

17 вопрос

Последовательные интерфейсы AMD HyperTransport, Intel QuickPath Interconnect и RapidIO.

HyperTransport

Определение

HyperTransport (ранее известная как Lightning Data Transport (LDT)) — двунаправленная последовательно-параллельная компьютерная шина с высокой пропускной способностью и малыми задержками. Для разработки и продвижения данной шины был образован консорциум HyperTransport Technology. Технология используется:

компаниями AMD и Transmeta в x86-процессорах;

PMC-Sierra, Broadcom и Raza Microelectronics — в процессорах AMD, MIPS;

nVidia, VIA, SiS, ULi/ALi, Apple Computer и HP — в наборах системной логики для ПК;

HP, Sun Microsystems, IBM и iWill — в серверах;

Cray, Newisys и PathScale — в суперкомпьютерах;

компанией Cisco — в маршрутизаторах.

Рассмотрим шину

Высокоскоростная шина ввода-вывода HyperTransport (HT) предназначена для использования в компьютерных системах, прежде всего в качестве внутренней локальной шины. В сравнении с шиной PCI интерфейс HyperTransport позволяет снизить число проводников на системной плате, устранить задержки, связанные с монополизацией шины устройствами с низкой производительностью, уменьшить энергопотребление и повысить пропускную способность.

Шина HyperTransport организована на различных уровнях:

- на физическом уровне шина представлена линиями данных, управления, тактовыми, а также контроллерами и стандартными электрическими сигналами;
- на уровне передачи данных определяется порядок инициализации и конфигурирования устройств, установления и прекращения сеанса связи, циклического контроля адекватности данных, выделения пакетов для передачи данных;
- на уровне протокола определены команды выделения виртуальных каналов связи, правила управления потоком данных;

- на уровне транзакций команды протокола конкретизированы в управляющие сигналы, например чтения или записи;
- на уровне сессии определены правила управления энергопотреблением и прочие команды общего характера.

Физические устройства в рамках интерфейса HyperTransport подразделяются на несколько типов:

- save («пещера») - оконечное устройство на двунаправленном канале связи;
- tunnel («туннель») - устройство на двунаправленном канале связи, установленное «на проходе» (но не мост);
- bridge («мост») - устройство на двунаправленных каналах связи, один из которых считается главным и связывает устройство с контроллером шины (Host), а другие соединяют с прочими устройствами.

В минимальной конфигурации (ширина канала 2 бит, на каждый бит требуется две физические линии) потребуется 24 контакта (8 для данных + 4 для тактовых сигналов + 4 для линий управления + 2 сигнальных + 4 заземления + 1 питания + 1 сброса), в максимальной конфигурации (ширина канала 32 бит) речь идет уже о 197 выводах. Для сравнения укажем, что спецификация PCI 2.1 предусматривает 84 вывода, а PCI-X - 150 выводов.

Физически технология HyperTransport основывается на улучшенной версии низковольтных дифференциальных сигналов. Для всех линий (данных, управления, тактовых) используются шины с дифференциальным сопротивлением 100 Ом. Уровень сигнала составляет 1,2 В (в отличие от 2,5 В, установленных спецификацией IEEE LVDS). Благодаря этому длина шины может достигать 24 дюйма (около 61 см) при полосе пропускания на одной линии до 800 Мбит/с. Спецификация HyperTransport предусматривает разделение «восходящих» (Upstream) и «нисходящих» (Downstream) потоков данных (асинхронность). Пакет, объединяющий адреса, команды и данные, всегда кратен 32 бит. Поэтому обеспечивается его безошибочная передача по масштабируемым каналам шириной от 2 до 32 бит. Пиковая пропускная способность соединения HyperTransport достигает 12,8 Гбайт/с (по 6,4 Гбайт/с на нисходящий и восходящий каналы шириной 32 бит при частоте 800 МГц и передаче данных по фронту и спаду сигнала). Для сравнения укажем, что пиковая пропускная способность системной шины (200 МГц) процессора AMD Athlon составляет 2,128 Гбайт/с. Особенностью технологии HyperTransport является совместимость с устройствами PCI на уровне протоколов.

