

# ***SMP, Multicore и HyperThreading***

В.А. Савельев

факультет математики, механики и компьютерных наук

# *Симметричный МультиПроцессор*

- Отдельный компьютер со следующими характеристиками:
  - **два или более** процессора сравнимых возможностей
  - процессоры **совместно используют** одну и ту же память и устройства ввода/вывода
  - процессоры соединены с шиной или другим внутренним соединителем
  - время доступа к памяти примерно одинаково
  - все процессоры выполняют **одинаковые функции** (поэтому и симметричный)

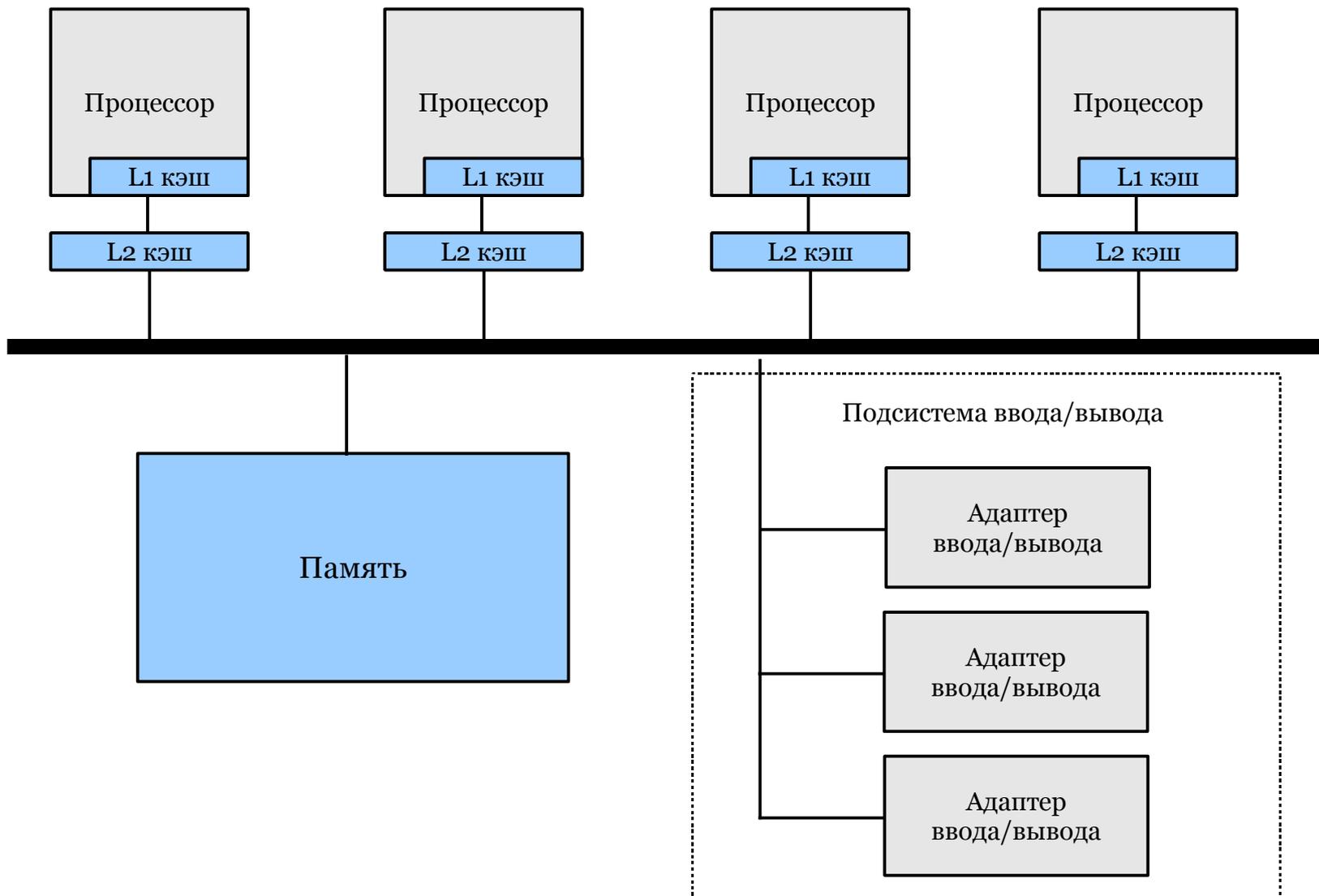
# *Преимущества SMP*

- **Производительность**
  - когда можно работать параллельно
- **Доступность**
  - процессоры выполняют одинаковые функции, поэтому проблемы с одним из них не приводят к краху системы
- **Масштабируемость**
  - пользователь может добавлять процессоры, повышая производительность системы

# *Организация SMP*

- Общая шина (или шина с разделением по времени)
- Многопортовая память
- Центральное управляющее устройство

