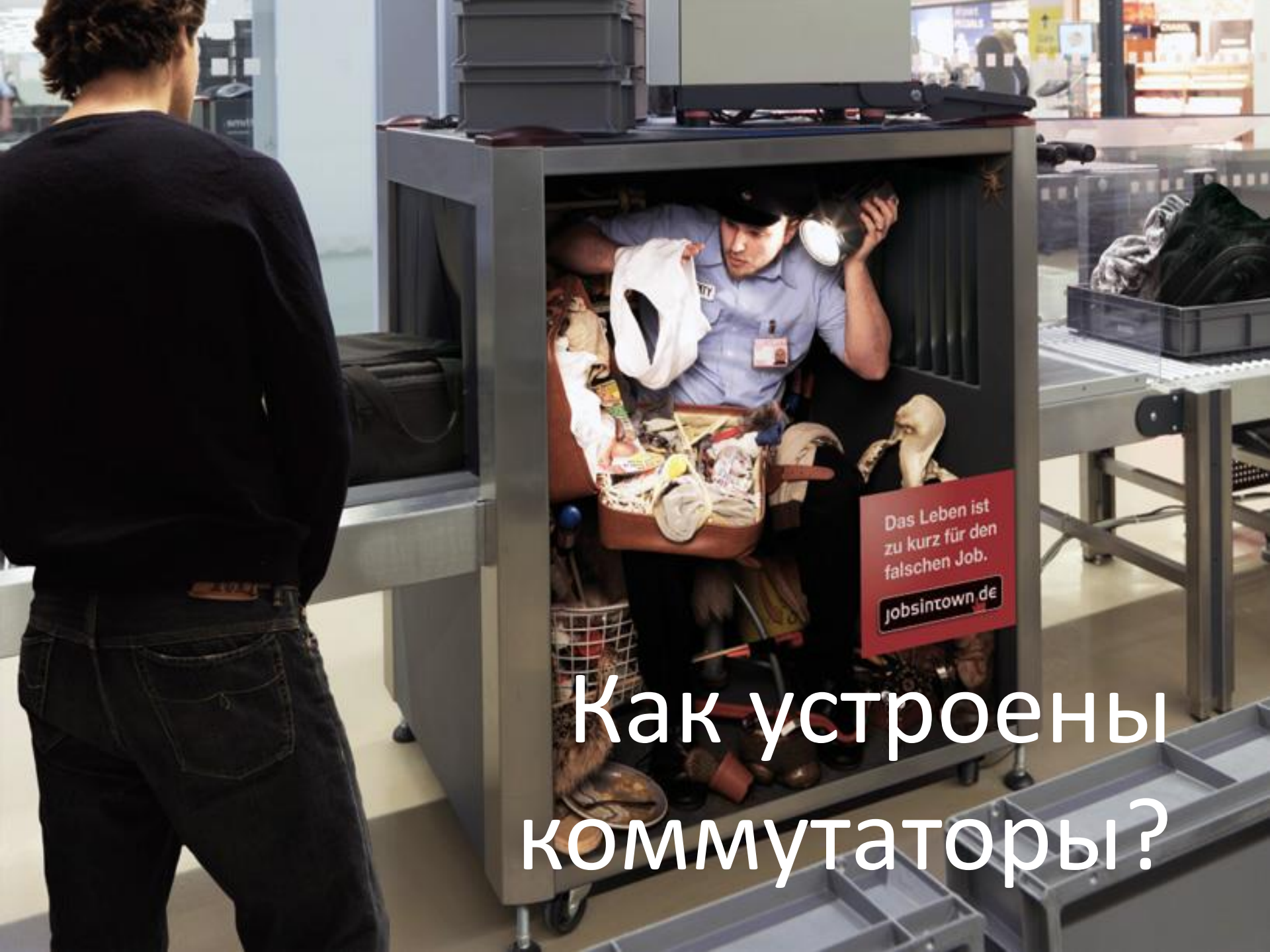
АРХИТЕКТУРА СОВРЕМЕННЫХ ЭВМ

Лекция 8:
Устройство коммутаторов

ВМК МГУ им. М.В. Ломоносова, Кафедра АСВК
Доцент, к.ф.-м.н. Волканов Д.Ю.

План лекции

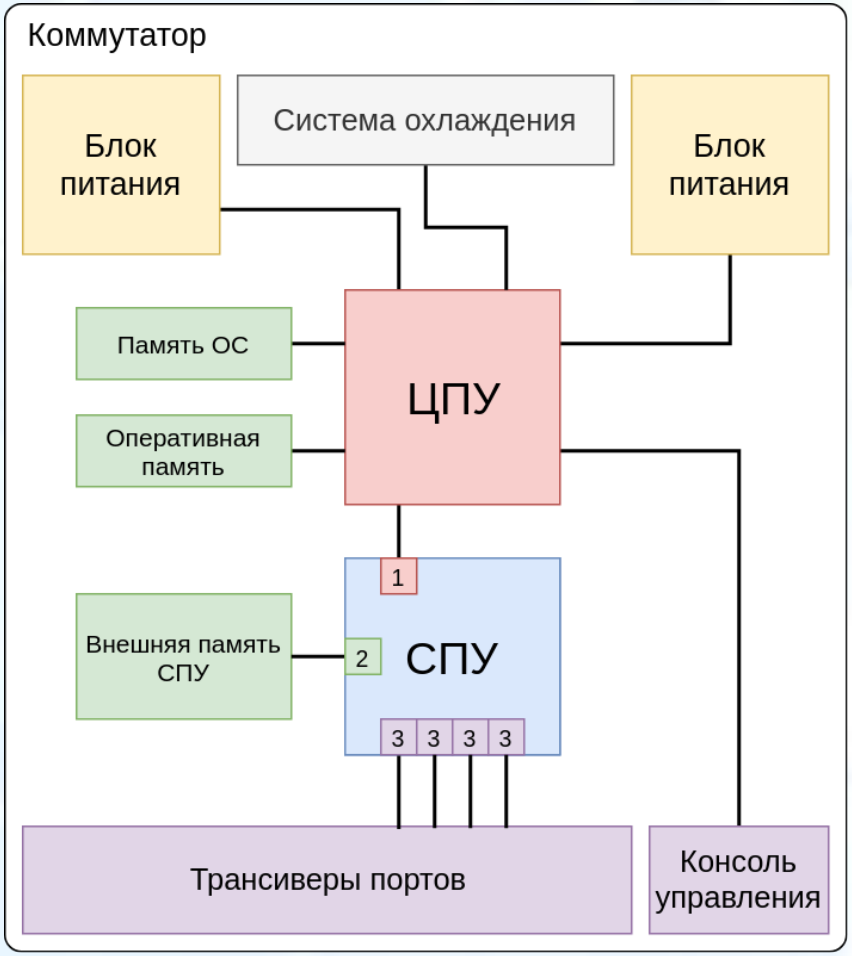
- Общая схема коммутатора
- Поколения коммутаторов
- Основные функции сетевого процессора
- Обзор существующих сетевых процессоров



Как устроены коммутаторы?

Типовое устройство коммутатора

- **Сетевое процессорное устройство (СПУ)** – встроенная полупроводниковая система, оптимизированная для выполнения операций передачи данных
- Функции СПУ:
  - получение пакета;
  - выделение заголовка из пакета;
  - классификация пакета;
  - модификация заголовка и принятие решения о пути следования пакета;
  - управление очередями;
  - передача пакета.



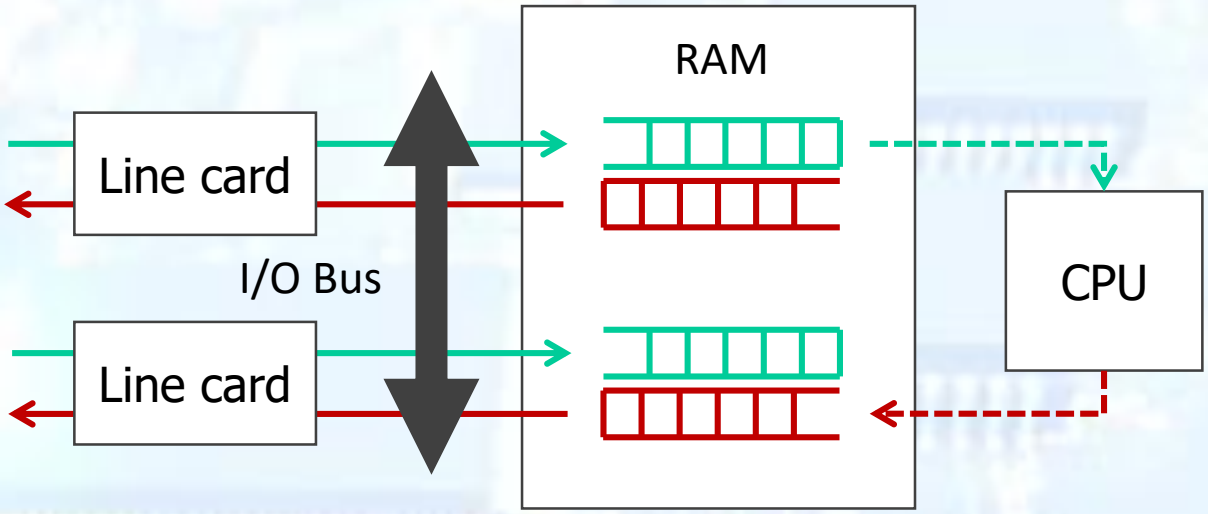
Классификация коммутаторов по поколениям (1)

- Поколения отражают достигнутые характеристики производительности, а не коренные изменение технологии
- Эволюция достигается за счёт изменения баланса между стоимостью и сложностью коммутационного устройства
- Каждое из поколений заняло свою нишу и продолжает использоваться сегодня

Классификация коммутаторов по поколениям (2)

