

# Транспьютеры

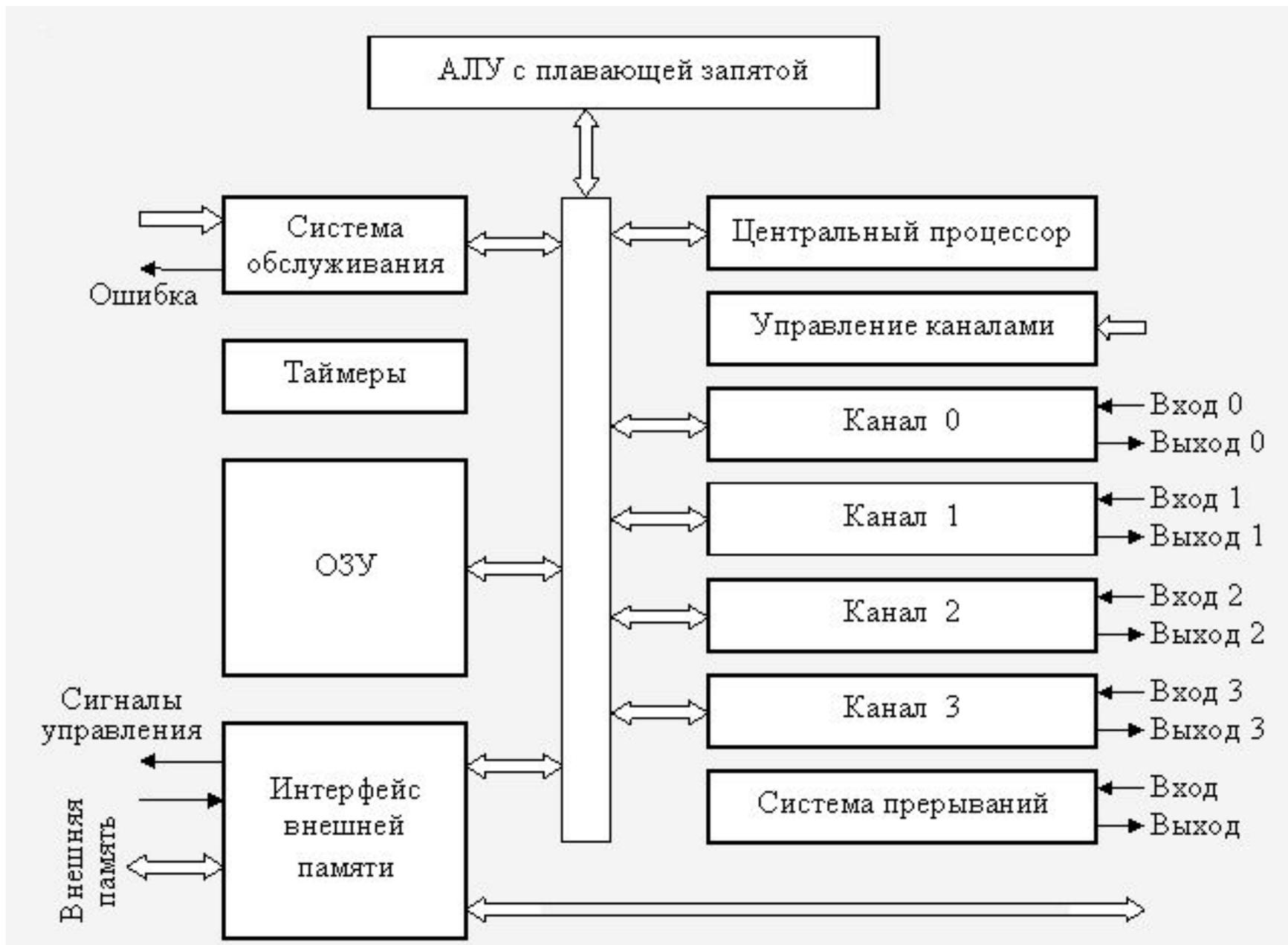
Транспьютер (transputer) — элемент построения многопроцессорных систем, выполненный на одном кристалле сверхбольшой интегральной схемы, продукт английской компании INMOS Ltd ([STMicroelectronics](#)).

**транспьютер = передатчик (transfer) + вычислитель (computer)** - указывает на присутствие встроенных скоростных каналов для обмена данными с соседними.



32 битный  
транспьютерный  
чип **IMS T414B**

# Базовая архитектура транспьютера



## **Базовая архитектура транспьютера включает:**

- Центральный процессор;
- АЛУ для операций с плавающей запятой;
- Каналы связи с другими транспьютерами и внешними устройствами (линки),
- Внутрикристалльную память (ОЗУ);
- Интерфейс для подключения внешней памяти;
- Интерфейс событий (систему прерываний);
- Логику системного сервиса (систему обслуживания);
- Таймеры.

# Линки транспьютера

**Линки** (link — связь) — синхронный примитив, ассоциация с трубой.

Линки бывают двух типов — физические и логические. Операции над линками обоих типов осуществляются одними и теми же командами.