Intel QuickPath Interconnect

Intel QuickPath Interconnect (QuickPath, сокр. QPI, ранее Common System Interface, CSI) — последовательная кэш-когерентная шина типа точка-точка разработанная фирмой Intel для соединения процессоров в многопроцессорных системах и для передачи данных между процессором и чипсетом. QPI создавалась в ответ на разработанную ранее консорциумом во главе с фирмой AMD шину HyperTransport.

Шина QuickPath была создана для замены применявшейся ранее шины Front Side Bus, которая осуществляла связь между центральным процессором и северным мостом материнской платы. Первые процессоры с интерфейсом QuickPath были выпущены на рынок в 2008 году. По состоянию на начало 2010 года, внешний интерфейс QuickPath используется только в сериях процессоров Xeon и Core i7 с ядром Nehalem для разъема LGA 1366, а также будет использоваться в следующем поколении Itanium (ядро Tukwila). При этом чипсеты для разъема LGA 1366 используют шину DMI для связи между северным и южным мостом. Процессоры для разъема LGA 1156 не имеют внешнего интерфейса QuickPath, поскольку чипсеты для данного разъема поддерживают только однопроцессорную конфигурацию, а функциональность северного моста встроена в сам процессор (и, следовательно, для связи процессора с аналогом южного моста используется шина DMI). Однако внутри процессора LGA 1156 связь между ядрами и встроенным контроллером PCIe осуществляется через встроенную шину QuickPath.

Каждое соединение шины QuickPath состоит из пары односторонних каналов, каждый из которых физически реализован как 20 дифференциальных пар проводов. Данные передаются в виде пакетов (дейтаграмм). Пропускная способность одного канала составляет от 4,8 до 6,4 GT/s (гигатранзакций в секунду). Одна передача содержит 16 бит полезной нагрузки, следовательно, теоретическая суммарная пропускная способность одного соединения (в двух направлениях) — от 19,2 до 25,6 гигабайт в секунду (то есть от 9,6 до 12,8 гигабайт/с в каждую сторону); при этом один процессор может иметь несколько соединений.

<https://www.intel.com/content/www/us/en/io/quickpath-technology/quick-path-interconnect-introduction-paper.html>

RapidIO

RapidIO — это высокопроизводительный интерфейс передачи данных для соединения микросхем в рамках одной печатной платы, а также для соединения между собой нескольких печатных плат. Данный интерфейс был разработан для применения во встраиваемых системах.

Основными конкурентами RapidIO являются HyperTransport, Infiniband и PCI Express, которые, однако, предназначены для решения других задач.

Интерфейс RapidIO разработан компаниями Mercury Computer Systems и Motorola (ныне Freescale Semiconductor) как развитие шины, применявшейся в многопроцессорных системах цифровой обработки сигналов компании Mercury.

Спецификации интерфейса RapidIO разработаны организацией «RapidIO Trade Association». На настоящий момент последней версией спецификации является 4.

Обзор протокола

Протокол RapidIO определён трёхуровневой спецификацией:

Физический уровень: электрическая спецификация, PCS/PMA, протокол канального уровня для надёжного обмена пакетами

Транспортный уровень: маршрутизация, многоадресная рассылка, и программная модель

Логический уровень: логический ввод/вывод, система обмена сообщениями, глобальная разделяемая память (CC-NUMA), управление потоком, потоковая передача данных.

Системные спецификации включают:

Инициализацию системы

Работу с ошибками / Горячую замену

Физический уровень

Электрическая спецификация RapidIO основана на промышленном стандарте Ethernet и стандартах Optical Interconnect Forum:

XAUI для шин со скоростью 1,25, 2,5 и 3,125 ГБд (1, 2 и 2,5 Гбит/с)

OIF CEI 6+ Гбит/с для шин со скоростью 5,0 и 6,25 ГБд (4 и 5 Гбит/с)

10GBASE-KR 802.3-ар ("длинный") и 802.3-ba ("короткий") для шин со скоростью 10,3125 ГБд (9,85 Гбит/с)

Уровень PCS/PMA RapidIO поддерживает две формы кодирования/разбивания по кадрам:

8b/10b для шин со скоростью до 6,25 ГБд

64b/67b, сродни тому, что используется в Interlaken, для шин со скоростью свыше 6,25 ГБд

Каждый элемент RapidIO, участвующий в работе, отправляет и получает три вида данных: пакеты, управляющие символы и последовательность простоя.