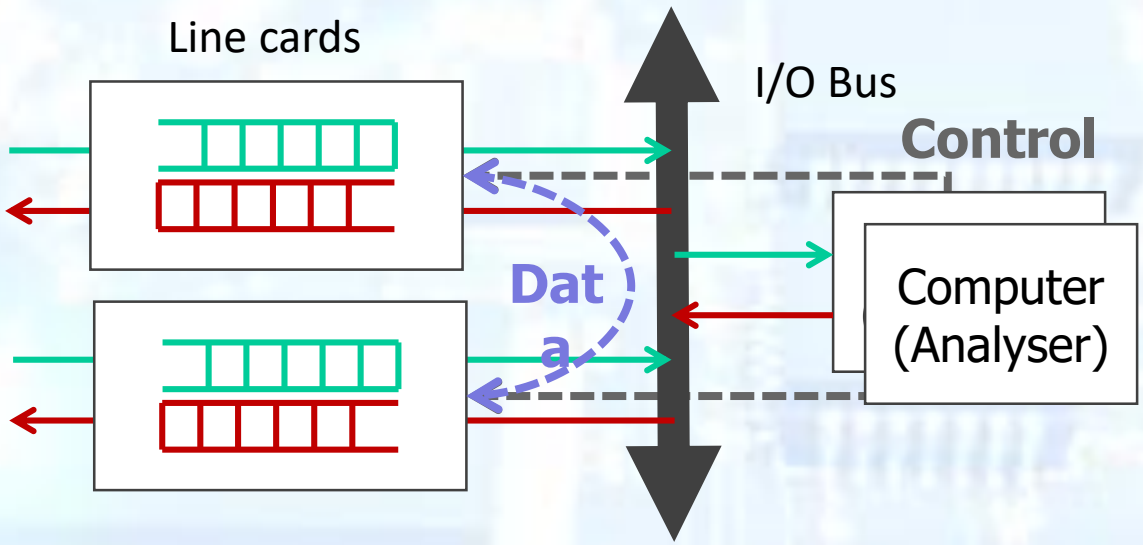
1. Интерфейсные Процессоры Сообщений – SISD компьютеры с несколькими сетевыми интерфейсами *[см. William Yeager]*
2. Распределённая MIMD архитектура с собственными контроллерами на интерфейсах
3. Переход от архитектуры с единой шиной передачи данных к коммутационным матрицам

Первое поколение коммутаторов



- Большинство домашних Ethernet коммутаторов и маршрутизаторов
- Bottleneck'ом может быть шина данных или процессор – в зависимости от типа трафика и производительности этих компонентов

Второе поколение коммутаторов

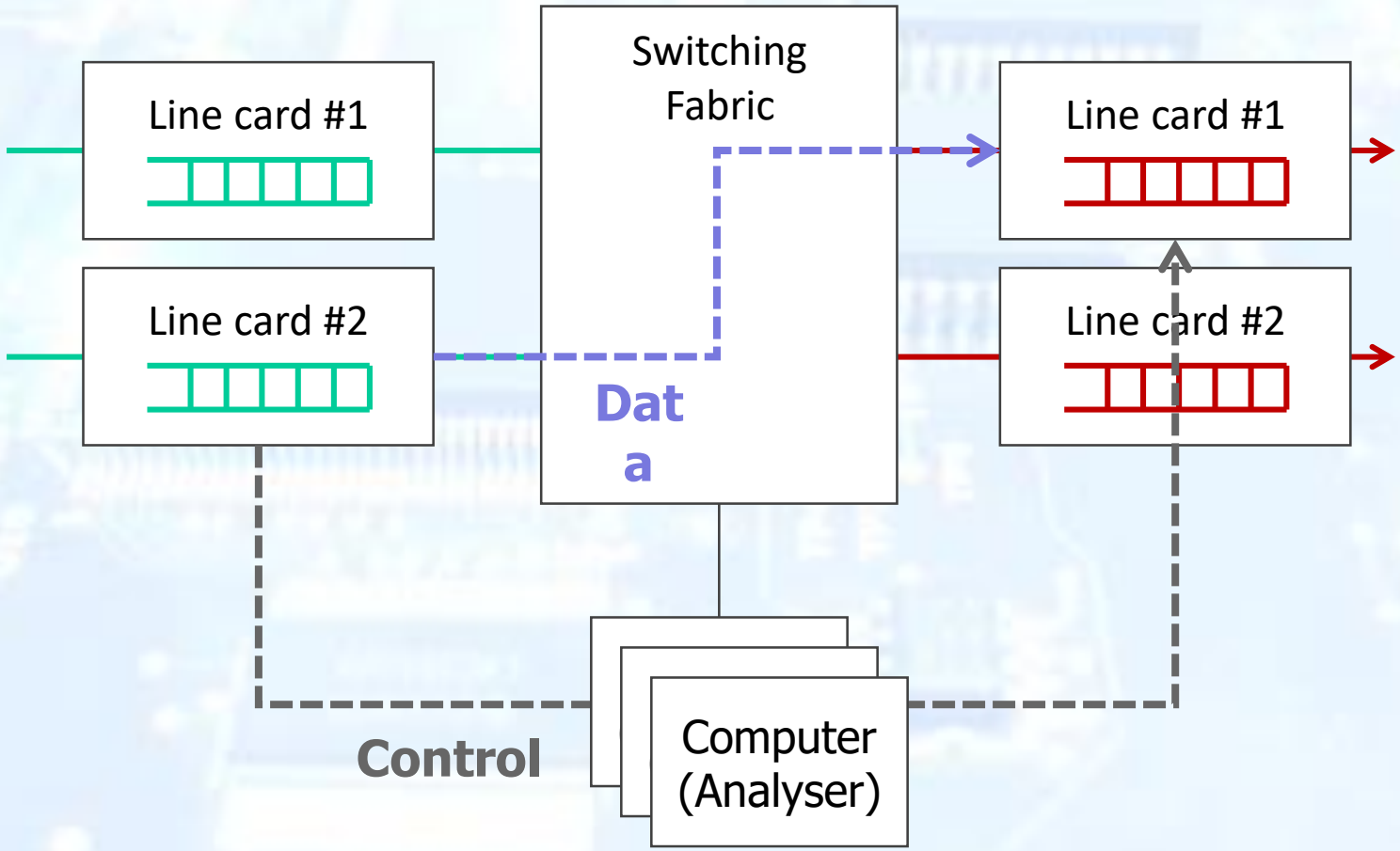


- Умные сетевые интерфейсы с собственным контроллером и встроенной памятью
- Данные пересылаются между картами напрямую, минуя оперативную память
- Слабое место – общая шина передачи данных

К какому поколению относятся решения в духе NFV?

- Самые современные реализации программных коммутаторов и некоторых ***виртуальных сетевых функций*** (VNF) – коммутаторы второго поколения
- Анализ пакетов занимает больше времени, чем их передача между интерфейсами
- Пример: NVWare – CGNAT с пропускной способностью до 80 Gbit/sec

Третье поколение коммутаторов



Третье поколение коммутаторов

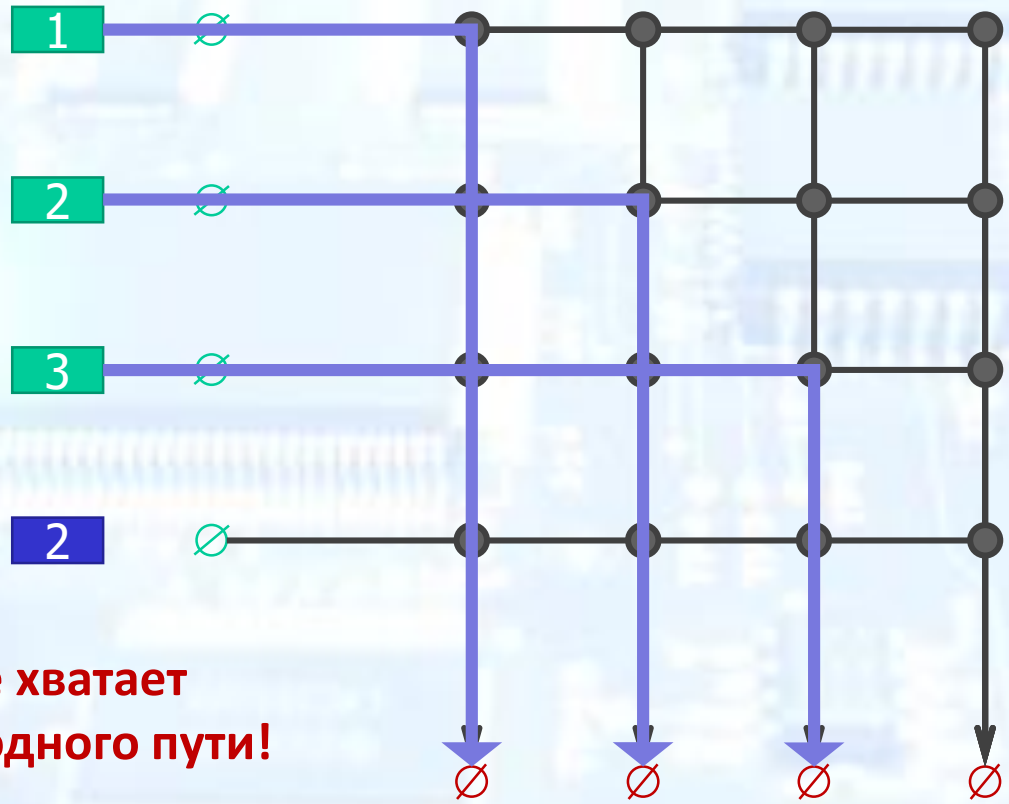
Если во втором поколении для передачи пакетов используется общая шина, то в третьем – ***коммутационная матрица***

Коммутационная матрица способна передавать между N интерфейсами сразу несколько пакетов одновременно (имеет K transmission planes):

- Для шины передачи данных $K = 1$
- Матрица переключателей $N \times N$: $1 \leq k \leq N$ (в зависимости от нагрузки)
- Матрица переключателей $N \times N^2$: $\mathrm{k} = N$ (вне зависимости от нагрузки)