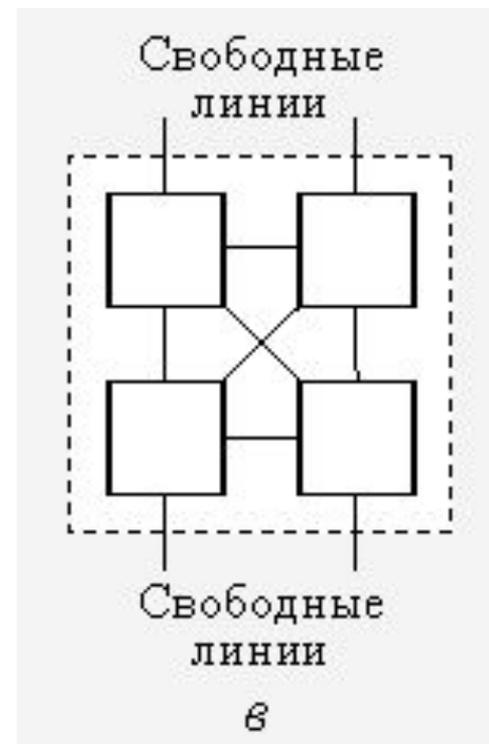
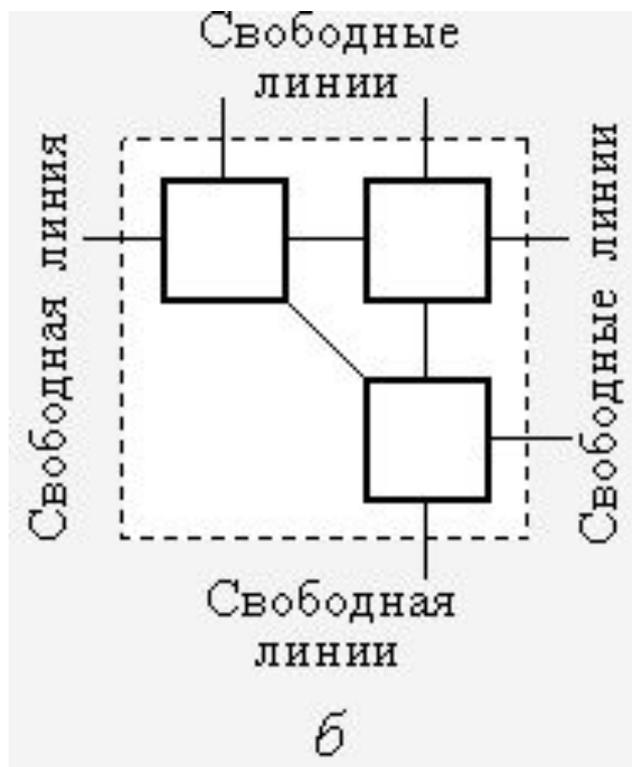
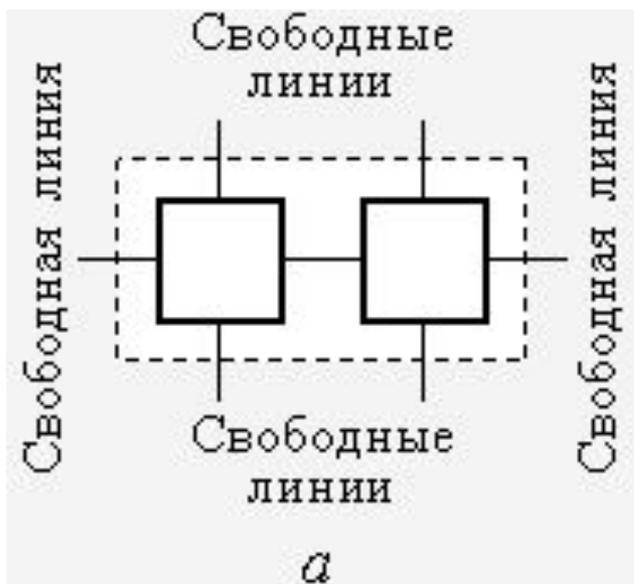
**Физический линк** представляет собой последовательный интерфейс RS432, реализованный на кристалле процессора.

Современные транспьютеры имеют **четыре физических линка**.

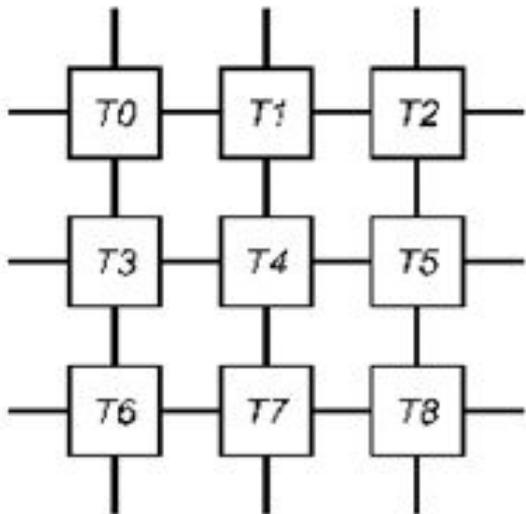
Физические линки могут передавать данные со скоростью до 20 Мбит/с и могут использоваться как для соединения транспьютеров между собой, так и для подключения внешних устройств. Благодаря этому физический линк может использоваться как для связи между процессами на разных транспьютерах, так и для синхронизации процесса с внешними событиями и даже просто для ввода-вывода.

# Системы на основе транспьютеров

Параллельная система может создаваться из набора транспьютеров, которые функционируют независимо и взаимодействуют через последовательные каналы связи.

