

*Все идеи и алгоритмы, описываемые в данной статье, являются результатом моей независимой и полностью самостоятельной интеллектуальной деятельности. Как автор, разрешаю свободно использовать, изменять, дополнять все идеи и алгоритмы любому человеку или организации в любых типах проектов при обязательном указании моего авторства.*

© Балыбердин Андрей Леонидович 2019 Rutel@Mail.ru

**Синхронная Символьная Иерархия**  
**Принципы формирования виртуальных каналов**  
(разделение единого физического канала на множество виртуальных)

Автор : Балыбердин А.Л.

Новосибирск, 2022 г.

Обязательным атрибутом современной сети, является разделение физических линий связи на множество отдельных виртуальных каналов. Каналы могут быть и совсем виртуальными, как в случае с пакетной коммутацией, так и прописаны в формате передаваемых конструкций (PDH, SHD и т.д.). Поскольку плезиохронные каналы появились во времена голосовой телефонии и других потребителей не было, виртуальные каналы имели одинаковую скорость 64КБит\сек. Для современных сетей требуются каналы с различной скоростью и различными свойствами.

В статье «ССИ\_Синхронный поток\_Эффект проскальзывания (1)» показан принцип формирования базового потока символов, теперь будет описан принцип его разделения на отдельные плезиохронные потоки с произвольной скоростью передачи символов.

Ограничения: Сумма скоростей виртуальных каналов не должна быть больше скорости физического канала. Время коммутации не должно быть больше единиц периодов передачи символов в виртуальном канале и стремиться к половине периода передачи в конкретном виртуальном канале (Чем медленнее виртуальный канал, тем больше времени его данные находятся в FIFO коммутатора)

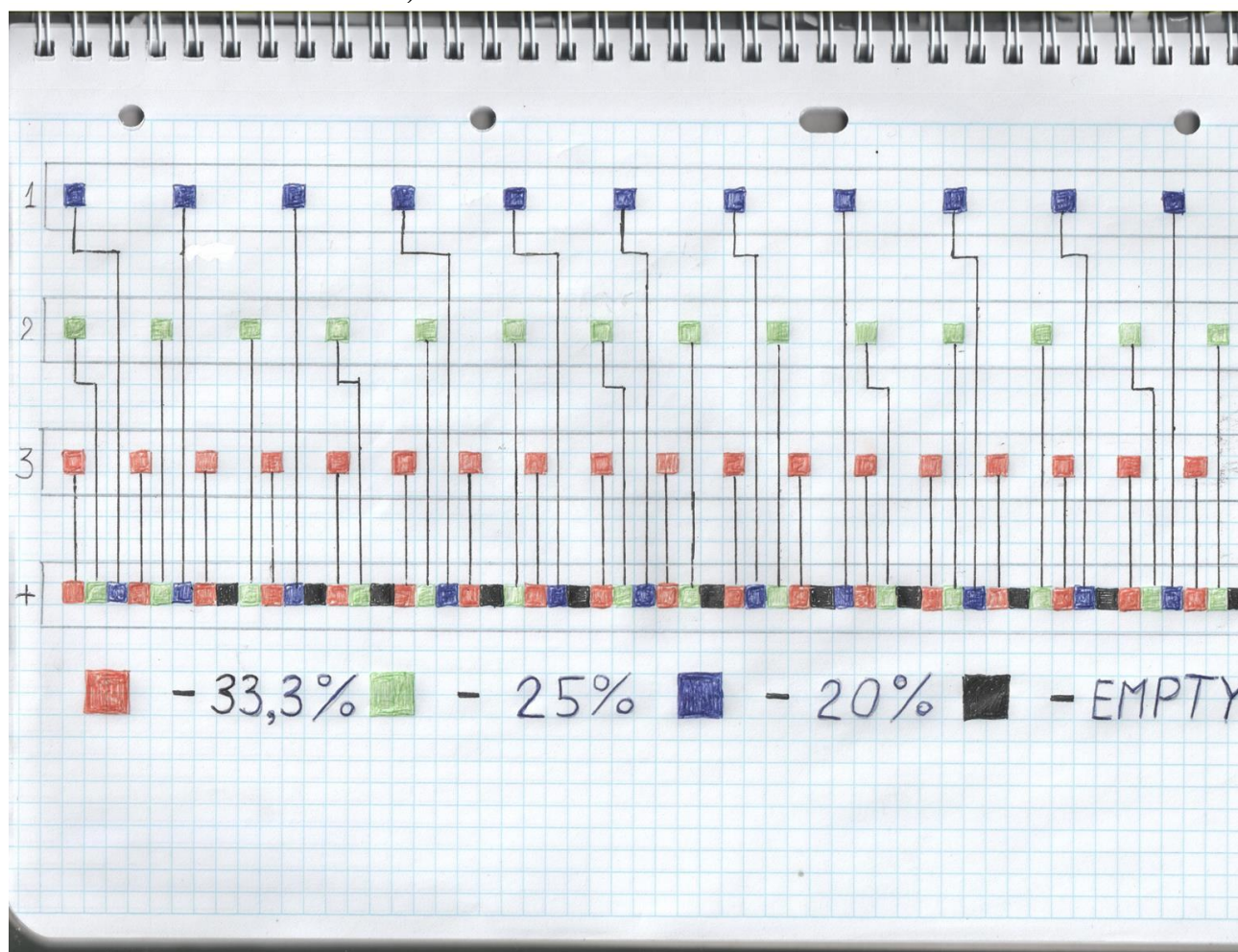
Ограничение на время коммутации существенно ограничивает варианты построения коммутатора. Самый простой вариант — это массив модулей много-портового FIFO (3-5 символов), число портов чтения (записи) равно числу внешних интерфейсов коммутатора. Каждое FIFO соединяет вход коммутатора с выходом, какой вход с каким выходом определяется в момент создания виртуального канала. Число модулей FIFO ограничивает (равно), число одновременных виртуальных каналов, создаваемых коммутатором.