Пакеты

Каждый пакет содержит в себе два значения, которые управляют обменом этим пакетом на физическом уровне. Первый из них — это acknowledge ID (ackID), уникальное

в пределах канала 5-и, 6-и или 12-битное число, которое используется для отслеживания пакетов, передающихся в канале. Пакеты передаются с последовательно возрастающими значениями askID. Из-за того, что askID зависит от канала, он хранится не в контрольной сумме, а на уровне протокола. Это позволяет менять askID с каждым каналом, через который проходит пакет, в то время как контрольная сумма пакета остается постоянной от конца до конца и служит для проверки целостности пакета. Когда пакет успешно получен, его получение подтверждается согласно askID. Передатчик обязан сохранять пакет до тех пор, пока его успешное получение не было подтверждено на другом конце.

Вторым значением является физический приоритет пакета. Физический приоритет состоит из бита-идентификатора Виртуального Канала (VC), битов приоритета и бита, означающего что пакет критически важен для потока запросов (CRF). Бит VC определяет, используются ли остальные биты для нумерации виртуального канала от 1 до 8, или же для определения приоритета внутри нулевого виртуального канала. Виртуальным каналам назначаются соответствующие минимальные гарантированные пропускные способности. Внутри виртуального канала 0, пакеты с высоким приоритетом, могут отправляться раньше пакетов с низким. Пакеты ответов должны обладать большим физическим приоритетом, чем пакеты запросов, во избежание взаимоблокировки.

Физический уровень RapidIO добавляет к пакетам двухбайтовый заголовок в начале каждого пакета, включающий в себя askID и физический приоритет, а также двухбайтовую контрольную сумму в конец, для проверки целостности пакета. Пакеты, большие 80 байт так же имеют промежуточную контрольную сумму после первых 80 байт. За одним исключением, контрольная сумма пакета служит для проверки целостности пакета на концах соединения.

Управляющие символы

Управляющие символы RapidIO могут отправляться в любой момент, даже внутри пакета. Это позволяет RapidIO иметь наименьшую возможную задержку управления, и достигать высокой пропускной способности с меньшими буферами, нежели у других протоколов.

Управляющие символы используются для отделения пакетов друг от друга (Start of Packet, End of Packet, Stomp), подтверждения получения пакетов (Packet Acknowledge, Packet Not Acknowledged), сброса (Reset Device, Reset Port) и для распространения событий внутри системы RapidIO (Multicast Event Control Symbol). Управляющие символы так же используются для управления потоком (Retry, Buffer Status, Virtual Output Queue Backpressure) и восстановления ошибок.

Процедура восстановления ошибок очень быстра. Как только получатель замечает ошибку передачи в полученном потоке данных, он заставляет связанный с ним передатчик отправить управляющий символ Packet Not Accepted. Как только на другом конце получают управляющий символ Packet Not Accepted, передача новых пакетов останавливается и отправляется управляющий символ Link Request/Port Status. askID, который надо использовать при следующей передаче указывается управляющим символом Link Response. Затем передача возобновляется.

Последовательность простая

Последовательность простая используется во время инициализации канала для оптимизации качества сигнала. Она так же передается, когда нет никаких других отправляемых пакетов или управляющих символов.

Транспортный уровень

Каждый конечный узел в RapidIO определяется уникальным идентификатором устройства (deviceID). Каждый пакет RapidIO содержит идентификаторы двух устройств. Первый из них — идентификатор пункта назначения (destID), который показывает, куда должен быть направлен пакет. Второй — идентификатор пункта отправления (srcID), который указывает, откуда пакет произошел. Когда конечный узел получает пакет RapidIO с запросом, который требует ответа, пакет ответа составляется переменной мест srcID и destID.

Свитчи RapidIO используют destID полученных пакетов для определения выходного порта или портов, которые должны быть использованы для дальнейшей передачи пакета. Обычно, destID используется в качестве индекса для массива контрольных значений. Операция индексирования быстра и выполняется низкими средствами. Свитчи RapidIO для маршрутной таблицы поддерживают стандартную программную модель, что упрощает управление системой.