# Общая шина



# *Общая шина*

- Простая конструкция
- Структура и интерфейсы не отличаются от однопроцессорных систем
- Допускается много процессоров и много систем ввода/вывода
- Свойства
  - адресация
  - арбитраж
  - временное разделение

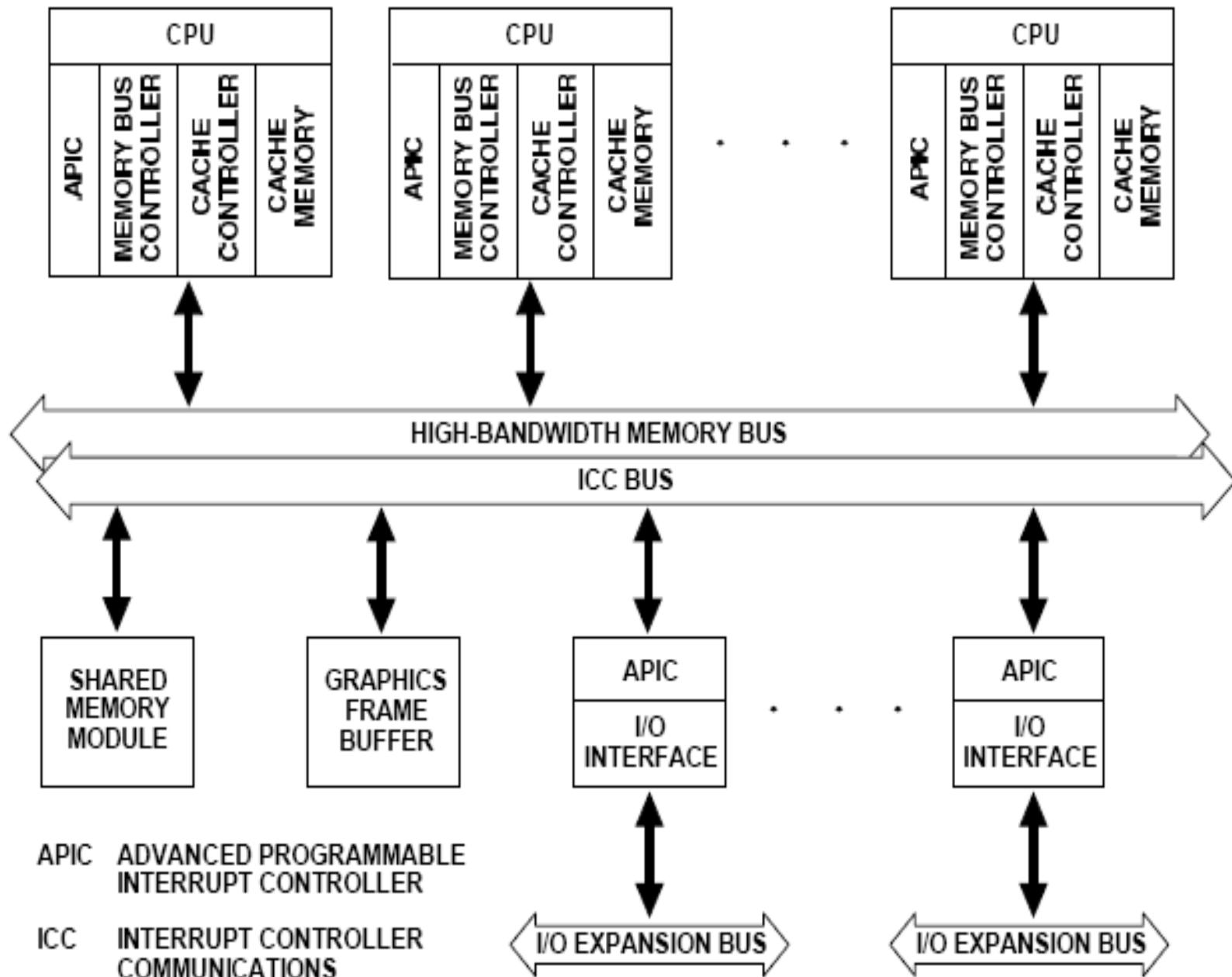
# *Общая шина*

- **Преимущества**
  - простота
  - гибкость
  - надежность
- **Недостатки**
  - производительность лимитируется циклом шины
  - каждый процессор должен иметь кэш
  - проблемы с согласованием кэшей

