

Архитектура и микроархитектура процессора



Архитектура процессора – это его программная модель, то есть программно-видимые свойства.

Микроархитектура процессора – это внутренняя реализация этой программной модели.

Микроархитектура

Переименование регистров(register renaming)

Исполнение по предложению
(Speculative execution)

Конвейеризация
(pipelining)

Продвижение данных
(data forwarding)

Предсказание переходов
(branch prediction)

Исполнение с изменением последовательности инструкции(out-of-order execution)

Конвейеризация (pipelining):

Предполагает разбивку выполнения каждой инструкции на несколько этапов, причем каждый этап выполняется на своей ступени конвейера процессора

Одновременно может обрабатываться несколько инструкций, и производительность процессора можно оценивать темпом выхода инструкций со всех его конвейеров

Для достижения максимальной производительности процессора
Надо обеспечить полную загрузку конвейеров с минимальным числом лишних штрафных циклов (penalty cycles).

Суперконвейерная архитектура в настоящее время имеет от 20 конвейеров

При записи промежуточных результатов устанавливается соответствие логических имен и физических регистров

Переименование регистров (register renaming):

Позволяет обойти архитектурное ограничение на возможность параллельного исполнения инструкций (доступно всего лишь 8 общих регистров)

Т.о., одновременно может исполняться несколько инструкций, ссылающихся на одно и то же логическое имя регистра, при условии, что между ними нет фактических зависимостей по данным.

Продвижение данных (data forwarding):

Подразумевает начало исполнения инструкции до готовности всех операндов. При этом выполняются все возможные действия, и декодированная инструкция с одним операндом помещается в исполнительное устройство, где дожидается готовности второго операнда, выходящего с другого конвейера.

Исполнение с изменением последовательности инструкции (out-of-order execution):

Изменяется порядок внутренних манипуляций данными, а внешние (шинные) операции ввода-вывода и записи в память выполняются в порядке предписанном программным кодом.

- 1) Свойственно RISC-архитектуре,
- 2) Блокирует несовершенства разрядностей при совместимости различного программного кода

Предсказание переходов (branch prediction):

Позволяет продолжать выборку и декодирование потока инструкций после выборки инструкций ветвления (условного перехода), не дожидаясь проверки условия.

Исполнение по предположению (Speculative execution):

предсказанные после перехода инструкции не только декодируются, но и по возможности исполняются до проверки условий перехода.

+ если сбывается – удача; - если нет – конвейер простаивает несколько тактов



Reduced (restricted) Instruction
Set Computer

$\exists \{ \Delta \square \}$

Complete Instruction
Set Computer

□

RISC

CISC



- Имеет набор однородных регистров универсального назначения (**достаточно большое количество**);
- Коды и инструкции имеют четкую структуру с фиксированной длиной;
- Небольшие затраты на декодирование и исполнение инструкций за минимальное число тактов синхронизации;
- **Унификация регистров.**

- Состав и набор их регистров существенно неоднородны;
- **Широкий набор команд усложняет декодирование инструкций и, как следствие, расходуются аппаратные средства;**
- **Возрастает число тактов, необходимых для выполнения инструкций.**



В процессорах семейства x86, применяется комбинированная архитектура – CISC-процессор имеет RISC-ядро