Crossbar Switching Fabric

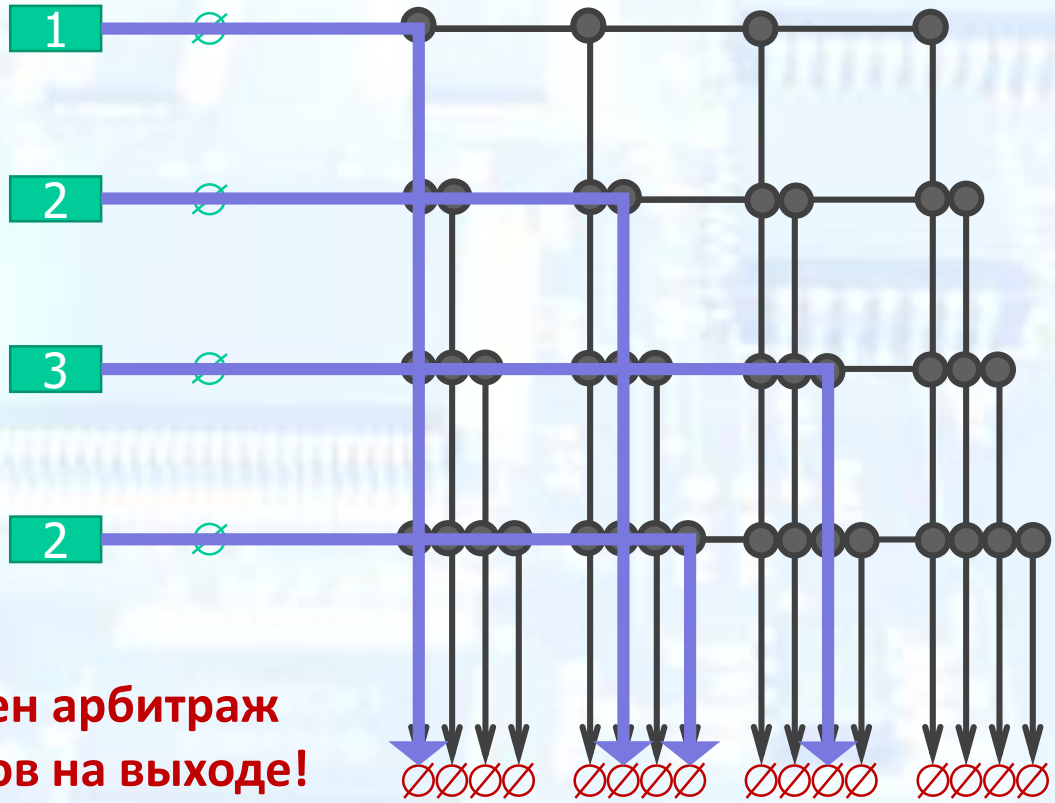
N^2 переключателей



Не хватает свободного пути!

Crossbar Switching Fabric

N^3 переключателей



Нужен арбитраж пакетов на выходе!

Режимы коммутации

Δ_f -- задержка коммутации пакета (FIFO)

Δ_s -- задержка сериализации пакета

Δ_h -- задержка обработки пакета (FIFO)

Store-and-Forward: $\Delta_f = \Delta_s + \Delta_h$

Обработка начинается после получения всего пакета

Работает в терминах пакетов

Cut-Trough [& Fragment free]: $\Delta_f = d + \Delta_h$

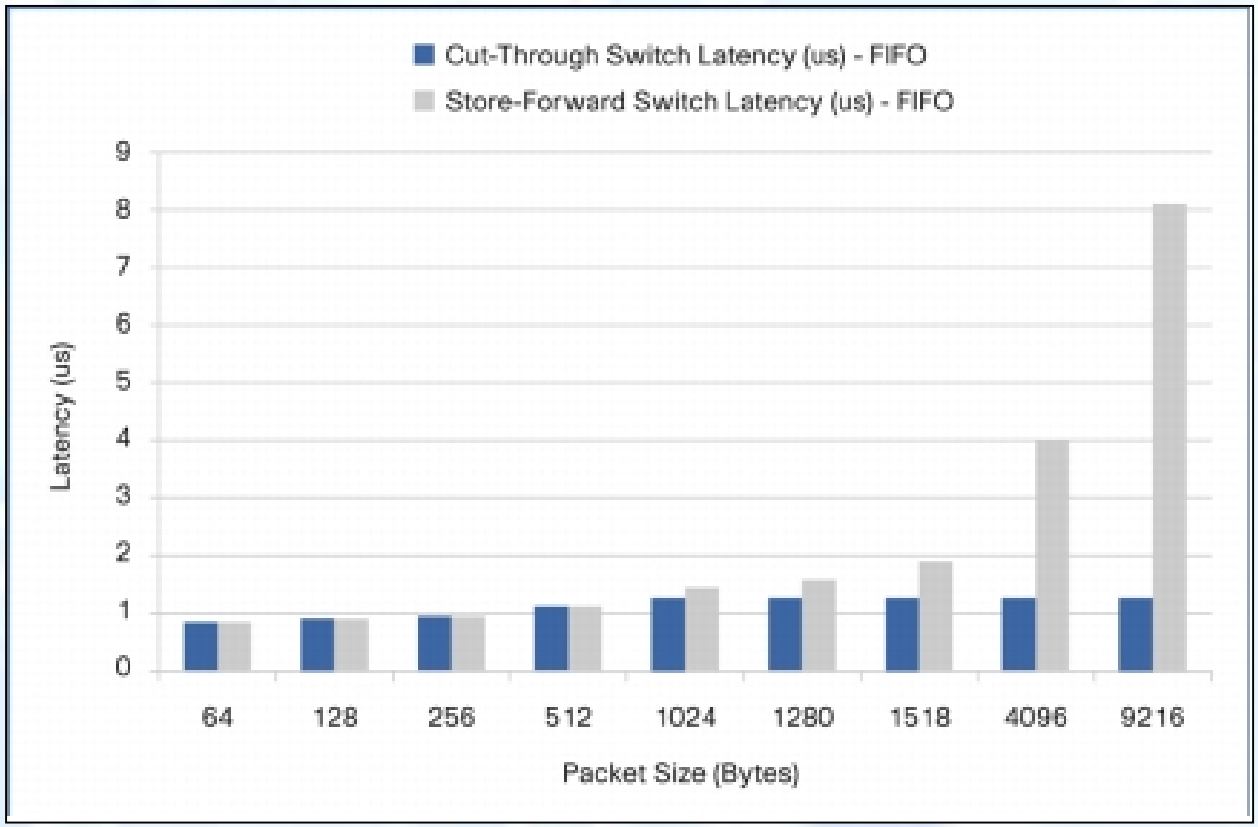
d – время сериализации битов заголовка

Обработка начинается сразу после получения *достаточного количества* битов заголовка пакета

Работает в терминах ***ячеек*** фиксированной длины

14

Режимы коммутации



Store-and-forward vs. Cut-Trough на канале в 10 Gb

15

Зачем нужна буферизация?

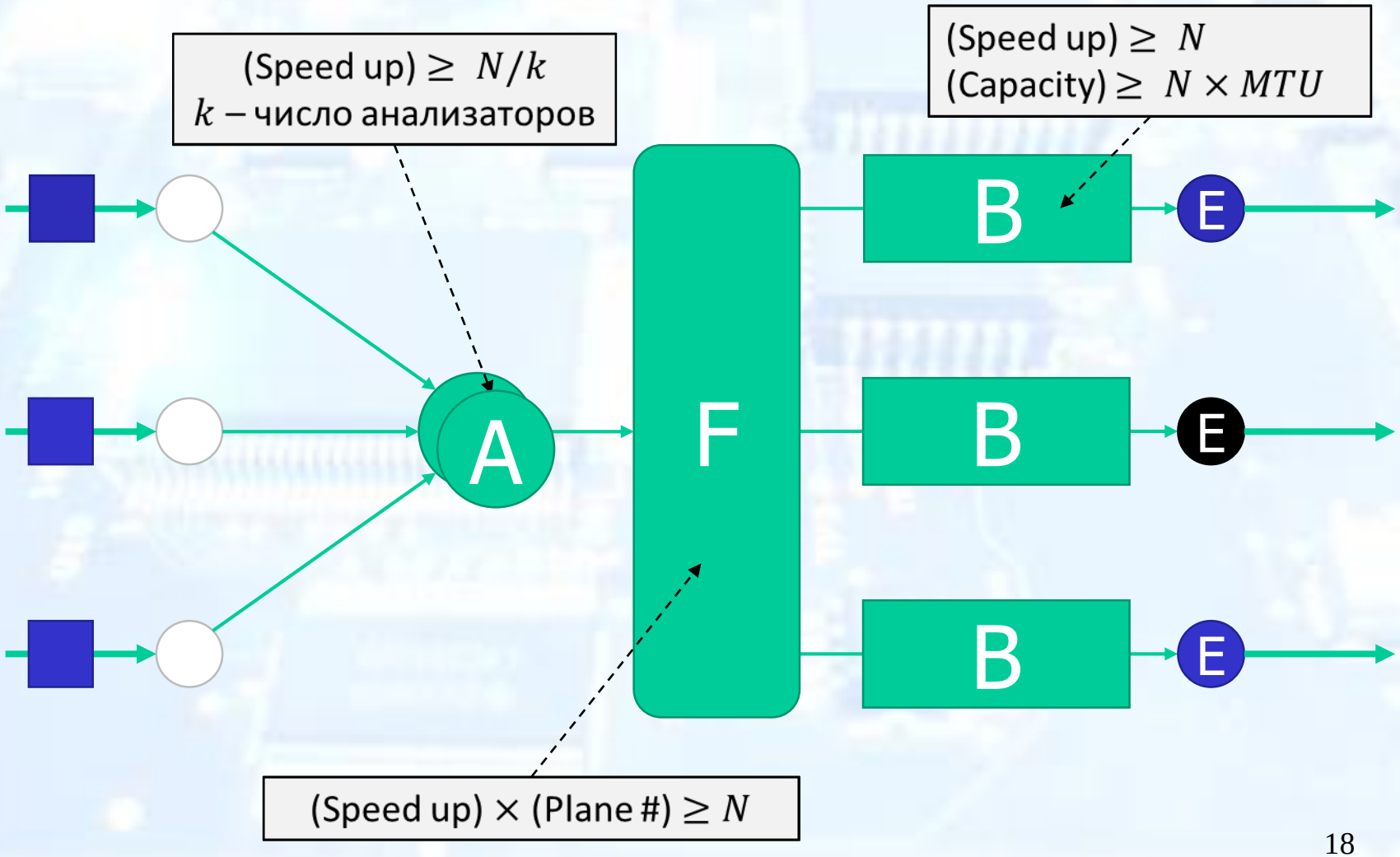
- Линии связи не могут передавать сразу несколько пакетов одновременно
- Что делать если несколько пакетов нужно отправить через одну и ту же линию?
  - Сбросить один из пакетов
  - Сохранить пакет в буфер
- Классическая задача проектирования коммутатора – где расположить буферы, чтобы собрать наилучшее устройство:
  - Максимальная производительность
  - Хорошая масштабируемость
  - Минимальная стоимость

Измерение производительности коммутаторов

RFCs 2544 & 6815

- Если распределение трафика таково, что для каждого выходного порта суммарная скорость поступления данных, которые нужно через него передать, не превосходит скорости подключённой к нему линии связи, то распределение называется ***приемлемым для коммутатора (admissible)***
- Если коммутатор не сбрасывает пакеты при поступлении трафика с приемлемым для него распределением, то говорят, что этот коммутатор удовлетворяет требованиям ***full backplane***

