

Конструкции шин однокристальных ARM-микроконтроллеров для интегрированных широкополосных систем

Александр КРУТИКОВ
a.krutikov@yeint.spb.ru

Современный рынок информационных технологий ознаменован быстрым развитием встраиваемых систем, предназначенных для решения задач управления в реальном времени. Однако во многих новых устройствах со встроенной сетевой организацией по передаче информации традиционная шинно-ориентированная архитектура просто не в состоянии управиться с арбитражем шины при широкополосной передаче больших массивов данных. Решением данной задачи стало применение «усовершенствованной высокопроизводительной шины» (Advanced High-performance Bus — AHB).

Высокопроизводительные коммуникационные системы, выполненные на базе 32-разрядных микропроцессоров ARM, широко применяются в современных вычислительных устройствах, связанных с передачей, сбором или отображением самой различной информации. Сегодня трудно найти область, в которой они не нашли бы применения — от систем промышленной автоматизации и медицинских мониторов до игровых комплексов в казино. Однако при применении 32-разрядных встраиваемых ARM решений все настойчивее возникает потребность в управлении в реальном времени широкополосной передачей потоков данных по сетям. Для однокристальных систем особенно важно передавать данные и управлять информационными потоками однозначным и определенным образом. Кроме того, важно, чтобы работа системы оставалась под полным программным управлением, задаваемым системным разработчиком, что зачастую является необходимым требованием для однозначного и надежного управления потоками. К сожалению, данное требование не всегда возможно выполнить в случае однокристальных систем, основанных на шинной архитектуре, поскольку в данных системах программист, как правило, не имеет доступа к распределению ресурсов системной шины, и, следовательно, требование полного программного контроля не выполняется.

Разработчики и изготовители чипов часто для снижения времени и стоимости разработки однокристальной системы заимствуют технические решения от архитектур уровня платы или системы. Поскольку пользовательское оборудование, такое как принтеры или карманные компьютеры, редко требует

немедленной реакции в реальном времени, традиционные решения для таких применений работают очень хорошо.

Однако для многих новых устройств со встроенной сетевой организацией по передаче информации традиционная шинно-ориентированная архитектура просто не в состоянии управиться с арбитражем шины при широкополосной передаче больших массивов данных. Это особенно справедливо для таких применений, как дисплеи человеко-машинных интерфейсов (HMI), кассовые терминалы, цветные принтеры и копиры, проекторы с сетевым подключением, видеокамеры охранных систем и т. п. Устройства, в реальном времени сочетающие обработку, отображение и сетевую передачу информации, нуждаются в непрерывном управлении пропускной способностью шины.

Преимущества и недостатки однокристальных систем с традиционной шиной

Разработчики однокристальных систем неохотно отказываются от архитектуры разделяемой шины, заимствованной из мира одноплатных систем, поскольку применение подобной архитектуры позволяет снизить технические требования и усилия на верификацию в цикле разработки и делает однокристальные системы самого высокого уровня интеграции почти столь же простыми, как карты расширения в системах с материнской платой. Использование шин общей архитектуры позволяет разработчикам сосредоточиться на проблемах более высокого уровня.

Использование компаний ARM Limited шины общей архитектуры в «усовершенст-

вованной шинной архитектуре микроконтроллера» (Advanced Microcontroller Bus Architecture — AMBA) позволило лицензиатам сосредоточиться на имеющемся приложении и быстро выйти на рынок.

Микропроцессоры, контроллеры прямого доступа к памяти, контроллеры памяти и другие блоки соединяются через «усовершенствованную высокопроизводительную шину» (Advanced High-performance Bus — AHB). Узлы меньшей производительности — универсальные асинхронные приемопередатчики (UART), платы ввода-вывода общего назначения (GPIO) и таймеры — очень хорошо подходят для соединения через AHB.

Однако для многих встроенных приложений, на которые нацелены основанные на ARM однокристальные системы, требуется не только непрерывная работа с приложениями в реальном времени, но и высокоскоростной доступ к сетевому окружению.

