

# Нужен ли RAID на USB?

09 мая 2016



Количество имеет свойство переходить в качество. Характеристики производительности универсальной последовательной шины не являются исключением из этой фундаментальной истины. Достижение планки в 10 Гб/сек открыло для USB3.1 ряд новых сфер применения, для которых ранее эта технология считалась недостаточно производительной. Нас интересует насколько эффективным будет [RAID-массив, подключенный к шине USB3.1](#).

Сразу оговоримся, что эта статья не призвана дать однозначный ответ. Сегодня мы построим несколько теоретических моделей, для практического подтверждения или опровержения которых потребуется достаточно большой цикл публикаций и бенчмарков. Архитектура и стратегия функционирования RAID-контроллеров, как правило является секретом производителя. Этим она отличается от унифицированных моделей *Bus Master IDE* и *AHCI*, программная модель которых открыта и унифицирована «с точностью до бита».



**Рис.1** Плата USB RAID и контроллер ASM1352R: режимы RAID устанавливаются переключателями, согласно нанесенной таблице

## Постановка задачи

Попытка оценить ситуацию «в целом» приведет к беспредметным рассуждениям. Поэтому, к общему будем идти от частного и рассмотрим конкретные типы операций в конкретных типах RAID массивов при фиксированном количестве накопителей.

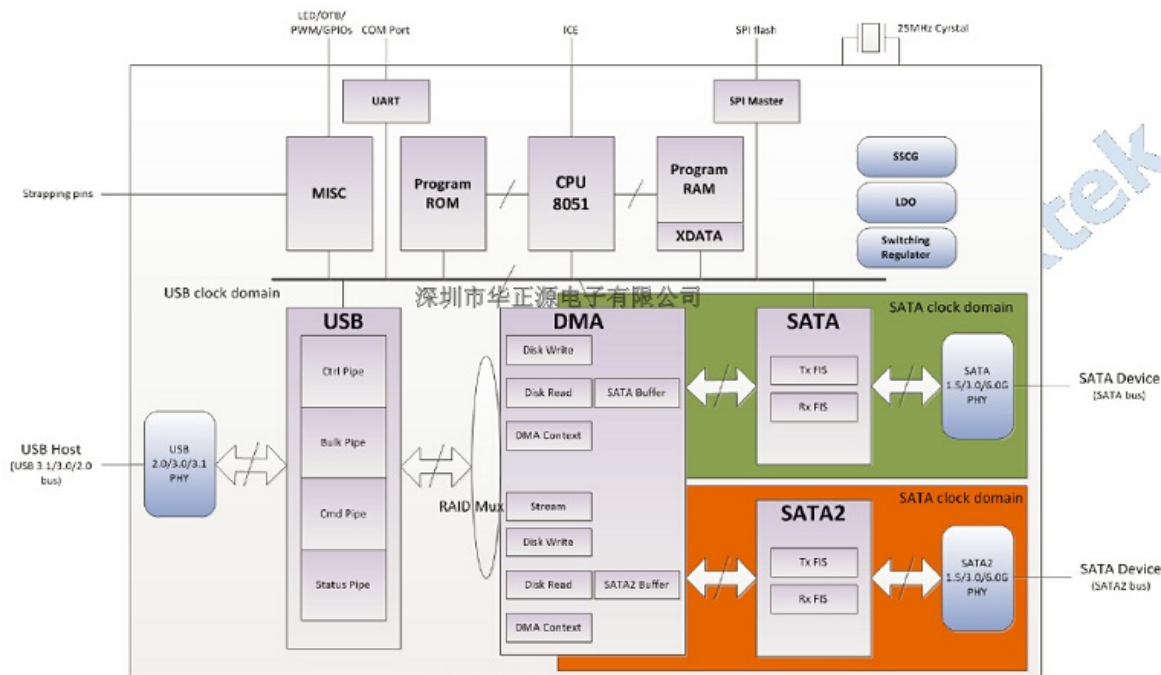


Figure 1: Functional Diagram of ASM1352R

Рис 2. Функциональная схема контроллера ASM1352R: режим отладки на основе технологического UART — неизменный атрибут контроллеров ASMedia