Модули приемника (передатчика) должны выдавать на адресные входы коммутатора номер FIFO к которому происходит обращение. Для примера, зададим точность установки скорости передачи виртуального канала в один символ в секунду при скорости физического канала 100G и размере символа в 100 бит потребуется 1E9 тайм-слотов. Максимальное число отдельных виртуальных каналов ограничивается числом FIFO модулей. Формирование последовательности опроса FIFO производится с учетом скорости передачи в каждом виртуальном канале. Необходимо гарантировать что FIFO размером 3-5 символов никогда не переполнится. При этом не снижать КПД использования физического канала больше чем на единицы процентов (допускается использование для пользовательских данных до 95 % пропускной способности физического канала на постоянной основе).

Думаю алгоритмов раскладки достаточно много, лично я нашел два. Первый больше подходит для пояснения принципа раскладки, но сильно неудобен для реализации. Второй более подходит для практической реализации.

**Мультиплексирование (разделение) канала связи (1)**  
(Алгоритм симметричен, одинаков для приемника и передатчика)

1. Сортируем создаваемые каналы по убыванию скорости передачи.
2. Для каждого канала создаем счетчик (регулирует скорость передачи) и назначаем FIFO, где хранятся передаваемые символы.
3. Каждый такт синхронизации в каждый счетчик прибавляем константу пропорциональную скорости передачи, константа =  $V$  (требуемая) /  $V$  (физического потока).
4. Каждый такт проверяем счетчики, в очередности убывания скоростей привязанных к ним каналов, на переполнение (значение больше единицы), пока не найдем первый счетчик, содержащий значение больше единицы.
5. Из найденного счетчика вычитаем единицу, а в суммарный поток добавляем один символ из буфера передаваемых символов (создан в связке со счетчиком).

