

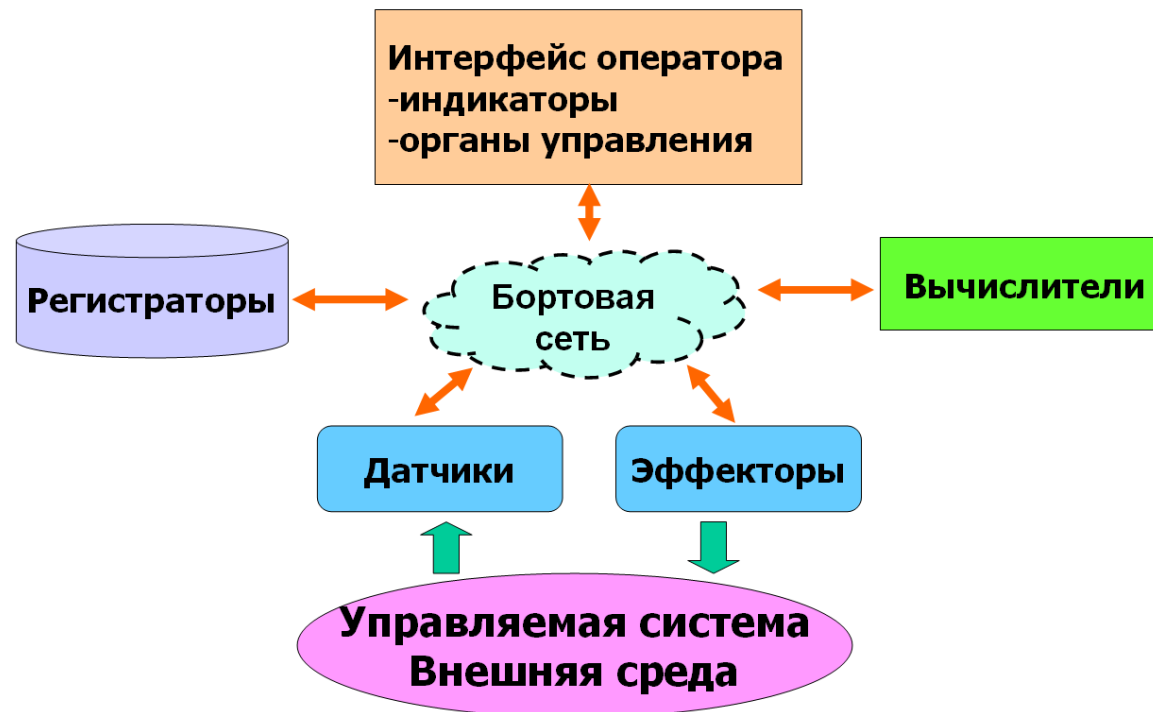
ИНФОРМАЦИОННО-УПРАВЛЯЮЩИЕ СИСТЕМЫ РЕАЛЬНОГО ВРЕМЕНИ

Лекция 9: *Коммутируемые сети*

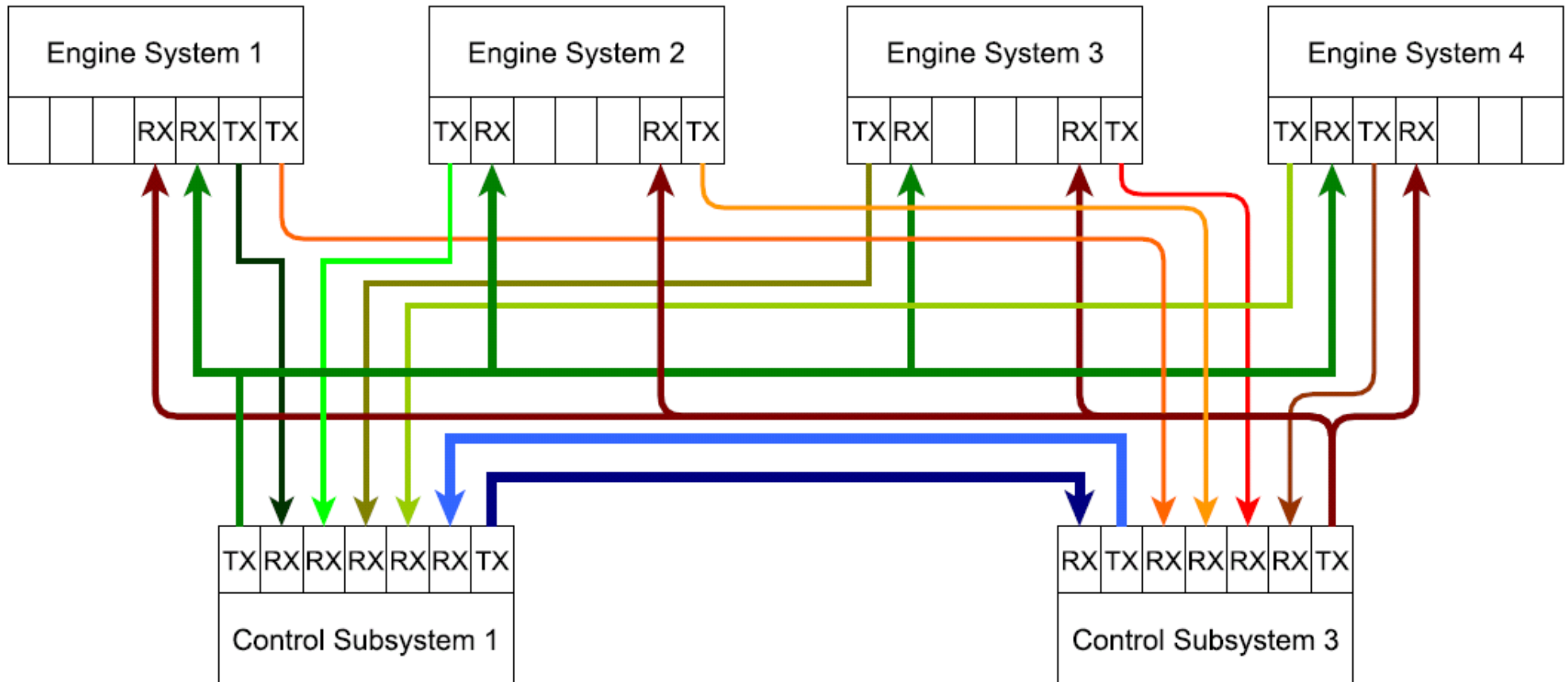
Кафедра АСВК,
Лаборатория Вычислительных Комплексов
Балашов В.В.

Бортовые сети

- Бортовые сети – обеспечение связи между бортовыми подсистемами
 - Надежная доставка
 - Соблюдение требований реального времени

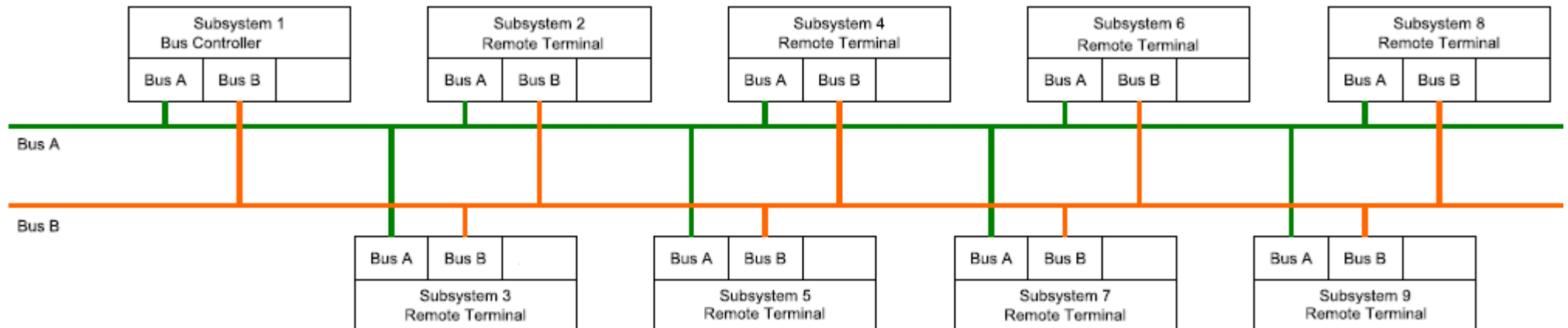


Каналы точка-точка



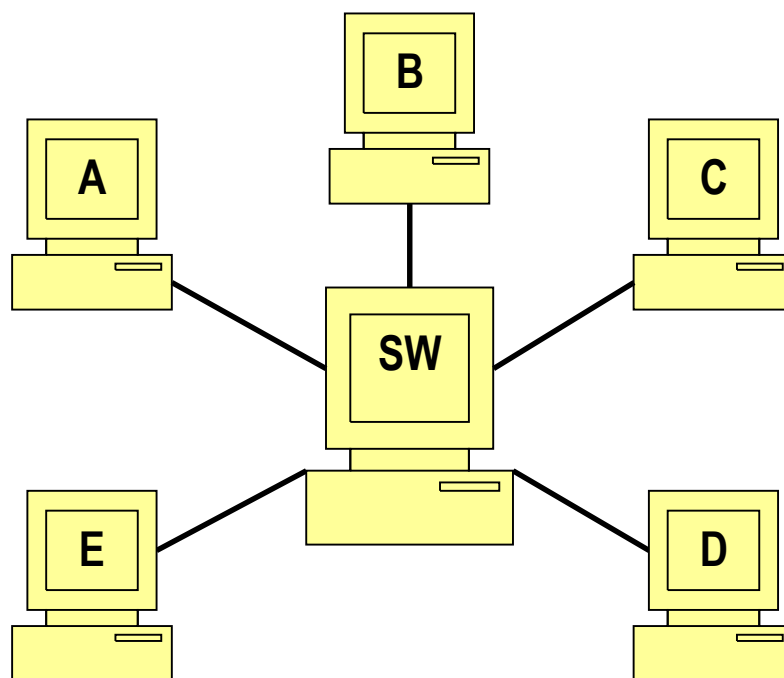
- Много кабелей
- Пропускная способность: недоиспользование, нехватка, сложность наращивания
- Сложно закладывать резерв
- Проблемы с передачей данных по сложному маршруту
- Низкая адаптивность (невозможна реконфигурация)