Буферизация на выходе
Output Queuing

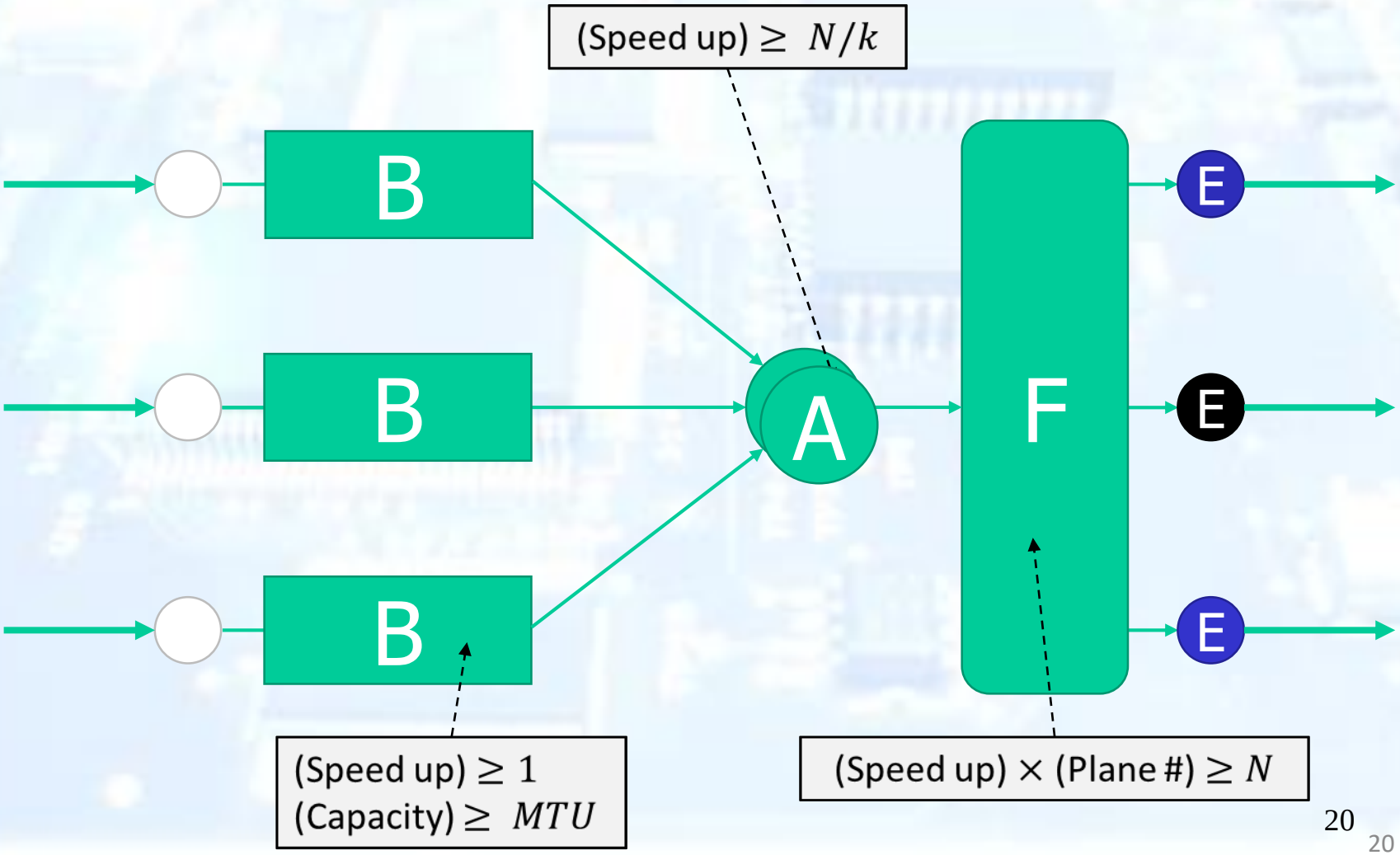


18

Производительность коммутатора

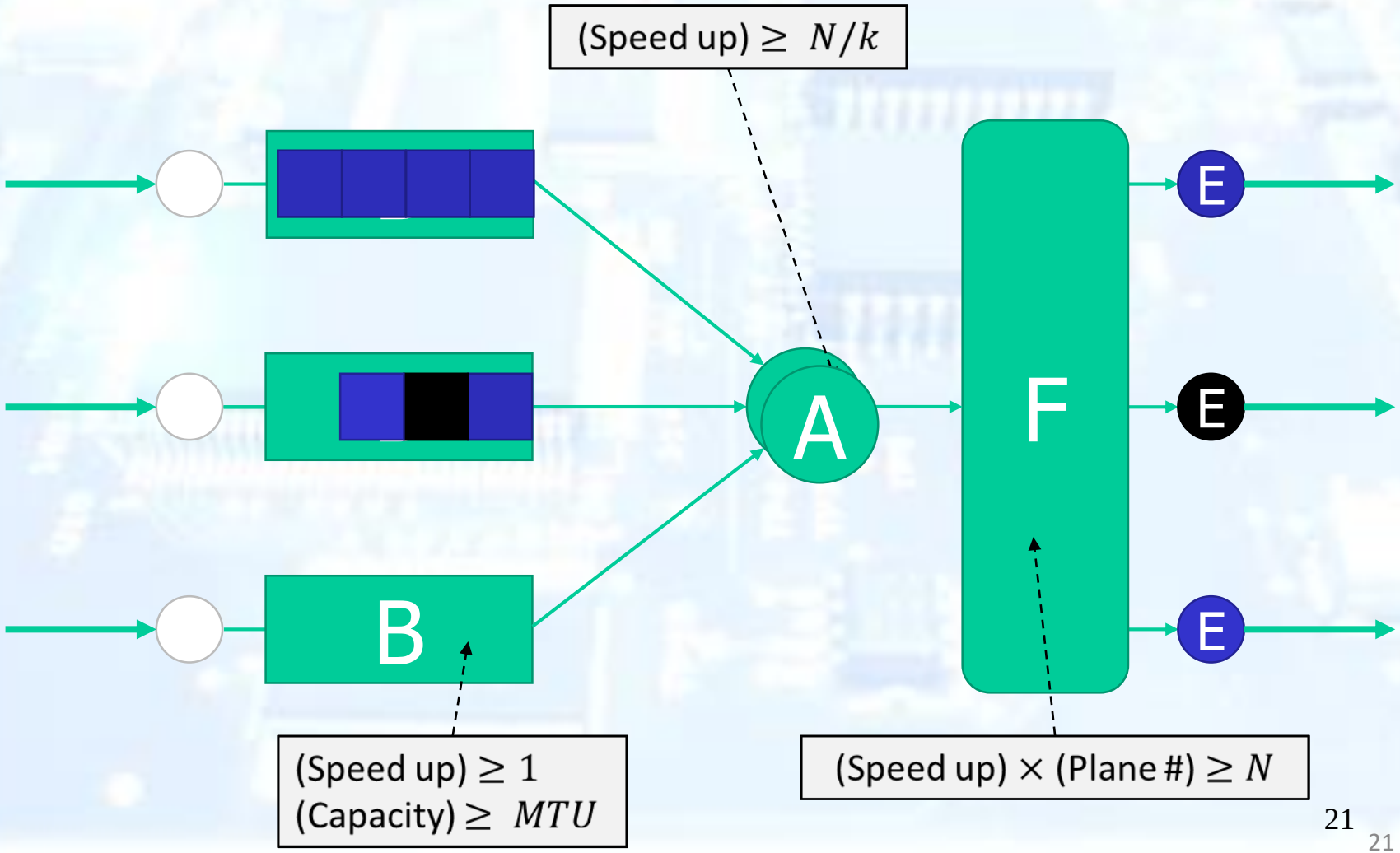
- Анализаторы должны работать с ускорением N/k, где k – количество анализаторов
- Скорость работы матрицы должна превышать скорость линий связи в N раз
- Буферные блоки должны работать с ускорением не менее N, а их объём – составлять не менее N MTU
  - Каковы должна быть частота и ширина шины доступа к памяти, чтобы поддерживать работу современных коммутаторов?

