Министерство образования Республики Беларусь
Учреждение образования
Гомельский государственный университет им. Ф. Скорины

Физический факультет

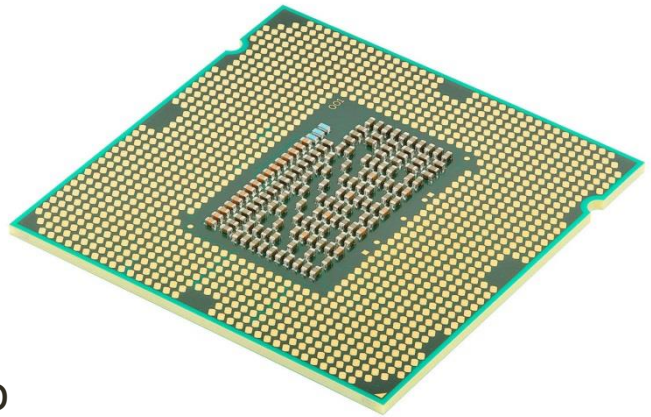
«Архитектура и ПО вычислительных систем»

Лекция – Структура современной ВС (Центральный процессор)

Лектор – ст. преподаватель Грищенко В.В.

Центральный процессор (central processing unit, CPU) — электронный блок либо интегральная схема (микропроцессор), исполняющая машинные инструкции (код программ), главная часть аппаратного обеспечения компьютера или программируемого логического контроллера. Иногда называют микропроцессором или просто процессором.

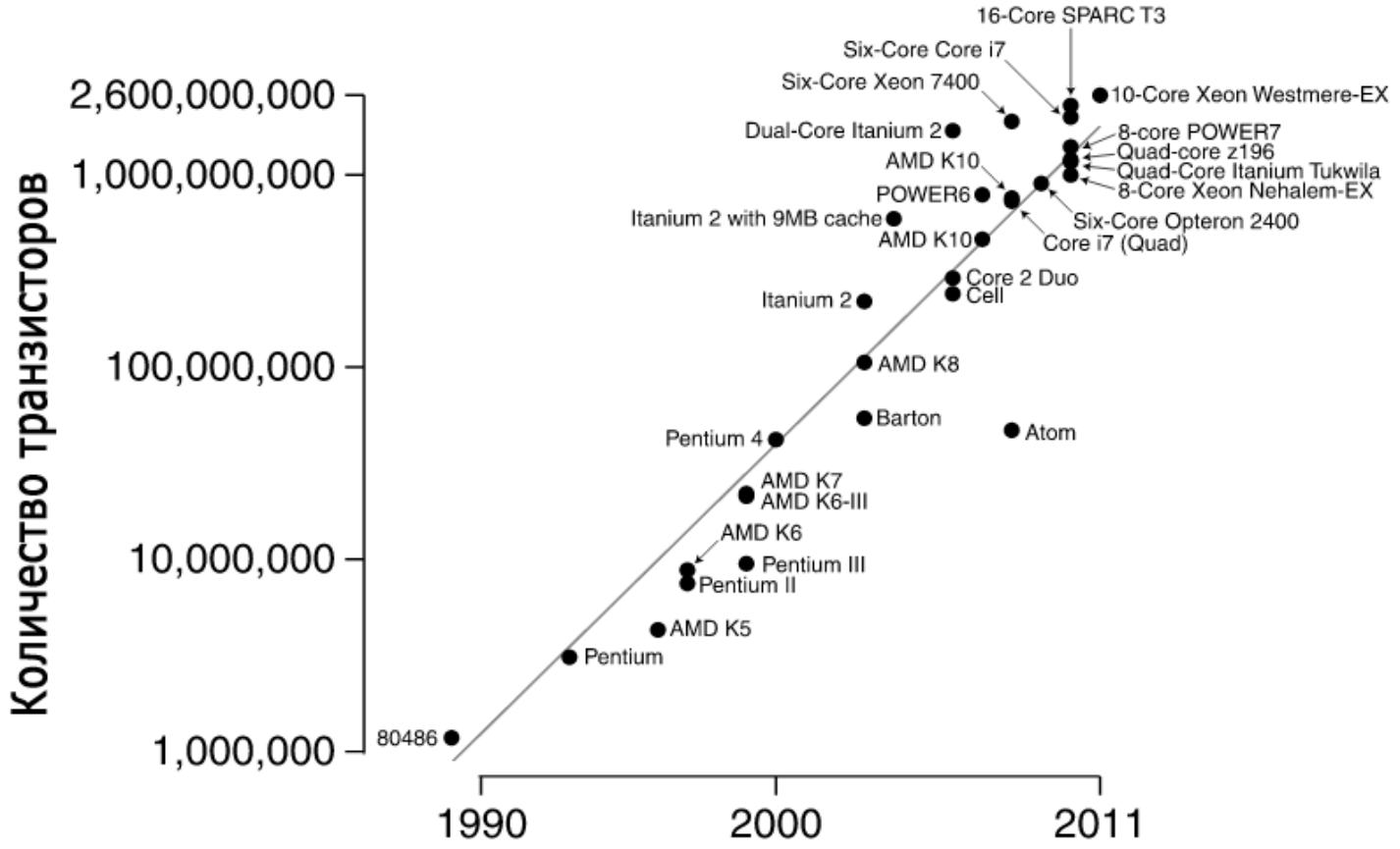
Изначально термин центральное процессорное устройство описывал специализированный класс логических машин, предназначенных для выполнения сложных компьютерных программ. Вследствие довольно точного соответствия этого назначения функциям существовавших в то время компьютерных процессоров, он естественным образом был перенесён на сами компьютеры. Начало применения термина и его аббревиатуры по отношению к компьютерным системам было положено в 1960-е годы.