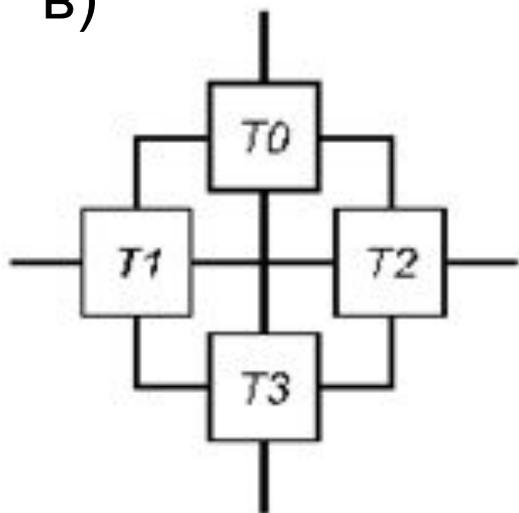


б)

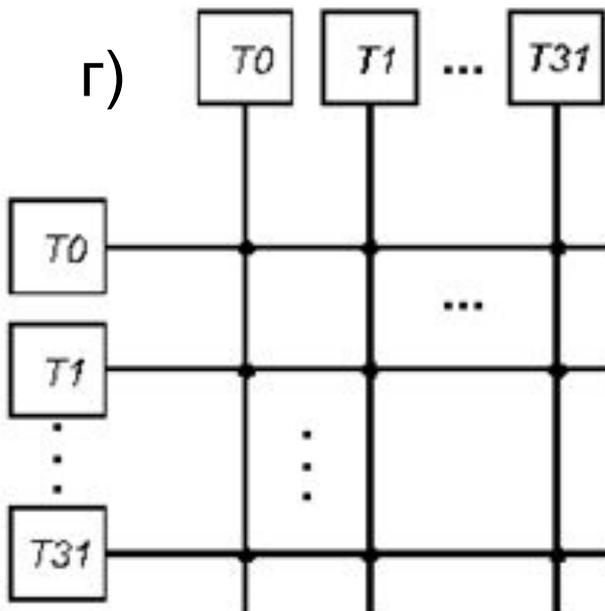


Для образования транспьютерных систем требуемого размера каналы различных транспьютеров могут соединяться непосредственно (рис. б, в) или через коммутаторы типа координатного переключателя на 32 входа и выхода, который обеспечивает одновременно 16 пар связей (рис. г). Такие переключатели могут настраиваться программно или вручную и входят в комплект транспьютерных СБИС. Размер систем не ограничен, а структура системы может быть сетевой, иерархической или смешанной.

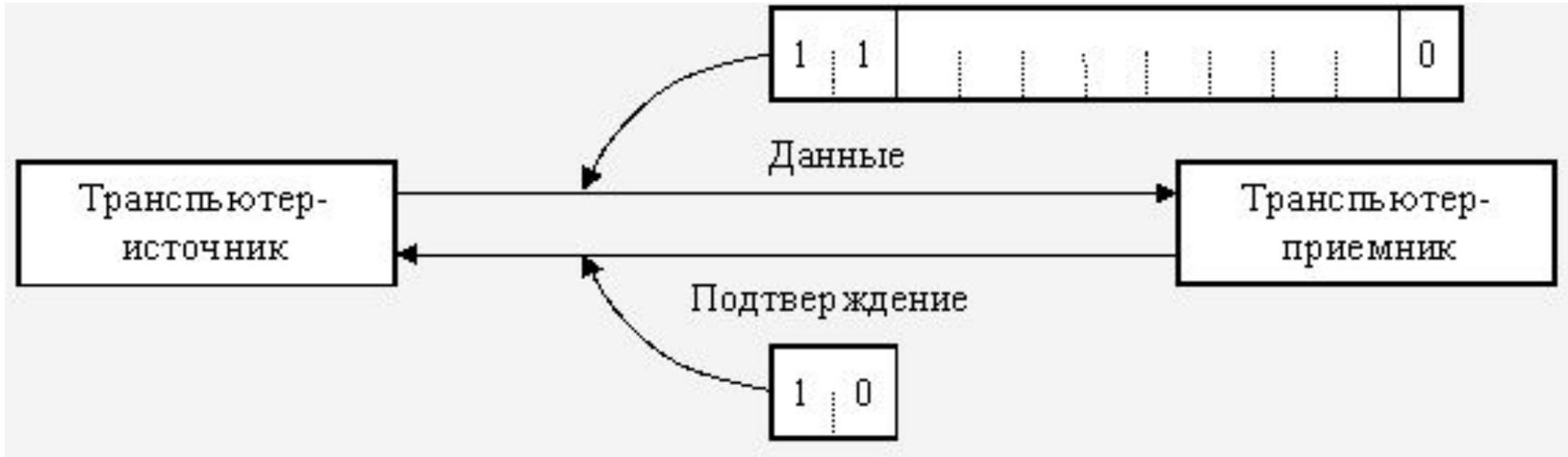
в)



г)



# Организация ввода/вывода в транспьютерной системе



Передача информации производится синхронно под воздействием либо общего генератора тактовых импульсов (ГТИ), либо локальных ГТИ с одинаковой частотой следования импульсов. Информация передается в виде пакетов. Каждый раз, когда пересылается пакет данных, приемник отвечает пакетом подтверждения

Язык программирования Оккам назван в честь английского философа XIV века Уильяма Оккамского, а его сентенция, известная как бритва Оккама, является девизом проекта.