Буферизация на входе
Input Queuing



20

Буферизация на входе
Input Queuing

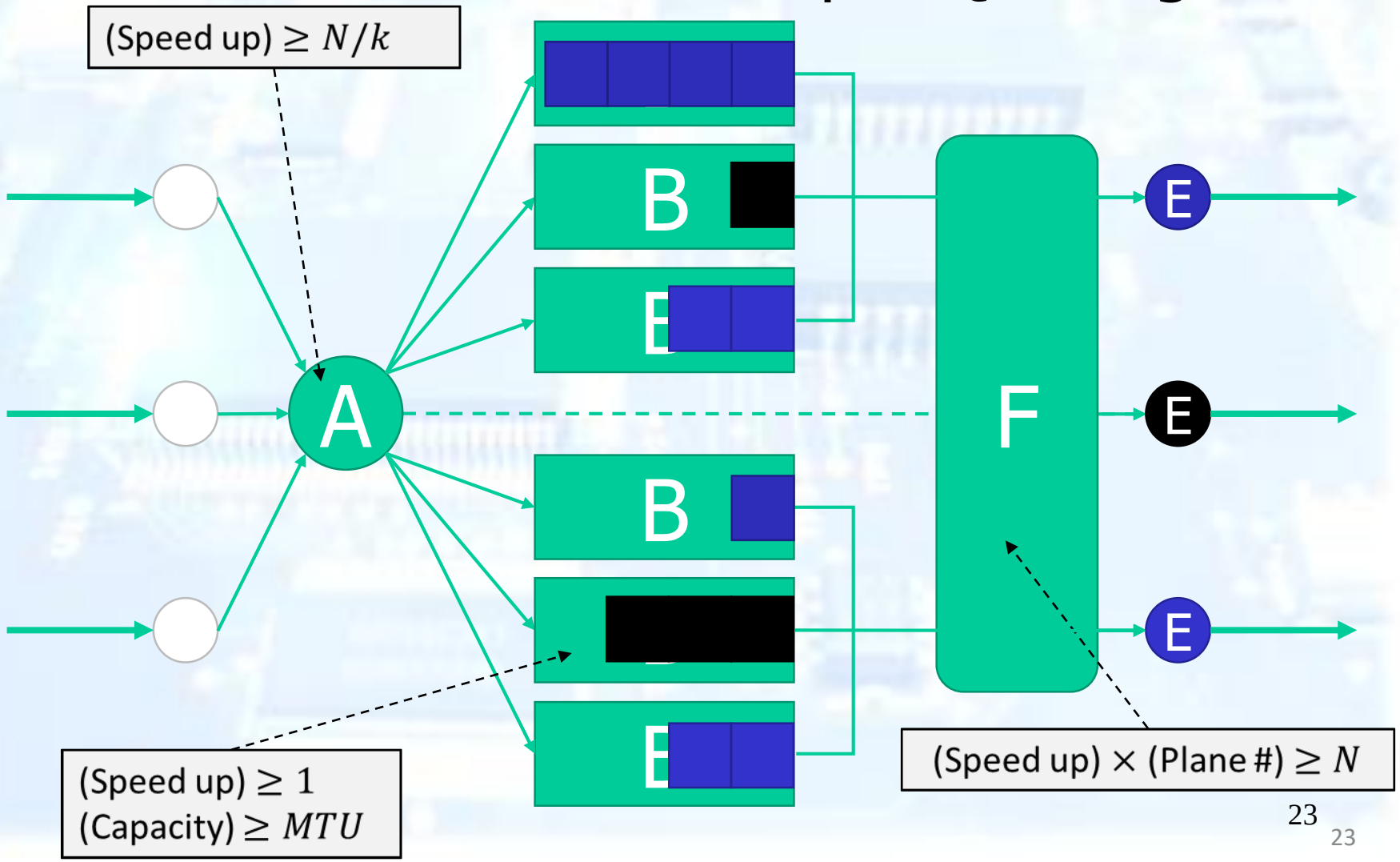


21

Буферизация на входе Input Queuing

- Нет необходимости в сверх-быстрой памяти
- Если пакеты из нескольких входных портов начинают конкурировать за один и тот же вход коммутационной фабрики, возникает блокировка пакетов, находящихся за ними – ***Head Of Line (HOL) Blocking***
- При равномерном распределении маршрутов передачи пакетов производительность IQ-коммутатора равна менее 59% показателя коммутатора с буферизацией на выходе
- M. Karo; M. Hluchyj; S. Morgan ***Input Versus Output Queuing on a Space-Division Packet Switch (1987)***

Виртуальная буферизация на выходе
Virtual Output Queuing



23

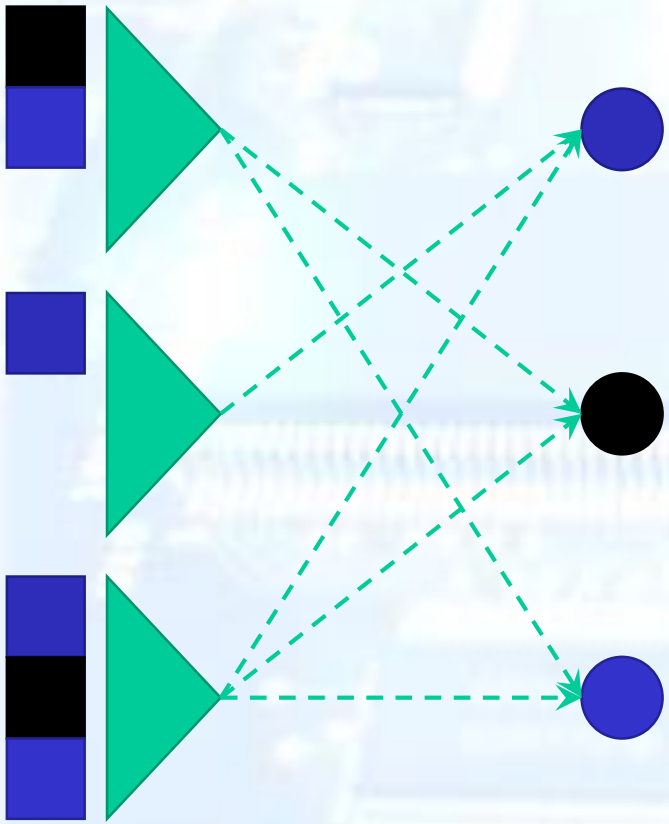
Виртуальная буферизация на выходе
Virtual Output Queuing

N. McKeown A. Mekkittikul V. Anantharam J. Walrand

Achieving 100% Throughput in an Input-Queued Switch

- Проблема HOL blocking не возникает
- Появляется N^2 очередей пакетов
- Коммутационная матрица с N^2 входами не требуется, но необходим алгоритм быстрого арбитража для выбора нужной очереди на каждом из интерфейсов
- Сложные динамические алгоритмы арбитража затруднительно реализовать в аппаратуре

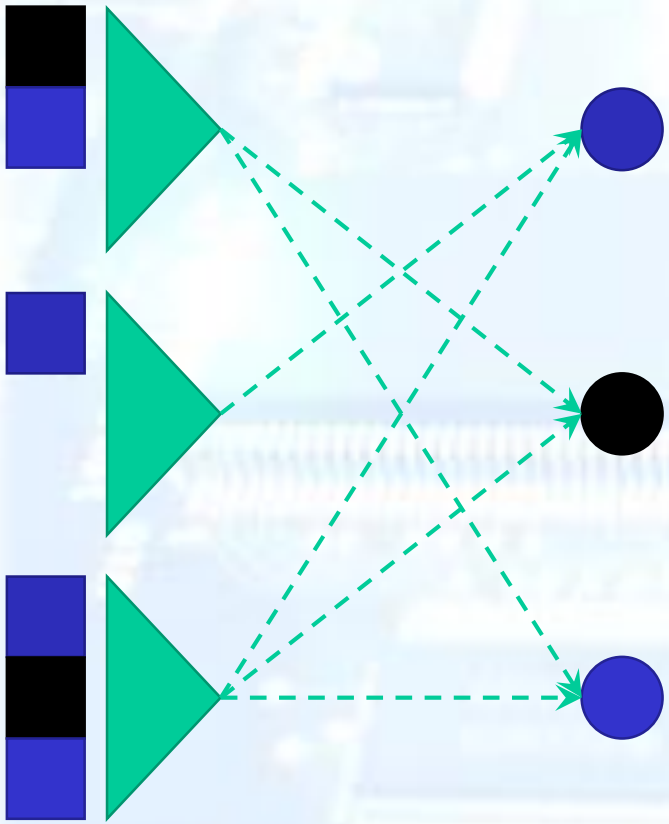
Алгоритмы арбитража коммутационной матрицы



Предположения модели:

- Коммутационная матрица имеет N ***плоскостей***
- Пакеты разбиваются на ***ячейки*** фиксированной длины (cells) при входе и восстанавливаются на выходе из матрицы
- Коммутационная матрица работает по ***тактам*** – на каждом такте она может по одной ячейке из каждого входа и поместить по одной ячейке на каждый свой выход

Алгоритмы арбитража коммутационной матрицы



Найти такой алгоритм арбитража коммутационной матрицы, при котором:

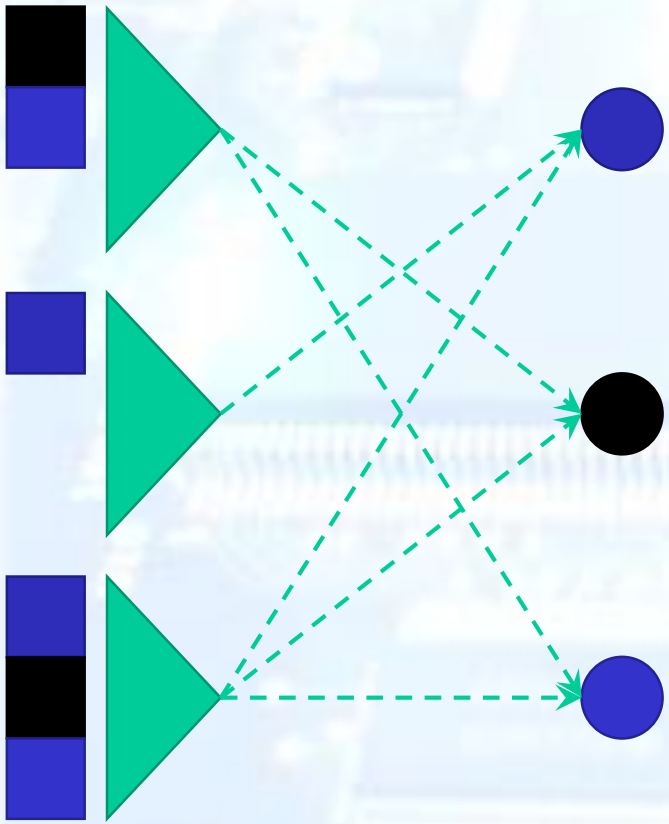
- ***(Производительность)***

Требования full backplane выполнялись бы без ускорения матрицы

- ***(Справедливость)***

Никогда не происходило бы удушение потоков трафика

Алгоритмы арбитража коммутационной матрицы



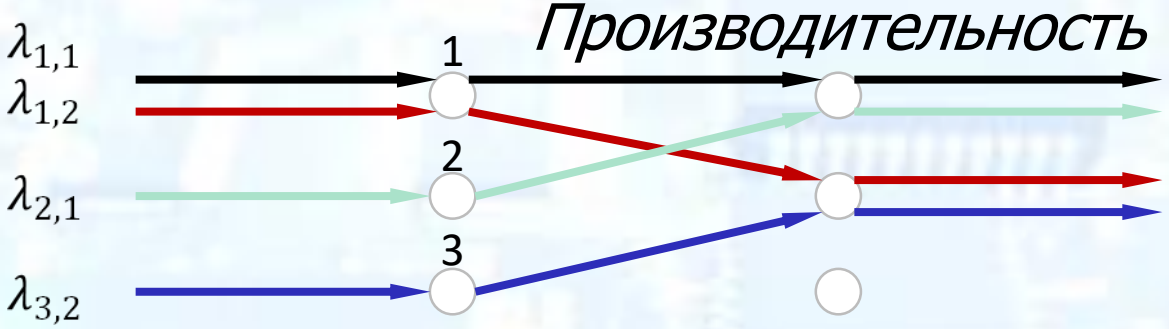
Гипотеза:

•Подходящий алгоритм должен передавать как можно большее количество ячеек на каждом такте работы матрицы – задача ***поиска наибольшего паросочетания***

Гипотеза неверна – алгоритм не обладает ни высокой производительностью, ни справедливостью планирования