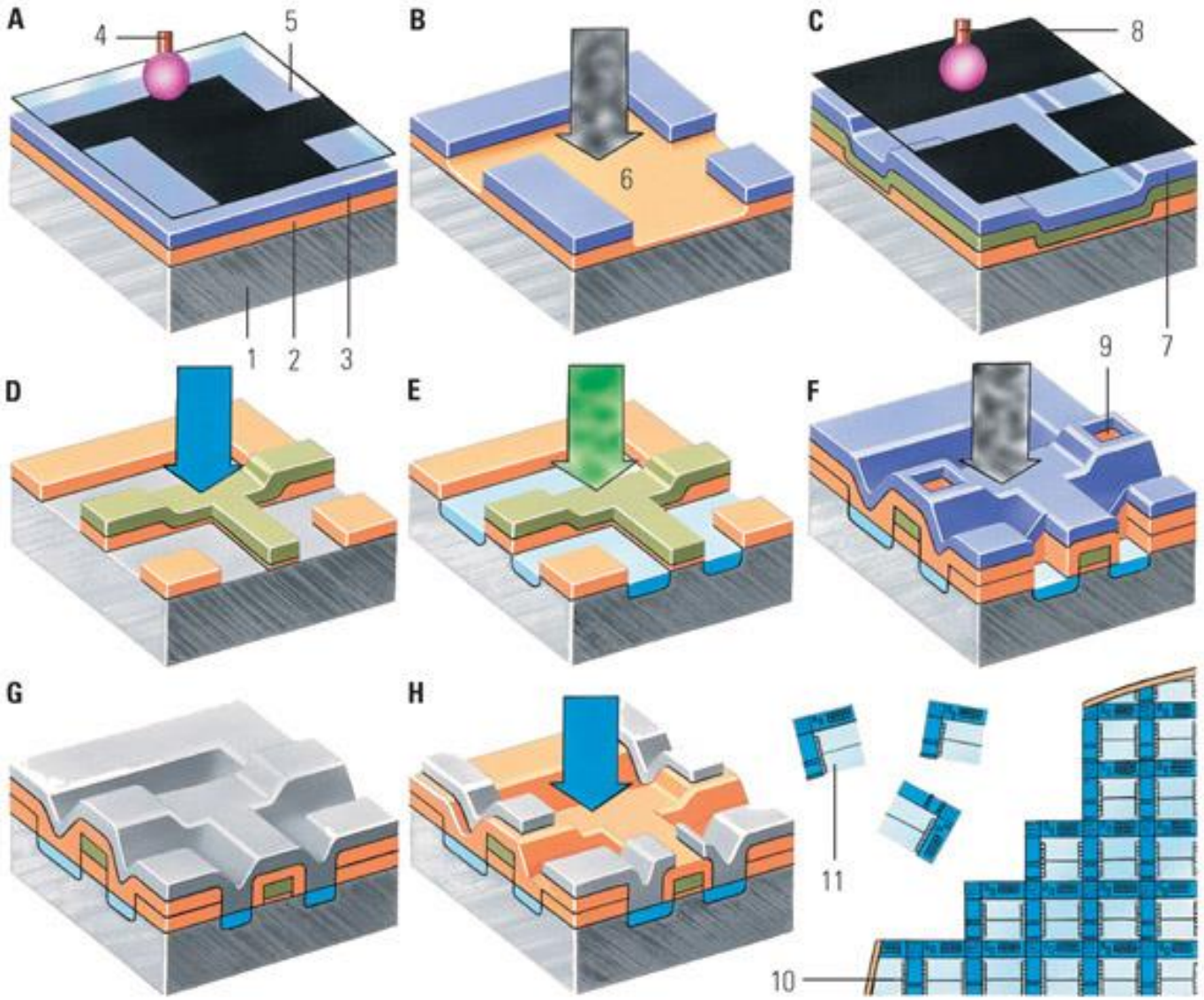
Главными характеристиками ЦПУ являются: тактовая частота, производительность, энергопотребление, нормы литографического процесса, используемого при производстве (для микропроцессоров) и архитектура.

Экономическим обоснованием роста производительности вы-числительной системы является необходимость снижения стоимости получения результатов вычислений. Стоимости по времени их получения и затраченным на их получение энергии. Уменьшение размера транзистора и длины их межсоединий – наиболее эффективный путь.

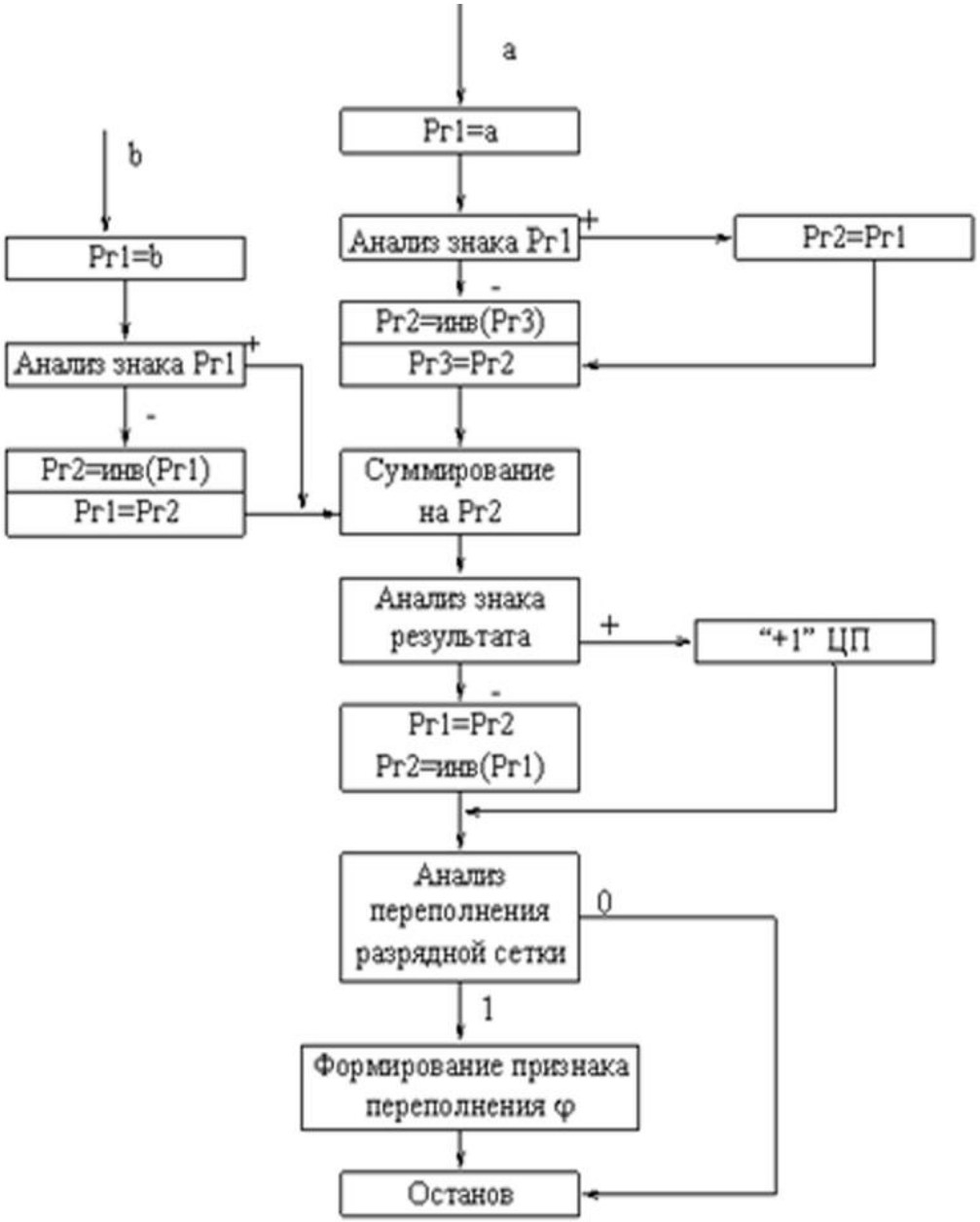
Закон Мура эмпирическое наблюдение, изначально сделанное Гордоном Муром, согласно которому (в современной формулировке) количество транзисторов, размещаемых на кристалле интегральной схемы, удваивается каждые 24 месяца. Часто цитируемый интервал в 18 месяцев связан с прогнозом Давида Хауса из Intel, по мнению которого производительность процессоров должна удваиваться каждые 18 месяцев из-за сочетания роста количества транзисторов и быстродействия каждого из них.