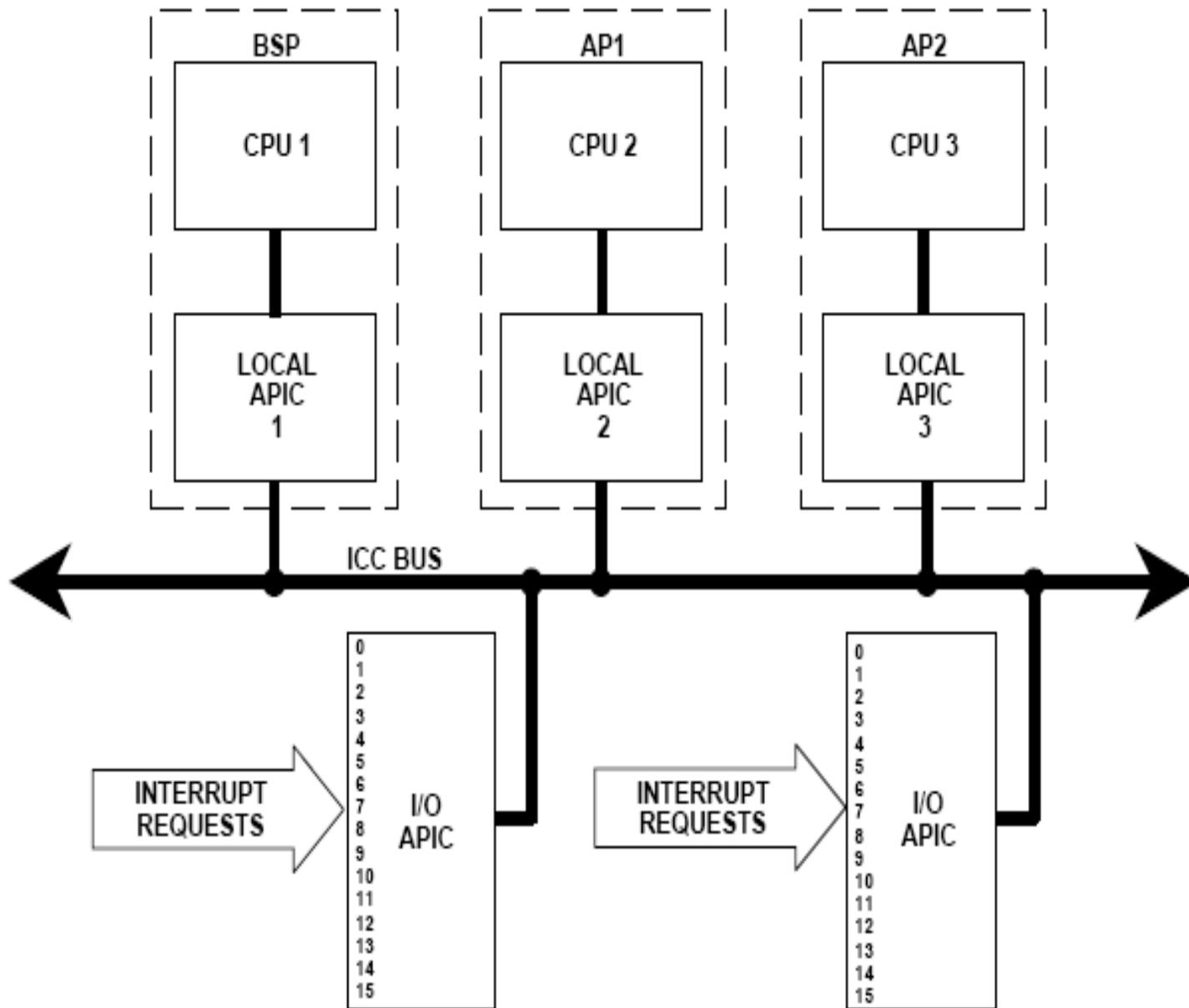
# *Intel MP*

- именно общая шина положена в основу Intel MultiProcessor. Спецификация содержит:
  - поддержку SMP для Intel-совместимых процессоров
  - поддержку симметричной обработки прерываний ввода/вывода с помощью APIC
  - минимальная поддержка MP в BIOS
  - структуру конфигурационной таблицы MP
  - правила подключения шин ввода/вывода к MP
  - требования к вторичным кэшам и шине памяти