Неприменимость алгоритма для наибольшего паросочетания

Производительность



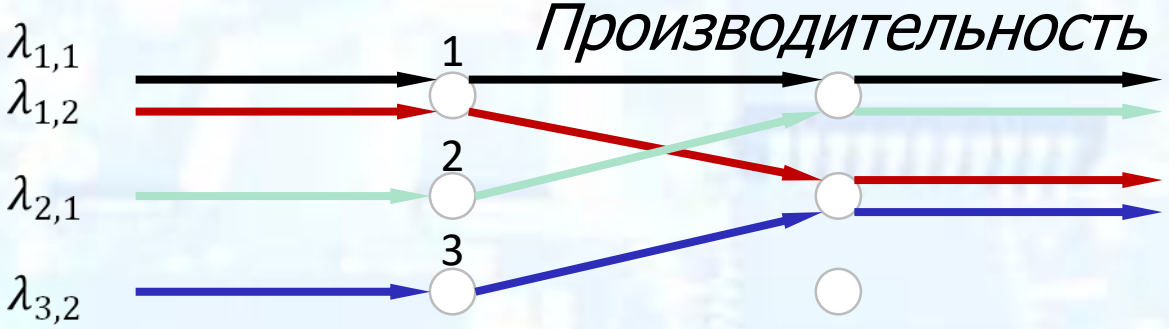
(A1) Пусть пропускная способность каждой из линий связи равна 1, скорость каждого из потоков

$$\lambda_{1,1} = \lambda_{1,2} = \lambda_{2,1} = \lambda_{3,2} = \frac{1}{2} - \delta$$

(A2) Пусть в ситуациях, когда существует несколько наибольших паросочетаний алгоритм выбирает один из них с равной вероятностью

Неприменимость алгоритма для наибольшего паросочетания

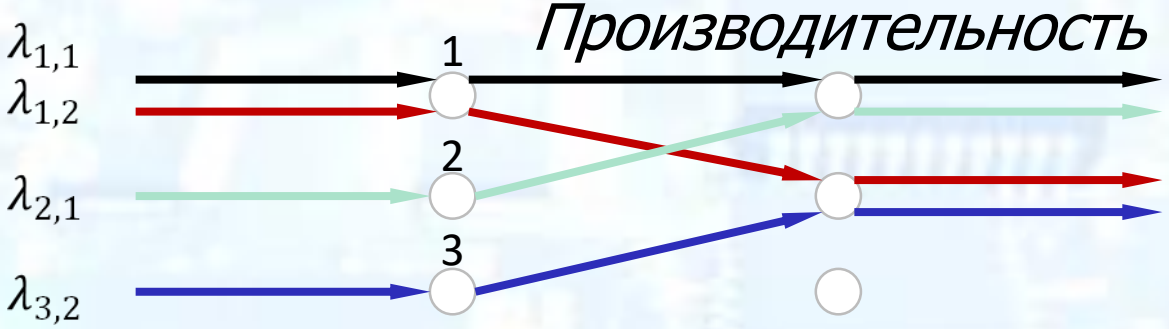
Производительность



- Расчитаем максимальную скорость передачи для входа 1:
- Пусть потоки $\lambda_{2,1}$ и $\lambda_{3,2}$ готовы к передаче:
  - Из **(A1)** вероятность такого события $(1/2 - \delta)$^2
  - Тогда есть три наибольших паросочетания:
    $\lambda_{1,1}\&\lambda_{3,2}$; $\lambda_{1,2}\&\lambda_{2,1}$; $\lambda_{2,1}\&\lambda_{3,2}$
  - Из **(A2)** вероятность выбора первого входа 2/3
- Если хотя бы один из $\lambda_{2,1}$ и $\lambda_{3,2}$ не готов к передаче:
  - Алгоритм выбирает вход 1 с вероятностью 1

Неприменимость алгоритма для наибольшего паросочетания

Производительность



- По формуле полной вероятности наибольшая скорость передачи данных, поступивших на коммутатор через вход 1:

$$\frac{2}{3}\left(\frac{1}{2}-\delta\right)^2+\left(1-\left(\frac{1}{2}-\delta\right)^2\right)=1-\frac{1}{3}\left(\frac{1}{2}-\delta\right)^2$$

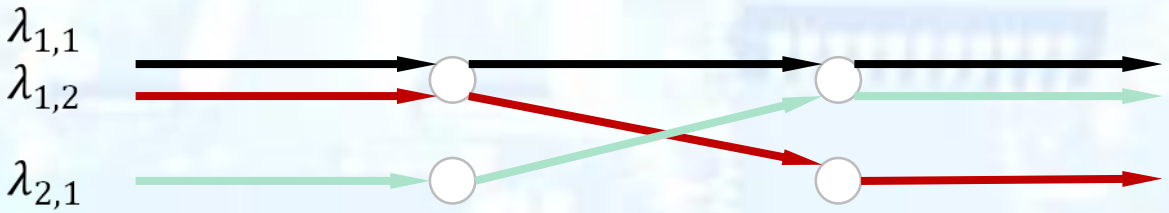
- Требования full backplane нарушаются, если скорость поступления данных на вход 1 превышет наибольшую скорость передачи:

$$1-2\delta>1-\frac{1}{3}\left(\frac{1}{2}-\delta\right)^2$$

- Выполнено при $\delta < 0.0358$

Неприменимость алгоритма для наибольшего паросочетания

Справедливость



- Пусть скорость поступления данных из потоков равна пропускной способности канала $\lambda_{1,1} = \lambda_{1,2} = \lambda_{2,1} = 1$
- Алгоритм для поиска наибольшего паросочетания всегда будет выбирать потоки $\lambda_{1,2}$ и $\lambda_{2,1}$
- Поток $\lambda_{1,1}$ будет испытывать ***удушение (starvation)***

Алгоритм арбитража Oldest Cell First

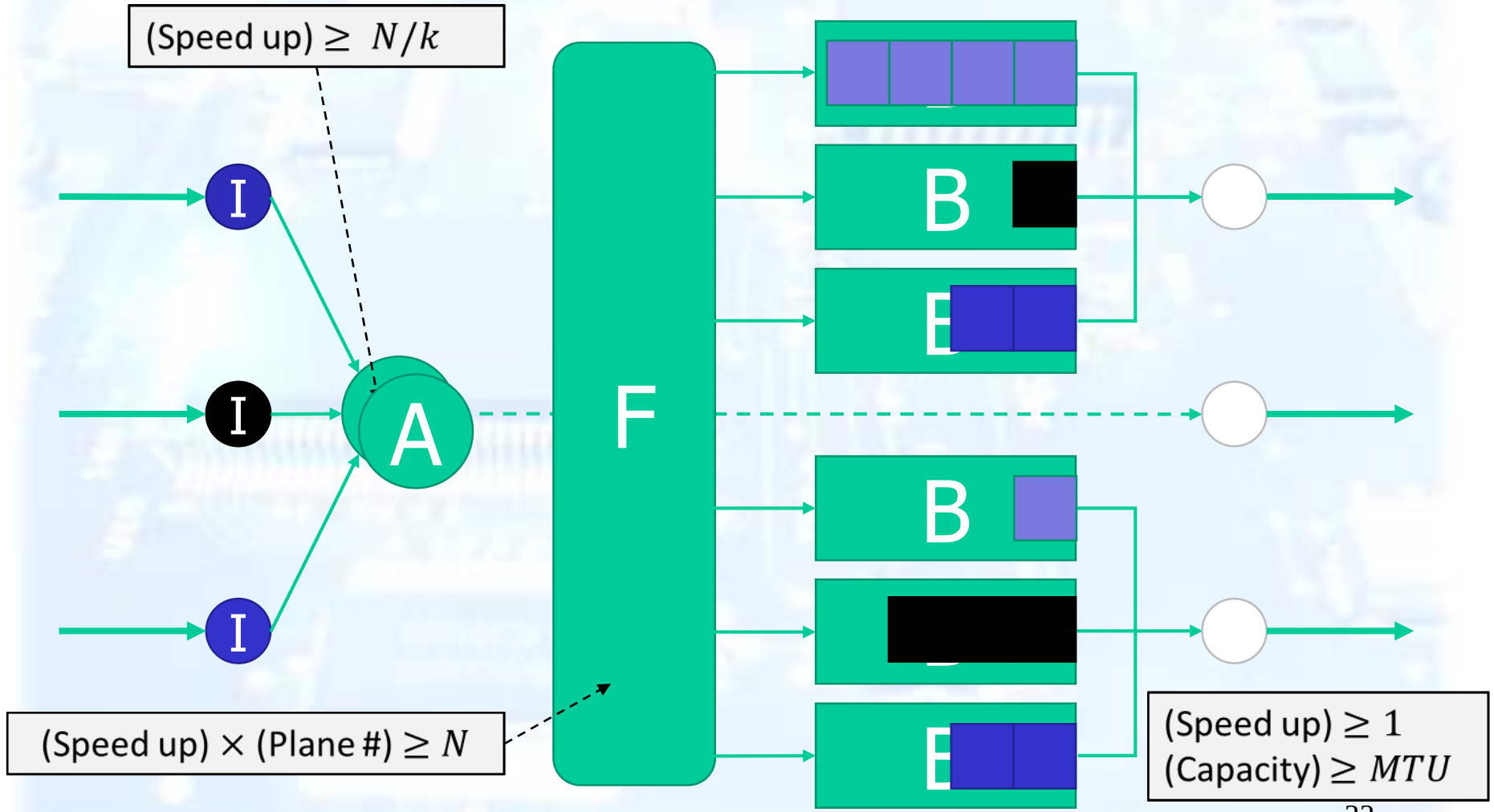
- Каждой из очередей присваивается собственный весовой коэффициент – количество тактов, когда находящаяся в ведущей позиции ячейка не была выбрана
- Алгоритм арбитража выбирает паросочетания таким образом, чтобы максимизировать сумму весов выбранных им очередей

Недостатки:

- Высокая сложность реализации

32

Множественная буферизация на выходе
Multiple Output Queuing



33

Множественная буферизация на выходе
Multiple Output Queuing

- Не нужен алгоритм арбитража коммутационной матрицы
- Необходимо усложнение логики для распределения пакетов по очередям на выходе из коммутационной матрицы и дополнительный планировщик пакетов для выборки данных из этих очередей

34

Практика построения коммутационных устройств

When the rubber meets the road…

- Модели редко используются в чистом виде
- Производители пытаются искать компромиссы между стоимостью и утилизацией каналов под различной нагрузкой
- На сегодняшний день наиболее распространены модели Combined Input-Output Queuing (CIOQ)
- Вместо полноценной коммутационной матрицы часто используются её упрощения – knockout switches или сети из переключателей

Архитектура сети Классический подход vs ПКС

Классический подход подразумевает последовательную передачу данных через ряд «узлов», каждый из которых **повторяет идентичные сложные вычисления,** выполняя и управление сетью (Control Plane), и передачу данных (Data Plane).