NetBurst: до и после

ДО

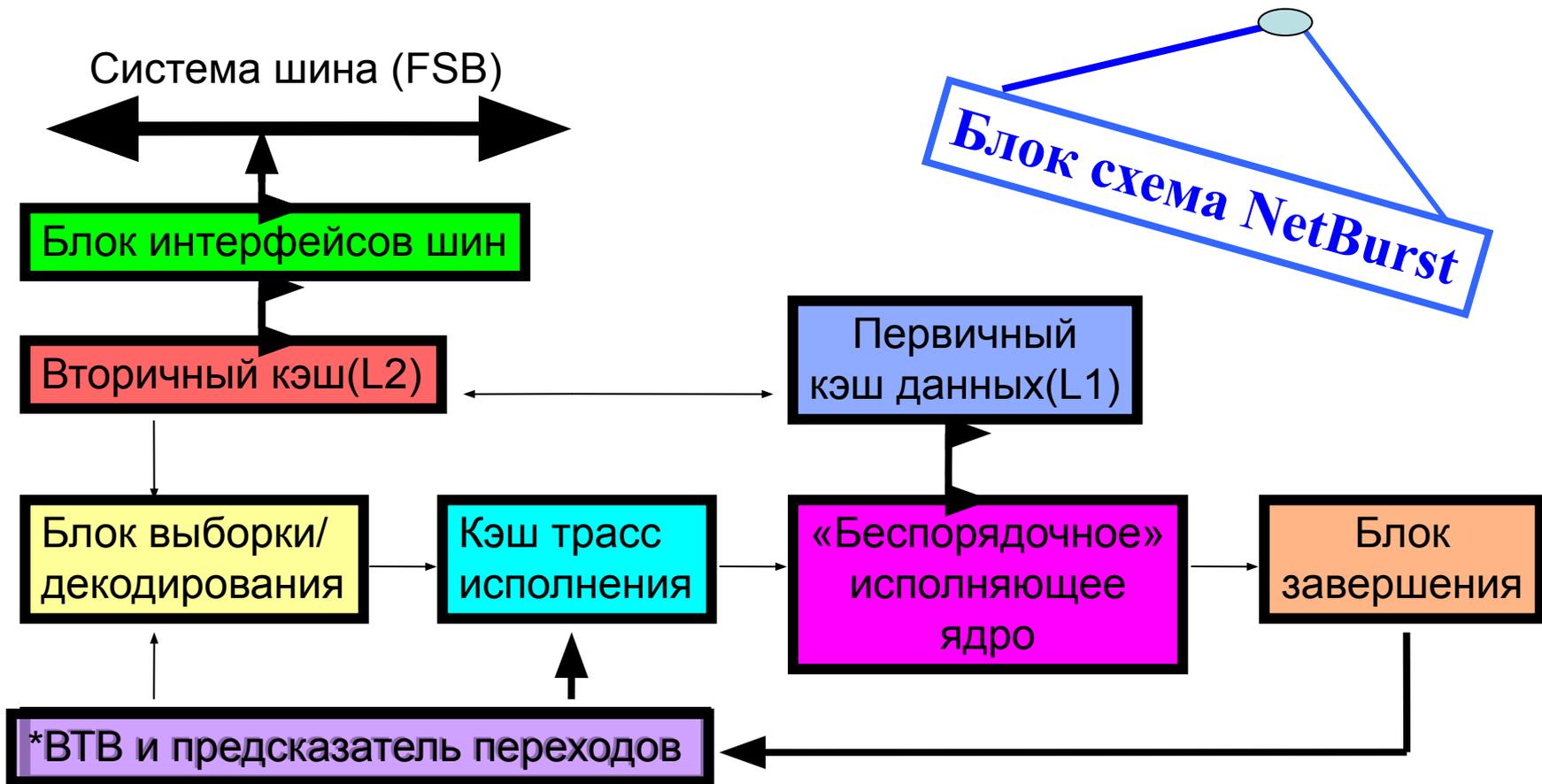
Поколение P6: суперскалярная гиперконвейерность.
Боролись за рост тактовой частоты.

NetBurst: отсутствие первичного кэша инструкций, в котором хранились копии фрагментов ОЗУ и вторичного кэша, содержащие ранее исполненные инструкции и следующие за ними строки.

Cache L1 instructions => Executive Trace Cache (кэш трасс исполнений)



Трассами называют последовательности микроопераций, в которые декодированы инструкции.