# Intel MP

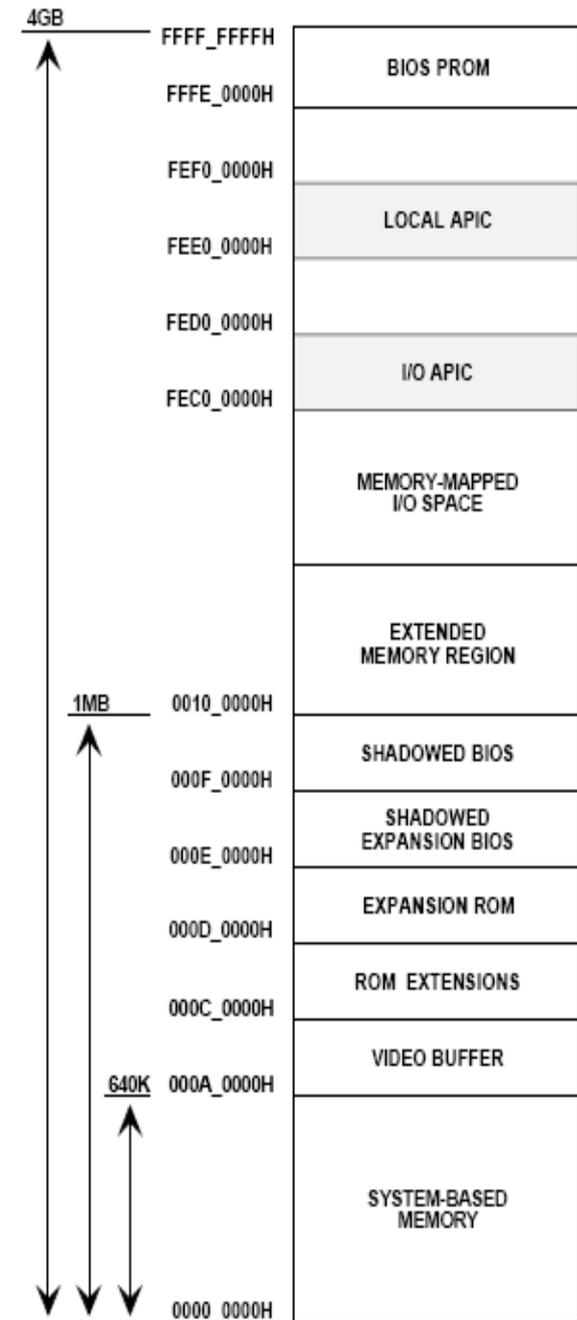


# Системные процессоры и конфигурация APIC



# Карта памяти

- Intel SMP использует стандартное для архитектуры IBM PC/AT распределение памяти. Отображаемые на память устройства ввода/вывода занимают верхние адреса. I/O APIC и локальный APIC занимают, соответственно, 0FEC0\_0000h и 0FEE0\_0000h



PART OF THIS SPECIFICATION

# *Захват шины*

- Для защиты целостности данных при критических операциях с памятью спецификация предписывает поддерживать сигнал LOCK#
  - Процессор устанавливает LOCK# на все время пока осуществляется цикл *чтение-изменение-запись*.
  - Сигнал заставляет блокировать все запросы на доступ к шине от других процессоров или устройств, способных управлять шиной (bus-master)
  - Игнорируются и очищаются кэши

# Захват шины

- Сигнал LOCK# устанавливается, если:
  - выполняется инструкция XCHG
  - изменяются дескрипторы сегментов
  - изменяются элементы страничных таблиц (PDE и PTE)
  - выполняется установка флаг B(usy) в TSS
  - при пересылке вектора прерываний из контролера прерываний в процессор
  - перед инструкцией, обращающейся к памяти, стоит префикс LOCK



Прикладной программист этого не делает

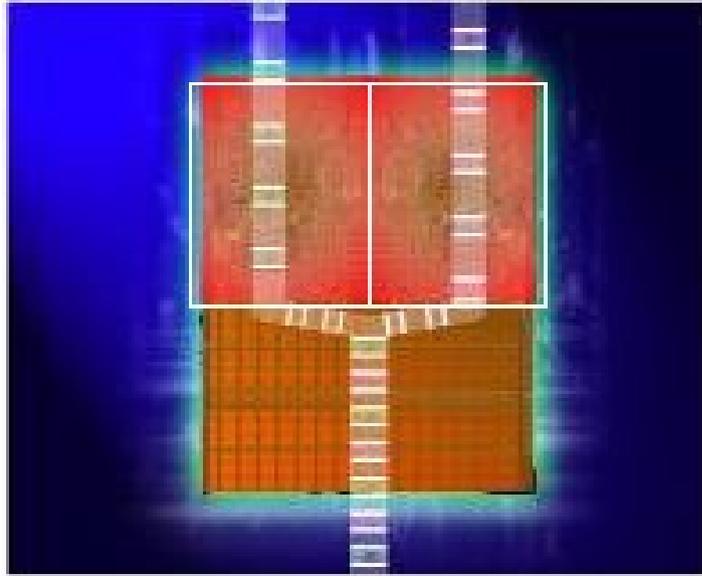
# *Префикс LOCK используется перед*

- инструкциями проверки и изменения битов (BTS, BTR и BTC).
- инструкциями обмена (XADD, CMPXCHG и CMPXCHG8B)
- следующими арифметическими и логическими инструкциями
  - INC, DEC, NOT, NEG
  - ADD, ADC, SUB, SBB, AND, OR, XOR

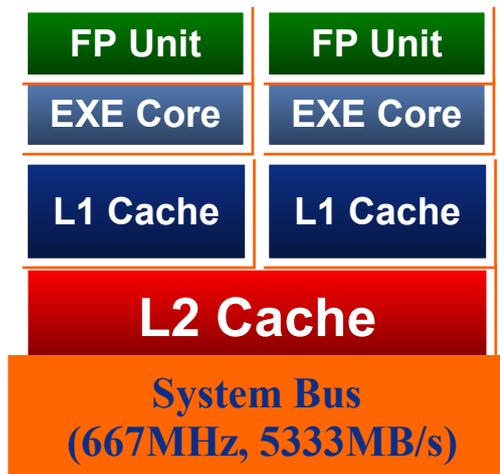
# *Псевдоблокировка шины*

- Система должна поддерживать и безопасность операций с невыровненными данными. Способ не указывается. De facto должен поддерживаться и сигнал PLOCK#
- С учетом кэширования памяти, сигнал PLOCK# будет возникать только если данные пересекают границу 64-байтовой строки кэша.

