Шелеметьева Я. В.

ИССЛЕДОВАНИЕ ТЕХНОЛОГИИ УДАЛЕННОГО ПРЯМОГО ДОСТУПА К ПАМЯТИ В АРХИТЕКТУРАХ ВЫСОКОПРОИЗВОДИТЕЛЬНЫХ СИСТЕМ

Аннотация. *В данной статье описываются основные принципы технологии удаленного прямого доступа к памяти (RemoteDirectMemoryAccess — RDMA, англ.) в архитектурах высокопроизводительных вычислительных систем.Особое внимание уделеноалгоритмам передачи данных в сеть/из сети: рассмотрен стандартный алгоритм передачи данных (на примере сетей, построенных на основе стека протоколов TCP/IP), и алгоритм с использованием технологии RDMAв указанных системах. Исследованы наиболее популярные вычислительные архитектуры, поддерживающие режим удаленного прямого доступа к памяти, показана эффективность применения технологии RDMA в высокопроизводительных сетевых архитектурах на примере сетей 10 GigabitEthernet. С использованием теории множеств, на основе расчетов рабочих тактов процессора при выполнении обработки сетевых команд и данных осуществляется сравнение производительности узлов вычислительных систем, передача данных в которых происходит по стандартным алгоритмам и узлов, в которых реализована технология RDMA. Обоснована целесообразность использования технологии RDMA в вычислительных узлах высокопроизводительных архитектур. В архитектурах современных компьютерных систем вычислительные узлы объединены между собой сетевыми технологиями, и значительный объем данных передается по сети. Одним из методов повышения эффективности работы вычислительных узлов, а значит и сетевой вычислительной системы в целом, является разгрузка их процессоров от сетевых вычислений с помощью режима удаленного прямого доступа к памяти (RDMA).Режим RDMA — это технология, которая позволяет передавать сетевые данные, минуя центральный процессор, напрямую из буферов сетевого адаптера в буферы приложения, тем самым освобождая процессор от обработки сетевых данных В вычислительных архитектурах высокопроизводительных систем объем входящего и исходящего сетевого трафика очень высок, и обработка сетевых данных занимает большое количество процессорного времени, поэтому предложенная технология удаленного прямого доступа к памяти актуальна для разгрузки процессора конкретного вычислительного узла, а значит, и для повышения производительности системы в целом. По сравнению со стандартной передачей данных, в которой процессор вычислительного узла системы полностью обрабатывает сетевые данные, технология RDMA позволяет освободить процессор от сетевой нагрузки и направить всю производительную мощность на решение внутренних задач, тем самым повысив скорость решения задач и надежность системы в целом.*

Ключевые слова: *высокопроизводительные вычислительные архитектуры, удаленный доступ, прямой доступ, сетевые технологии, производительность процессоров, RDMA, сетевой адаптер, TCP/IP, передача данных, GigabitEthernet.*

Введение

В архитектурах современных компьютерных систем вычислительные узлы объединены между собой сетевыми технологиями, и значительный объем данных передается по сети. Одним из методов повышения эффективности работы вычислительных узлов, а значит и сетевой вычислительной системы в целом, является разгрузка их процессоров от сетевых вычислений с помощью режима удаленного прямого доступа к памяти (RDMA).Режим RDMA — это технология, которая позволяет передавать

DOI: 10.7256/2305-6061.2013.3.10773

сетевые данные, минуя центральный процессор, напрямую из буферов сетевого адаптера в буферы приложения, тем самым освобождая процессор от обработки сетевых данных [1,9].

1. Технология прямого удаленного доступа к памяти на примере сетей 10 GigabitEthernet

В сетевых технологиях, построенных на основе стека протоколов TCP/IP и не осуществляющих режим RDMA, если приложение передает данные в сеть, оно отправляет сообщение с этими данными протоколу транспортного уровня TCP. Протокол TCP копирует сообщение в свои буферы, сегментирует сообщение (если это необходимо) на пакеты, добавляет свои заголовки и передает данные протоколу сетевого уровня IP. Протокол IP также копирует данные в свои буферы, добавляет сетевые заголовки и передает данные в буфер сетевого адаптера, где добавляются заголовки канального уровня и сообщение передается в сеть. В узле-приемнике происходят обратные действия. Заголовки пакета “разбираются»: начиная с канальных, заканчивая заголовками приложения. За копирование между буферами протоколов отвечает операционная система и, соответственно, центральный процессор [2].Алгоритм передачи данных в сеть для FastEthernet представлен на рис.1.