Интеграция каналов, мультиплексирование трафика



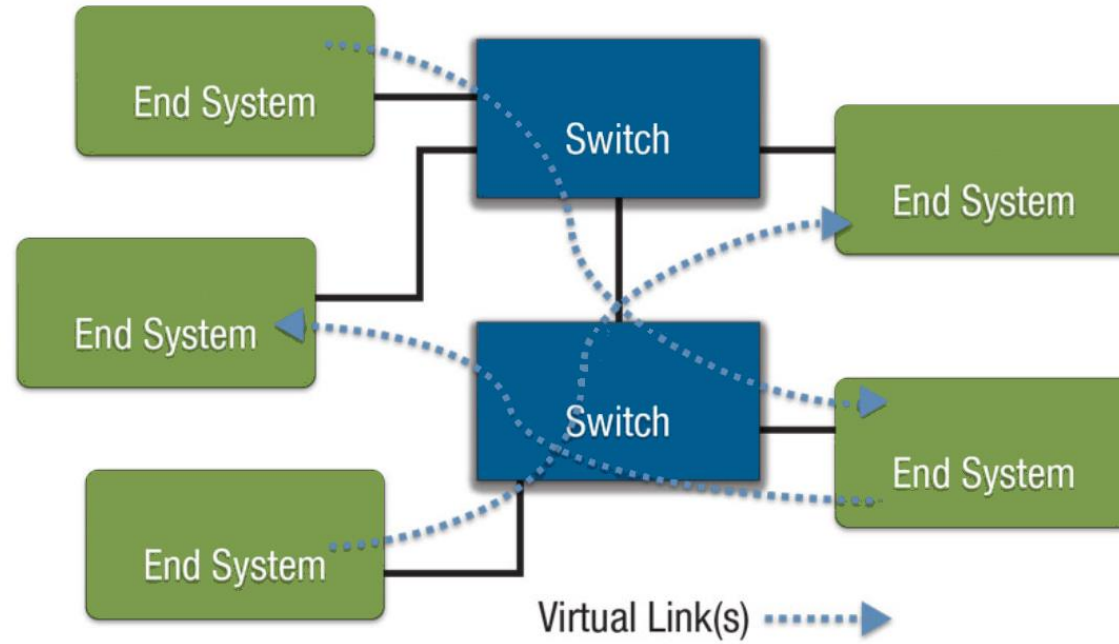
- Много каналов точка-точка → общая шина для многих потоков данных
- Проблема коллизий при доступе к шине
 - синхронизация доступа (нужно единое время)
 - *либо* централизованное управление (накладные расходы...)
- Последовательная обработка запросов => задержки
- Нет устойчивости к «генерации» в канале при выходе абонента из строя

Коммутатор: параллелизм обмена



- Дуплексные (двунаправленные) каналы
- Частичный параллелизм обмена
 - $A \rightarrow B \parallel B \rightarrow C \parallel C \rightarrow D$ – нет конфликта
 - $A \rightarrow B, C \rightarrow B$ – конфликт на линии $SW \rightarrow B$; как делить линию?
- Неустойчивость к «генерации»
- Проблема мультиплексирования потоков данных при отправке
- Оставшиеся конфликты – как лечить?
 - синхронизация доступа (нужно единое время)
 - централизованное управление (накладные расходы...)
 - верхние оценки задержек (а если между A и B большой поток?)

Виртуальные каналы



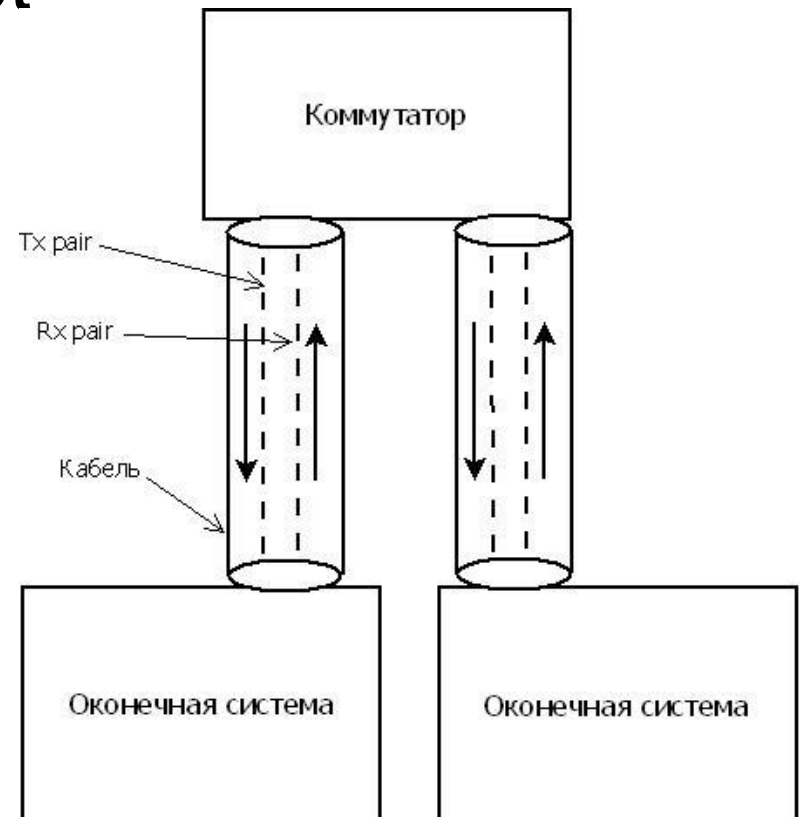
- Разделение пропускной способности, разграничение потоков данных
 - => пригодность для сложного трафика из множества потоков данных
- Гарантированные верхние границы задержек
- Резервирование, гибкость реконфигурации
- Реализация: согласованные действия отправителя и коммутаторов
- Устойчивость к «генерации»
 - коммутатор сбрасывает слишком частые кадры

Протоколы коммутируемых сетей с поддержкой ВК

- AFDX (на базе 100 Мбит Ethernet)
- FC-RT (на базе Fibre Channel)
- Программно-конфигурируемые сети

Стандарт AFDX

- Avionics Full-Duplex Ethernet (AFDX) – стандарт построения бортовых сетей на основе протокола Ethernet
 - Основан на протоколе Ethernet
 - Полнодуплексная передача данных
 - Теоретическая пропускная способность – до 100 Мбит/с на одном физическом соединении



Архитектура сети AFDX

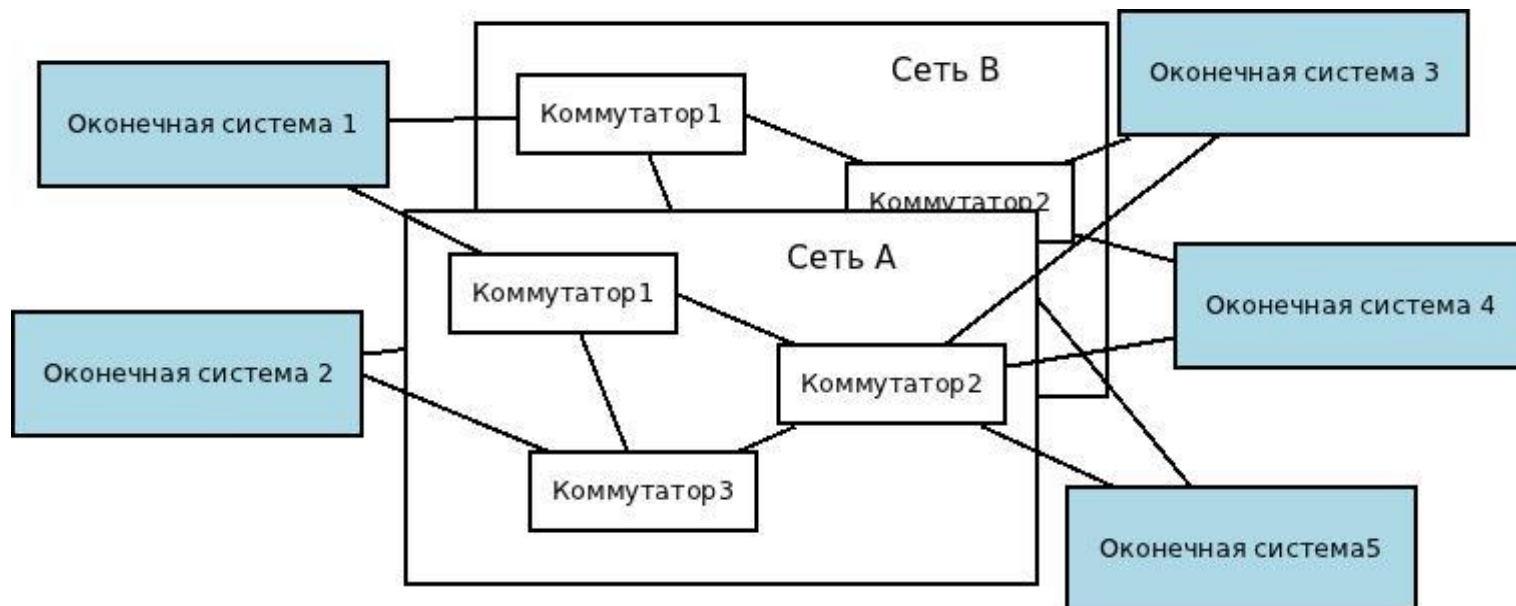
- Компоненты:

- Абоненты (бортовые подсистемы, отправители и получатели данных)
- Оконечные системы – интерфейс между абонентами и сетью
- Коммутаторы и физические соединения



Архитектура сети AFDX

- Дублирование сети для увеличения надежности передачи
 - Кадры передаются одновременно в обе сети
 - При диагностировании ошибки (например, несовпадение контрольной суммы) в одной сети данные берутся из другой сети
 - На конечной системе производится сброс кадра в случае, если кадр уже пришел из другой сети

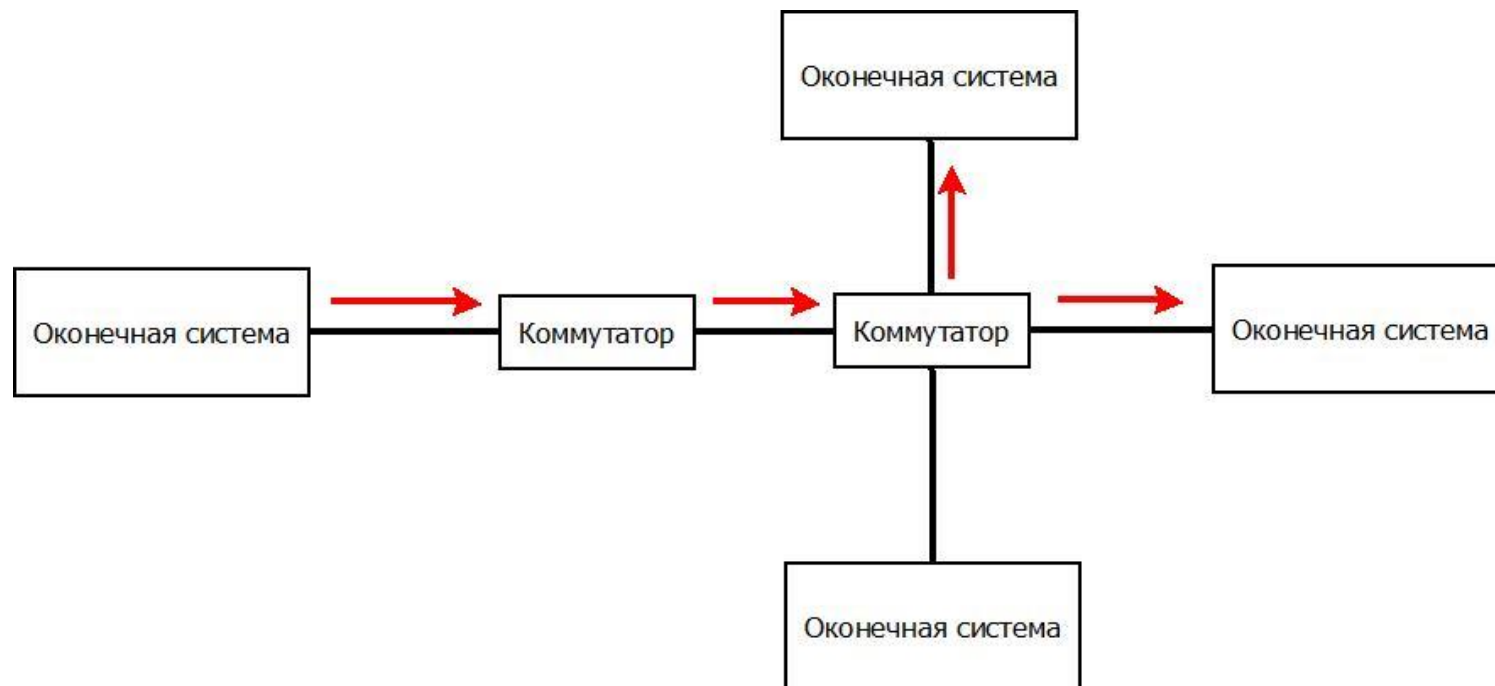


Стек протоколов

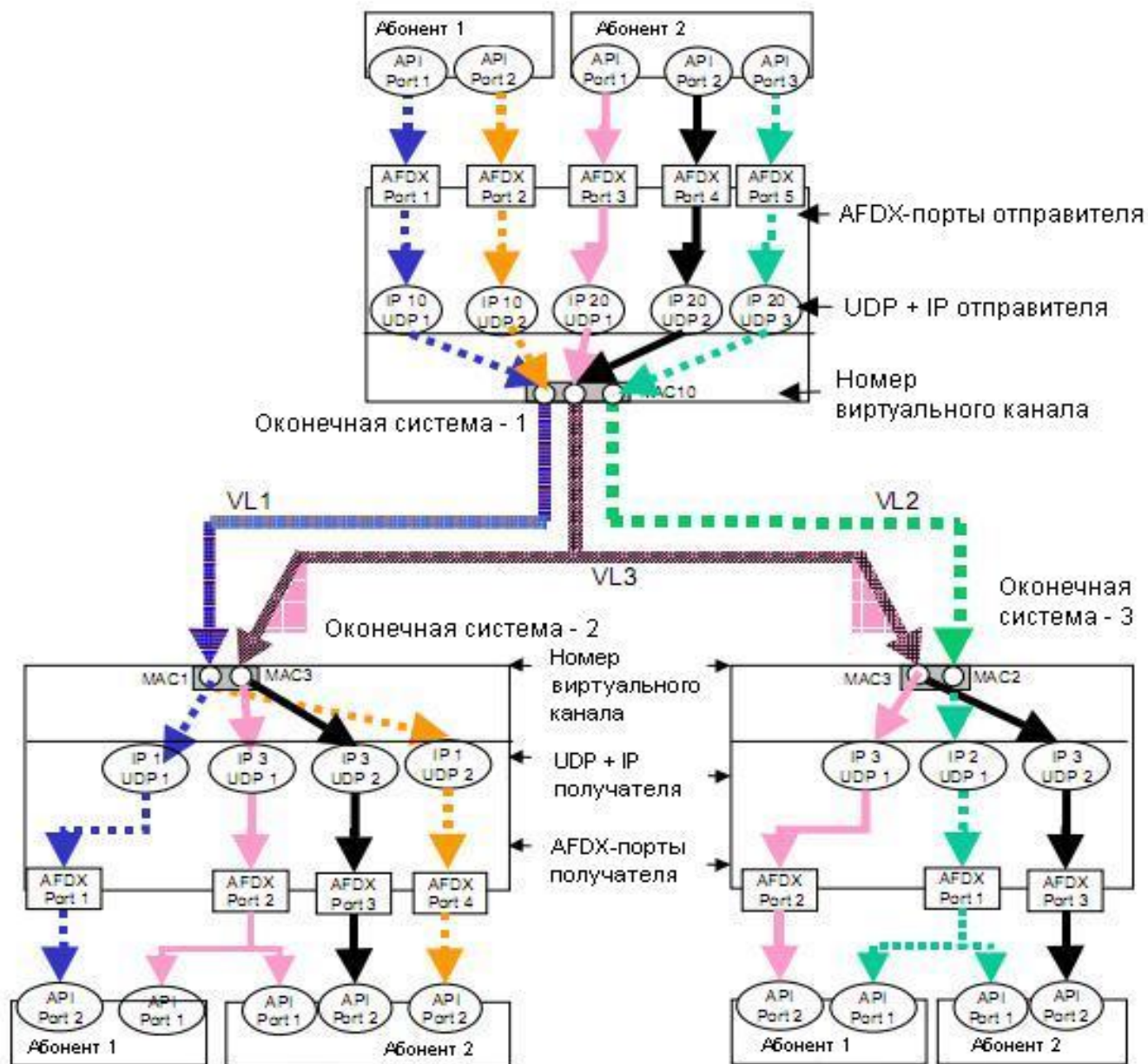
- Канальный уровень
 - Ethernet
 - Виртуальные каналы
 - Маршрутизация
- Сетевой уровень
 - IP (без маршрутизации)
- Транспортный уровень
 - UDP

Стек протоколов

- Виртуальные каналы
 - Одна оконечная система – отправитель; одна или более оконечная система – получатель
 - Маршрут следования кадров виртуального канала прописан статически в коммутаторах



Передача данных

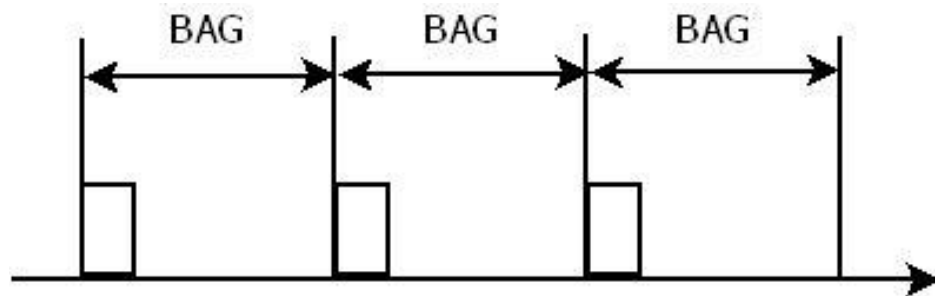


Параметры виртуальных каналов

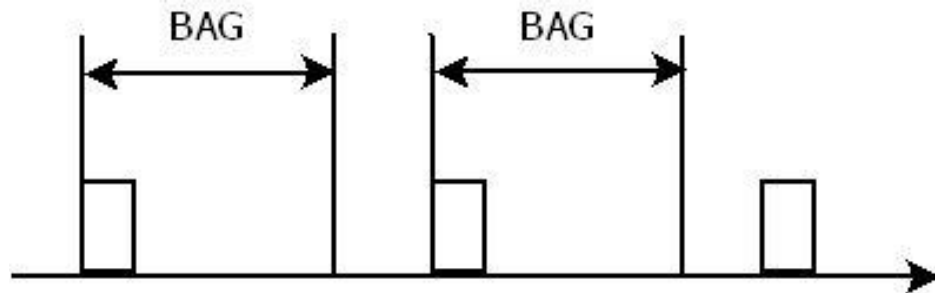
- Для каждого виртуального канала вводятся следующие параметры:
 - *BAG* – Bandwidth Allocation Gap – минимальный интервал времени между началами выдачи последовательных кадров на одном виртуальном канале (1-128 мс, является степенью двойки)
 - *L_{max}* – максимальный размер кадра (≤ 1518 байт)
 - *J_{max}* – максимально допустимое отклонение между кадрами от *BAG*