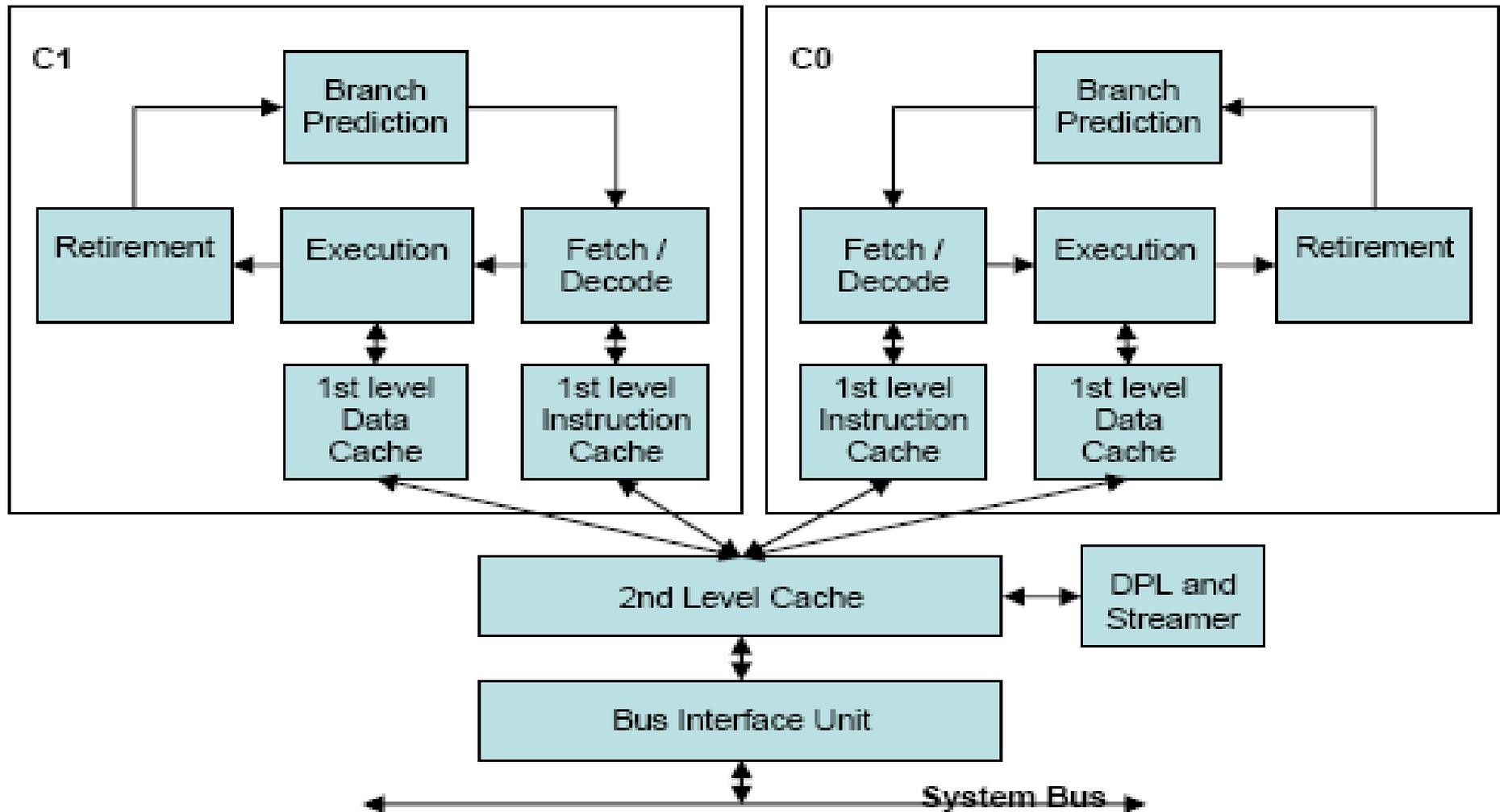
# Многоядерные процессоры



- Процессор CoreDuo
  - два исполняющих ядра
  - два L1 кэша
  - один общий L2 кэш