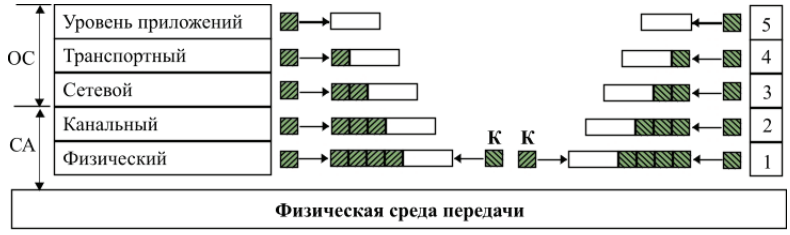

Рис.1. Алгоритм стандартной передачи данных, где ОС — операционная система, СА — сетевой адаптер, К — концевик, добавляемый физическим уровнем

Такой алгоритм передачи данных подходит для сетей, где скорость передачи не больше 1 Гбит/сек. Для сетей, в которых скорость превышает данное значение, центральный процессор не справляется с вычислениями. С увеличением скорости передачи в сети, затраты процессора на межбуферное копирование, сборку/разборку пакетов, проверку контрольной суммы также увеличиваются, и процессор не справляется со всеми задачами, которые на него возложены.

Для решения данной проблемы было предложено перенести вычисления, связанные с добавлением сетевых заголовков, проверкой контрольных сумм и целостности пакетов в сетевые адаптеры, в которых реализована технология RDMA, позволяющаяпередавать данные напрямую в память без участия центрального процессора. В формате пакетов, передаваемых с помощью RDMA технологий, указан буфер памяти приложения, в который должны быть доставлены входящие данные [3,8]. Алгоритм работы протокола RDMA представлен на рис. 1.2.

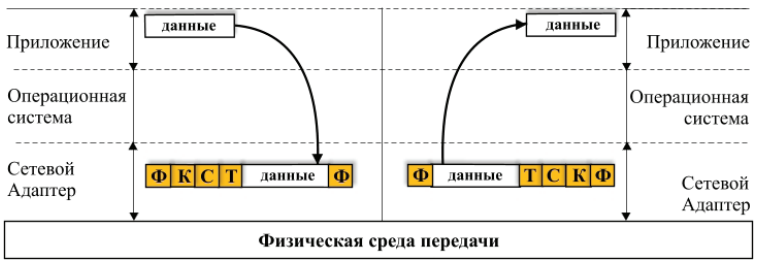

Рис.2. Алгоритм работы протокола RDMA, где Ф — заголовок физического уровня, К — заголовок канального уровня, С — заголовок сетевого уровня, Т — заголовок транспортного уровня

DOI: 10.7256/2305-6061.2013.3.10773

Из рисунков 1 и 2 видно, что в алгоритме передачи данных с технологией RDMA, количество копий данных между буферами операционной системы значительно уменьшилосьпо сравнению со стандартным алгоритмом передачи данных. Данные из буферов приложения (т.е. буферов операционной системы) помещаются сразу в буфер сетевого адаптера, который добавляет все требуемые заголовки. На приемной стороне, соответственно, происходят обратные действия: адаптер, разобрав сетевые заголовки, помещает входящие данные из своих буферов напрямую в буферы приложения [3]. *Таким образом, в алгоритме передачи данных с RDMA технологией становится на два межбуферных копирования меньше: пропадает необходимость в передаче данных между буферами приложения и буферами транспортного уровня и далее — между буферами транспортного и сетевого уровней.*

2. Сравнение производительности вычислительных узлов, в которых осуществляется стандартная передача данных и узлов, в которых реализована технология RDMA

В настоящее время существует ряд архитектур высокопроизводительных систем, в которых реализован режим RDMA: 10 GigabitEthernet, FibreChannel, Myrinet, InfiniBand.Рассмотрим эффективность технологии RDMA на примере 10 GigabitEthernet.