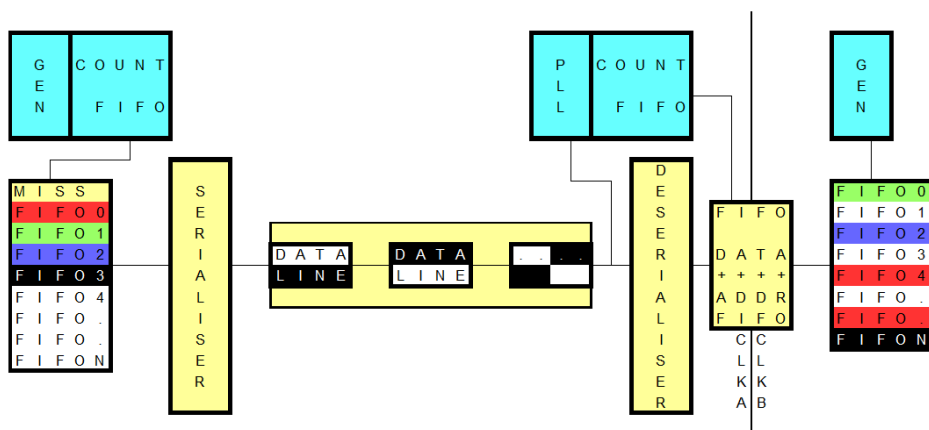


Данный алгоритм проверен на программной модели, максимальное расстояние между соседними символами в пределах виртуального канала составляет 2.3 периода передачи символа в пределах данного виртуального канала.

## Мультиплексирование (разделение) канала связи (2)

(Алгоритм симметричен, одинаков для приемника и передатчика)

1. Создаем по одному счетчику для каждого приемника и передатчика физического канала (примерно, как в «ССИ\_Синхронный поток\_Эффект проскальзывания (1)»), с помощью служебных символов синхронизируем их относительно потока символов в данном физическом канале. Для выбранного примера счетчик от 0 до  $10E9 - 1$ .
2. Получается, что каждый символ будет иметь уникальный номер (в пределах одной секунды)
3. При обращении к коммутатору преобразовываем содержимое счетчика в адрес (номер) FIFO.



### Алгоритм преобразования — Счетчик <:> Адрес FIFO:

1. Диапазон значений в  $10E9$ , по сути является номерами тайм-слотов в передаваемой периодической структуре. Остается только распределить эти тайм-слоты максимально равномерно и пропорционально скорости каждого виртуального канала.
2. Каждому адресу FIFO соответствует несколько значений счетчика. Ноль если канал не создан, единица если канал имеет скорость один символ в секунду и так далее.
3. Представим весь диапазон значений счетчика как некоторое адресное пространство и присвоим каждому каналу непрерывный диапазон значений равный символьной скорости виртуального канала. Выглядит как выделение памяти в современных вычислительных системах.
4. Каждый модуль FIFO откликается на любое значение в пределах этого диапазона, выдавая или записывая символ.
5. Просто подавать счетчик на адресные входы FIFO нельзя, данные будут передаваться непрерывными последовательностями раз в секунду. Для равномерного распределения опросов FIFO нужно «перевернуть» значение счетчика (поменять местами старшие разряды с младшими).
6. FIFO также должно сравнивать этот «перевернутый» счетчик с диапазоном значений, занятый данным виртуальным каналом.

## 7. Пример работы алгоритма преобразования «Счетчик- адрес FIFO»



Второй алгоритм распределения тайм-слотов гораздо эффективнее в реализации чем первый и соответствует всем требованиям ССИ, но не был проверен на программной или аппаратной модели.

### Темы для НИР

1. Проверить равномерность получаемых виртуальных потоков.
2. Практически измерить максимальное число символов в FIFO, для разного числа виртуальных каналов и различных вариантов скоростей.
3. Проверить зависимость неравномерности символьной скорости от последовательности (места расположения) виртуальных каналов в пространстве комбинаций счетчика.
4. Проверить работу алгоритма, когда число комбинаций счетчика не равно степени двойки.