Рассмотрим гипотетическую платформу, оборудованную интерфейсом USB3.1 к которому подключен RAID контроллер. В свою очередь, к RAID контроллеру подключено два накопителя. Интерфейс USB3.1, связывающий хост-компьютер и контроллер, обеспечивает трафик 10 гигабит в секунду. Два накопителя стандарта SATA III, подключенных к контроллеру, способны создать трафик  $2 \times 6 = 12$  гигабит в секунду. Указаны теоретические величины, также, пока не учитываем фактор избыточного кодирования при передаче данных. Для упрощенной модели, примем, что суммарный трафик, создаваемый накопителями, примерно в 1.2 раза выше, чем пропускная способность интерфейса между контроллером и хост-компьютером.

## Уточнение постановки задачи

Как было сказано выше, для выполнения предметного анализа необходимо уточнить уровень RAID массива и вид операции. Для начала анализировать будем два простейших вида массивов: RAID0 и RAID1, и только в вариантах с двумя накопителями. Напомним базовые принципы.

RAID0 (strip) обеспечивает увеличение производительности за счет распараллеливания операций. Дисковое пространство разделяется на блоки, типовой размер таких блоков от 0.5 до десятков килобайт, в зависимости от реализации. Четные блоки хранятся на первом накопителе, нечетные на втором. При обработке запроса на чтение или запись, каждый из накопителей обрабатывает «свои» блоки, в теории, параллельно и синхронно. Теоретически, одинаковые накопители должны выполнять одинаковые операции за одинаковое время. На практике, в действие вступает фактор разброса параметров, наличие отремапленных дефектных секторов, рассинхронизация вращения

шпинделей и многое другое, поэтому фактор масштабирования производительности всегда будет немного меньше ожидаемого.

*RAID1 (mirror)* обеспечивает надежность хранения за счет дублирования информации на двух накопителях. Выполнение запроса на запись будет состоять в записи одинаковых данных по одинаковому адресу, на оба диска. Примечательно, что в случае чтения, RAID1 может повышать производительность, используя параллельное чтение двух дисков. Но поскольку деления на четные и нечетные блоки в данном случае нет, в отличие от RAID0, читаемые диапазоны секторов для двух дисков здесь будут различны, что требует большего интеллекта от контроллера и позволяет говорить об асинхронной работе накопителей.



*Рис 3. Еще один вариант реализации RAID-массива, сформированного на USB-шине с помощью контроллера ASM1352R от ASMedia Technology*

Для двух простейших уровней RAID-массивов, рассмотрим по две операции: чтение и запись. Итого, 4 случая, для каждого из которых попытаемся смоделировать влияние дефицита USB-трафика.

### RAID0, чтение данных

Этот и следующий случаи в наибольшей степени критичны к дефициту шинного трафика. Поскольку данные расположены с повторяющимся чередованием, от контроллера требуется достаточно мало: синхронный и параллельный запуск операций на двух накопителях. Интеллект контроллера, необходимый для сборки одного результирующего блока из двух фрагментов, прочитанных с двух дисков, в данном случае минимален. Поэтому, есть основания считать, что критичность к остальным факторам, в том числе трафику шины USB, здесь будет максимальной.

## RAID0, запись данных

Этот случай повторяет ранее рассмотренный, за исключением направления передачи данных.

## RAID1, чтение данных

Напомним, смысл RAID1 в надежности хранения данных, а прибавка производительности за счет параллелизма чтения здесь является только бонусом, требующим дополнительного интеллекта от контроллера и асинхронной работы накопителей (в отличие от RAID0). Есть основания полагать, что полнота реализации такой прибавки здесь в большей степени зависит от локального интеллекта контроллера, чем от шинного трафика USB.

