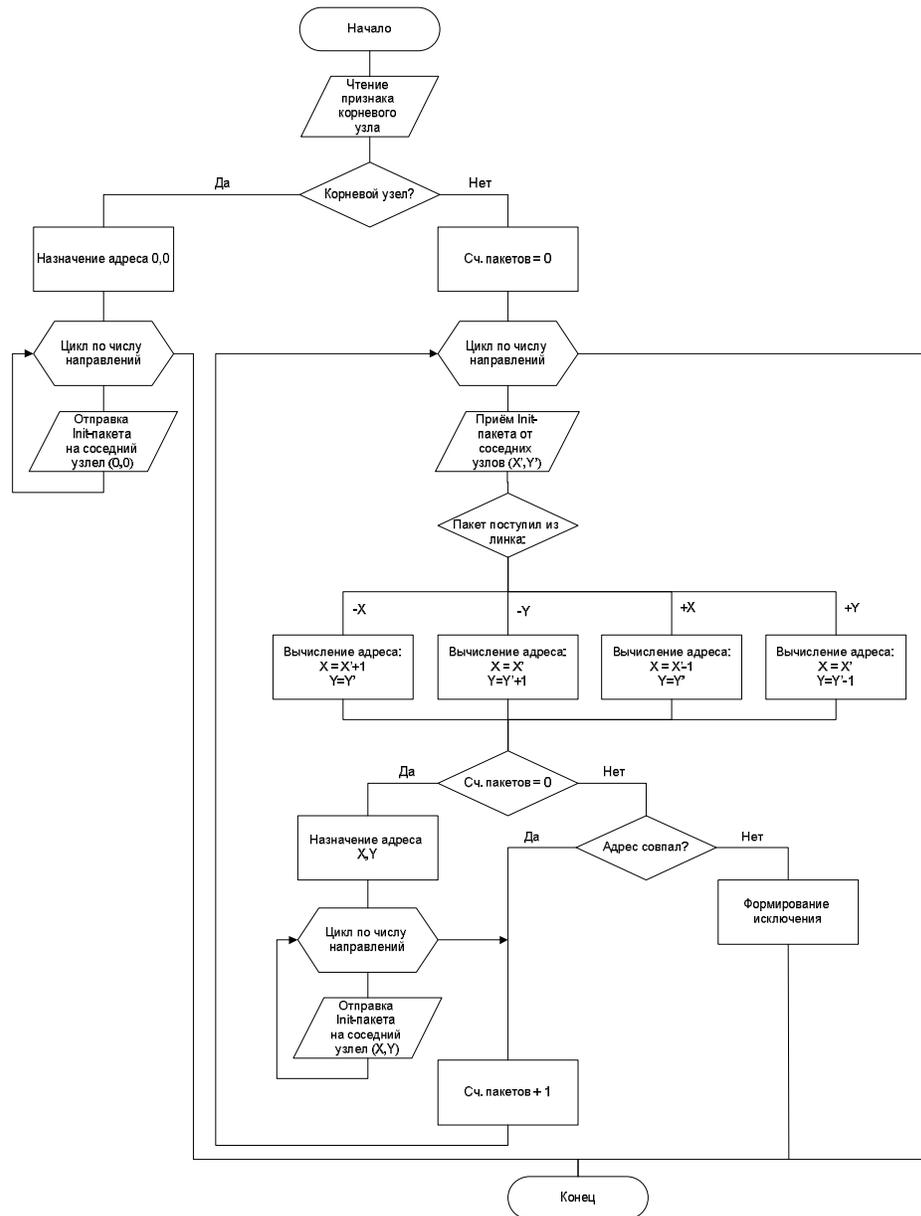


Рис. 7. Алгоритм автоматического назначения физического адреса узла в структуре многомерного тора

Приведённый алгоритм работает следующим образом. После включения электропитания и выполнения представленного ранее алгоритма инициализации связи по всем линкам осуществляется чтение признака корневого узла.

Данный признак задаётся с помощью переключки BootStrap только на маршрутизаторе одного узла МВС, имеющего нулевые координаты. Именно этот узел первым рассылает по всем соседним узлам Init-пакеты, содержащие его координаты.

Другие узлы ожидают поступления первого Init-пакета с любого направления и на основании содержащегося в нём адреса соседнего узла (X' , Y') осуществляют расчёт своего физического адреса (X , Y) и рассылку Init-пакетов по всем соседним узлам. Все остальные Init-пакеты, пришедшие от соседних узлов, используются лишь для контроля правильности назначенного адреса. В случае выявления несоответствия, свидетельствующего об ошибке в схеме соединений коммуникационной сети, формируется исключительная ситуация.