«Бритва (лезвие) Оккама» — методологический принцип, получивший название по имени английского монаха-францисканца, философа Уильяма Оккамского (Ockham, Oskam, Ossam; ок. 1285-1349):

**«Не следует привлекать новые сущности без самой крайней на то необходимости»**

Этот принцип формирует базис методологического **редукционизма**, также называемый принципом бережливости, или законом экономии.

Однако то, что называют **«Бритвой Оккама»**, не было сформулировано Оккамом, он всего лишь озвучил принцип, известный ещё со времён Аристотеля и в логике носящий название **«закон достаточного основания»**. Принцип выражается в словах

**«То, что можно объяснить посредством меньшего, не следует выражать посредством большего».**

В современной науке под бритвой Оккама обычно понимают более общий принцип, утверждающий, что если существует несколько логически непротиворечивых определений или объяснений какого-либо явления, то следует считать верным самое простое из них.

На языке теории информации принцип «Бритвы Оккама» гласит,

**САМЫМ ТОЧНЫМ СООБЩЕНИЕМ ЯВЛЯЕТСЯ  
СООБЩЕНИЕ МИНИМАЛЬНОЙ ДЛИНЫ.**

**Основой языка** являются: средства описания параллелизма выполняемых процессов; средства описания межпроцессорного обмена данными; средства описания размещения процессов по единицам оборудования.

Язык обеспечивает описание простых операций пересылки данных между двумя точками, а также позволяет явно указать на параллелизм при выполнении программы несколькими транспьютерами.

**Основным понятием** программы на языке Оссам является процесс, состоящий из одного или более операторов программы, которые могут быть выполнены последовательно или параллельно. Процессы могут быть распределены по транспьютерам вычислительной системы, при этом оборудование транспьютера поддерживает совместное использование транспьютера несколькими процессами.

**Оккам - программы состоят из трех примитивных процессов:**

- $v := e$  присвоение значения выражения  $e$  переменной  $v$
- $c ! e$  вывести значение выражения  $e$  в канал  $c$
- $c ? v$  ввести из .канала  $c$  в переменную  $v$

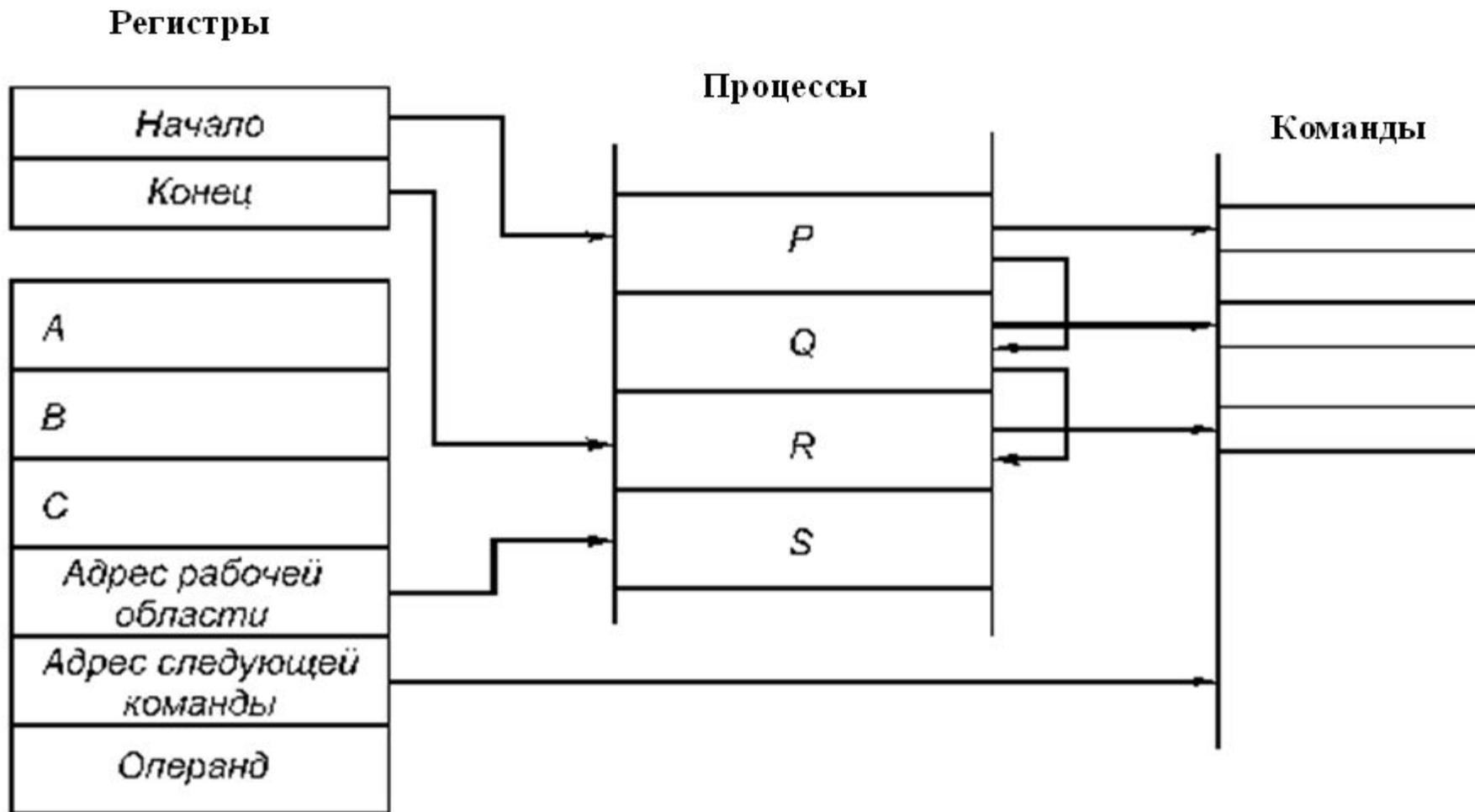
**Примитивные процессы объединяются и формируют конструкции:**