* ВТВ = Buffer Table Branch

Толстыми линия изображены наиболее используемые пути

Кэш трасс совместно с блоком выборки и декодирования образуют *устройство предварительной обработки*, выполняющее функции:

- Предварительную выборки инструкций, которые предполагается исполнить;
- Декодирование инструкции в микрооперации;
- Генерацию кодов для сложных инструкций;
- Доставку декодированных инструкций из кэша трассы;
- Предсказание переходов, использует статические и динамические методы

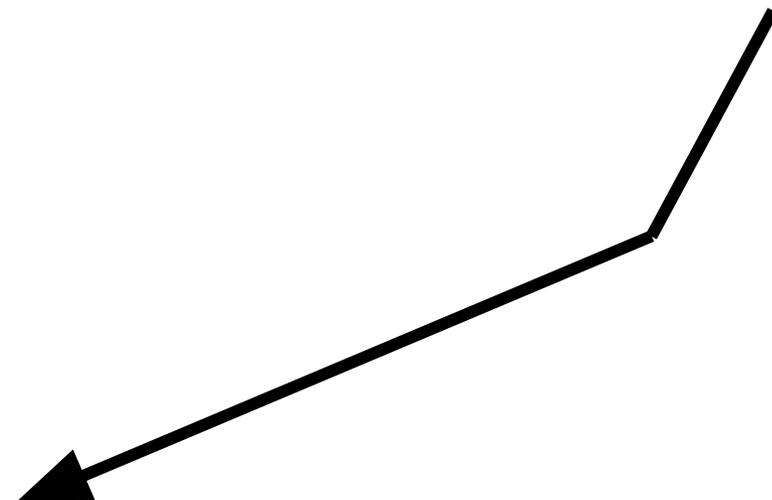
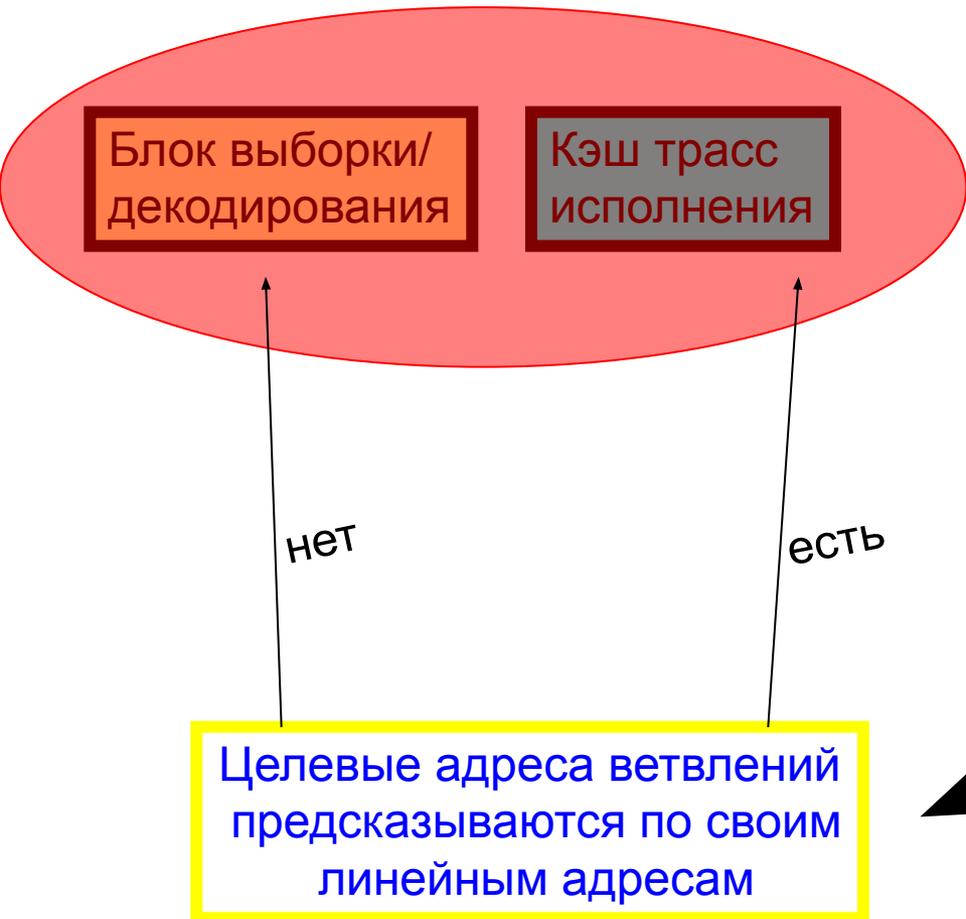
Блок выборки/
декодирования