Учитывая, что вероятность одновременного отказа нескольких линий связи на каждом узле пренебрежимо мала, а применение предложенного метода в ВКС с топологией многомерный тор обеспечивает множество путей обхода отказавших линий связи, получаемое значение коэффициента готовности (K_r), в основном определяется наработкой на отказ вычислительных узлов. По результатам оценки эффекта от его применения была получена зависимость K_r МВС от числа узлов, приведённая на рис. 8 ($MTBF_{узел} = 10^6 ч$, $MTBF_{линии\ связи} = 10^5 ч$, $MTTR = 1 ч$).

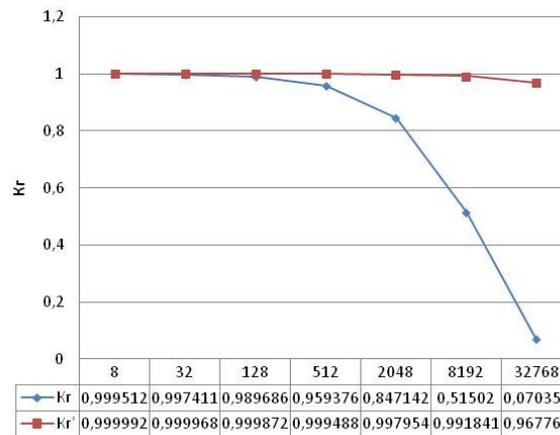


Рис. 8. Зависимость коэффициента готовности МВС от её размера (K_r (синий график) – без использования предложенного метода; K_r' (красный график) – с применением метода обеспечения отказоустойчивости ВКС с топологией многомерный тор)

Таким образом, предложенный метод обеспечения отказоустойчивости ВКС с топологией многомерный тор позволил довести уровень $K_r > 0,99$ даже для МВС большого размера.

Апробация приведённых алгоритмов позволила подтвердить их работоспособность. В настоящее время они реализованы в аппаратуре серийно выпускаемого сетевого оборудования Ангара.

Библиографический список

1. Chao H.J., Liu B. High Performance Switches and Routers, Wiley-IEEE Press, Hoboken, NJ, 2007, 640 p.
2. Симонов А.С., Семенов А.С., Макагон Д.В. Направления развития высокоскоростной коммуникационной сети для многопроцессорных

вычислительных систем // Труды МАИ. 2019. № 108. URL:

<http://trudymai.ru/published.php?ID=109525>. DOI: [10.34759/trd-2019-108-14](https://doi.org/10.34759/trd-2019-108-14)

3. Слущкин А.И., Симонов А.С., Жабин И.А., Макагон Д.В., Сыромятников Е.Л. Разработка межузловой коммуникационной сети ЕС8430 «Ангара» для перспективных российских суперкомпьютеров // Успехи современной радиоэлектроники. 2012. № 1. С. 6 – 10.

4. Жабин И.А., Макагон Д.В., Поляков Д.А., Симонов А.С., Сыромятников Е.Л., Щербак А.Н. Первое поколение высокоскоростной коммуникационной сети «Ангара» // Научноёмкие технологии. 2014. Т. 15. № 1. С. 21 – 27.

5. Жабин И.А., Макагон Д.В., Симонов А.С., Сыромятников Е.Л., Фролов А.С., Щербак А.Н. Кристалл для «Ангары» // Суперкомпьютеры. 2013. № 4 (16). С. 46 – 49.

6. Брехов О.М., Балян А.В. Метод анализа функционирования вычислительных сетей с применением многоуровневого внедрения неисправностей // Труды МАИ. 2016. № 89. URL: <http://trudymai.ru/published.php?ID=73394>

7. Брехов О.М., Балян А.В. Методика внедрения неисправностей для анализа работы протокола резервирования бортового маршрутизатора // Труды МАИ. 2016. № 81. URL: <http://trudymai.ru/published.php?ID=57913>

8. Романов А.М. Обеспечение масштабирования и заданного уровня отказоустойчивости систем управления роботов // Труды МАИ. 2020. № 111. URL: <http://trudymai.ru/published.php?ID=115194>. DOI: [10.34759/trd-2020-111-19](https://doi.org/10.34759/trd-2020-111-19)