SDN

ПКС – вынос задач управления в единый центр: разделение уровней управления сетью и передачи данных, что позволяет решить «классические» проблемы.

Недостатки:

–Значительное увеличение CAPEX и OPEX.

–Снижение эффективности использования ресурсов сети.

–Высокие требования к количеству и квалификации персонала.

–Зависимость от конкретных производителей.

Преимущества ПКС:

✓Сокращение CAPEX/OPEX

✓Централизация управления

✓Ускорение вывода на рынок новых сервисов

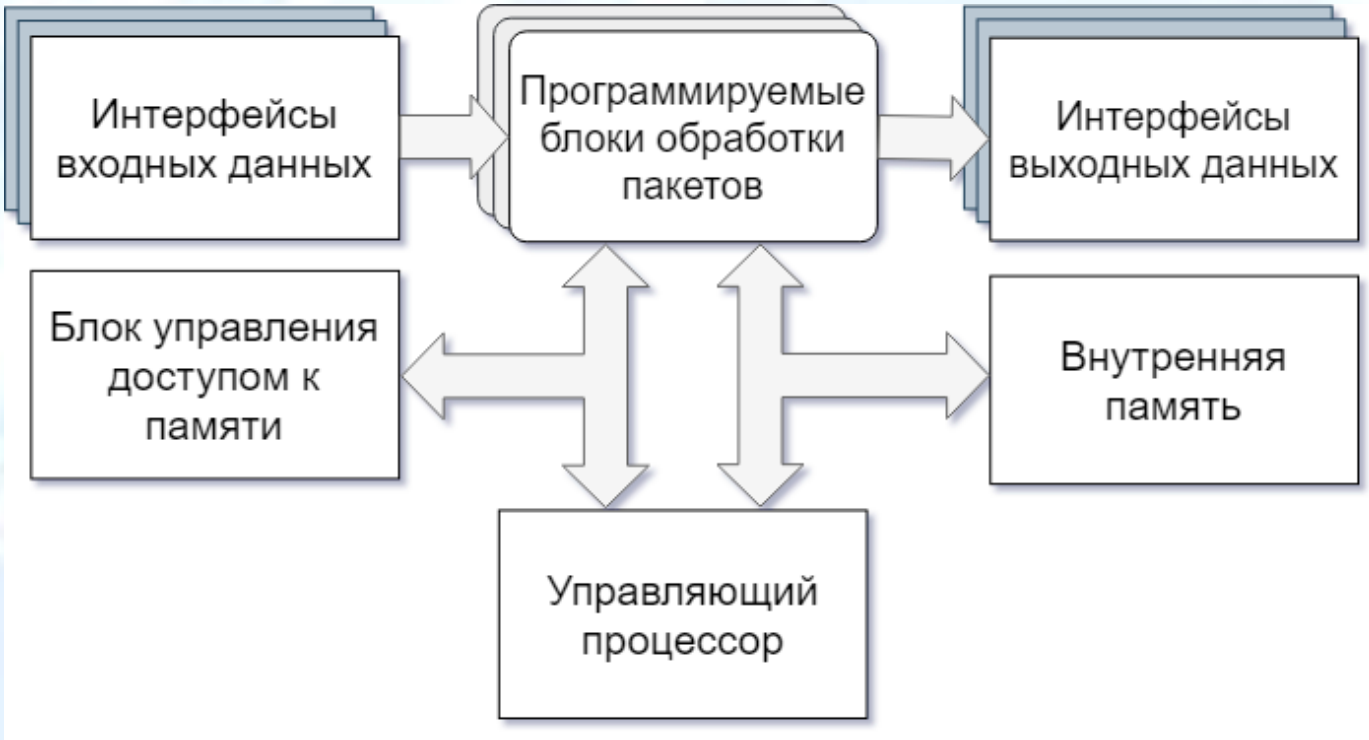
✓Использование стандартного оборудования

✓Повышение эффективности использования каналов связи

Ключевые особенности технологии ПКС

- Изоляция контура управления от контура передачи данных
- Унифицированный интерфейс для приложений управления
- Унифицированный интерфейс для контура передачи данных
- Централизация управления
  - понятие состояния сети
  - резкое сокращение времени сходимости
  - топология на L2 и L3

Обобщенная архитектура СПУ



14 байт	20 байт	80 байт	
Ethernet	IP заголовок	TCP заголовок	Полезная нагрузка

Рассматриваемые СПУ

- Barefoot Tofino
- Barefoot Tofino 2
- Mellanox NP-5
- Mellanox SwitchX-2
- Huawei ENP
- Innovium Teralynx 7
- Nokia FP4
- Cisco NPU
- Juniper Q5
- Broadcom Tomahawk 3
- Broadcom Trident 3

Критерии обзора СПУ

- Год выпуска
- **Программируемость СПУ**
- **Тип СПУ**
- **Ключевые особенности архитектуры** (6 критериев)
- Характеристики кристалла (3 критерия)
- Тип интерфейса к ЦПУ
- Управляющий процессор на кристалле (если предусмотрен)
- Производительность
- Допустимые конфигурации сетевых интерфейсов
- Стоимость

40

Программируемость СПУ

- **Устройства с фиксированной функциональностью**
  - Фиксированный стек протоколов и программа обработки пакетов
- **Конфигурируемые устройства**
  - Загрузка программы обработки пакетов в рамках предопределенных протоколов передачи данных

Broadcom Tomahawk

- **Программируемые устройства**
  - Определение новых протоколов передачи данных в загружаемой программе

Barefoot Tofino, Broadcom Trident, Mellanox NP-5, Cisco NPU…

Подходы к построению коммутаторов

Коммутатор на ядрах общего назначения

Достоинства

- Гибкость настройки и модификации функциональности
- Простота внесения изменений

Недостатки

- Плохое соотношение стоимость / производительность
- При скорости выше 10 Гб/сек потеря пакетов 5-6 %
- Высокое энергопотребление

Схемы специального назначения

Достоинства:

- Наилучшее соотношение стоимость/производительность
- Возможность достижения высокой скорости обработки данных
- Низкое энергопотребление

Недостатки:

- Высокая сложность программирования сервисов
- Необходимость наличия глубокой экспертизы в разработке сетевых устройств
- Необходимость полной переделки при переходе на новый стек протоколов

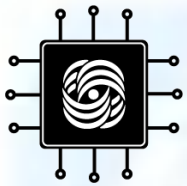
Сетевые процессоры (NPU)

Достоинства:

- Гибкость в программировании новых сервисов
- Промежуточное положение по соотношению стоимость /производительность
- Низкое энергопотребление
- Возможность быстрого развития линейки устройств
- Длительное время нахождения на рынке
- Соответствие имеющемуся опыту разработки в РФ

Недостатки:

- Длительный цикл разработки



Типы программируемых СПУ

- **Многопроцессорная ИС на базе процессоров общего назначения**
  - Гибкость программирования
  - Невысокая скорость обработки пакетов
- **ASIC**
  - Аппаратная реализация основных функций СПУ (низкая гибкость программирования)
  - Высокая скорость обработки пакетов
- **Сетевой процессор**
  - Специализация к задачам обработки пакетов
  - Компромисс по возможностям программирования

Cisco NPU

Barefoot Tofino, Innovium Teralynx, Broadcom Trident, Broadcom Tomahawk

Mellanox NP-5, Huawei ENP, Juniper Q5, …

Сравнение СПУ по общим критериям

СПУ	Программируемость	Производительность	Интерфейсы	Стоимость	Стоимость коммутатора
Mellanox NP-5	+, Си	240 Гбит/с	До 100 GbE	1000$	?
Mellanox SwitchX-2	+, Си	До 2 Тбит/с	До 56 GbE	Нет данных	15000$
Huawei ENP	+	480 Гбит/с	До 100 GbE	Нет в продаже	6000$
Nokia FP4	+	2,4 Тбит/с	До 400 GbE	Нет данных	Нет данных
Barefoot Tofino	+, P4	6,5 Тбит/с	До 100 GbE	Нет данных	8000$

Сравнение характеристик кристалла СПУ

СПУ	Тех. процесс	Интерфейсы с ЦПУ
Mellanox NP-5	28 нм	PCI Express, Ethernet 1×10GbE
Mellanox SwitchX-2	16 нм	PCI Express Gen3
Huawei ENP	16 нм	Нет данных
Nokia FP4	16 нм	Нет данных
Barefoot Tofino	16 нм	4×PCI Express Gen3, 1 или более Ethernet до 100 GbE

Сравнение ключевых особенностей архитектур конвейеров СПУ

СПУ	Состав конвейера	Типы ядер СПУ
Mellanox NP-5	Конвейер из 5 функционально специализированных стадий	Специализированные векторные процессоры
Mellanox SwitchX-2	Нет данных	Нет данных
Huawei ENP	Явной структуры конвейера нет, процессоры объединены в группы, которые могут параллельно выполнять разные задачи	Процессоры со специализированными инструкциями для обработки заголовков Ethernet, IP
Nokia FP4	Нет данных	Нет данных
Barefoot Tofino	Конвейеры входной и выходной обработки, разделенные коммутационной матрицей. Функционально специализированные стадии трех типов	Специализированные процессоры для каждого из типов стадий

Организация конвейера

Два основных подхода:

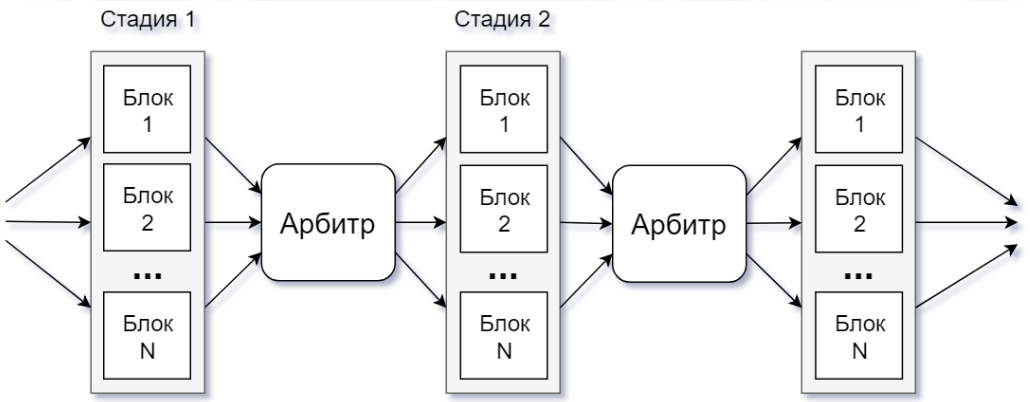
- процессорные ядра общего назначения внутри стадий — Cisco NPU
- специализация ядер к функциям обработки пакетов — Barefoot Tofino, Mellanox NP-5, Huawei ENP, …