- SEQ** - последовательное выполнение
- IF** - условное выполнение
- WHILE** - цикл
- PAR** - параллельное выполнение
- ALT** - альтернативное выполнение

Все остальные процессы могут быть получены иерархическим построением (через ранее определённые). Для этой цели Оссам предоставляет набор конструкторов процессов: **SEQ, PAR, IF, WHILE, ALT**.

При этом действует правило, согласно которому составной процесс типа SEQ или PAR считается выполненным, когда завершены все составляющие его процессы. Процессы могут быть поименованы и вызваны по имени с передачей параметров. Процессы SEQ, PAR, IF и ALT могут быть реплицированы (размножены) при помощи репликатора FOR. Процесс ALT (как и PAR) привносит в язык индетерминизм, так как считается, что при одновременном выполнении нескольких условий точно предсказать дальнейший ход событий невозможно. Конструкция сама является процессом и может использоваться как компонента другой конструкции; другими словами, Оссам является иерархическим языком с блочной структурой.

# Обработка параллельных процессов



Каждый канал транспьютера физически состоит из двух одноразрядных каналов, один для работы в прямом, другой - для работы в обратном направлении, обозначаемые как link.in и link.out.

В транспьютере, кроме вычислительного стека ЦП для целочисленной арифметики, имеется стек для работы над данными с плавающей запятой с регистрами AF, BF, CF. Список команд транспьютера включает 110 команд. Они делятся на две группы: с прямой адресацией (один байт) и с косвенной адресацией (два или более байтов).

В ситуации, изображенной на рисунке 2, имеется четыре активных процесса, причем процесс S выполняется, а процессы P, Q и R ожидают выполнения в планировочном списке. Команда транспьютера **start process** создает новый активный процесс, добавляя его в конец планировочного списка.

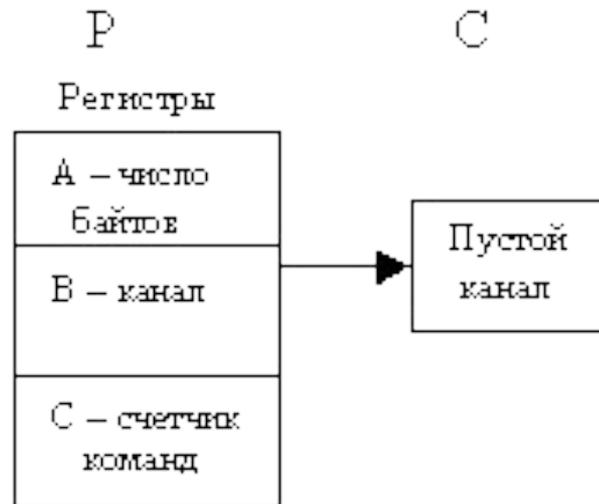
Команда **end process** завершает текущий процесс, убирая его из планировочного списка. В Оккаме конструкция **PAR** - параллельного запуска процессов может закончиться только тогда, когда завершатся все ее компоненты параллельного процесса. Каждая команда **start process** увеличивает их число, а **end process** уменьшает.

# Связь транспьютеров

В процессоре реализован ряд операций для поддержки обмена сообщениями, наиболее важными из которых являются: Input message; Output message

## Коммуникация по внутренним каналам

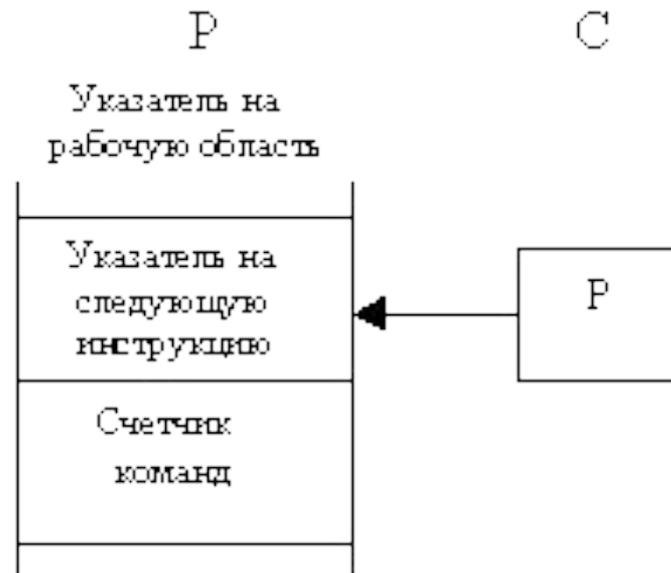
В каждый момент времени внутренний канал (одно слово памяти) содержит или идентификатор процесса, или специальное значение empty. До начала использования канал инициализируется значением empty.