Кэш трасс
исполнения

Способен
хранить до
12 К операций

Кэш трасс совместно с блоком
выборки и декодирования образуют
устройство предварительной обработки,
выполняющее функции:

- Предварительную выборки инструкций,
которые предполагается исполнить;



ВТВ и предсказатель переходов

Статическое предсказание

□ Условные переходы назад сбудутся (типовой цикл)

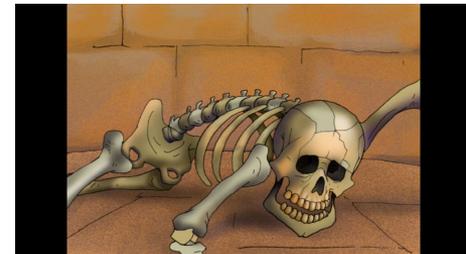
□ Условные переходы вперед – нет

Кроме того, данный тип предсказаний используется тогда, когда линейный адрес инструкции отсутствует в буфере ВТВ

Динамическое предсказание

□ Накопление статистики прохождения данных инструкций, по которой и принимается решение о том, какую ветвь прорабатывать конвейеру.

+ Branch hints: 3Eh – будет, 2Eh - нет



«Беспорядочное»
исполняющее
ядро

Имеет пиковую
пропускную способность,
превышающую
возможности блока
предварительной
обработки и блока
завершения

По сравнению с R6, стало
более
производительным по
количеству
тактов, требуемых для
исполнения
микроинструкций

Программная модель процессора: Введение

Процессор может работать в одном из двух режимов и переключаться между ними достаточно быстро, как в ту, так и в другую сторону:

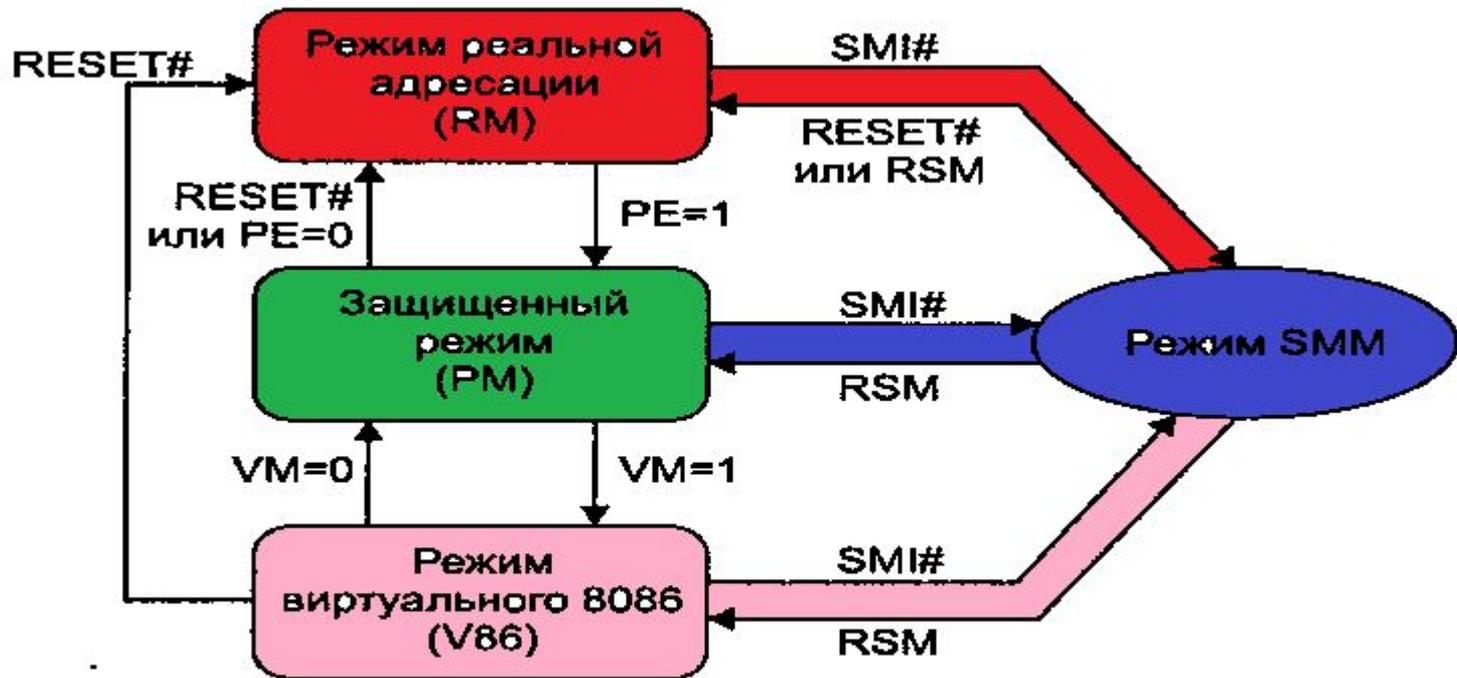
Real Address Mode — режим реальной адресации (или просто реальный режим — Real Mode). В этом режиме возможна адресация до 1 Мбайт физической памяти

Protected Virtual Address Mode — защищенный режим виртуальной адресации (или просто защищенный режим — Protected Mode). В этом режиме процессор позволяет адресовать до 4 Гбайт физической памяти, через которые при использовании механизма страничной адресации могут отображаться до 64 Тбайт виртуальной памяти каждой задачи.