В сетях 10 GigabitEthernet, функционирующих по правилам стека протоколов TCP/IP выделяют 5 уровней сетевых протоколов, которые, добавляя сетевые заголовки, вносят служебную информацию в формат пакета, передаваемого по сети. В алгоритме передачи данных с использованием технологииRDMA, процессор освобождается от обработки заголовков сетевого и транспортного уровней, тогда как в стандартных алгоритмах передачи данных процессором указанные заголовкиобрабатываются. Таким образом, *одним из критериев сравнения алгоритмов стандартной передачи данных и технологии передачи данных с RDMAпри равном объеме данных, переданных по сети/в сеть, являетсяколичество байт сетевых данных, обрабатываемых процессором*. Для сравнения технологий передачи данных (стандартная передача данных и передача с RDMA) воспользуемся теорией множеств [5].

Рабочее множествоS_i — служебные данные (или сетевые заголовки), выражаемые в битах, которые вносят протоколы стека TCP/IP для осуществления передачи данных по сети.

Мощность рабочих множеств $|S_i|$ каждого уровня равна объему ($V_{служ.}$) данных служебной информации, вносимой протоколами. $|S_i| = \{V_{служ.}\} = \{V_{заголовок\ 1} + V_{заголовок\ 2} + V_{заголовок\ n}\}$.

Рабочее множество сетевого уровня IP приведено на рис. 3 [4].

Рабочее множество сетевого уровняS_{ip}— совокупность сетевых заголовков, вносимых протоколом IP для осуществления передачи данных по сети.

Мощность множества протокола сетевого уровня $|S_{ip}|$ равна сумме объемов всех служебных заголовков $V_{служip}$ этого протокола и составляет 160 бит (20 байт): $|S_{ip}| = \{V_{служip}\} =$ 20 байт.

Протокол транспортного уровня TCP вносит в формат сетевого пакета служебную информацию общим объемом $V_{служtcp}$160 бит (20 байт). Рабочее множество транспортного уровня TCP приведено на рис. 4. Мощность множества протокола транспортного уровня $|S_{tcp}|$ равна сумме объемов всех служебных заголовков $V_{служtcp}$ этого протокола и составляет 160 бит (20 байт): $|S_{tcp}| = \{V_{служtcp}\} = 20$ байт.

Мощность служебных множеств протоколов |S|, обрабатываемых операционной системой и задействующих процессор, равна сумме мощностей служебных множеств транспортного $|S_{tcp}|$ и сетевого протоколов $|S_{ip}|$ и составляет 320 бит (40 байт):

$|S| = |S_{ip}| + |S_{tcp}| =$ 40 байт.

В 10 GigabitEthernet системах максимальное количество данных, которое может быть передано по сети составляет 8192 байт, минимальное — 512 байт [6]. Таким образом, в 10 GigabitEthernet системах, поддерживающих технологиюRDMA, процессор освобожден от обработки **40 байт данных**, что составляет 7,8% от общего объема пакета Ethernet минимальной длины.

Для того чтобы оценить значимость разгрузки процессора в вычислительном узле от обработки 40 байт данных, вносимых заголовками транспортного и сетевого уровней схематично опишем

DOI: 10.7256/2305-6061.2013.3.10773

Наименование полей заголовка		Номер версии	Длина заголовка	Тип сервиса	Общая длина	Идентификатор
Объем служебных данных $V_{служ.}$ в битах		4	4	8	16	16
Флаги	Смещение фрагмента	Время жизни	Протокол	КПК	Адрес отправителя	Адрес получателя
3	13	8	8	16	32	32