Транспортный уровень RapidIO поддерживает любую топологию сети, от простых дерева и ячеистой сети, до n-размерных гиперкубов, многомерных тороидов и менее известные структуры вроде узловых сетей.

Транспортный уровень RapidIO позволяет использовать виртуализацию (например, конечный узел RapidIO может поддерживать множество deviceID). Части идентификатора пункта назначения каждого пакета могут использоваться для определения конкретных виртуальных устройств внутри узла.

Логический уровень

Логический уровень RapidIO состоит из нескольких спецификаций, каждая из которых предоставляет форматы пакетов и протоколы для передач с разной семантикой.

Логический ввод/вывод

Уровень логического ввода/вывода определяет форматы пакетов для чтения, записи, записи с ответом и различных атомарных операций. Примеры атомарных операций: установить, очистить, увеличить, уменьшить, поменять местами, проверить и поменять местами и сравнить и поменять местами.

Передача сообщений

Спецификация передачи сообщений определяет Дверные Звонки и Сообщения. Дверные Звонки передают 16-битный код события. Сообщения передают вплоть до 4 килобайт данных, разбитых на 16 пакетов, каждый из которых может нести максимум 256 байт. Пакеты ответа должны отправляться для каждого запроса Дверного Звонка или

Сообщения. Значение статуса в ответном пакете означает выполнение, ошибку или повтор. Статус ответа "повтор" запрашивает отправителя запроса послать пакет еще раз. Ответ "повтор" на логическом уровне позволяет множеству отправителей обращаться к малому количеству разделяемых получателей, что приводит к большой пропускной способности при малой мощности.

Управление потоком

Спецификация Управления Потоком определяет форматы пакетов и протоколы для простых операций управления потоком готов/занят. Пакеты управления потоком могут происходить с концов соединения и из свитчей. Получение пакета управления потоком "занят" останавливает передачу потока или потоков, пока не будет получен пакет управления потоком "готов", или произойдет тайм-аут. Пакеты управления потоком так же могут использоваться как общий механизм для управления ресурсами системы.

CC-NUMA

Спецификация Глобальной Разделяемой Памяти определяет форматы пакетов и протоколы для работы с кешем связной разделяемой памяти в сети RapidIO

Потоковая передача данных

Спецификация Поточковой Передачи Данных поддерживает передачу сообщений с форматами пакетов и семантикой, отличными от спецификации Передачи Сообщений. Формат пакета Поточковой Передачи поддерживает передачу до 64 килобайт данных, разделённых по множеству пакетов. Каждая передача связана с Классом Услуг и Идентификатором Потока, что делает возможными тысячи уникальных потоков между конечными узлами.

Спецификация Поточковой Передачи Данных также определяет форматы пакетов и семантики для управления потоком с Расширенным Заголовком, для управления производительностью внутри системы клиент-сервер. Каждый клиент использует пакеты управления потоком с расширенным заголовком для информирования сервера о величине работы, которая будет отослана серверу. Сервер отвечает пакетами управления потоком с расширенными заголовками, которые используют протоколы основанные на готовности/занятости, рейтинге или кредите для управления тем как быстро и в каком объеме клиенты отошлют серверу работу.

Инициализация системы

Системы с известной топологией могут быть инициализированы соответствующем системе методом без влияния на совместимость. Спецификация инициализации системы RapidIO поддерживает инициализацию систем, в которых топология неизвестна или динамична. Алгоритмы инициализации системы поддерживают наличие запасных хостов, для того, чтобы инициализации могла не иметь единой точки отказа.

Каждый хост системы рекурсивно проходит по всей сети RapidIO, захватывая владение устройствами, выделяя идентификаторы устройств конечным узлам и обновляя таблицы

маршрутизации на свитчах. При происхождении конфликта владения, побеждает хост с наивысшим deviceID. Проигравший хост освобождает устройства и отступает, ожидая победивший хост. Победивший хост продолжает проход, включая завладение проигравшим хостом. Затем, проигравший хост исследует систему путем чтения таблиц маршрутизации свитчей и регистрирует на каждом конечном узле для того, чтобы узнать конфигурации системы. Если победивший хост не завершит обход за известный период времени, проигравший хост считает, что выигравший хост потерпел неудачу и сам завершает обход.