Для таких приложений требуется однокристальная система, которая формирует сигналы управления, собирает и передает данные по сети в реальном времени. В зависимости от природы сети и требований к пропускной способности, они могут исчерпать возможности существующей шинной архитектуры однокристальных систем до предела.

Например, встроенное высокопроизводительное приложение с соединением по сети может обрабатывать видеопоток от камеры или передавать графическую информацию от сервера на принтер через Ethernet-соединение и в то же время обновлять изображение на местном дисплее с жесткими требованиями по времени сканирования и обновления. При работе с внешним ЖК-монитором контроллер должен знать точное число байт,

которое надо передать по шине, порядок, в котором информация пересылается по шине, и определенное временное окно, в которое следует передать данные на дисплей, причем в определенном порядке. Необходимо также постоянно передавать на ЖК-монитор информацию об изменении изображения.

Концепция разделяемой шины для удовлетворения таких требований к однокристальным системам недостаточна. В типовой АНВ-системе все первичные ресурсы — «bus masters», что означает, что, когда шина свободна, они могут захватить ее на время, необходимое для завершения задачи. Но в однокристальных системах на основе ARM у программиста нет возможности напрямую контролировать долю ресурсов шины, которую они могут захватить, когда управление переходит к ним.

Существует несколько способов распределения приоритетов этих операций на шине с разделяемой архитектурой: арбитраж в гирляндной цепи, центральный параллельный арбитраж, распределенный арбитраж путем самовыбора или обнаружения столкновений и арбитраж шины с множественными запросами к шине. Однако при захвате шины очередным мастером шины все другие операции откладываются. Не суще-

ствует механизма, через который несколько устройств могут получить к шине доступ в той степени, в какой это нужно, чтобы удовлетворить требования приложения без ограничения возможности других важных операций формировать определенный клик в реальном времени.

Одна общая технология, применяемая в AMBA-окружении для обработки таких ситуаций, — это использование каналов арбитража. Если имеется шесть устройств bus master, то в шину закладывается шесть каналов арбитража. Но вместо того, чтобы присваивать каждый канал конкретному устройству, расположенная на кристалле логика арбитража распределяет каналы на основе числа мастер-устройств, запрашивающих доступ к шине. Если шину запрашивают четыре мастер-устройства, шесть каналов распределяется между ними, гарантируя, что каждому мастеру будет предоставлен равный доступ к шине.

Это, однако, не решает основной проблемы — как предоставить достаточную полосу шины для выполнения определенной задачи. Если одной операции нужны три канала, а другим операциям в совокупности — только два, каждой будет выделено равное количество из имеющегося пространства каналов. В результате несколько каналов используют-

ся с неполной загрузкой, другие вовсе не используются, а некоторые будут перегружены, влияя на способность однокристальной системы реагировать на события определенным образом и с достаточно низкой задержкой.

Решение: система программного управления пропускной способностью шины

Для решения указанных выше задач необходима схема программного распределения полосы пропускания, которая предоставляет одному мастер-устройству ресурсы шины, которые ему требуются в данный момент, а оставшуюся часть пространства шины отдает другим мастер-устройствам, которым тоже может потребоваться доступ к шине. И поскольку со временем ситуация может измениться, необходим некий механизм, регулярно перераспределяющий ресурсы шины.

Среди различных решений, существующих на сегодня, наиболее интересны предлагаемые компанией NetSilicon (www.netsilicon.com). Компания существует с 1985 года (с 1989 входит в мировую корпорацию Digi International — www.digi.com) и является одним из мировых лидеров в области поставки ARM-решений, с годовым оборотом свыше \$100 млн в год. Микропроцессоры ARM компании NetSilicon используются в продукции таких крупных компаний, как Siemens, Wago, Mitsubishi Electric, National Instruments, Omron, Philips, Schneider Electric, Hirschmann, Phoenix Contact и еще более 750 компаний.