Рис. 3. Рабочее множество сетевого уровня IP

Наименование полей заголовка		Порт отправителя	Порт получателя	№ в последовательности	№ подтверждения
Объем служебных данных $V_{служ.}$ в битах		16	16	32	32
Смещение данных	Резерв	Контрольные биты	Окно	КПК	Указатель срочности
4	6	6	16	16	16

Рис. 4. Рабочее множество сетевого уровня TCP

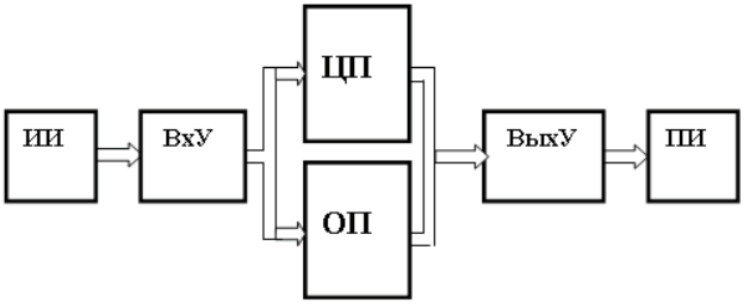

Схема 1. Описание вычислительной системы передачи информации

вычислительную систему передачи информации с точки зрения теории информации (схема 1). На схеме 1 введены обозначения: ИИ и ПИ — источник и приемник информации соответственно; ЦП — центральный процессор; ОП — оперативная память; Вх. У и Вых. У — соответственно входное и выходное устройства, которые осуществляют функции кодирования и декодирования. Данной схемой можно описать алгоритм передачи данных от приложения (ИИ) в сетевой адаптер (ПИ). Входное и выходное устройства в данном случае — драйверы операционной системы и сетевого адаптера соответственно. Три основных функции вычислительной системы передачи информации — это обработка, хранение и ввод-вывод данных, за которые отвечают, соответственно, процессор (или процессоры), память и периферийные устройства. Если освободить процессор от обработки части входящего и исходящего сетевого трафика, можно направить процессорные ресурсы на решение внутренних задач вычислительного узла, тем самым увеличив скорость их решения; снизить задержки сети при интенсивном входящем и исходящим трафике, а так же в целом передать или принять больше сетевых данных от вычислительных устройств системы.

Одной из основных характеристик процессора является разрядность шины данных — количество обработанной информации за такт работы. 64-разрядный процессор за один такт обрабатывает 64

DOI: 10.7256/2305-6061.2013.3.10773

бит (8 байт) данных. Количество тактов (n), которое требуется для обработки сетевого пакета длиной (l) находится из (1), где 8 байт — количество обрабатываемых данных за 1 рабочий такт (разрядность шины данных) [7].

n = l / 8 байт (1)

В алгоритмах передачи данных, поддерживающих технологию RDMA, процессор обрабатывает на 40 байт данных меньше (обработкой заголовков сетевого и транспортного уровня занимается сетевой адаптер), поэтому общая длина сетевого пакета, который обрабатывает процессор, так же уменьшается на 40 байт. Для вычислительных узлов, реализующих технологию RDMA, (1) примет следующий вид:

n = (l — 40 байт) / 8 байт (2)

В табл. 1 представлена зависимость количества тактов 64-разрядного процессора от длины пакета (l) для архитектуры 10 GigabitEthernet, в которой реализована технология RDMA и для архитектуры 10 GigabitEthernet, в которой осуществляется стандартная передача данных.

Длина пакета в байтах	512	1536	2560	3584	4608	5632	6656	7680	8192
Кол-во тактов процессора без RDMA	64	192	320	448	576	704	832	960	1024
Кол-во тактов процессора с RDMA	59	187	315	443	571	699	827	955	1019

Табл. 1 Зависимость количества тактов 64-разрядного процессора от обработки пакетов разной длины

Из таблицы видно, что с увеличением длины пакета растет число тактов процессора, но общее число тактов, в случае если в вычислительном узле реализована технология RDMA, всегда меньше, чем в вычислительном узле, где RDMA не поддерживается.