Параметры виртуальных каналов: *BAG*

- Использование *BAG* для достижения максимальной выделенной пропускной способности:



- Альтернативный вариант:



- В дальнейшем рассматривается только первый вариант передачи, без промежутков между соседними *BAG*-интервалами

Пропускная способность виртуальных каналов

- Вычисление:

- $Bandwidth = L_{max} / BAG$

- $BAG = 32 \text{ мс}$

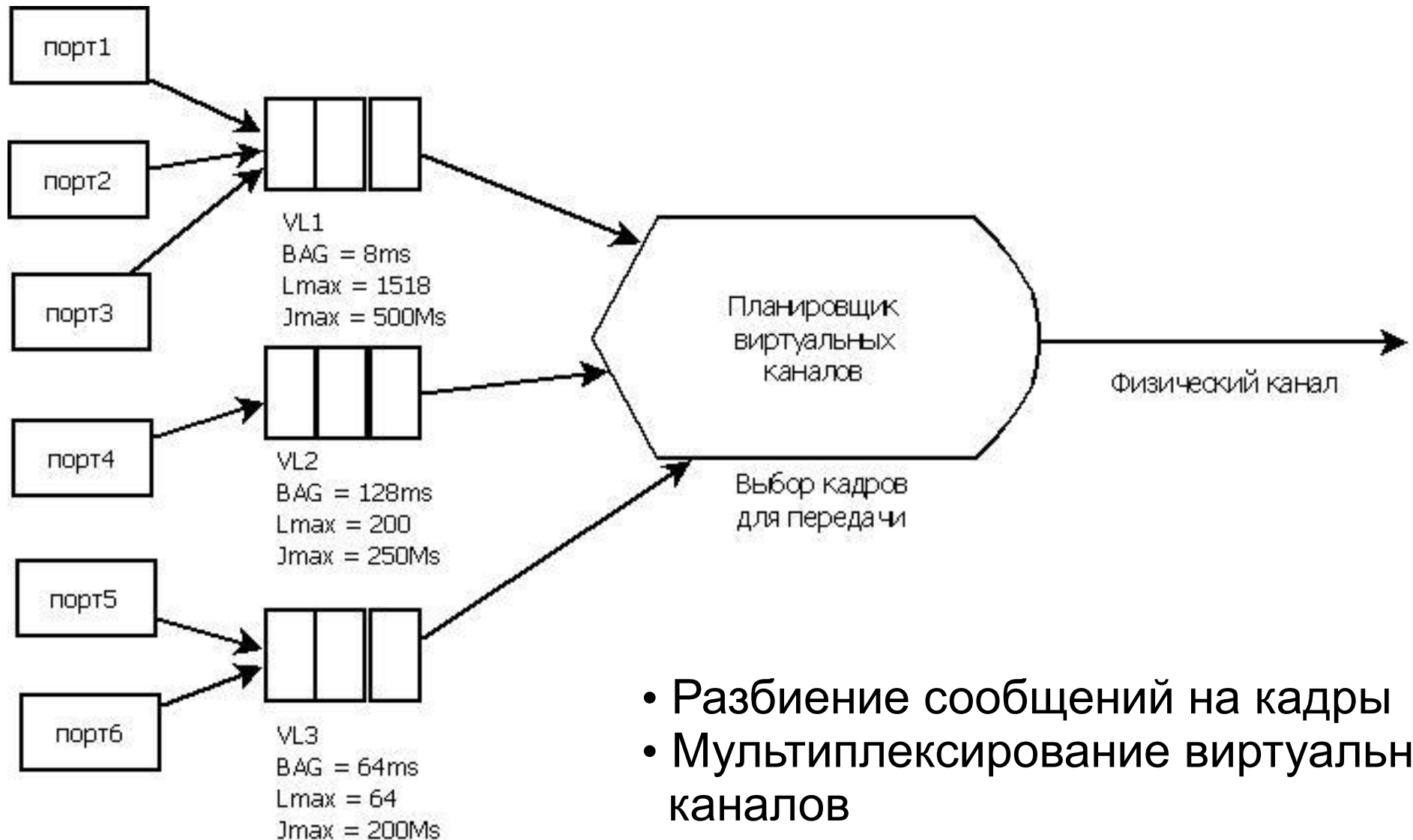
- $L_{max} = 200 \text{ байт}$

- $Bandwidth = 200 \text{ байт} / (32 / 1000) \text{ сек} = 6250 \text{ байт/сек}$

- Ограничение на зарезервированную пропускную способность на физическом канале (проводе):

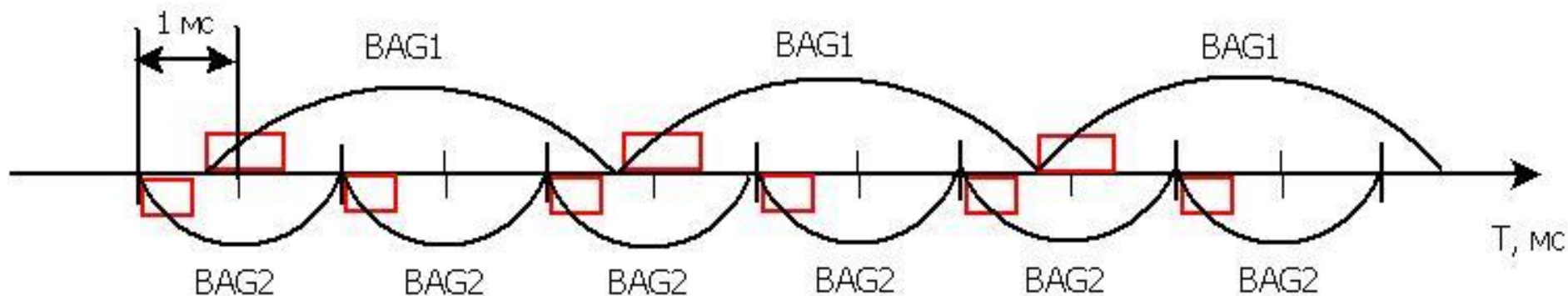
- $\sum_{VL=1..n} L_{VL,max} / BAG_{VL} \leq 100 \text{ Мбит} / \text{сек}$

Управление виртуальными каналами



Формирование трафика

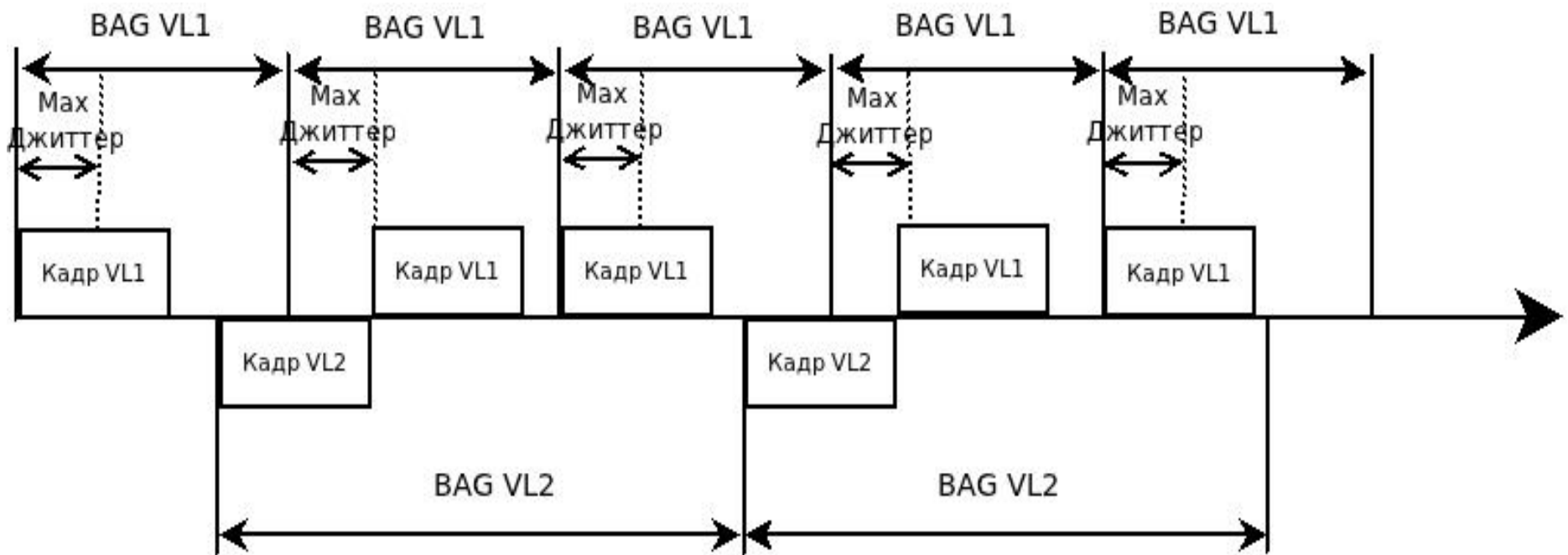
- При формировании трафика на отправителе – мультиплексирование
- При мультиплексировании определяется значение джиттера
- С нулевым джиттером:



Кратность BAG-ов, известность всех L_{max} , отсутствие джиттера *готовности кадров* в каждом ВК
=> исключено взаимное влияние ВК,
отсутствует джиттер *выдачи кадров* в каждом ВК

Формирование трафика

- Мультиплексирование с ненулевым джиттером



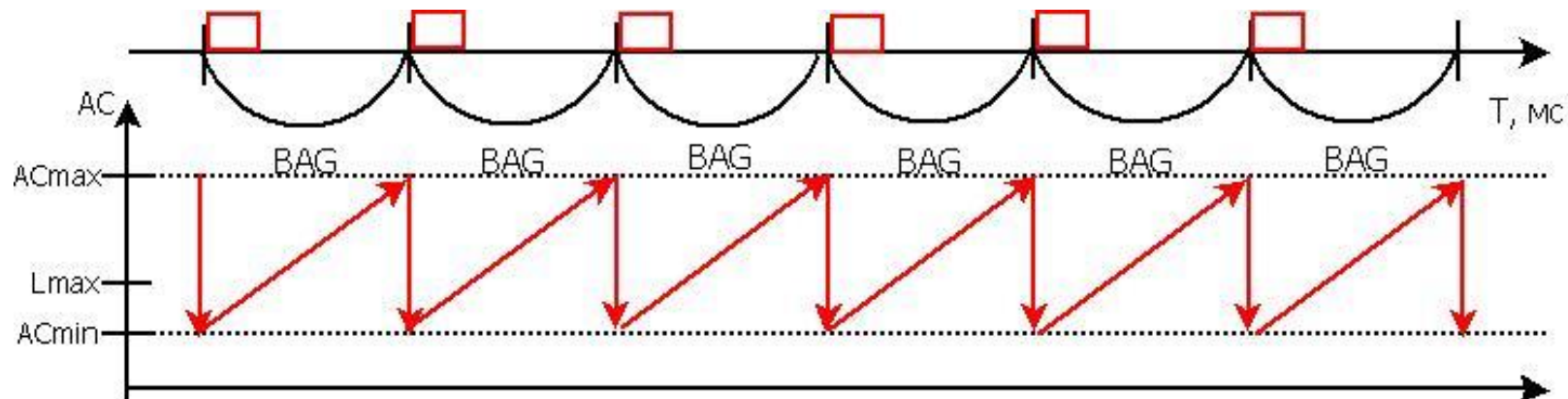
Такого варианта стремятся избегать

Коммутатор

- **Функции коммутатора:**
 - Маршрутизация кадров по пути следования виртуальных каналов (пути виртуальных каналов конфигурируются статически)
 - Фильтрация трафика (контроль целостности кадра, контроль следования кадра по виртуальному каналу)
 - Контроль трафика
 - размер кадра (не должен превышать L_{max})
 - BAG, J_{max}
 - нарушение => сброс кадра

Контроль трафика на коммутаторе

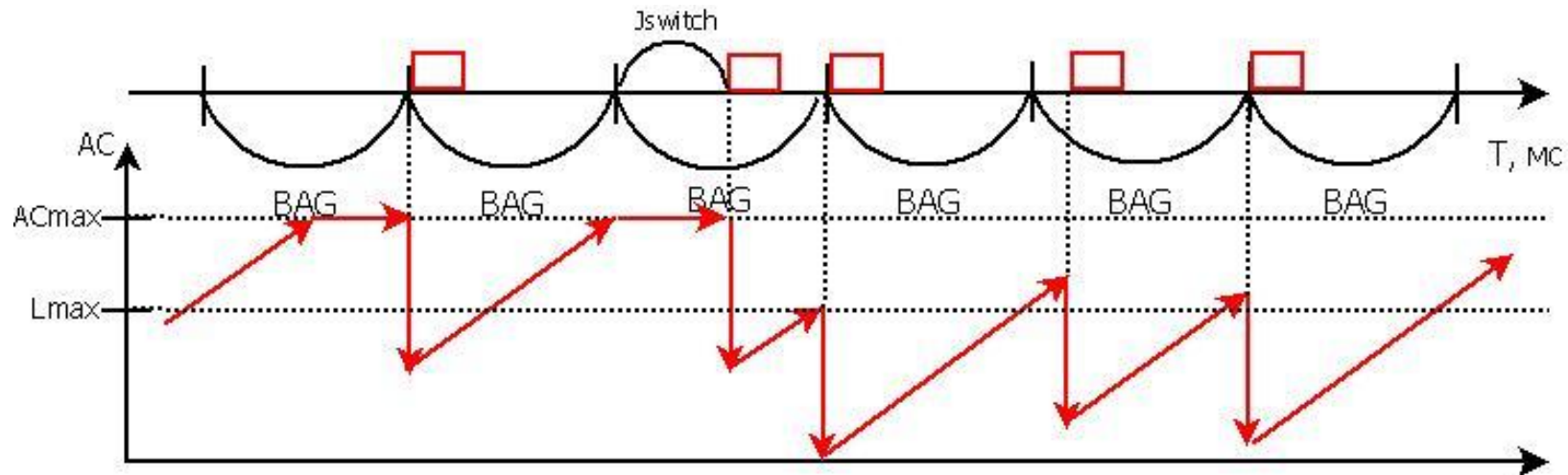
- Контроль времен прихода кадров на соответствие BAG и J_{max} :
 - Производится на входном порту коммутатора
 - Используется алгоритм, основанный на вычислении кредита
 - AC – кредит, растет с течением времени до значения AC_{max}
 - При приходе кадра AC уменьшается на размер кадра; если кредита не хватает – кадр сбрасывается



Контроль трафика на коммутаторе

- Кредит соответствует количеству байт, которые пропускает канал
 - За время BAG кредит увеличивается на $Lmax$
 - $ACmax$ – соответствует количеству байт, которое позволяет пропустить 2 кадра за $(BAG - Jmax)$

Обработка ненулевого джиттера:



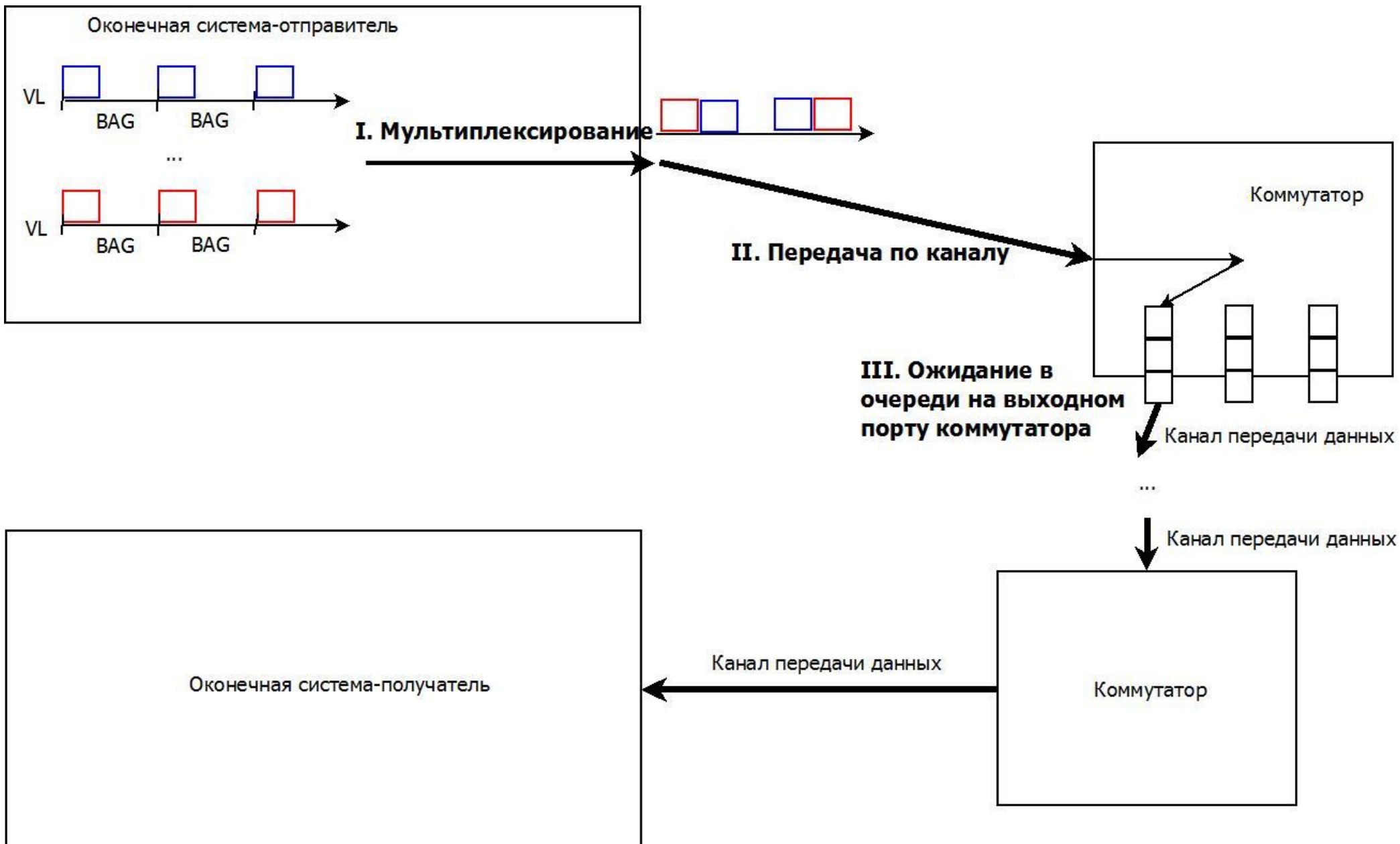
Задачи проектирования сети AFDX

- Дано: потоки данных, требования к их передаче в реальном времени
 - размер сообщения
 - частота передачи
 - макс. допустимый джиттер (end-to-end)
 - макс. допустимая задержка (end-to-end)
- Требуется:
 - построить систему виртуальных каналов:
 - маршруты
 - параметры (BAG, Lmax)
 - рассчитать конфигурационные параметры сетевых устройств – коммутаторов, абонентов (в т.ч. Jmax)