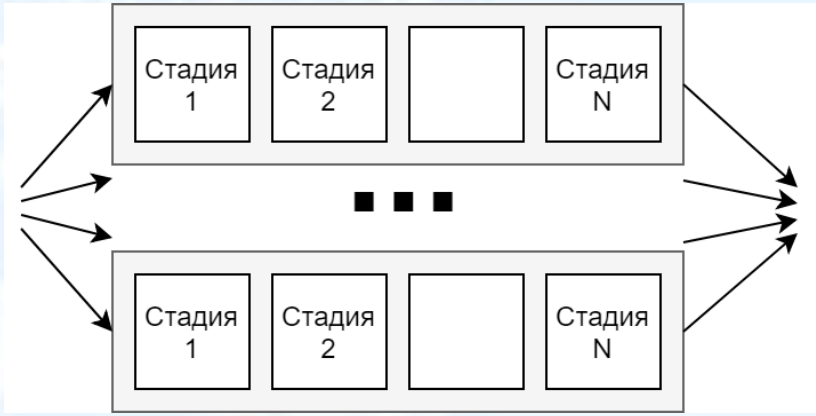
Механизм “разворота” пакетов:

- повторный проход пакета по конвейеру — Barefoot Tofino, Mellanox NP-5, Juniper Q5
- понижает пропускную способность конвейера

Параллелизм СПУ

- Параллелизм на уровне стадий конвейера
- Параллелизм конвейеров
- Комбинированные подходы





Память СПУ

Стратегии размещения данных:

- тела пакетов – внешняя память,
- таблицы классификации – внутренняя память

Broadcom Trident, Broadcom Tomahawk, Barefoot Tofino, Mellanox NP-5

- все данные во внешней памяти

Huawei ENP, Juniper Q5

Память тел пакетов: DDR SDRAM, RL DRAM

Память таблиц классификации: SRAM, TCAM

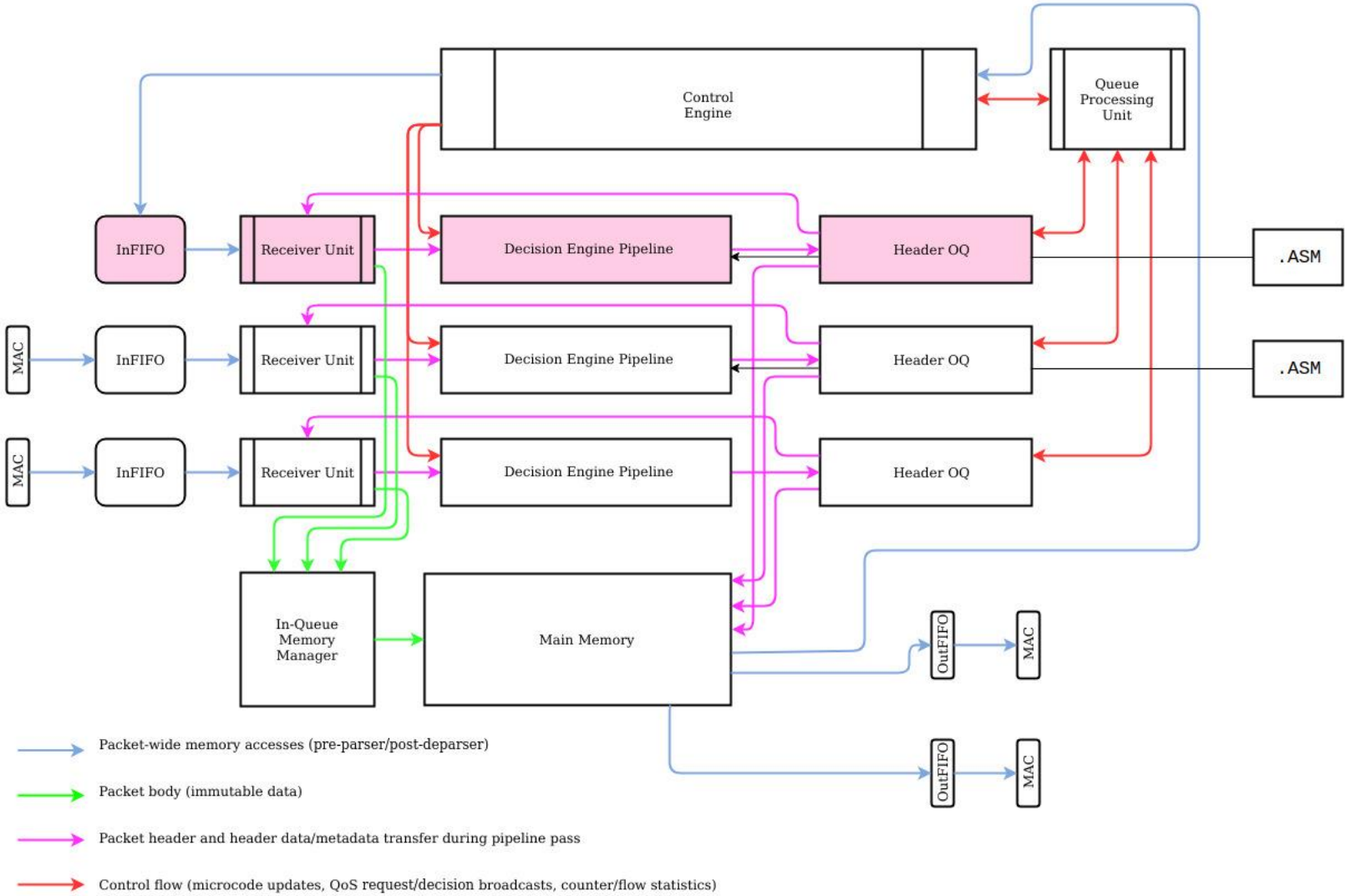
Основные тенденции

- Программируемость разные производители понимают по-разному
- Наибольшая производительность у устройств ASIC
- Принципы построения конвейеров:
  - разделение на 2 части (ingress, egress);
  - коммутационная матрица и репликатор пакетов между частями конвейера;
  - функциональная специализация стадий;
  - масштабируемая архитектура из однотипных конвейеров.
- Размещение тел пакетов во внешней памяти, таблиц классификации – во внутренней

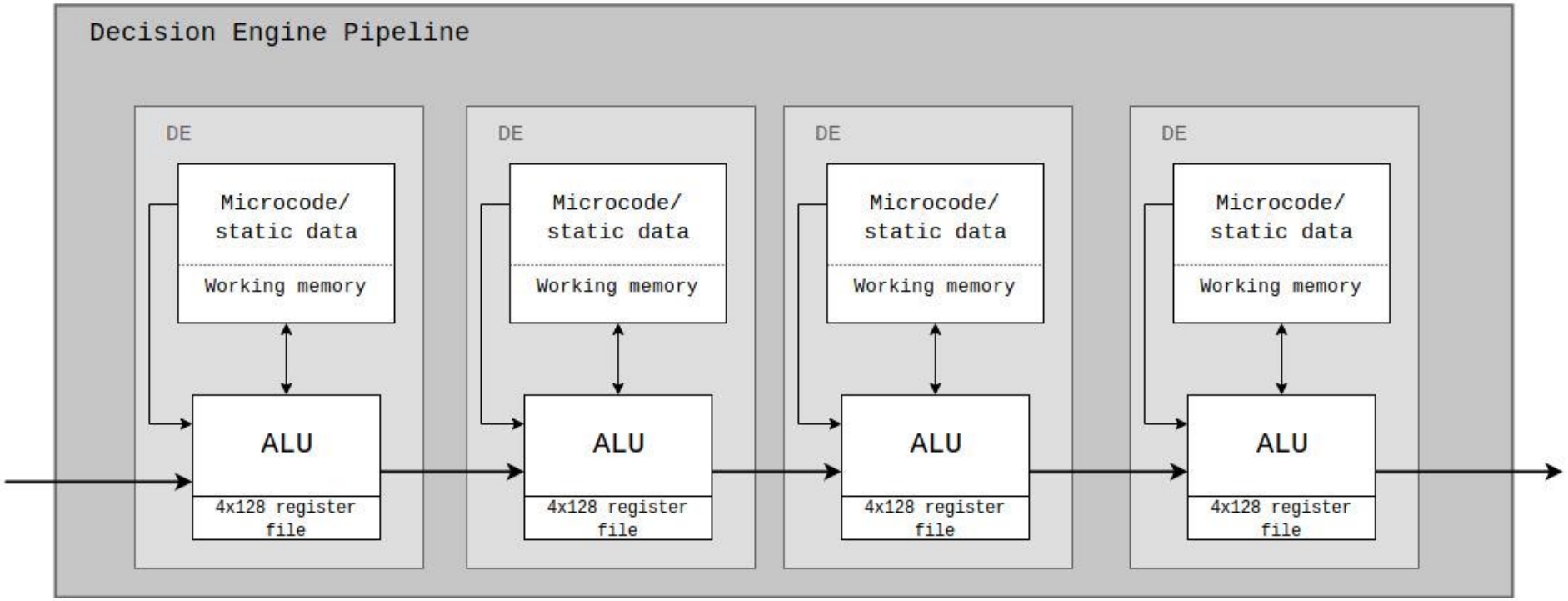
Выводы

- Разработка СПУ типа сетевой процессор
- Масштабируемая архитектура (набор однотипных конвейеров или конвейер стадий с однотипными ядрами)
- Функциональная специализация стадий конвейера
- Использование блоков памяти на кристалле для хранения таблиц классификации

Архитектура со специализированными ядрами



Архитектура со специализированными ядрами



Характеристики предлагаемой архитектуры

- 24 порта по 10 Гбит/с *(+ 4 порта по 100 Гбит/с)*
- Интерфейс PCIe Gen2 x8
- 24*8 специализированных 64-бит потоковых процессоров
- Рабочая частота: 800 МГц
- Общий объем ОЗУ на кристалле: 64 МБайт
- Поддержка таблиц до 32 тыс. элементов.*
- Размер кристалла до 160 мм2.*
- Тех. процесс TSMC 28 нм HPC+.*
- Энергопотребление до 45 Вт.*

* - *характеристики будут уточнены при проектировании*

Спасибо за внимание!