*Рис. 6. Вывод в пустой канал*

Процесс Р выполняет инструкцию вывода в "свободный" канал С. Оценочный стек содержит указатель на сообщение, адрес канала С и число байтов в сообщении.

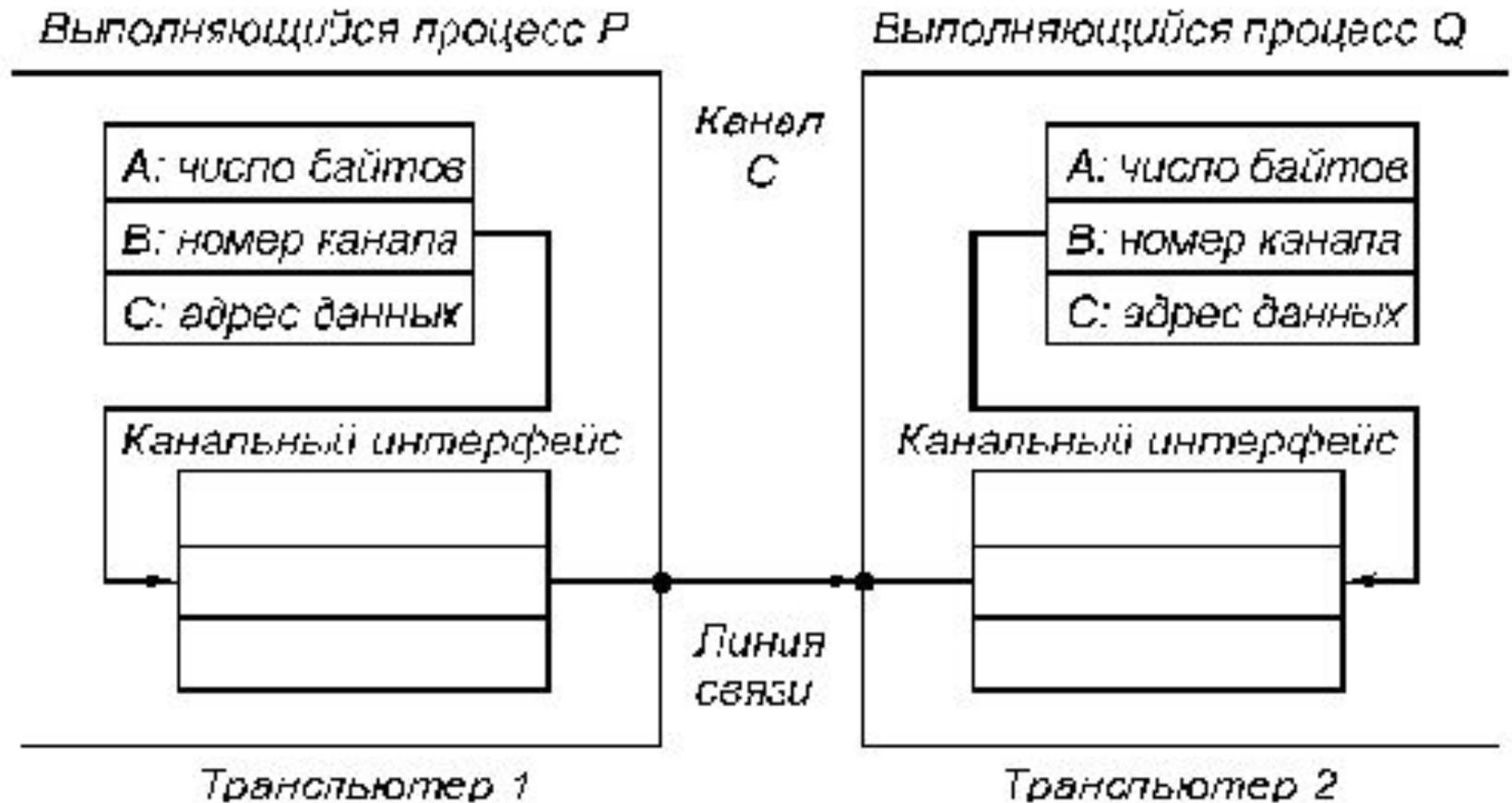
После выполнения инструкции вывода канал С содержит адрес рабочей области процесса Р, а адрес передаваемого сообщения записывается в рабочую область процесса Р. Процесс Р выводится из списка диспетчеризации, и процессор начинает выполнение следующего процесса из списка диспетчеризации