Оценка длительности передачи кадра

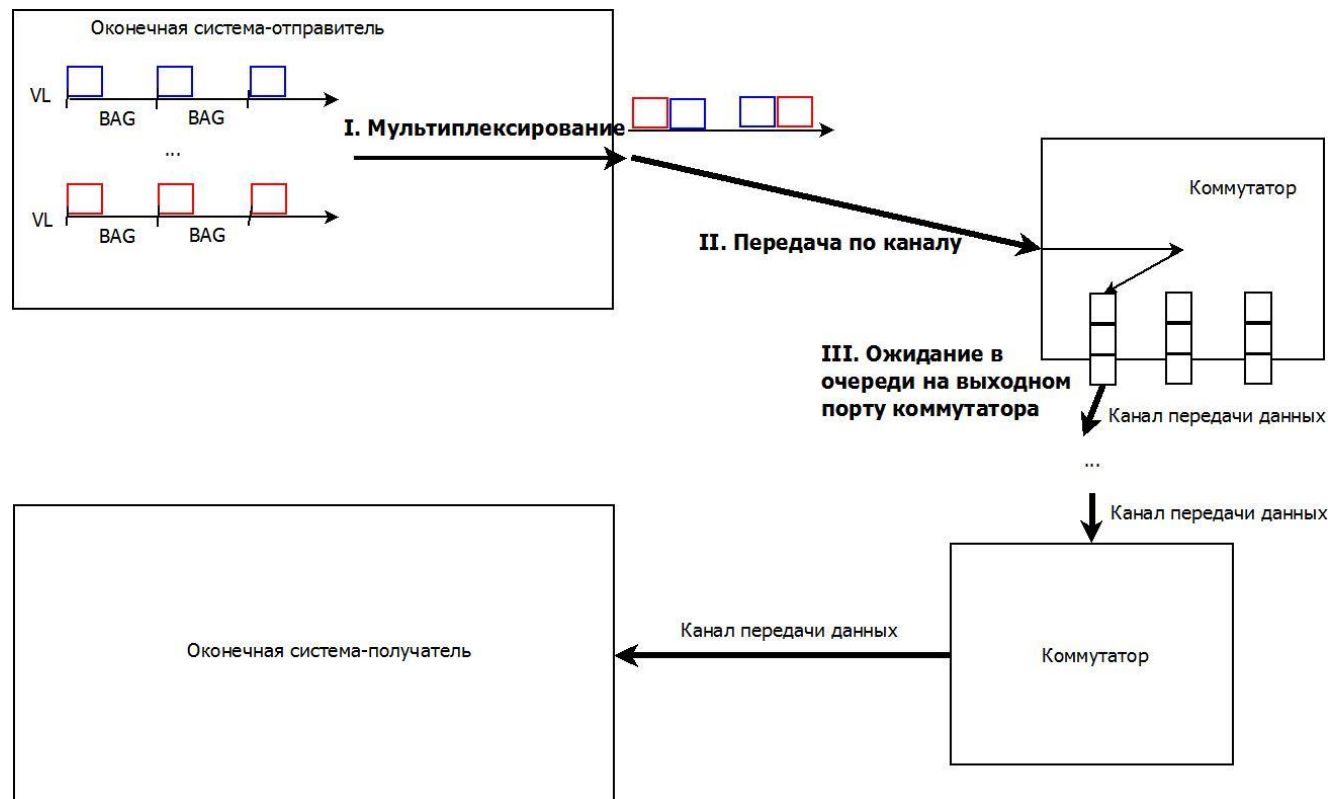
- Необходима для оценки длительности передачи сообщения
 - Актуальность: требования реального времени – длительность не должна превышать заданных значений
 - Длительность вычисляется с момента поступления кадра для выдачи в канал до момента поступления кадра на окончательную систему-получатель

Оценка длительности передачи кадра



Оценка длительности передачи кадра

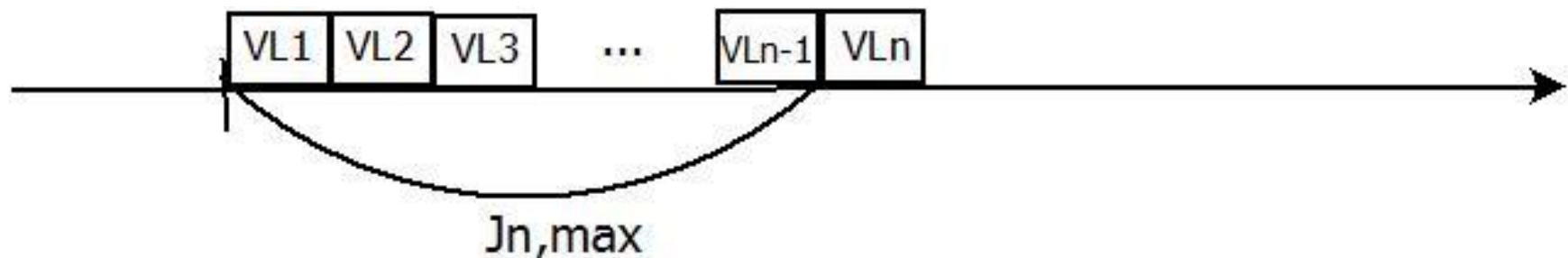
- Длительность передачи кадра:
 - максимальный джиттер на отправителе
 - длительность передачи по каналам
 - задержки на выходных портах коммутаторов



Оценка длительности передачи кадра

- Мультиплексирование

- При мультиплексировании может возникать джиттер
- Максимальная задержка – при максимальном джиттере
- Максимальное значение джиттера – при ожидании кадров всех других виртуальных каналов



Оценка длительности передачи кадра

- Мультиплексирование

- Вычисление максимального джиттера на отправителе или на выходе коммутатора

$$J_{\max} \leq J_T + \frac{\sum_{i \in VLs} L_{i, \max}}{R}$$

- VLs – множество виртуальных каналов, формируемых на оконечной системе-отправителе
- R – скорость выдачи данных на канал (100 Мбит/сек)
- J_T – технический джиттер (время обработки кадра), в AFDX равен 40 мкс

Оценка длительности передачи кадра

- Длительность передачи кадров по каналам
 - R – скорость выдачи данных на канал (100 Мбит/сек)
 - n – количество каналов передачи данных на пути следования кадра
 - формула расчета длительности передачи кадра по каналам:

$$t_{links} = n \cdot \frac{L_{max}}{R}$$

Оценка задержки кадра на коммутаторе

- В очереди перед кадром: кадры с других ВК, идущие на тот же выходной порт
- Сколько этих кадров?
 - интервал накопления M : макс. интервал после предыдущего кадра данного ВК
 - $M = BAG + \text{накопленный макс. джиттер}$
 - число кадров с другого ВК, набежавших за интервал M : см. формулы анализа времени отклика
 - какие-то из набежавших кадров уже выданы в выходной порт к моменту прихода «нашего» кадра (\Rightarrow их выдачи не нужно ожидать)
- Метод «trajectory approach»: аналогично расчету времени отклика для планирования задач с фиксированными приоритетами

Оценка длительности и джиттера передачи сообщения

$$Dur_{\max}(msg) = \mu + \delta + \Delta$$

$\mu = \text{const}$ (разбиение и сборка сообщения)

δ - время выдачи сообщения в канал (выдача последнего кадра)