NetSilicon разработала новую систему управления полосой пропускания взамен той, что используется в AMBA-архитектуре. Система основана на использовании 16-слотового арбитра шин со скользящим приоритетом (рис. 1) с набором алгоритмов замены содержимого кэша через программируемые псевдослучайные (или вращающиеся) приоритеты. Например, в случае системы NS9750 производства NetSilicon (рис. 2) шесть устройств bus master вместо того, чтобы подключаться для распределения шести каналов АНВ, разделяют доступ к 16-слотовой схеме распределения шины. Через назначенные регистры в модуле управления системой разработчику системы становятся доступны три способа распределения ресурсов в однокристальной системе.

На самом верхнем уровне всякий раз, когда мастер-устройство посылает запрос на доступ к шине, оно получает ответ в порядке поступления запросов до тех пор, пока не будут задействованы все шесть мастер-устройств. Затем, в зависимости от необходимой полосы пропускания, каждому мастер-устройству выделяется определенное число слотов, и оно получает исключительный доступ к этим слотам. Например, если четыре слота в NS9750 выделены центральному процессору, четыре слота — контроллеру сети Ethernet, четыре слота — мосту BBUS, три слота — ЖК-монитору и три слота — для PCI/Cardbus, такое рас-

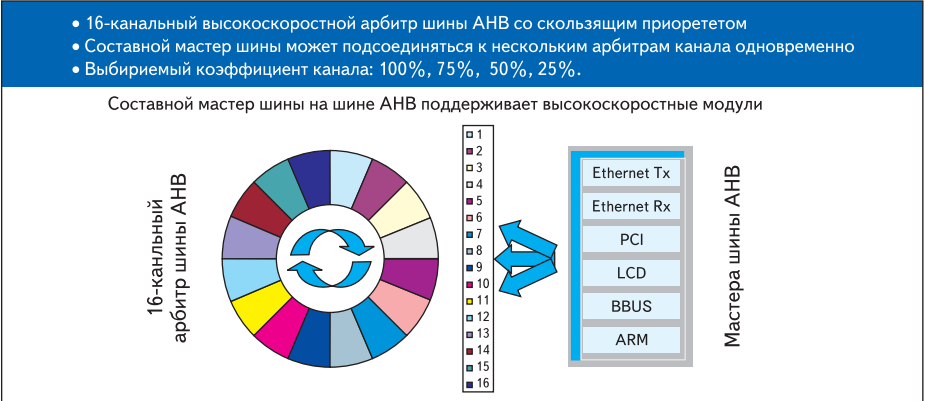


Рис. 1. Система контроля пропускной способности шин АНВ в микропроцессорах типа NS9XXX