Таким образом, в вычислительных узлах с 64-х разрядным процессором, в которых применяется технология RDMA, количество тактов, которое требуется для обработки одного входящего сетевого пакета (n_1), уменьшится на 5 единиц:

$n_1 = (V_{служір} + V_{служтср}) /8байт = (20 байт + 20 байт) / 8 байт = 5$ (3)

При интенсивном входящем и исходящем трафике полученное значение является значительным показателем. Приведем пример. Приложению необходимо передать сетевые данные (сообщение) общим объемом (V) 3,9 Мбайт (4 096 000 байт). Для передачи такого объема, процессор сегментирует сообщение на пакеты, объемом (v) 1536 байт каждый (число взято из данных о максимальной и минимальной длине пакета 10 GigabitEthernet). Общее число пакетов (p), на которое разобьётся сообщение, рассчитывается из (4) и равно 2 667 пакетов.

$p = V/v$=4 096 000 байт / 1536байт= 2 667 (4)

Учитывая, что на обработку 1 пакета данных в алгоритме, поддерживающем технологию RDMA, требуется на 5 тактов работы процессора меньше, чем в стандартном алгоритме передачи данных (3), можнорассчитать, насколько меньше тактов (n_o), процессор потратит на обработку сообщения объемом (V) 3,9 Мбайт (4 096 000 байт), при объеме передаваемого пакета (v) 1536 байт (5):

$n_o = V/ v*n_1$=4 096 000 байт / 1536байт*5 = 13 335 (5)

13 335 тактов, “сэкономленных” на обработке сетевых заголовков — значительный показатель для передачи сообщения объемом 3,9 Мбайт в высокопроизводительных сетях. На графике 1показано, сколько рабочих тактов процессора можно “сэкономить” на обработке сетевых пакетов, используя технологию RDMA. Из графика видно, что чем больше пакетов передано по сетив/из вычислительного узла, тем на большее число уменьшится количество рабочих тактов процессора.

DOI: 10.7256/2305-6061.2013.3.10773

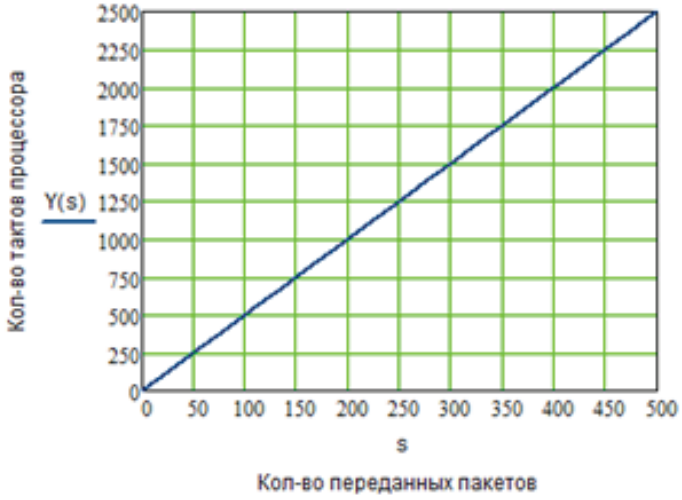

График 1. Зависимость количества сохраненных тактов процессора от числа обработанных пакетов.

Заключение

Критерии сравнения стандартного алгоритма передачи данных и алгоритма с технологией RDMA– количество обрабатываемых процессором сетевых данных, количество тактов процессора, освобожденных от обработки сетевых данных и направленных на решение внутренних задач

Ввычислительных архитектурах высокопроизводительных систем объем входящего и исходящего сетевого трафика очень высок, и обработка сетевых данных занимает большое количество процессорного времени, поэтому предложенная технология удаленного прямого доступа к памяти актуальна для разгрузки процессора конкретного вычислительного узла, а значит, и для повышения производительности системы в целом. По сравнению со стандартной передачей данных, в которой процессор вычислительного узла системы полностью обрабатывает сетевые данные, технология RDMA позволяет освободить процессор от сетевой нагрузки и направить всю производительную мощность на решение внутренних задач, тем самым повысив скорость решения задач и надежность системы в целом.