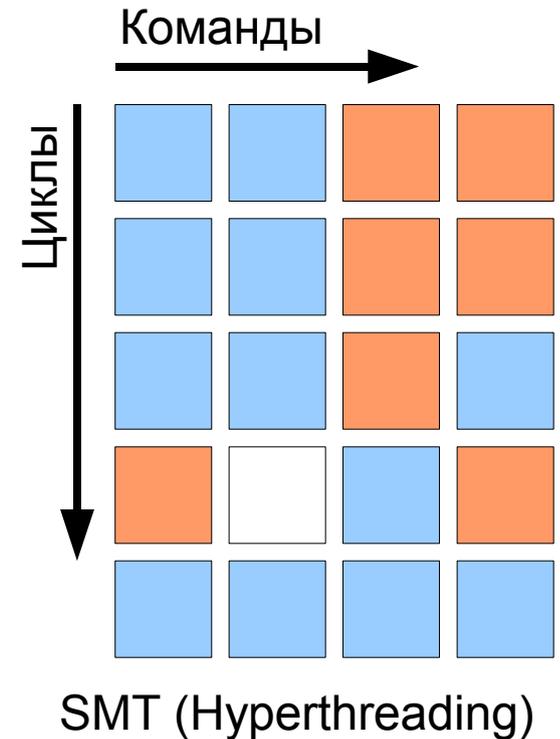
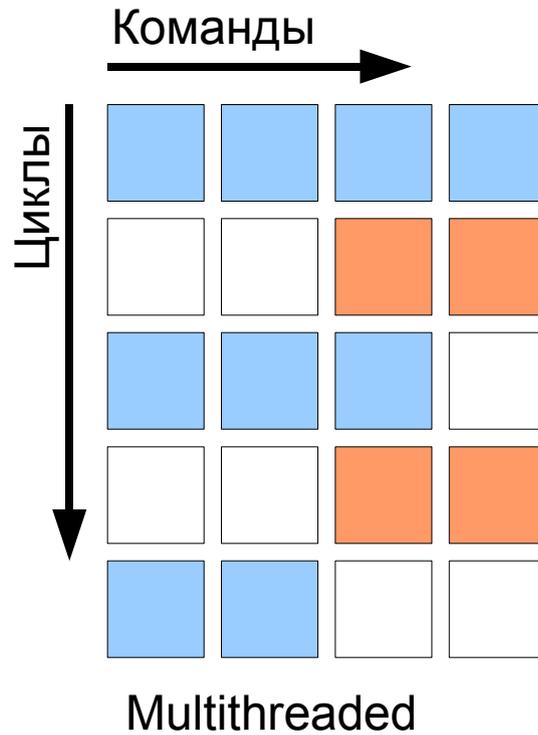
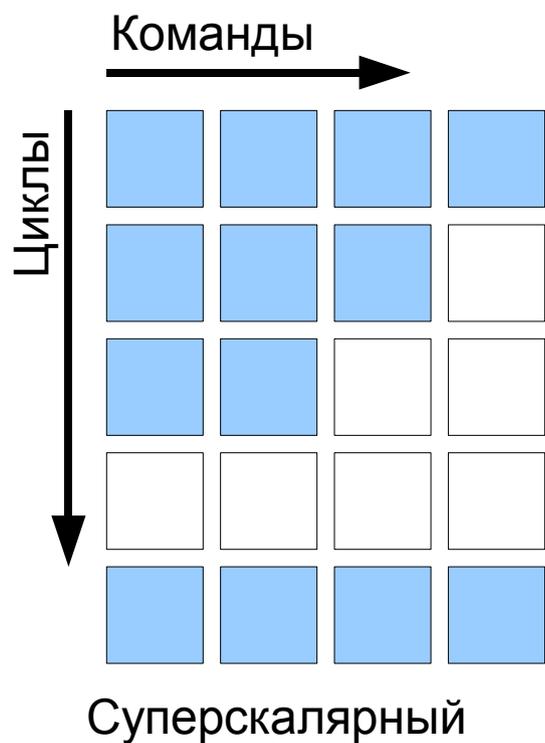
# Микроархитектура



# *Hyper-Threading*

- МОЖЕТ ИСПОЛНЯТЬ ДВЕ КОМАНДЫ ИЗ РАЗНЫХ ПОТОКОВ ЗА ТАКТ
- ИСПОЛНЕНИЕ ДВУХ ПОТОКОВ ПРОИСХОДИТ НА ОДНИХ И ТЕХ РЕСУРСАХ ПРОЦЕССОРА
- ВЕРХНИЙ УРОВЕНЬ ЭМУЛИРУЕТ ДВА НЕЗАВИСИМЫХ ПРОЦЕССОРА (ДВА СОСТОЯНИЯ ПРОЦЕССОРА И КОНТРОЛЛЕРА ПРЕРЫВАНИЙ)
- ТЕХНОЛОГИЯ УВЕЛИЧИВАЕТ ПЛОЩАДЬ ПРИМЕРНО НА 5% ПРОЦЕНТОВ, А ПРОИЗВОДИТЕЛЬНОСТЬ - НА 30%

# HyperThreading



SMT-процессоры объединяют свойства суперскалярных и многонитевых процессоров. Они позволяют в одном цикле выполнять инструкции разных нитей.