9. Budruk R., Anderson D., Shanley T. PCI Express System Architecture, Addison-Wesley Professional, Boston, MA, 2003, 1120 p.
10. Winkles J. Sizing of the Replay Buffer in PCI Express Devices, MindShare, Inc., Cedar Park, TX, 2003, 12 p.
11. Bing Li B., Ding Y., Liu Y. Circuit Design of PCI Express Retry Mechanisms // WSEAS Transactions on Circuits and Systems, 2014, vol. 13, Art. no. 17, pp. 165 - 174.
12. Aguilar M., Veloz A., Gumin M. Proposal of Implementation of the «Data Link Layer» of PCI-Express // 1st International Conference on Electrical and Electronics Engineering, IEEE, 2004, pp. 65 – 69.
13. Слепов Н.Н. Оценка показателей ошибок цифровых линий передачи // Электроника: Наука, Технология, Бизнес. 2002. № 5. С. 22 – 31.
14. Kaddoum G., Chargé P., Roviras D., Fournier-Prunaret D. Analytical calculation of BER in communication systems using a piecewise linear chaotic map // 18th European Conference on Circuit Theory and Design, IEEE, 2007, pp. 691 – 694.
15. Anatomy of an Eye Diagram: Application Note № 11410-0053, Anritsu Company, 2010, 20 p.
16. Dmitriev-Zdorov V., Miller M.T., Ferry C. The Jitter-Noise Duality and Anatomy of an Eye Diagram, UBM Electronics, DesignCon 2014: Where the Chip Meets the Board, 2014, 25 p.
17. Dally W.J., Brian T. Principles and Practices of Interconnection Networks, Morgan Kaufmann, San Francisco, CA, 2004, 550 p.

18. Duato J., Yalamanchili S., Ni L. Interconnection Networks: An Engineering Approach, Morgan Kaufmann, San Francisco, CA, 2003. p. 650.
19. Duato J., Robles A., Silla F., Beivide R. A Comparison of Router Architectures for Virtual Cut-Through and Wormhole Switching in a NOW Environment // Proceedings 13th International Parallel Processing Symposium and 10th Symposium on Parallel and Distributed Processing, IEEE, 1999, pp. 1 – 8.
20. Alnajjar D., Suguiy M. A Comprehensive Guide for CRC Hardware Implementation, 2015. URL:
https://www.researchgate.net/publication/282133684_A_Comprehensive_Guide_for_CRC_Hardware_Implementation.html
21. Lu W., Wong S. A Fast CRC Update Implementation // IEEE Workshop on High Performance Switching and Routing, 2003, pp. 113 – 120.
22. Zanwar C., Patil R.A., Godbole P.D. Routing Algorithms for Interconnection Networks: A Review // 2016 International Conference on Information, Communication and Computing Technology, 2016, pp. 1 – 5.
23. Ferdous J., Salihi R. Routing: Internet Routing Protocols and Algorithms, 2015. URL:
https://www.researchgate.net/publication/281490293_Routing_Internet_Routing_Protocols_and_Algorithms/citation/download.html

Fault tolerance insuring technique for communication network of multidimensional torus topology

Simonov A.S.*, Zhabin I.A.**

*Scientific Research Center of Electronic Computing Technology, NICEVT,
125; Varshavskoye shosse, Moscow, 117587, Russia*

**e-mail: simonov@nicevt.ru*

***e-mail: zhabin@nicevt.ru*

Abstract

The article describes a method for fault tolerance ensuring of a communication network with a multidimensional torus topology, tested while developing the Angara high-speed interconnect router. The high-speed data transmission is associated with a number of difficulties that should be overcome by the hardware designer to achieve acceptable fault tolerance. Problematics of the issue becomes especially acute with an increase in the number of nodes in the system, since the probability of failures is greatly enhanced with expansion in the number of connections.

The work on the fault tolerance ensuring was performed in the following areas:

- quality improving of the transmitted signal by parameters fine-tuning of the high-speed transceivers;
- automated control of the parameters of the signals eye diagram and optimal parameters selection of the transmitting and receiving sides of the data transmission channel;

- developing a fault-tolerant link integrity control algorithm with the ability of the damaged data retransmission;
- developing a data flow control mechanism to control the credit information of the virtual channels;
- developing network layer algorithms for network state monitoring, and quick routing tables rebuilding.

The algorithms and mechanisms described in the article continuity ensuring of the transmitted data flow and their guaranteed delivery over the communication lines between the nodes, exclusion of losses, duplications and distortions of packets during transmission. The conducted measurements have shown that the bit error rate (BER) was about 10⁻¹², which corresponded to the permissible values.

The methods described in the article allow obtaining the availability factor of more than 0.99 for large computing systems. At present, they are implemented in the equipment of the serially produced Angara network equipment, and are also widely applied in the design of the second generation Angara high speed interconnect router.

Keywords: fault tolerance, communication network.