Δ - длительность передачи кадра

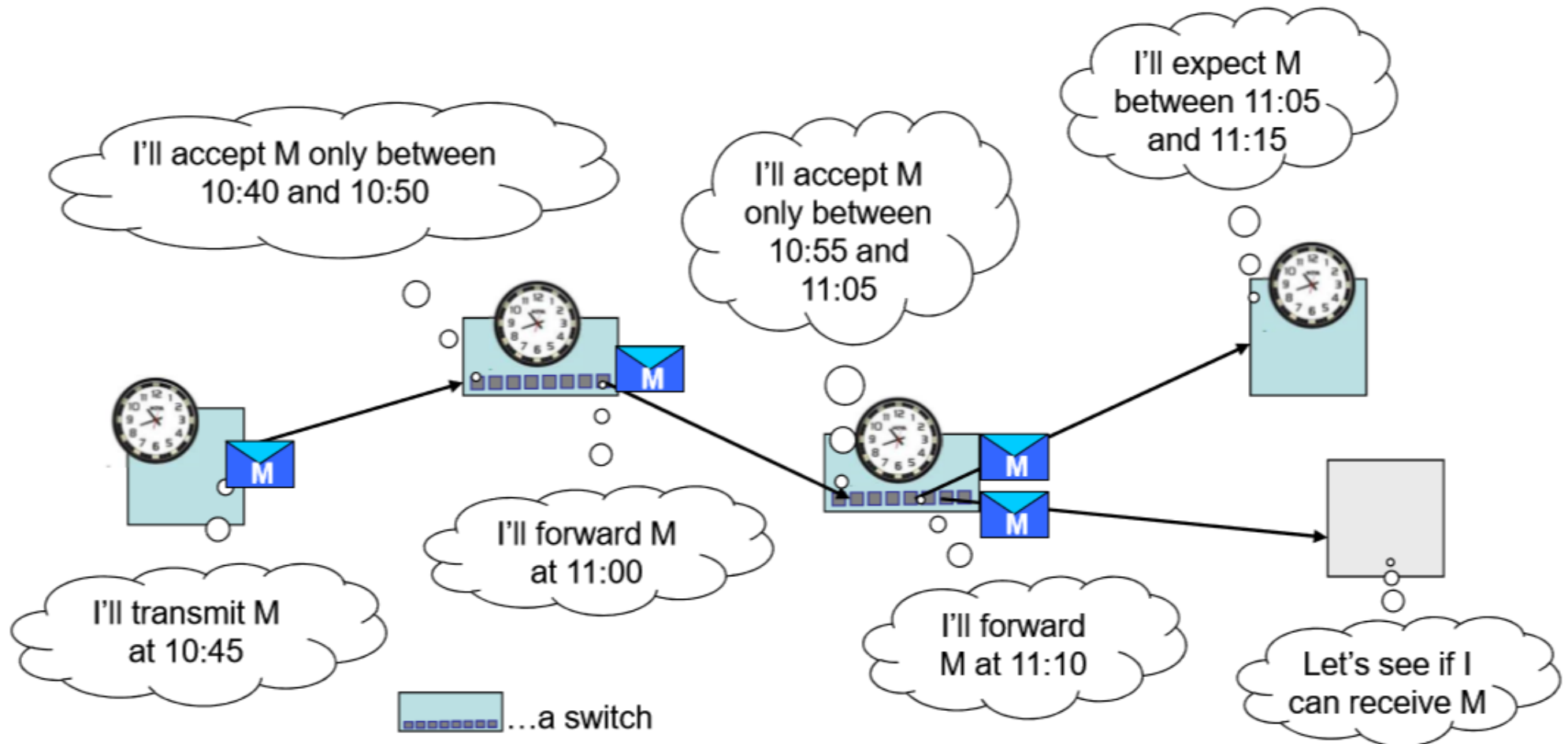
$$J(msg) = Dur_{\max}(msg) - Dur_{\min}(msg)$$

Ethernet с временной синхронизацией

- Опыт эксплуатации AFDX:
 - «+»: использование отработанной технологии Ethernet
 - «-»: сложная схема расчета задержек, наличие очередей на коммутаторах
- Новый стандарт бортовых сетей на базе Ethernet:
 - максимальная предсказуемость задержек за счет глобального расписания использования сети
 - единое время у всех абонентов
 - тактирование времени
 - сквозное расписание передачи данных
 - на каждом такте в каждый канал выдается одно сообщение
 - на коммутаторах нет очередей

Time-triggered Traffic Timing

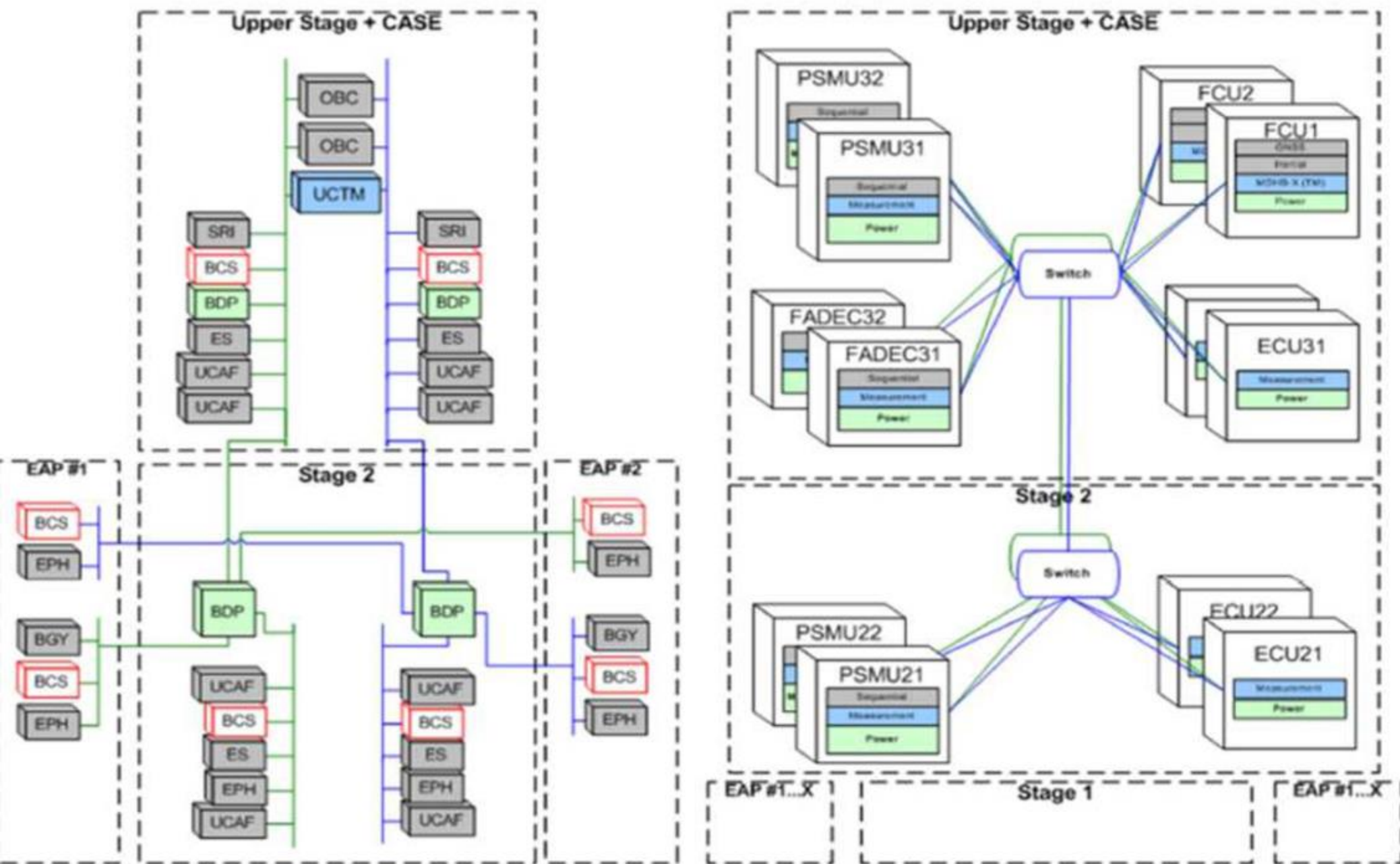
- Full control of timings in the system
- Defined latency and sub-microsecond jitter
- Minimum memory needs



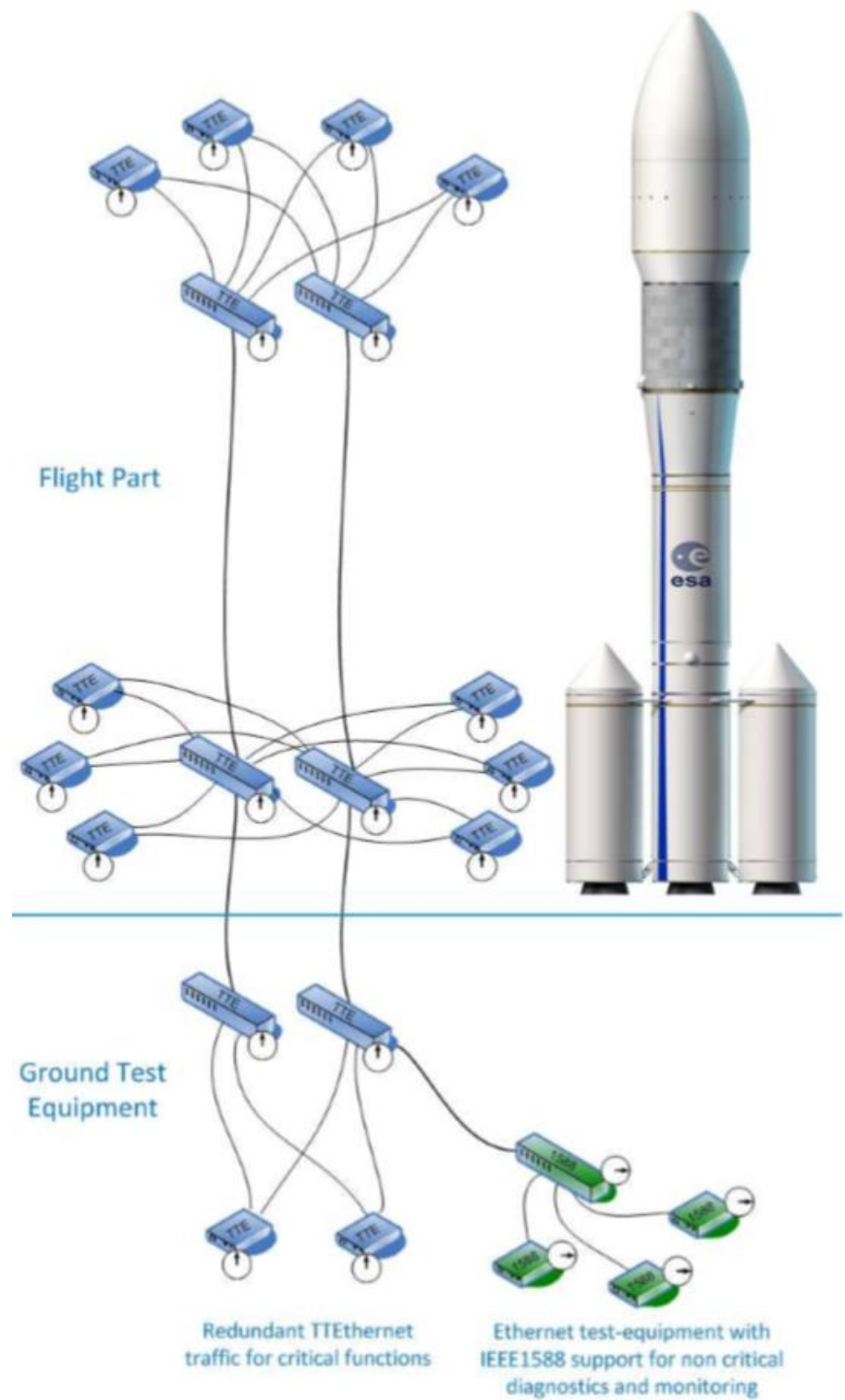
Ariane 5 → Ariane 6: MIL STD-1553B → TTEthernet