## RAID1, запись данных

Это самый интересный случай. Поскольку операция записи дублируется для двух накопителей и записываемые данные совпадают, нет смысла дважды передавать по USB интерфейсу одинаковые данные. Если RAID контроллер «достаточно умен», чтобы учесть это, требования к трафику универсальной последовательной шины сократятся вдвое, до уровня одного накопителя.

## Фактор интеллекта

Очевидно, если шинный трафик в дефиците, то им нужно распорядиться рационально. Одним из примеров такого подхода является технология *Bulk Streams*, реализованная в третьем поколении USB. Контроллеру предоставляется информация о нескольких потоках запросов ввода-вывода (а не только одном текущем запросе). Это дает возможность выполнить последовательность операций ввода-вывода в таком порядке, который обеспечит максимальную производительность. Практика показывает, что сортировка [очереди команд NCO](#) не потеряла своей актуальности даже с приходом SSD дисков.

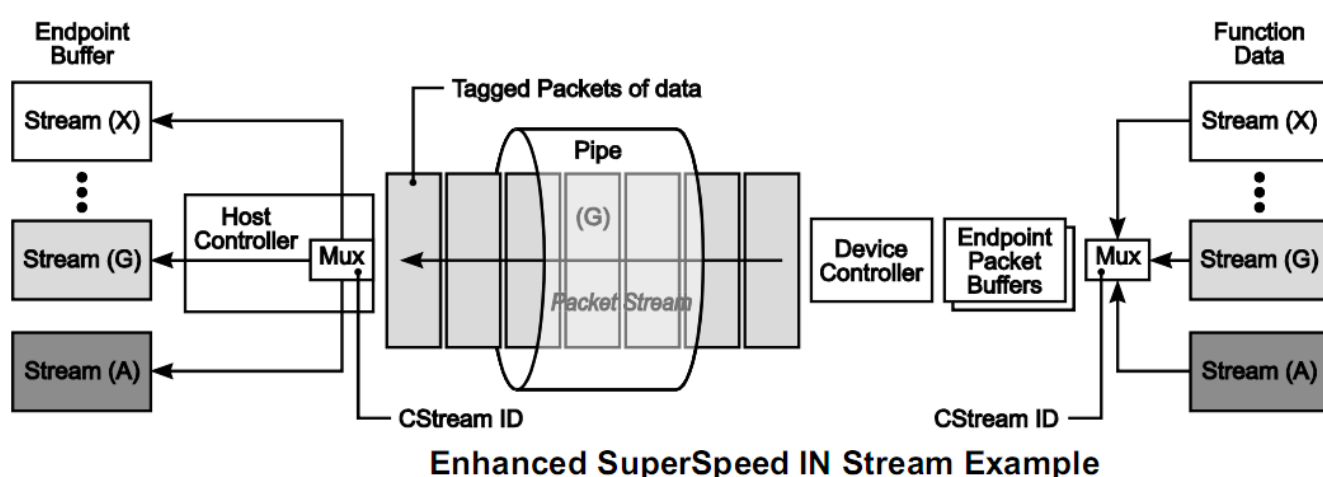


Рис. 4 Технология Bulk Streams в действии: накопителю доступна информация о нескольких потоках запросов ввода-вывода одновременно, что дает возможность оптимизировать последовательность выполнения операций чтения и записи

## Вместо послесловия

Нашу упрощенную модель придется несколько откорректировать, если топология USB-подключений насыщена и в распоряжении контроллера дисков находится только часть теоретического трафика USB. Наличие сбойных или долгодитаемых секторов также вносит свои поправки, даже после их успешного *ремапа*. А более фатальные ошибки, сопровождающиеся переходом накопителя в состояние таймаута на несколько секунд, сделают все рассуждения о производительности бессмысленными. Именно поэтому, *анализ функционирования RAID массива должен быть комплексным*. Звучит банально, но верно...

Теги: [Мнения](#)