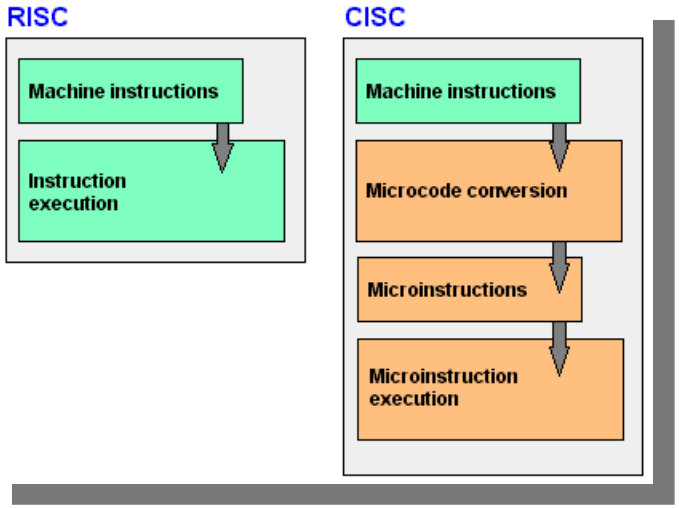


Упрощенная процедура изготовления чипа



Схему реализации процессором операции сложения можно продемонстрировать следующим рисунком.



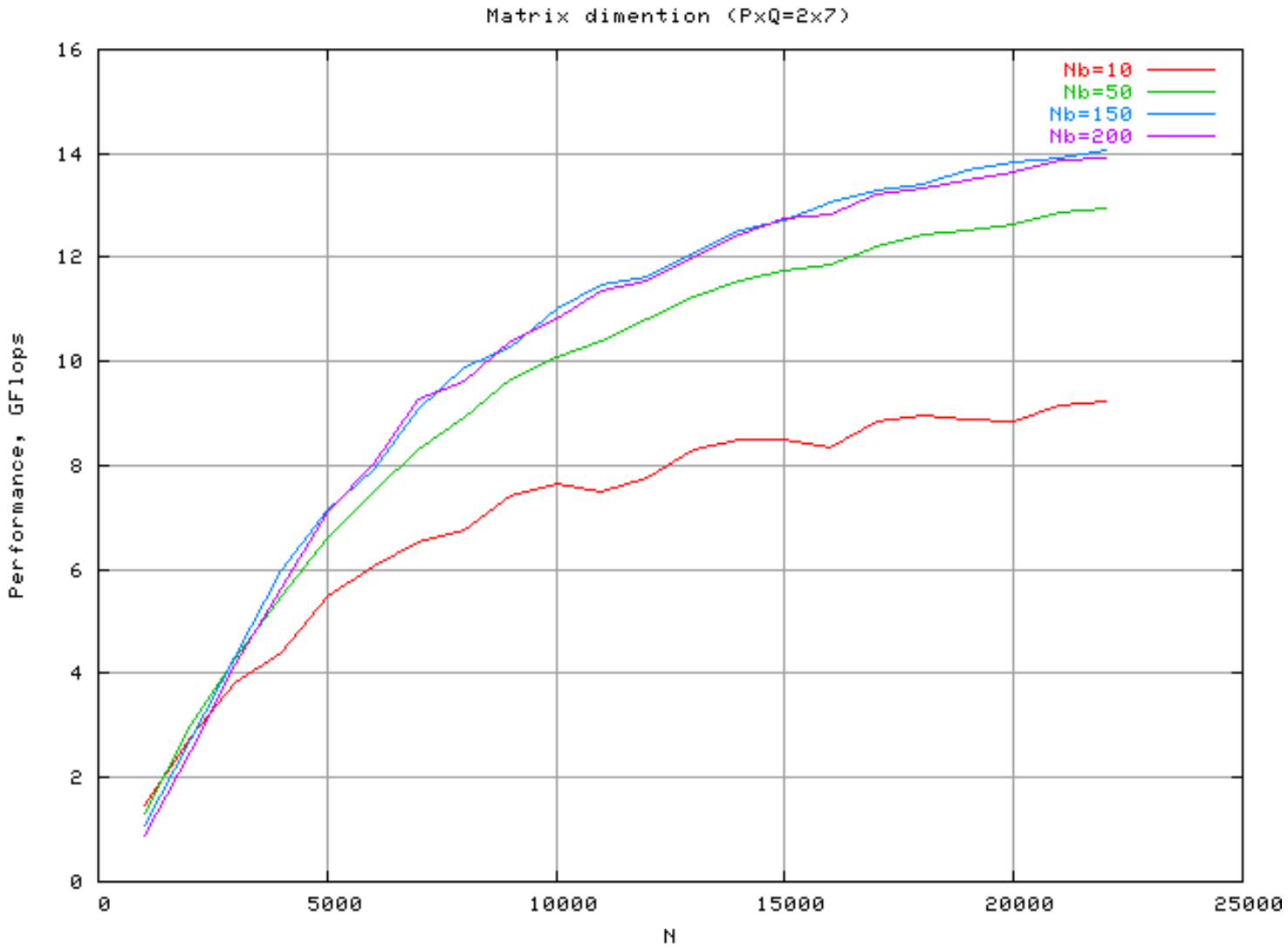


CISC (Complex instruction set computing, компьютер с полным набором команд) — концепция проектирования процессоров, которая характеризуется следующим набором свойств:
•нефиксированное значение длины команды;
•арифметические действия кодируются в одной команде;
•небольшое число регистров, каждый из которых выполняет строго определённую функцию.
Многие компиляторы не задействовали все возможности таких наборов инструкций, а на сложные методы адресации уходит много времени из-за дополнительных обращений к медленной памяти.
Было показано, что такие функции лучше исполнять последовательностью более простых инструкций, если при этом процессор упрощается и в нём остаётся место для большего числа регистров, за счёт которых можно сократить количество обращений к памяти.

RISC (*restricted (reduced) instruction set computer,* компьютер с сокращённым набором команд) — архитектура процессора, в котором быстродействие увеличивается за счёт упрощения инструкций, чтобы их декодирование было более простым, а время выполнения — меньшим.
В первых архитектурах, причисляемых к RISC, большинство инструкций для упрощения декодирования имеют одинаковую длину и похожую структуру, арифметические операции работают только с регистрами, а работа с памятью идёт через отдельные инструкции загрузки (load) и сохранения (store). Эти свойства и позволили лучше сбалансировать этапы конвейеризации, сделав конвейеры в RISC значительно более эффективными и позволив поднять тактовую частоту.