*Рис. 7.*

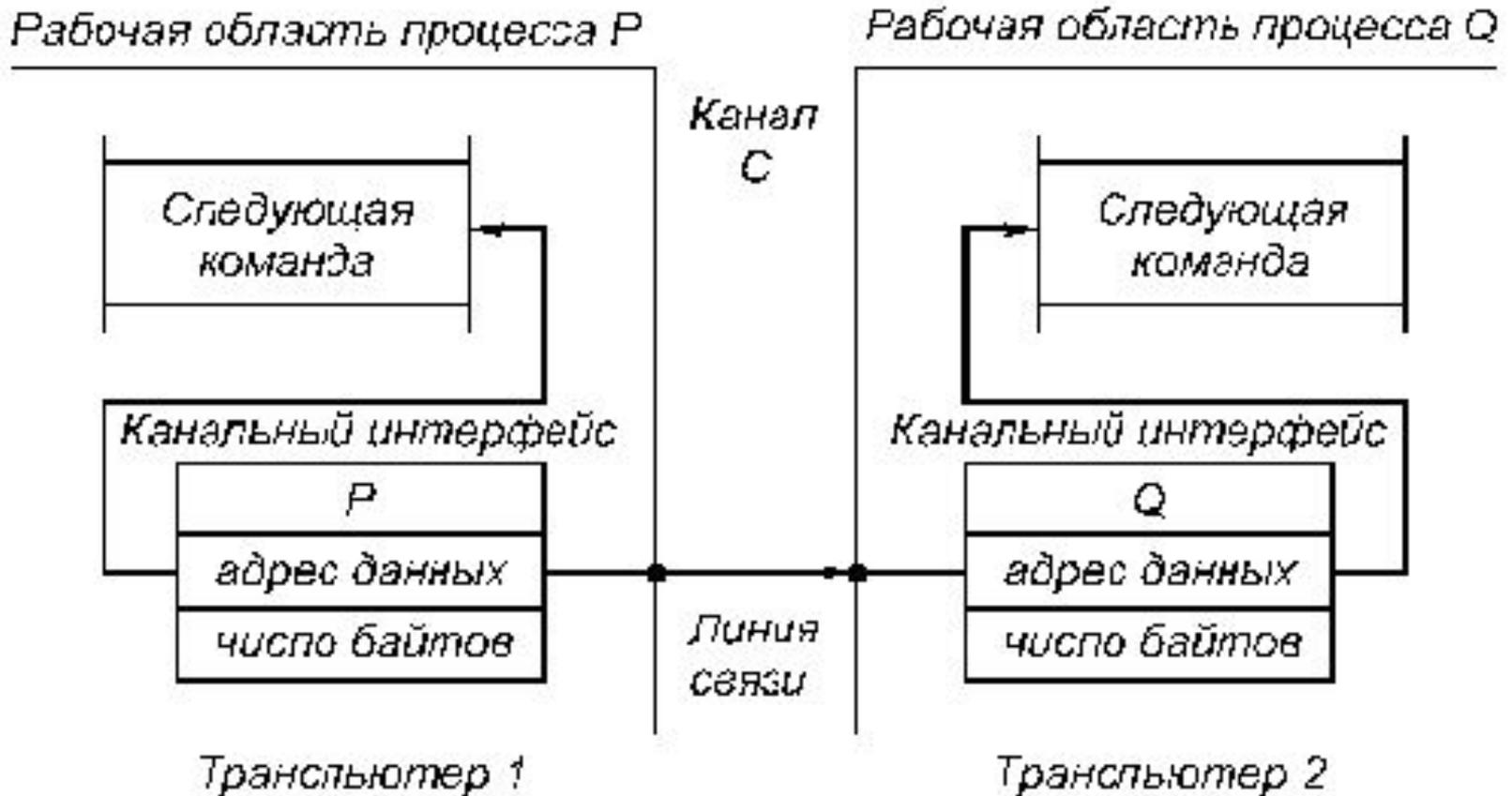


# Структура связи между транспьютерами по внешнему каналу



Процессы Р и Q выполняются на разных транспьютерах и взаимодействуют через канал С, реализованный с помощью линии связи между транспьютерами.

Пусть  $P$  передает данные, а  $Q$  принимает. Когда процесс  $P$  выполняет инструкцию вывода, регистры интерфейса связи транспьютера, на котором выполняется этот процесс, инициализируются и процесс  $P$  исключается из списка диспетчеризации. Таким же образом, когда процесс  $Q$  выполняет инструкцию ввода, инициализируются регистры интерфейса связи транспьютера, выполняющего процесс  $Q$ , и процесс  $Q$  исключается из списка диспетчеризации



Когда оба канальных интерфейса инициализированы, происходит копирование данных по межтранспьютерной линии связи. После этого процессы P и Q включаются в соответствующие списки диспетчеризации

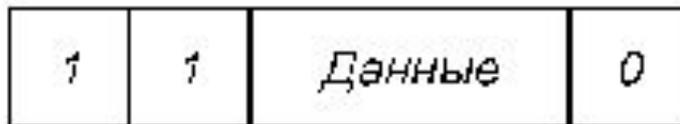
*Рабочая область процесса P*



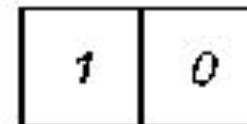
*Рабочая область процесса Q*



После отправления пакета данных транспьютер ожидает получения пакета подтверждения от принимающего транспьютера. Пакет подтверждения показывает, что процесс-получатель готов принять этот байт и что канал принимающего транспьютера может начать прием следующего байта.