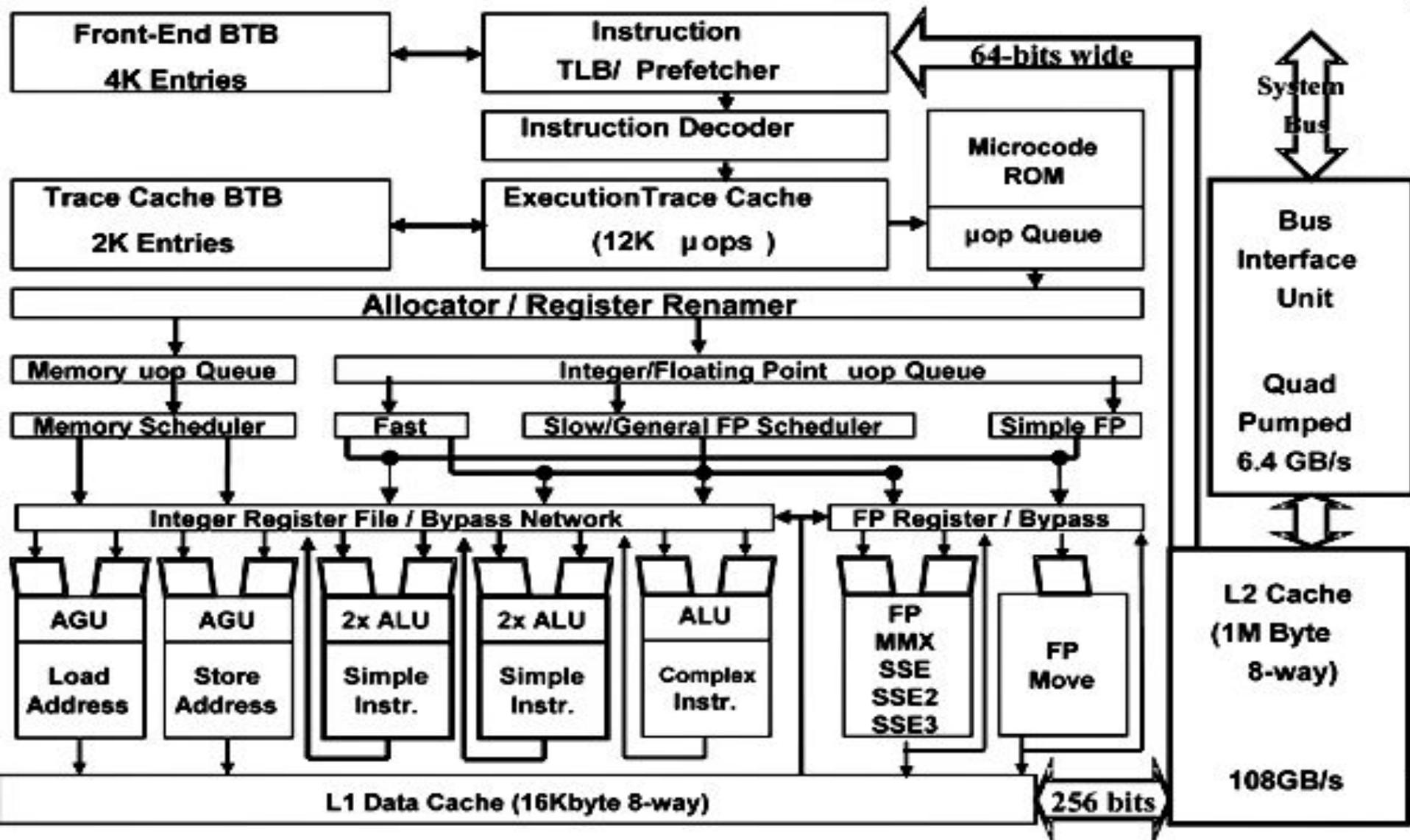
Существенным дополнением является Virtual 8086 Mode — режим виртуального процессора 8086. Этот режим является особым состоянием задачи защищенного режима, в котором процессор функционирует как 8086. На одном процессоре в таком режиме может параллельно исполняться несколько задач с изолированными друг от друга ресурсами. При этом использование физического адресного пространства памяти управляется механизмами сегментации и трансляции страниц. Попытки выполнения недопустимых команд, выхода за рамки отведенного пространства памяти и разрешенной области ввода-вывода контролируются системой защиты.

«Неофициальный» режим Big Real Mode, он же Unreal Mode, который поддерживают все 32-разрядные процессоры, позволяет адресоваться ко всему 4-гигабайтному пространству памяти. В этом режиме инструкции исполняются так же, как и в реальном режиме, но с помощью дополнительных сегментных регистров FS и GS программы получают непосредственный доступ к данным во всей физической памяти.

особый режим системного управления *System Management Mode (SMM)*, при котором процессор выходит в иное, изолированное от остальных режимов пространство памяти. Этот режим используется в служебных и отладочных целях.



Граф переходов между режимами процессора



Используемая литература.

- Книга «Процессоры «Pentium 4, Athlon и Duron»», авторы Михаил Гук, Виктор Юров
- Книга «Ассемблер. Учебник для ВУЗов», авторы Михаил Гук, Виктор Юров
- Книга «Архитектура ЭВМ», автор Мюллер
- <http://www.intuit.ru/department/hardware/csorg/8/2.htm>
- <http://ar-pc.ru/pc/200636/>
- soft-tlt.ru/programmsi1.html
- <http://cs.njit.edu/~sohn/cs650/>





Регистры процессора



Иркутский
государственный университет