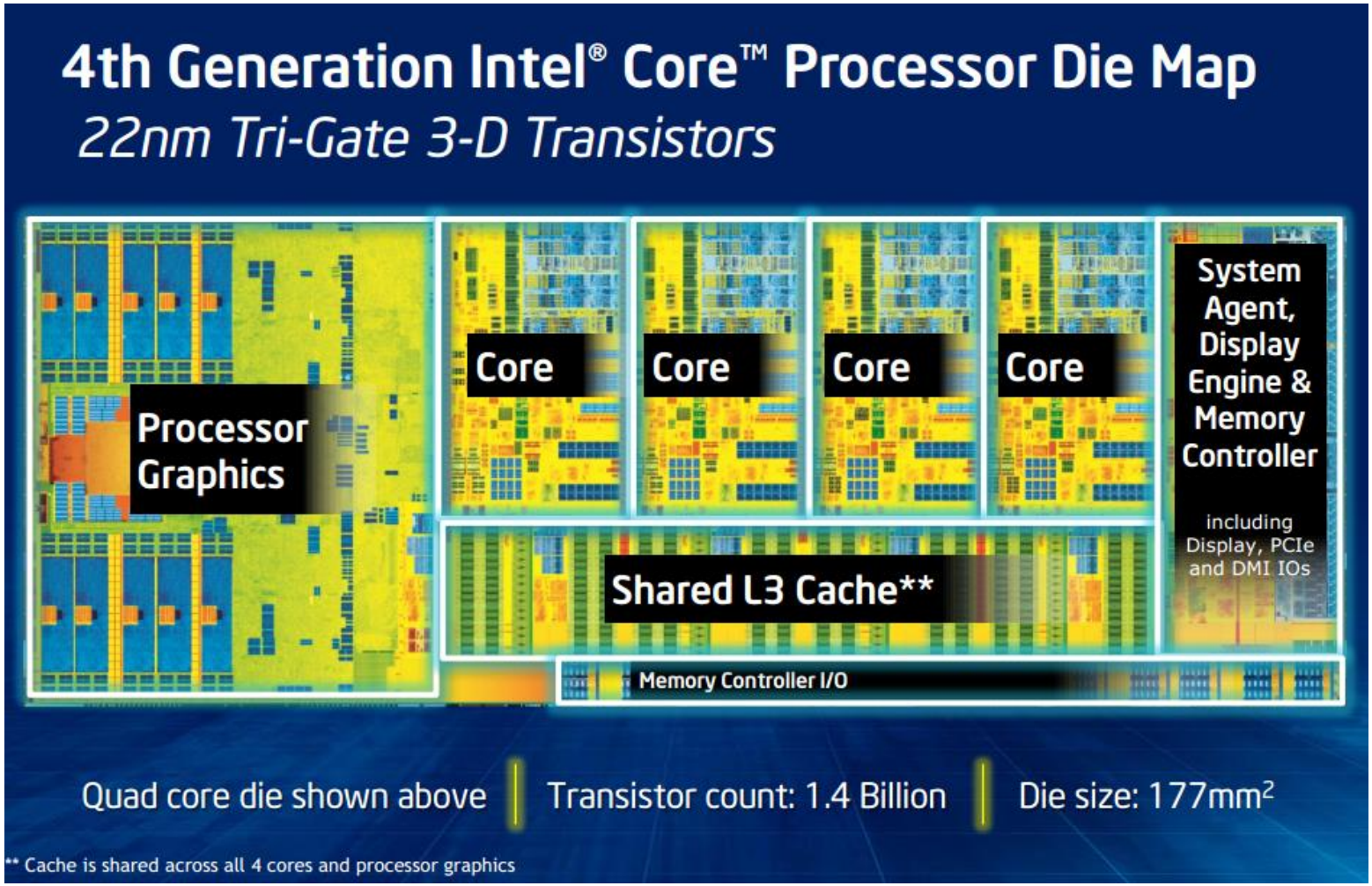
В более поздних 32-разрядных процессорах (начиная с Pentium Pro) появилось **PAE** (Physical Address Extension) — расширение адресов физической памяти до 36 бит (возможность адресации 64 Гбайт ОЗУ). Это изменение не затронуло разрядности задач — 32-бита.
MMX. Дополнительный «мультимедийный» (англ. Multi-Media eXtensions) набор инструкций, выполняющих по несколько характерных для процессов кодирования/декодирования потоковых аудио/видеоданных действий за одну машинную инструкцию. Впервые появился в процессорах Pentium MMX. Обеспечивает только целочисленные вычисления.
SSE (англ. Streaming SIMD Extensions — потоковое SIMD-расширение) — это SIMD (англ. Single Instruction, Multiple Data — «одна инструкция — множество данных») набор инструкций, разработанный Intel и впервые представленный в процессорах серии Pentium III. Поддерживает вычисления с плавающей точкой. SSE состоит из восьми 128-битных регистров (с xmm0 до xmm7). Каждый регистр определяет 4 последовательных значения с плавающей точкой одинарной точности. SSE включает в себя инструкции, которые производят операции со скалярными и упакованными типами данных. SSE2 – SSE5 - улучшенное расширение SSE. Расширение x86 инструкций SSE5 от AMD, названное SSE5 привносят в классическую x86 архитектуру некоторые возможности, доступные ранее исключительно в RISC процессорах.
AVX. Следующий набор расширений от Intel. Поддерживается обработка чисел с плавающей запятой упакованных в 256-битные "слова". Для них вводится поддержка тех же команд, что и в семействе SSE. 128-битные регистры SSE XMM0 - XMM15 расширяются до 256-битных YMM0-YMM15
AVX 2 - дальнейшее развитие AVX. Целочисленные команды SSE начинают работать с 256-битными AVX регистрами.
AES. Расширение системы команд AES — реализация в микропроцессоре шифрования AES.
Intel Post 32 nm processor extensions - набор инструкций Intel, позволяющий конвертировать числа с половинной точностью в числа с одинарной и двойной, аппаратно получать истинно случайные числа и обращаться к регистрам FS/GS.
3DNow! Набор инструкций для потоковой обработки вещественных чисел одинарной точности. Поддерживается процессорами AMD начиная с K6-2. Процессорами Intel не поддерживается. Инструкции 3DNow! используют регистры MMX в качестве операндов (в один регистр помещается два числа одинарной точности), поэтому, в отличие от SSE, при переключении задач не требуется отдельно сохранять контекст 3DNow!.

Влияние дополнительных наборов команд на производительность системы в различных приложениях.



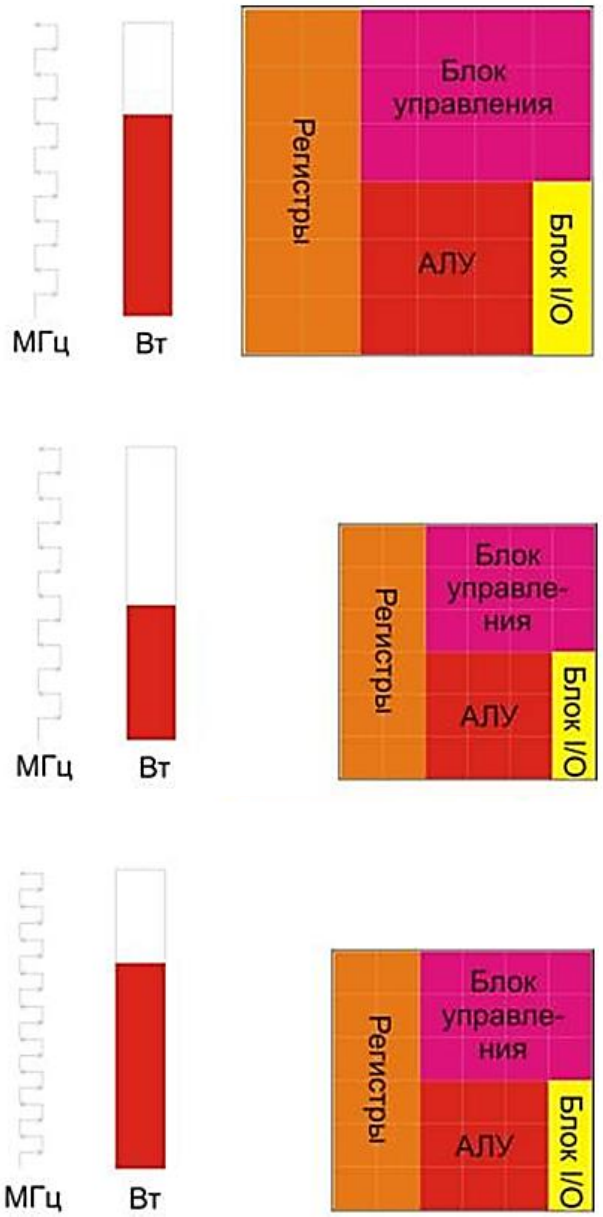
http://www.softrew.ru/instructions/266-sovremennye-processory.html

Расположение составных модулей в процессорах.