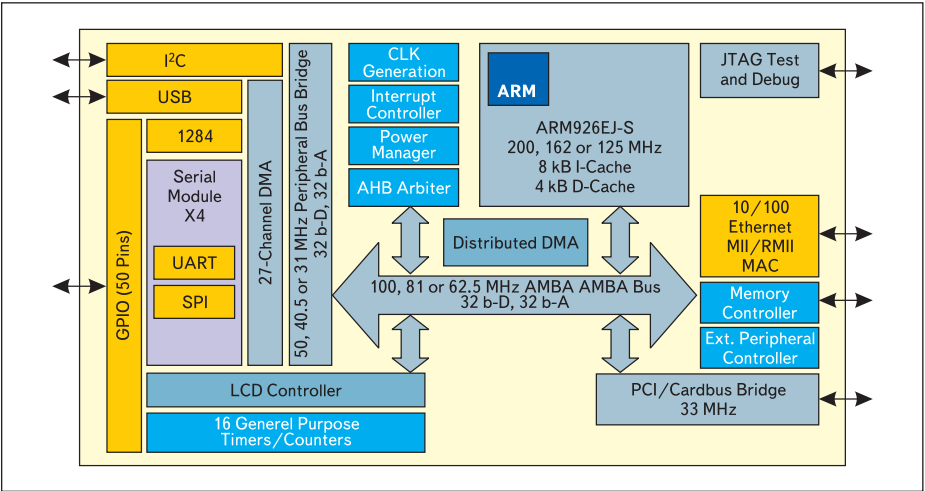


Рис. 2. Блок-схема микропроцессора NS9750

пределение по мере необходимости будет пересматриваться программным обеспечением системы при ее работе, что можно связать с числом циклов шины АНВ. Если ситуация за следующий цикл оценки не изменится, распределение остается таким же. Если изменилась — вырабатывается новое распределение слов для мастер-устройств.

Для еще более точного управления ресурсами шины такая схема циклического арбитража предоставляет два дополнительных уровня программируемости: доля пропускной способности шины, предоставленная центральному процессору ARM, и степень использования полосы для каждого из 16 слотов.

Вместо возможности управлять всеми ресурсами шины при работе в качестве мастер-устройства ядру ARM926EJ-S системы NS9750 по умолчанию выделено только 50% полной пропускной способности шины, или 8 слотов из 16. Это гарантирует, что в распоряжении остальных пяти мастер-устройств в любой момент времени находится по крайней мере 50% шины. Однако при прямом программном управлении ядру может быть «приказано» освободить часть выделенных ему ресурсов или же взять под свое управление дополнительные слоты на данный цикл арбитража шины, или на любое число циклов, определенное программистом.

Программист может также выбрать 100-, 75-, 50- или 25-процентный коэффициент использования пропускной способности шины на слот. Это означает, что для 25-процентного коэффициента данный конкретный слот будет опрашиваться только раз в четыре цикла, для 50-процентного — каждый второй цикл, для 75-процентного — каждые три цикла из четырех.

Программирование «вращающегося» арбитра шины

Программист может указать различные опции через несколько регистров, которые содержатся в модуле управления системой (рис. 3). Первый — это 16-входовый регистр конфигурирования запросов к шине (Bus Request Configuration — BRC). Каждый вход этого регистра представляет запрос шины для мастер-устройства и выделение слота. Каждый слот запрос/выделение в каждый момент времени предоставляется только одному мастер-устройству, однако каждое мастер-устройство может быть подключено к нескольким слотам запрос/выделение, в зависимости от требующейся этому устройству полосы. Когда одному мастер-устройству выделяется несколько каналов, эти каналы должны быть равномерно распределены среди 16 имеющихся.

В каждом слоте запрос/выделение имеет двухбитовое поле сокращения полосы (Bandwidth Reduction Field — BRF) для определения того, сколь часто каждый слот может арбитраживать системную шину — 100, 75, 50 или 25 процентов времени. Регистры BRC управляют передачей сигналов запроса на шину