Обход системы поддерживается в Linux подсистемой RapidIO.

Обработка ошибок

RapidIO разработан как устойчивая к ошибкам система с высокой доступностью, поддерживающая горячую замену. Определены условия для ошибок, требующих обнаружения и стандартные регистры для передачи статуса и информации об ошибке. Так же определен настраиваемый механизм изоляции, так что если обмен пакетами на канале невозможен, пакеты могут быть отброшены, во избежание заторов и обеспечения возможности провести диагностику и восстановительные меры. Определены внутрисполосные (запись пакета на порт) и внеполосные (прерывания) механизмы уведомлений.

16 вопрос

Системная шина как способ общения между центральными процессорами, основной памятью и другими компонентами узла вычислительной системы.

Системная шина- это основная интерфейсная система ПК, обеспечивающая сопряжение и связь всех его устройств между собой.

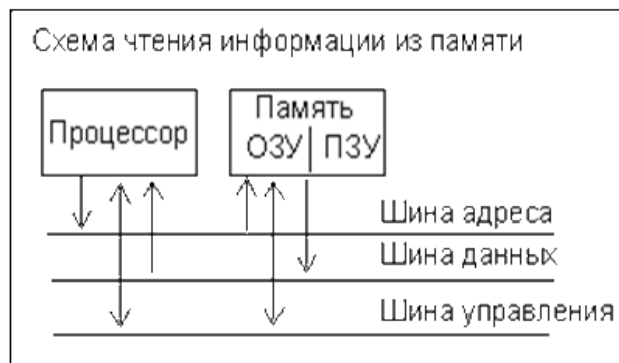
Основной функцией системной шины является передача информации между процессором и остальными устройствами ЭВМ. Все блоки, а точнее их порты ввода-вывода, через соответствующие разъемы подключаются к шине единообразно: непосредственно или через контроллеры (адаптеры).

Управление системной шиной осуществляется непосредственно, либо, чаще через контроллер шины. Обмен информацией между ВУ и системной шиной выполняется с использованием ASCII-кодов. Системная шина состоит из трех шин: шины управления, шины данных и адресной шины. По этим шинам циркулируют управляющие сигналы, данные (числа, символы), адреса ячеек памяти и номера устройств ввода-вывода. Важнейшими функциональными характеристиками системной шины являются: количество обслуживаемых ею устройств и ее пропускная способность, т.е. максимально возможная скорость передачи информации. Пропускная способность шины зависит от ее разрядности (есть шины 8-, 16-, 32- и 64-разрядные) и тактовой частоты, на которой шина работает.

Адресная шина. У процессоров Intel Pentium (а именно они наиболее распространены в персональных компьютерах) адресная шина 32-разрядная, то есть состоит из 32 параллельных линий. В зависимости от того, есть напряжение на какой-то из линий или нет, говорят, что на этой линии выставлена единица или ноль. Комбинация из 32 нулей и единиц образует 32-разрядный адрес, указывающий на одну из ячеек оперативной памяти. К ней и подключается процессор для копирования данных из ячейки в один из своих регистров.

Шина данных. По этой шине происходит копирование данных из оперативной памяти в регистры процессора и обратно. В компьютерах, собранных на базе процессоров Intel Pentium, шина данных 64-разрядная, то есть состоит из 64 линий, по которым за один раз на обработку поступают сразу 8 байтов.

Шина команд. Для того чтобы процессор мог обрабатывать данные, ему нужны команды. Он должен знать, что следует сделать с теми байтами, которые хранятся в его регистрах. Эти команды поступают в процессор тоже из оперативной памяти, из тех областей, где хранятся программы. Команды тоже представлены в виде байтов. Самые простые команды укладываются в один байт, однако, есть и такие, для которых нужно два, три и более байтов. В большинстве современных процессоров шина команд 32-разрядная (например, в процессоре Intel Pentium), хотя существуют 64-разрядные процессоры и даже 128-разрядные.



<https://lvk.cs.msu.su/~dimawolf/ArchEVM/Lecture04.pdf>