Та́ктовая частота́ — частота синхронизирующих импульсов синхронной электронной схемы, то есть количество синхронизирующих тактов, поступающих извне на вход схемы за одну секунду.

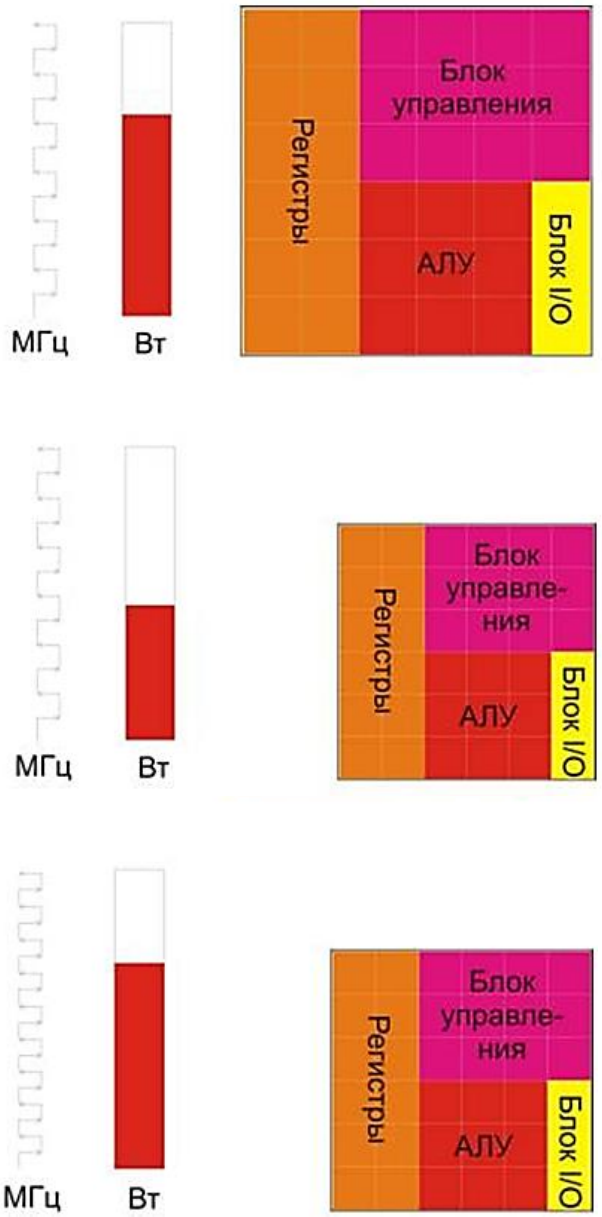
В самом первом приближении тактовая частота характеризует производительность подсистемы (процессора, памяти и пр.), то есть количество выполняемых операций в секунду. Однако системы с одной и той же тактовой частотой могут иметь различную производительность, так как на выполнение одной операции разным системам может требоваться различное количество тактов (обычно от долей такта до десятков тактов), а кроме того, системы, использующие конвейерную и параллельную обработку, могут на одних и тех же тактах выполнять одновременно несколько операций.



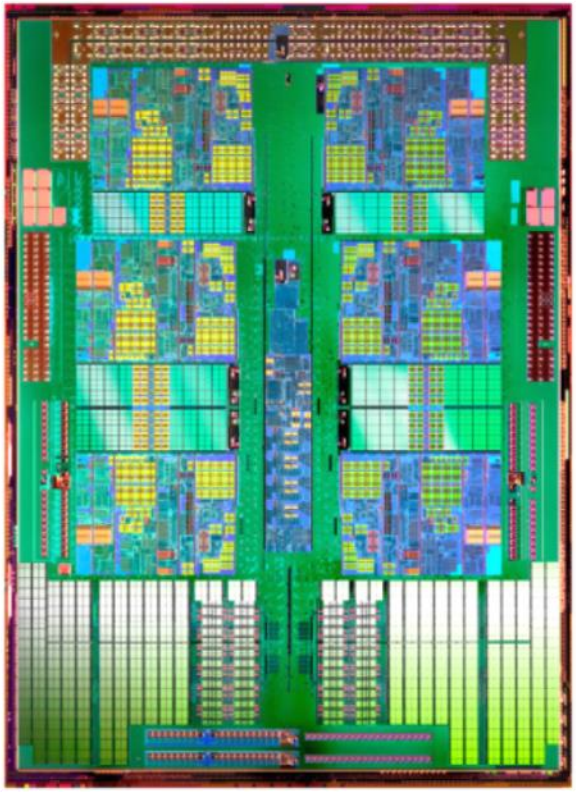
В 1961 году Рольф Ландауэр, исследователь из IBM, высказал мнение, что энергия в процессе вычислений расходуется не на что иное, как на уничтожение битов информации. На практике при стирании бита происходит выделение некоторого (очень малого) количества тепла. Но в классической фон-неймановской архитектуре значения битов в регистрах процессора переписываются огромное множество раз и объём выделяемой при этом энергии уже становится заметным.

Чем больше объём вычислений со стиранием информации, тем сильнее греются устройства, которые его производят.
…Процессор с быстродействием 100 петафлопс уже будет выделять около мегаватта тепла, а один зеттафлопсный процессор - приблизительно 10 гигаватт...
(Рольф Ландауэр)

Прогноз такого роста тепловыделения не оправдался, потому что производители электронных компонент учли законы физики и, наращивая вычислительную мощность, попутно снижают энергопотребление устройств.



FSB (Front Side Bus) — шина процессора, обеспечивающая связь ЦП с остальной периферией.
FSB работает в качестве магистрального канала между процессором и чипсетом.

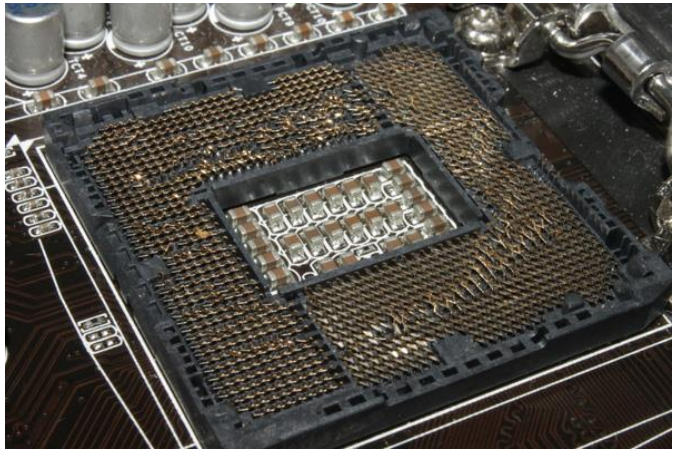


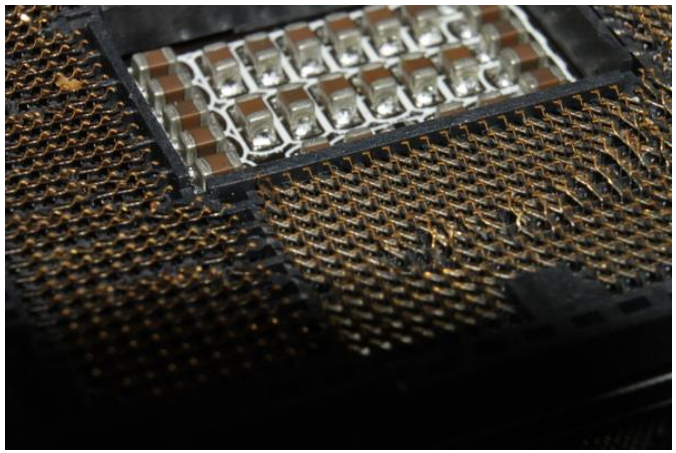
Множитель процессора (коэффициент умножения) — это число, на которое умножается частота шины.