References

1. Chao H.J., Liu B. *High Performance Switches and Routers*, Wiley-IEEE Press, Hoboken, NJ, 2007, 640 p.

2. Simonov A.S., Semenov A.S., Makagon D.V. *Trudy MAI*, 2019, no. 108. URL: <http://trudymai.ru/eng/published.php?ID=109525>. DOI: [10.34759/trd-2019-108-14](https://doi.org/10.34759/trd-2019-108-14)
3. Slutskin A.I., Simonov A.S., Zhabin I.A., Makagon D.V., Syromyatnikov E.L. *Uspekhi sovremennoi radioelektroniki*, 2012, no. 1, pp. 6 – 10.
4. Zhabin I.A., Makagon D.V., Polyakov D.A., Simonov A.S., Syromyatnikov E.L., Shcherbak A.N. *Naukoemkie tekhnologii*, 2014, vol. 15, no. 1, pp. 21 – 27.
5. Zhabin I.A., Makagon D.V., Simonov A.S., Syromyatnikov E.L., Frolov A.S., Shcherbak A.N. *Superkomp'yutery*, 2013, no. 4 (16), pp. 46 – 49.
6. Brekhov O.M., Balyan A.V. *Trudy MAI*, 2016, no. 89. URL: <http://trudymai.ru/eng/published.php?ID=73394>
7. Brekhov O.M., Balyan A.V. *Trudy MAI*, 2016, no. 81. URL: <http://trudymai.ru/eng/published.php?ID=57913>
8. Romanov A.M. *Trudy MAI*, 2020, no. 111. URL: <http://trudymai.ru/eng/published.php?ID=115194>. DOI: [10.34759/trd-2020-111-19](https://doi.org/10.34759/trd-2020-111-19)
9. Budruk R., Anderson D., Shanley T. *PCI Express System Architecture*, Addison-Wesley Professional, Boston, MA, 2003, 1120 p.
10. Winkles J. *Sizing of the Replay Buffer in PCI Express Devices*, MindShare, Inc., Cedar Park, TX, 2003, 12 p.
11. Bing Li B., Ding Y., Liu Y. Circuit Design of PCI Express Retry Mechanisms, *WSEAS Transactions on Circuits and Systems*, 2014, vol. 13, Art. no. 17, pp. 165 – 174

12. Aguilar M., Veloz A., Gumin M. Proposal of Implementation of the «Data Link Layer» of PCI-Express, *1st International Conference on Electrical and Electronics Engineering*, IEEE, 2004, pp. 65 – 69.
13. Slepov N.N. *Elektronika: Nauka, Tekhnologiya, Biznes*, 2002, no. 5, pp. 22 – 31.
14. Kaddoum G., Chargé P., Roviras D., Fournier-Prunaret D. Analytical calculation of BER in communication systems using a piecewise linear chaotic map, *18th European Conference on Circuit Theory and Design*, IEEE, 2007, pp. 691 – 694.
15. *Anatomy of an Eye Diagram: Application Note № 11410-0053*, Anritsu Company, 2010, 20 p.
16. Dmitriev-Zdorov V., Miller M.T., Ferry C. *The Jitter-Noise Duality and Anatomy of an Eye Diagram*, UBM Electronics, DesignCon 2014: Where the Chip Meets the Board, 2014, 25 p.
17. Dally W.J., Brian T. *Principles and Practices of Interconnection Networks*, Morgan Kaufmann, San Francisco, CA, 2004, 550 p.
18. Duato J., Yalamanchili S., Ni L. *Interconnection Networks: An Engineering Approach*, Morgan Kaufmann, San Francisco, CA, 2003. p. 650.
19. Duato J., Robles A., Silla F., Beivide R. A Comparison of Router Architectures for Virtual Cut-Through and Wormhole Switching in a NOW Environment, *Proceedings 13th International Parallel Processing Symposium and 10th Symposium on Parallel and Distributed Processing*, IEEE, 1999, pp. 1 – 8.
20. Alnajjar D., Suguiv M. *A Comprehensive Guide for CRC Hardware Implementation*, 2015.

URL:

https://www.researchgate.net/publication/282133684_A_Comprehensive_Guide_for_CRC_Hardware_Implementation.html

21. Lu W., Wong S. A Fast CRC Update Implementation, *IEEE Workshop on High Performance Switching and Routing*, 2003, pp. 113 – 120.
22. Zanwar C., Patil R.A., Godbole P.D. Routing Algorithms for Interconnection Networks: A Review, *2016 International Conference on Information, Communication and Computing Technology*, 2016, pp. 1 – 5.
23. Ferdaus J., Salihi R. *Routing: Internet Routing Protocols and Algorithms*, 2015. URL: https://www.researchgate.net/publication/281490293_Routing_Internet_Routing_Protocols_and_Algorithms/citation/download.html