# *Hyper-Threading*

- Особенности и недостатки
  - Поскольку обычно используется около 35% ресурсов процессора, технология позволяет увеличив площадь кристалла примерно на 5% процентов поднять производительность - на 30%
  - Поскольку производительность вырастает на 30%, а количество выполняемых потоков на 100%, то выигрывает система в целом, а не конкретное приложение
  - Существует опасность спада производительности при монополизации ресурсов одним потоком. Особенно опасны в этом смысле пустые циклы.

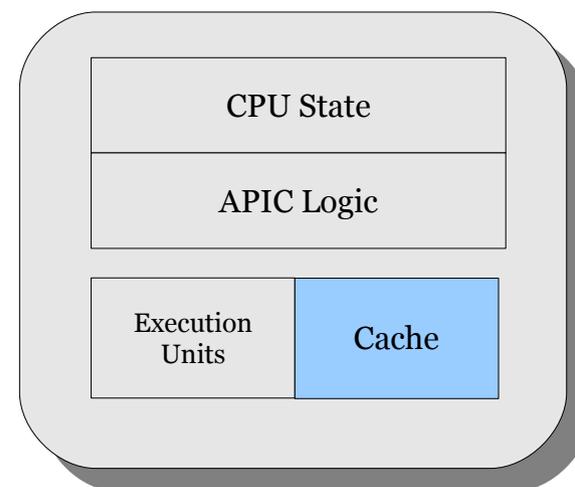
# *Hyper-Threading*

- Короткие циклы:
  - копирование памяти/строк (безопасно)
  - spin-блокировки (опасны)
- Для опасных коротких циклов — **новая команда в новых процессорах PAUSE**

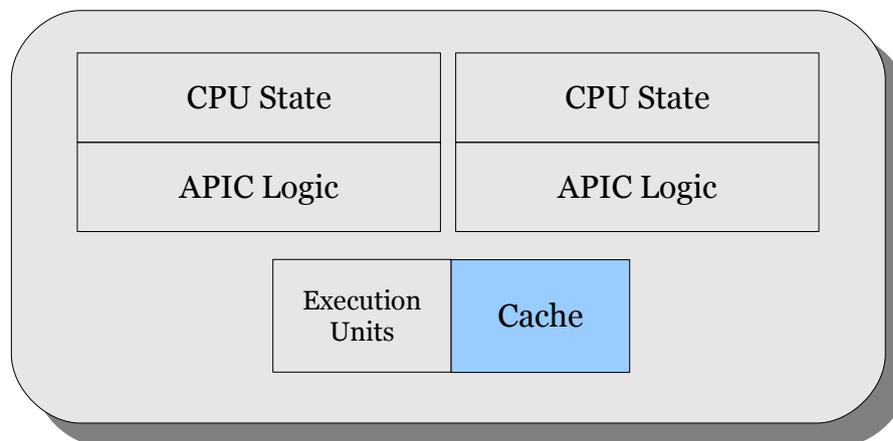
```
do {  
    _asm pause  
} while( sync_var != constant_value)
```