Тактовая частота процессора вычисляется как произведение частоты шины (FSB) на коэффициент умножения.

Напряжение на ядре (от 0.45 до 1.75 В)
Номинальное напряжение питания ядра процессора.
Этот параметр указывает напряжение, которое необходимо процессору для работы (измеряется в вольтах). Он характеризует энергопотребление процессора и особенно важен при выборе CPU для мобильной, нестационарной системы.

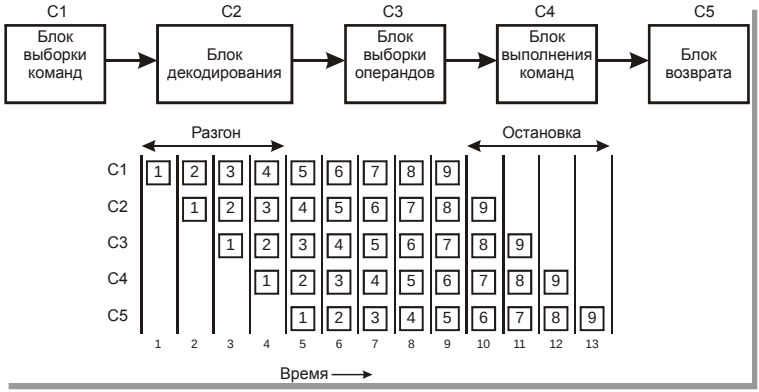
Максимальная рабочая температура (от 54.8 до 105.0 С)
Допустимая максимальная температура поверхности процессора, при которой возможна нормальная работа.
Температура процессора зависит от его загруженности и от качества теплоотвода. В холостом режиме и при нормальном охлаждении температура процессора находится в пределах 25-40°С, при высокой загруженности она может достигать 60-70 градусов.
Для процессоров с высокой рабочей температурой рекомендуются мощные системы охлаждения.





Рабочее значение	Минимальное рабочее значение	Максимальное рабочее значение
+ 3.3 V	+ 3.14 V	+ 4.5 V
+ 5 V	+ 4.75 V	+ 6.5 V
+ 12 V	+ 11.4 V	+ 14.5 V
– 12 V	– 10.8 V	– 14.5 V
+ 5 VSB	+ 4.75 V	+ 6.5 V

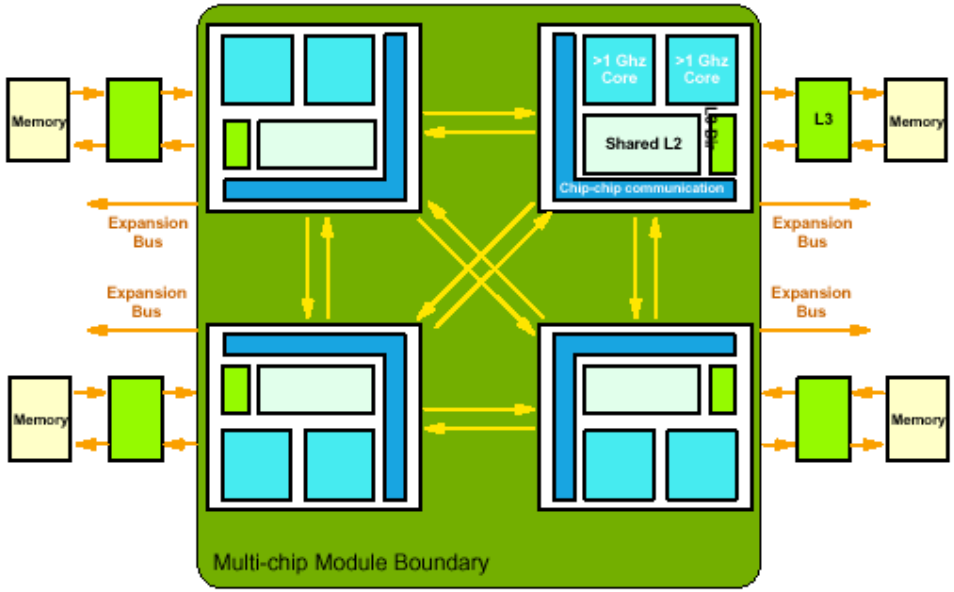
Производителям приходится решать задачу по увеличению производительности процессоров, то есть приближению их характеристик к значению потенциально возможной производительности.
К числу таких приемов можно отнести: *увеличение длины конвейера обработки команд процессора; подбор размера и структуры иерархии кэш-памяти процессора; увеличение внутренней частоты работы процессора, то есть увеличение максимального числа операций в секунду при том же наборе оборудования* и другие.



Другой эффективный путь – распараллелизация операций между несколькими процессорами или ядрами одного процессора:
•*SMP (symmetric multiprocessing)* – симметричная многопроцессорная архитектура
•*MPP (massive parallel processing)* – массивно-параллельная архитектура
•Гибридная архитектура *NUMA (nonuniform memory access, неоднородный доступ к памяти)*

Большинство современных процессоров обладают сходной архитектурой - это конвейерные суперскалярные процессоры с внеочередным (спекулятивным) исполнением инструкций, как RISC, так и x86.