Библиография

1. Bailey S. The Architecture of Direct Data Placement (DDP) and Remote Direct Memory Access (RDMA) on Internet Protocols. / Bailey S., Talpey T.— Fremont, CA: RFC, 2005.— 23 с.
2. Culley P. Marker PDU Aligned Framing for TCP Specification. / Culley P., Elzur U., Recio R.— Fremont, CA: RFC, 2007.— 74 с.
3. InfiniBandTM Architecture Specification Volume 1 Release 1.2.1 Final Release / подредакциейInfiniBandSM Trade Association — CA.: California USA, 2007.— 1727 с.
4. Recio R.A Remote Direct Memory Access Protocol Specification./Recio R., Metzler B., Culley P.— Fremont, CA: RFC, 2007.— 66 с.
5. Зяблов Н. А. Формализация задачи нахождения кратчайшего пути в информационно-вычислительных сетях. / Н. А. Зяблов, Н. С. Васяева URL: http://ify.ulstu.ru/sites/default/files/Zyablov_N.pdf.
6. Олифер, В. Г. Компьютерные сети. Принципы, технологии, протоколы. / Олифер В. Г., Олифер Н. А.— 3-е изд.— СПб.: Питер, 2010.— 943 с.
7. Портал технической литературы IBMhttp://www.redbooks.ibm.com/
8. Сайт разработчиков RDMAhttp://www.rdmaconsortium.org/.

DOI: 10.7256/2305-6061.2013.3.10773

9. Статья об основах режима RDMA и об основных положениях модели http://www.hpcwire.com/hpcwire/2006–09–15.html.
10. В. В. Голенков, Д. В. Шункевич, И. Т. Давыденко Семантическая технология проектирования интеллектуальных решателей задач на основе агентно-ориентированного подхода // Программные системы и вычислительные методы.— 2013.— 1.— С. 82–94. DOI: 10.7256/2305–6061.2013.01.7.

References

1. Bailey S. The Architecture of Direct Data Placement (DDP) and Remote Direct Memory Access (RDMA) on Internet Protocols. / Bailey S., Talpey T.— Fremont, CA: RFC, 2005.— 23 c.
2. Culley P. Marker PDU Aligned Framing for TCP Specification. / Culley P., Elzur U., Recio R.— Fremont, CA: RFC, 2007.— 74 c.
3. InfiniBandTM Architecture Specification Volume 1 Release 1.2.1 Final Release / podredaktsieiInfiniBandSM Trade Association — CA.: California USA, 2007.— 1727 s.
4. Recio R.A Remote Direct Memory Access Protocol Specification./Recio R., Metzler B., Culley P.— Fremont, CA: RFC, 2007.— 66 c.
5. Zyablov N. A. Formalizatsiya zadachi nakhozhdeniya kratchaishego puti v informatsionno-vychislitel'nykh setyakh. / N. A. Zyablov, N. S. Vasyaeva URL: http://ify.ulstu.ru/sites/default/files/Zyablov_N.pdf.
6. Olifer, V. G. Komp'yuternye seti. Printsipy, tekhnologii, protokoly. / Olifer V. G., Olifer N. A.— 3-e izd.— SPb.: Piter, 2010.— 943 s.
7. Portal tekhnicheskoi literatury IBMhttp://www.redbooks.ibm.com/
8. Sait razrabotchikov RDMAhttp://www.rdmaconsortium.org/.
9. Stat'ya ob osnovakh rezhima RDMA i ob osnovnykh polozheniyakh modeli http://www.hpcwire.com/hpcwire/2006–09–15.html.
10. V. V. Golenkov, D. V. Shunkevich, I. T. Davydenko Cemanticheskaya tekhnologiya proektirovaniya intellektual'nykh reshatelei zadach na osnove agentno-orientirovannogo podkhoda // Programmnye sistemy i vychislitel'nye metody.— 2013.— 1.— C. 82–94. DOI: 10.7256/2305–6061.2013.01.7.

DOI: 10.7256/2305-6061.2013.3.10773