*Пакет данных*



*Пакет подтверждения*

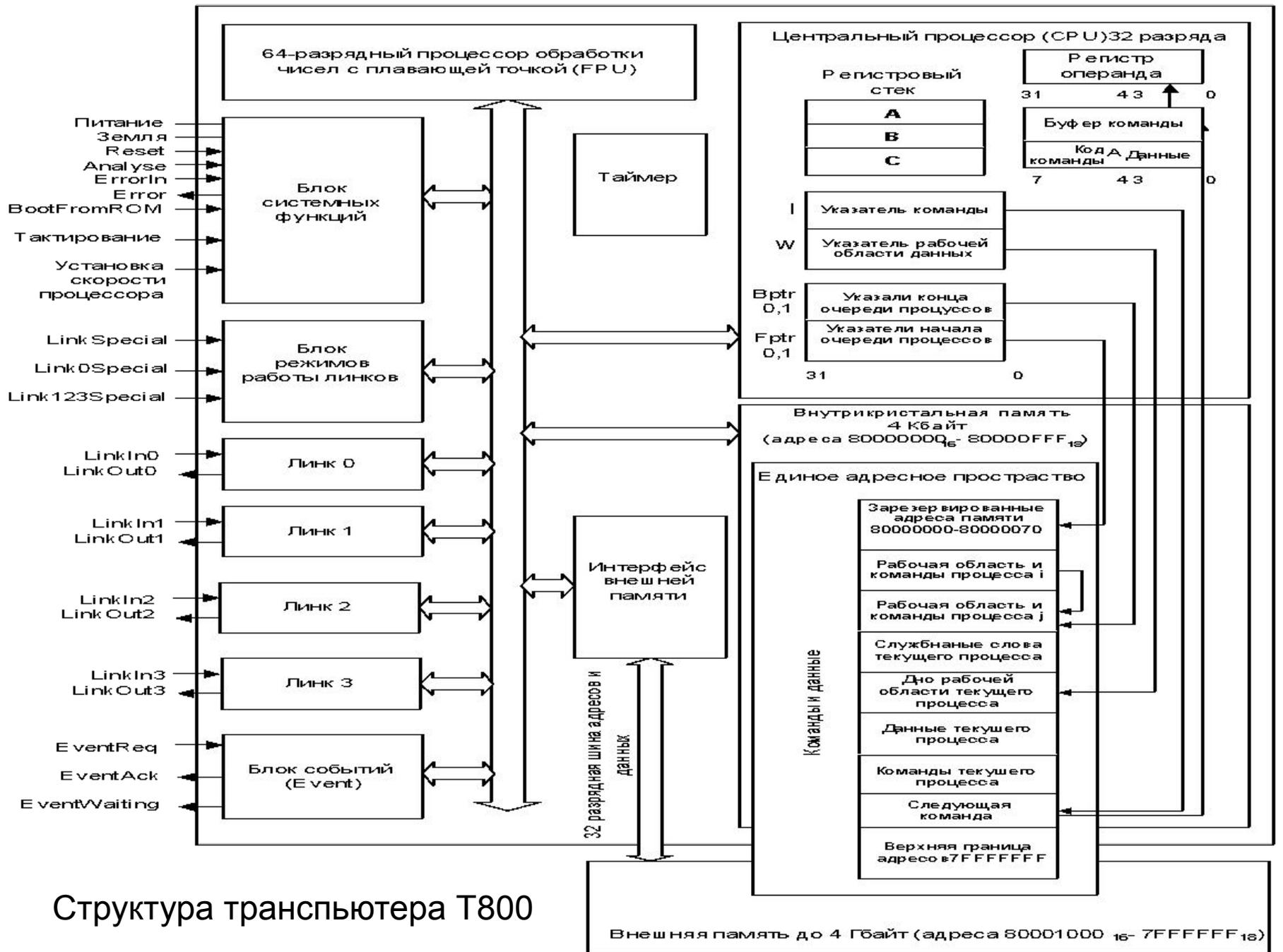
## ХАРАКТЕРИСТИКИ ТРАНСПЬЮТЕРОВ

Транспьютеры выпускаются фирмой INMOS, начиная с 1985 года. За исключением транспьютера T212, который имеет только две связи и 16-разрядные АЛУ, все остальные типы транспьютеров имеют 4 связи и 32 или 64-разрядные АЛУ.

Система команд транспьютеров относится к классу CISC.

В семействе транспьютеров только T9000 является суперскалярным, то есть может выполнять более 1 команды за такт.

| Тип трансп., год | Состав АЛУ, объем внутренней памяти          | Частота синхр., МГц | Быстродействие        | Число транз/, млн |
|------------------|--|---------------------|-----------------------|-------------------|
| T414 (1985)      | АЛУ-I-32 разр.<br>2 Кбайт                    | 15 20 10            | 0,1 МФлопс            | 0,2               |
| T800             | АЛУ-I-32 разр.<br>АЛУ-F-64 разр.<br>4 Кбайт  | 20 30               | 2,2 МФлопс            | 0,3               |
| T9000 (1995)     | АЛУ-I-32 разр.<br>АЛУ-F-64 разр.<br>16 Кбайт | 20 40 50            | 200 Мипс<br>25 МФлопс | 3,3               |



Структура транспьютера T800

В зависимости от модели транспьютер включает:

- ЦП, 32- или 16-разрядный;
- интерфейс внешней памяти;
- 2-4 линия- двунаправленных канала;
- программируемый блок событий (Event);
- таймер;
- внутреннее ОЗУ ёмкостью 2 или 4 Кбайта;
- блок режимов работы Линков;
- блок системных функций.

Некоторые модели могут содержать 64-разрядное устройство операций с плавающей точкой и интерфейсные схемы внешних устройств.