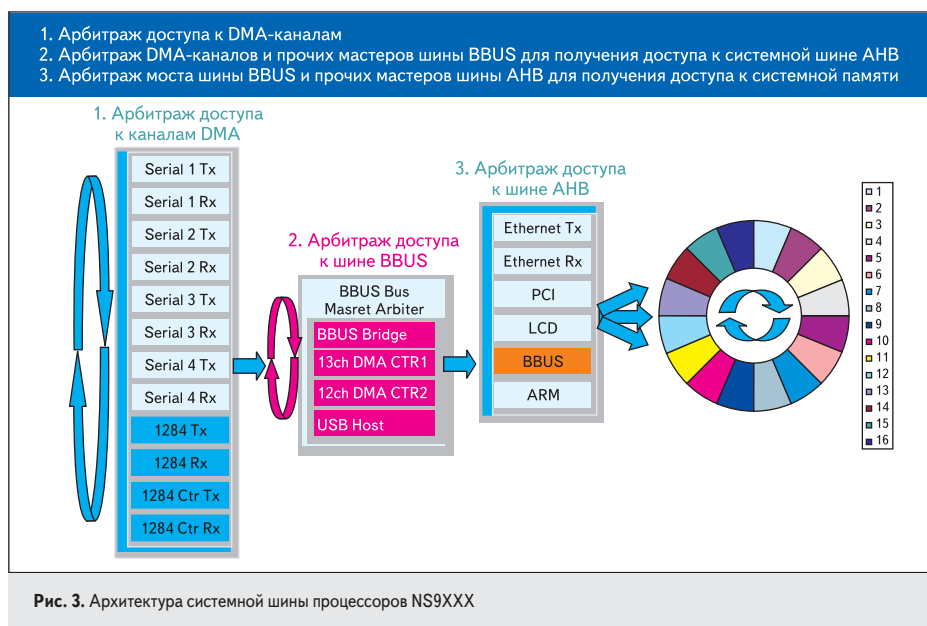


Рис. 3. Архитектура системной шины процессоров NS9XXX

во второй 16-входовый регистр запросов на шину (Bus Request Register — BRR). По умолчанию неразмещенные слоты в BRC блокируют соответствующие входы BRR от любых сигналов запроса доступа к шине.

Четвертый регистр используется для хранения сведений о том, у какого из мастер-устройств есть данные, ожидающие передачи на АНВ, а пятый регистр используется программистом для задания весового значения каждому запросу на шину и выделения слота, предоставленного конкретному мастер-устройству.

Циклический арбитраж в действии

В приведенном ранее примере, когда ЖК-монитор по определенному графику перераспределения конкретных арбитражей запрашивает дополнительный доступ к шине, программист может указать, что ЖК-монитору должен быть дан приоритет, основанный на природе потока данных, которые он должен обрабатывать. Если программист решит, что десять слотов должны быть предоставлены именно контроллеру ЖК-монитора, то шесть оставшихся слотов должны быть предоставлены остальным мастер-устройствам из исходного графика. В такой ситуации в распоряжении контроллера ЖК-монитора будет вдесятеро большая полоса, чем была бы в обычных условиях, и вдесятеро большая, чем у остальных мастер-устройств, чтобы обработать поток данных в данной конкретной ситуации.

Эта возможность может стать критически важной, когда устройство передает потоковые данные через Ethernet-соединение в то же время, в которое происходит обновление ЖК-монитора. Информацию на ЖК-мониторе требуется обновлять своевременно, и не прерывать этот процесс передачей данных через Ethernet.

В типовой шинной архитектуре AMBA, если ЖК-монитор запросил доступ к шине, ему придется ждать, пока мастер-устройство

Ethernet освободит ее, независимо от требования обновления информации. При новой программируемой схеме арбитража программист может «отложить» передачу данных через Ethernet, разрешив передачу с меньшей, но все еще приемлемой скоростью, и гарантируя, что ЖК-монитор получит достаточные ресурсы для своевременного обновления, так что изображение на экране не исчезнет.

Там, где требуется точное соблюдение временных параметров обслуживания ЖК-дисплея и полосы пропускания для гарантированного отображения актуальной информации, требования для Ethernet-протокола гораздо более мягкие и позволяют работать при более низких скоростях передачи. Однако поток данных не должен прерываться, что имело бы место в случае, если мастер шины ЖК-дисплея захватил управление шиной и освободил ее только тогда, когда закончилось бы обновление дисплея.

Преимущества архитектуры NS9xxx

Во встроенных широкополосных системах программисты должны оптимизировать передачу через канал нескольких параллельных потоков данных. Патентованное решение NetSilicon, описанное выше, дает в руки программистам мощное средство для удовлетворения конкретных требований их приложений.

С все более широким распространением субмикронных технологических процессов одна «система-на-кристалле» может включать в себя все более разнообразные функции. Это потребует, чтобы архитектуры RTOS и DMA предоставляли системным инженерам гибкое управление ресурсами системы. По мере того, как встроенные аппаратные решения продолжают проникать в область IP-сетей, NetSilicon полна решимости предлагать аппаратные и программные сервисы, призванные сделать такую интеграцию успешной.