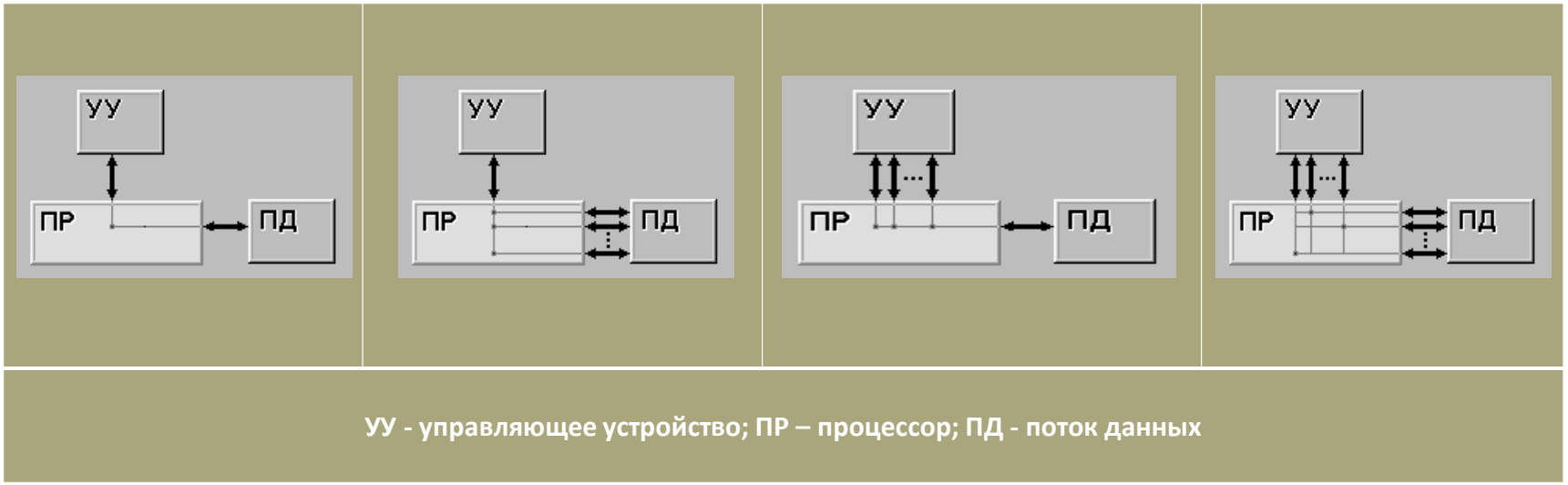
Сущность этого подхода заключается в том, что в процессоре присутствует несколько параллельно работающих функциональных устройств (FU), исполняющих по мере возможности инструкции из специального буфера, куда они поступают после декодирования. Плюсом таких процессоров является тот факт, что распараллеливание происходит независимо от программиста (по крайней мере, на языках высокого уровня) и нет необходимости использовать специальные алгоритмы и языковые конструции, использующиеся при разработке программ для машин с несколькими процессорами.

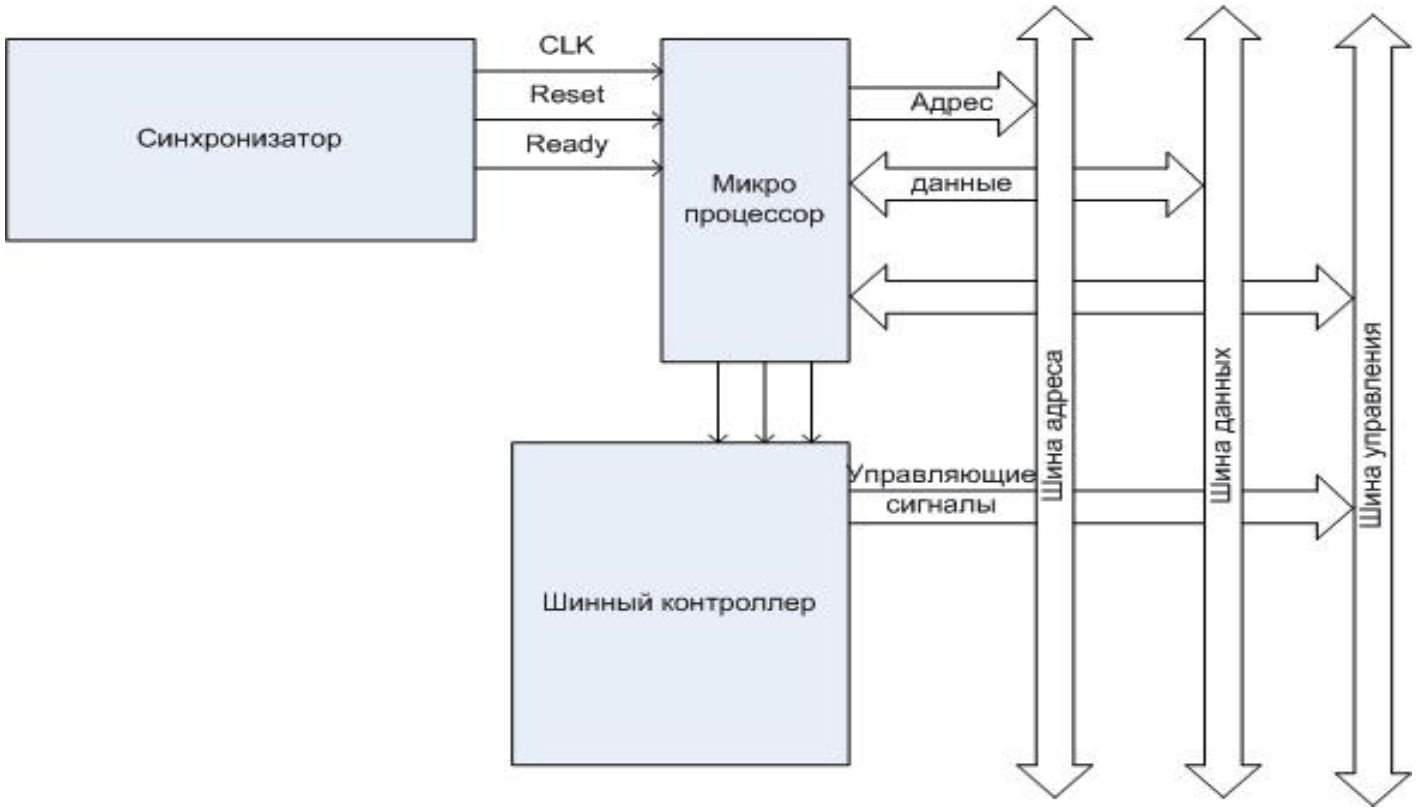


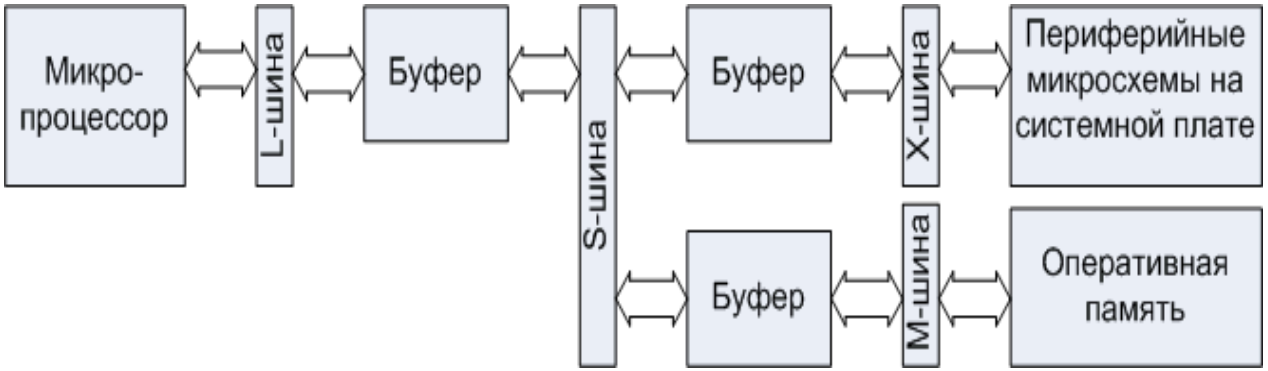
Механизм эффективно работает если команды, обрабатываемые конвейером, не противоречат друг другу, и одна не зависит от результата другой, то свободное ядро может осуществить параллельное выполнение следующей команды.