Launcher Application – Ariane 6



Интеграция с наземными системами



ТТ-Ethernet: планирование передачи данных по сети

- Обозначения:
 - $(s_i; f_i)$ – директивный интервал отправки сообщения (строится, например, по принципу «отправка один раз в период задачи)
 - $(HO_i; KO_i)$ – интервал отправки сообщения в выходной порт; HO, KO = начало отправки, конец отправки
 - $(HO_i; КП_i)$ – интервал передачи сообщения через сеть; $КП$ = конец приема
- Линейные ограничения:
 - $КП_i = HO_i + \langle \text{фиксир. задержки при передаче через сеть} \rangle$
 - $(HO_i; KO_i) \subset (s_i; f_i)$
 - $(HO_i; KO_i) \cap (HO_j; KO_j) = \emptyset$
 - $(HO_i; КП_i) \cap (HO_j; КП_j) = \emptyset$, при i, j – сообщениях от разных абонентов
- Учет расхождения времени: во всех ограничениях
 - $(HO_i; KO_i) \rightarrow (HO_i - \Delta; KO_i + \Delta)$
 - $(HO_i; КП_i) \rightarrow (HO_i - \Delta; КП_i + \Delta)$

TT-Ethernet: планирование передачи данных по сети

- Вычисление интервалов отправки:
 - Совокупность ограничений передается в решатель систем линейных ограничений
 - Любое найденное решение – корректное расписание выдачи
- Вычисление интервалов ожидания на коммутаторах
 - Интервал ожидания =
Интервал отправки +
фиксир. задержка передачи до коммутатора

Fibre Channel, профиль реального времени

- Базовая схема обмена данными позаимствована из AFDX
 - виртуальные каналы
 - 100% резервирование
- В каждом коммутаторе – несколько таблиц ВК
 - переключение в реальном времени между конфигурациями сети
 - нет избыточного резервирования пропускной способности для поддержки нескольких наборов ВК (втч. в разных режимах ИУС РВ)
 - поддержка статического набора возможных миграций задач
- Управление джиттером отправки целых сообщений
- Система приоритетов → поддержка нерегулярных (апериодических) сообщений
 - низкий приоритет → нет помех для сообщений из ВК
 - высокий приоритет → доставка без задержек (ценой задержки сообщений из ВК)
- Встроенные средства синхронизации времени

Программно-конфигурируемые сети на борту

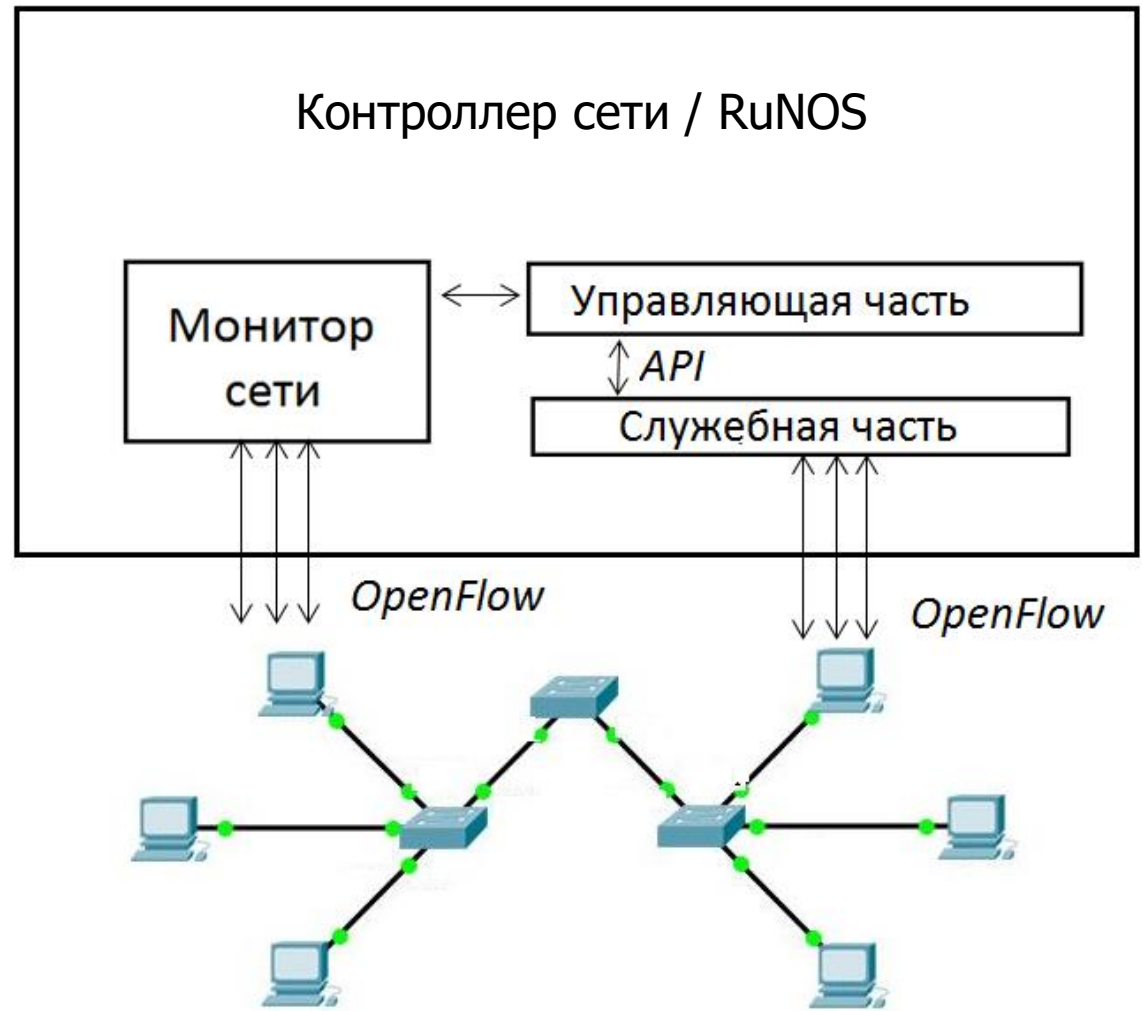
- Бортовая сеть управляется приложением, функционирующем на контроллере
- Синхронизация времени: например, протокол PTP
- Пассивный режим:
 - правила формируются заранее и загружаются на коммутаторы до старта системы
 - большую часть времени работы ИУС РВ коммутаторы функционируют автономно (без обращения к контроллеру)
- Активный режим:
 - контроллер постоянно осуществляет мониторинг обмена данными
 - обнаружены новые потоки данных → новые правила формируются и загружаются в коммутаторы
- Первый шаг внедрения ПКС на борт: схема выделения пропускной способности
 - воспроизведение схемы на основе алгоритма текущего ведра (по образцу AFDX)

Функциональность приложения для контроллера ПКС в ИУС РВ

- Построение маршрутов передачи сообщений между абонентами сети с обеспечением требуемого качества обслуживания, в т.ч. ограниченных задержек передачи данных и ограниченного джиттера
- Динамическая адаптация маршрутов в случае сбоев сети или миграции задач
- Формирование правил для коммутаторов, в т.ч.:
 - правил проверки трафика на соответствие требованиям качества обслуживания
 - правил маршрутизации
 - правил разделения пропускной способности сети между потоками данных

Приложение для контроллера ПКС-сети

- Служебная часть:
 - хранит информацию о наборе ВК
 - удаляет, добавляет, модифицирует ВК и их маршруты
 - (пере)программирует коммутаторы
- Управляющая часть
 - инициализирует предварительно рассчитанные наборы ВК для статических режимов
 - рассчитывает маршруты и параметры ВК в случае переключения в динамический режим (в т.ч. аварийный)
 - реализует жадный алгоритм маршрутизации с элементами перебора
- Сетевой монитор
 - выявляет сбои сети



Востребованность ПКС в ИУС РВ

- Отказоустойчивые системы с миграцией задач
 - сценарии миграции задач при множественных отказах не могут быть просчитаны заранее (комбинаторный взрыв числа сценариев)
 - AFDX: резервирование виртуальных каналов под сбойные режимы непродуктивно расходует пропускную способность сети
 - FC-RT: ограниченный объем памяти коммутаторов под таблицы ВК для сбойных режимов
- Динамическое формирование и смена режимов ИУС РВ
 - реконфигурируемость – «изюминка» ПКС
- Динамическое подключение оборудования и ПО
 - подключаемые в режиме plug-and-play устройства (датчики, устройства связи и т.п.)
 - ПО загружается с подключаемого устройства и выполняется на бортовом вычислителе с архитектурой ИМА в отдельном разделе
 - ВК для обмена бортового вычислителя с подключенным устройством автоматически настраиваются контроллером ПКС
 - пример: многофункциональный беспилотник со сменными датчиками

Спасибо за внимание!