# Отличия между технологиями

- это обычный процессор

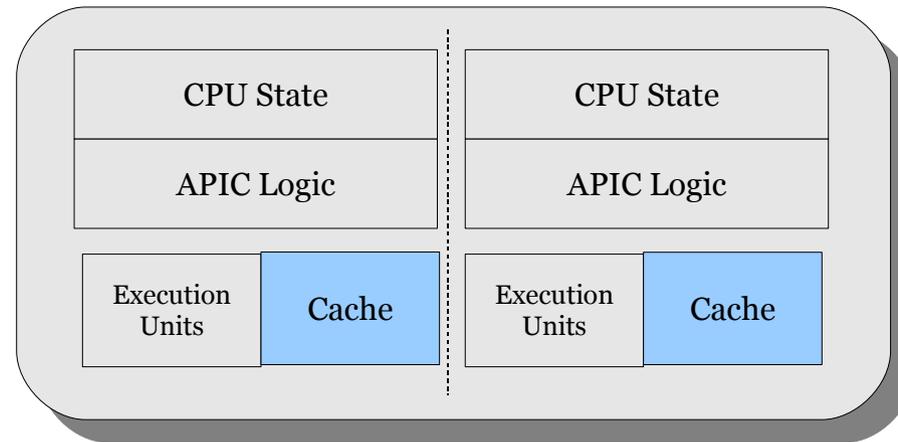


- это процессор HyperThreading

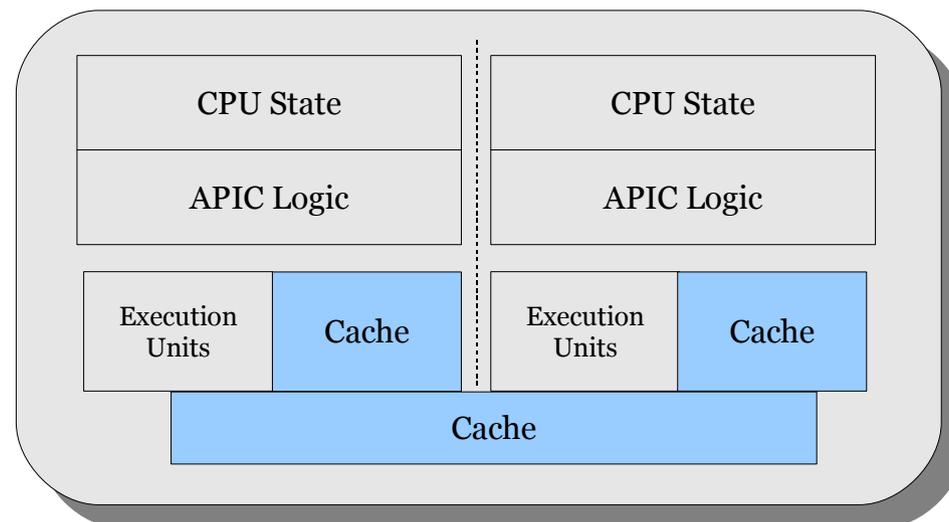


# Отличия между технологиями

- Это ранние процессоры Multicore

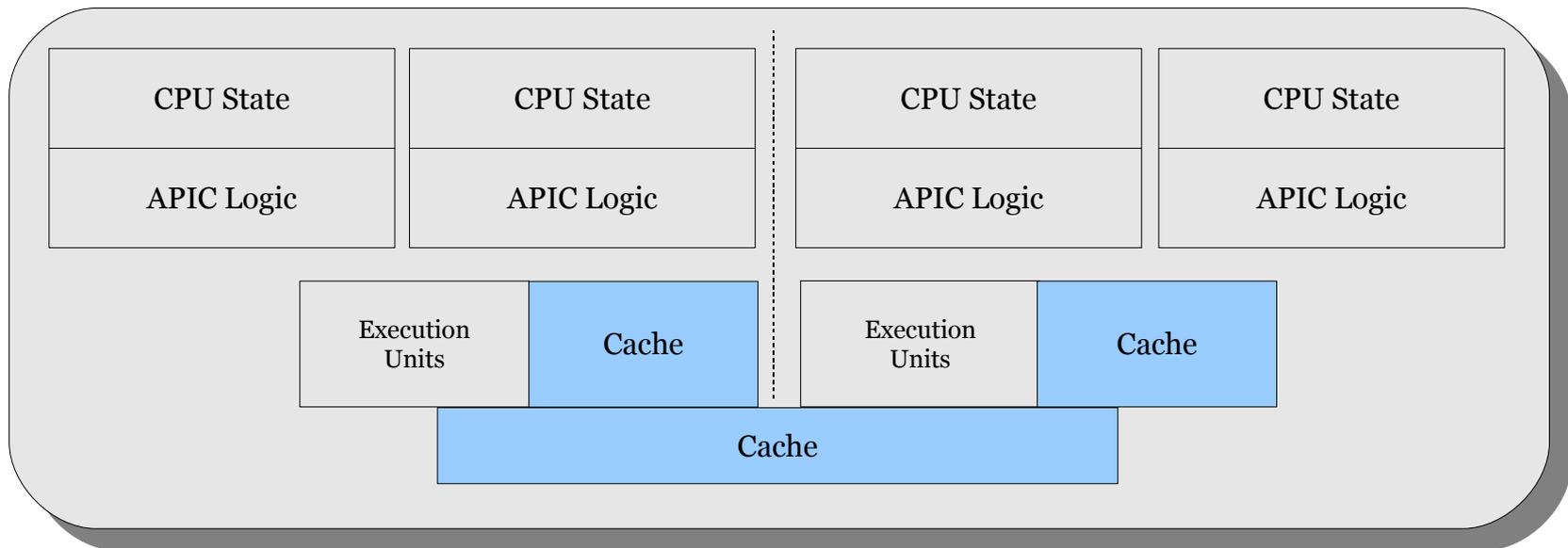


- Это современные процессоры Multicore



# Отличия между технологиями

- И, наконец, что такое Pentium Extrim и Xeon 7100



# *Библиография*

- 1. MultiProcessor Specification / Version 1.4, May 1997 – Intel Corp., 1997. – 97 p.**
- 2. Богачев К.Ю. *Архитектура процессоров* – Москва, 1999. – 127 с.**
- 3. *Developing Multithreaded Applications: A Platform Consistent Approach / Version 2.0, Feb 2005. – Intel Corp., 2005. – 128 p.***
- 4. Intel® 64 and IA-32 Architectures Software Developer's Manual – Vol. 1, 2a, 2b, 3a, 3b – Intel Corp., 2006. - 466 p.; 774 p.; 612 p.; 640 p.; 610 p.**
- 5. *Intel® 64 and IA-32 Architectures Optimization Reference Manual* – Intel Corp., 2006. – 490 p.**