Распараллелизация вычислений

Классификация Флинна. Классификация базируется на понятии потока, под которым понимается последовательность элементов, команд или данных, обрабатываемая процессором.
Флинн выделяет четыре класса архитектур: single instruction stream / single data stream – SISD, single instruction stream / multiple data stream – SIMD, multiple instruction stream / single data stream – MISD, multiple instruction stream / multiple data stream – MIMD.




Синхронизатор
CLK
Reset
Ready
Микро процессор
Адрес
данные
Управляющие сигналы
Шинный контроллер
Шина адреса
Шина данных
Шина управления


Микро-процессор
L-шина
Буфер
S-шина
Буфер
X-шина
Периферийные микросхемы на системной плате
Буфер
M-шина
Оперативная память

В реальном режиме при вычислении линейного адреса, по которому процессор собирается читать содержимое памяти или писать в неё, сегментная часть адреса умножается на 16 (или, то же самое, что и сдвиг влево на 4 бита) и суммируется со смещением (если процессору передаётся не полный адрес из двух 16-битных значений — сегмента и смещения, — а только 16-битное смещение, то сегмент берётся из одного из сегментных регистров). Таким образом, адреса 0400h:0001h и 0000h:4001h ссылаются на один и тот же физический адрес, так как 400h×16+1 = 0×16+4001h.

Такой способ вычисления физического адреса позволяет адресовать 1 Мб + 64 Кб − 16 байт памяти (диапазон адресов 0000h…10FFEFh). Однако в процессорах 8086/8088 всего 20 адресных линий, поэтому реально доступен только 1 мегабайт (диапазон адресов 0000h…FFFFFh), а при адресации выше (в диапазоне 100000h…10FFEFh) происходит «заворот» — старший единичный бит адреса игнорируется и происходит обращение к 64 килобайтам в начальных адресах (0000h…FFEFh).

Процессор 80286 имеет 24-битную адресную шину (возможна адресация 224 = 16 Мб памяти), поэтому в них переполнения не происходит. Компьютеры IBM PC/AT построены на процессоре Intel 80286, но, из соображений совместимости с IBM PC и IBM PC/XT, построенных на Intel 808x, в них был введён логический элемент (вентиль), управляющий работой 21-го адресного провода (A20). Этот логический элемент, получивший название "Gate A20", по умолчанию отключен, что соответствует режиму совместимости, но управляется через контроллер клавиатуры (микросхема Intel 8042).

В процессоре 80286, помимо реального режима, был реализован также защищённый режим. В защищённом режиме процессор может адресовать до 16 Мбайт физической памяти и 1 Гбайт виртуальной (16384 сегмента по 64 кбайт) за счёт изменения механизма адресации. Переключение из реального режима в защищённый происходит программно и относительно просто, однако для обратного перехода необходим аппаратный сброс процессора. Для отслеживания текущего режима работы процессора используется регистр слова состояния машины (MSW). Программы реального режима без модификаций в защищённом режиме исполняться не могут, так же как и программы BIOS машины.

Суть защищённого режима в следующем: программист и разрабатываемые им программы используют логическое адресное пространство, размер которого может составлять 1 гигабайт. Логический адрес преобразуется в физический адрес автоматически с помощью схемы управления памятью (MMU). При этом содержимое сегментного регистра не связано напрямую с физическим адресом, а является номером сегмента в соответствующей таблице. Благодаря защищённому режиму, в памяти может храниться только та часть программы, которая необходима в данный момент, а остальная часть может храниться во внешней памяти (например, на жёстком диске). В случае обращения к той части программы, которой нет в памяти в данный момент, операционная система может приостановить программу, загрузить требуемую секцию кода из внешней памяти и возобновить выполнение программы. Следовательно, становятся допустимыми программы, размер которых больше объёма имеющейся памяти, и пользователю кажется, что он работает с большей памятью, чем на самом деле.

Физический адрес формируется следующим образом. В сегментных регистрах хранится селектор, содержащий индекс дескриптора в таблице дескрипторов (13 бит), 1 бит, определяющий к какой таблице дескрипторов будет производиться обращение (к локальной или к глобальной) и 2 бита запрашиваемого уровня привилегий. Далее происходит обращение к соответствующей таблице дескрипторов и соответствующему дескриптору, который содержит начальный 24-битный адрес сегмента, размер сегмента и права доступа, после чего вычисляется необходимый физический адрес путём сложения адреса сегмента со смещением из 16-разрядного регистра.

Важность защищенного режима работы процессора для функционирования современных операционных систем

Линейная адресация памяти — схема адресации памяти компьютера в защищенном режиме (начиная с Intel 80386 и других совместимых x86-процессорах). Используется большинством современных многозадачных ОС.

Благодаря механизму линейной адресации можно создавать любое (ограниченное только размерами оперативной памяти) количество независимых виртуальных адресных пространств. Причём каждая страница линейного адресного пространства может находиться по любому физическому адресу или даже быть выгруженной на диск.

При использовании линейной адресации 32-битный логический адрес делится на три части:

Номер записи в каталоге страниц (номер таблицы страниц) — биты 31-22 (10 бит). Одна запись из каталога страниц определяет отображение 4 МБайт адресного пространства.
Номер записи в таблице страниц (номер страницы в таблице страниц) — биты 21-12 (10 бит). Одна запись из таблицы страниц определяет отображение 4 КБайт адресного пространства.
Смещение в рамках страницы — биты 11-0 (12 бит).
При использовании страниц по 4 МБайт вторая часть отсутствует. Смещение же в странице будут определять биты 21-0 (22 бита).

Для включения линейной адресации необходимо, находясь в защищенном режиме, установить бит PG в регистре CR0. Предварительно необходимо создать в памяти каталог страниц (англ. Page Directory, PD) и таблицы страниц (англ. Page Table, PT), после чего в регистр CR3 загрузить физический адрес каталога страниц.

Важность защищенного режима работы процессора для функционирования современных операционных систем

В качестве расширенной поддержки реального режима i386 позволяет одной или нескольким задачам работать в виртуальном режиме — режиме эмуляции режима реального адреса.

Важно понимать, что «виртуальный режим», несмотря на похожесть названия, является не «третьим режимом работы процессора» (то есть реальный, защищенный и виртуальный), а лишь режимом работы задачи в многозадачном окружении защищенного режима.

Виртуальный режим предназначается для одновременного выполнения программ реального режима (например, программы для DOS) под многозадачной операционной системой защищенного режима.

Выполнение в виртуальном режиме практически идентично реальному, за несколькими исключениями, обусловленными тем, что виртуальная задача выполняется в защищенном режиме:

- виртуальная задача не может выполнять привилегированные команды, потому что имеет низший уровень привилегий;
- все прерывания и исключения обрабатываются операционной системой защищённого режима (которая, впрочем, может инициировать обработчик прерывания виртуальной задачи), вместе с тем в задаче виртуального режима можно использовать:
- страничное преобразование, например, для:

1) расширения памяти, путем включения страниц в неиспользуемое адресное пространство
2) эмуляции расширений с переключением банков (например, EMS-памяти)
3) виртуальной развертки или свертки буферов внешних устройств (видеопамять, аппаратная EMS-память)

- эмуляцию внешних устройств через эмуляцию портов ввода-вывода
- отладку
- при выполнении нескольких задач виртуального режима, каждая из них может выполняться совершенно отдельно друг от друга, чего нельзя достигнуть в реальном режиме