

Структурная и функциональная организация ЭВМ (Computer Organization and Design)

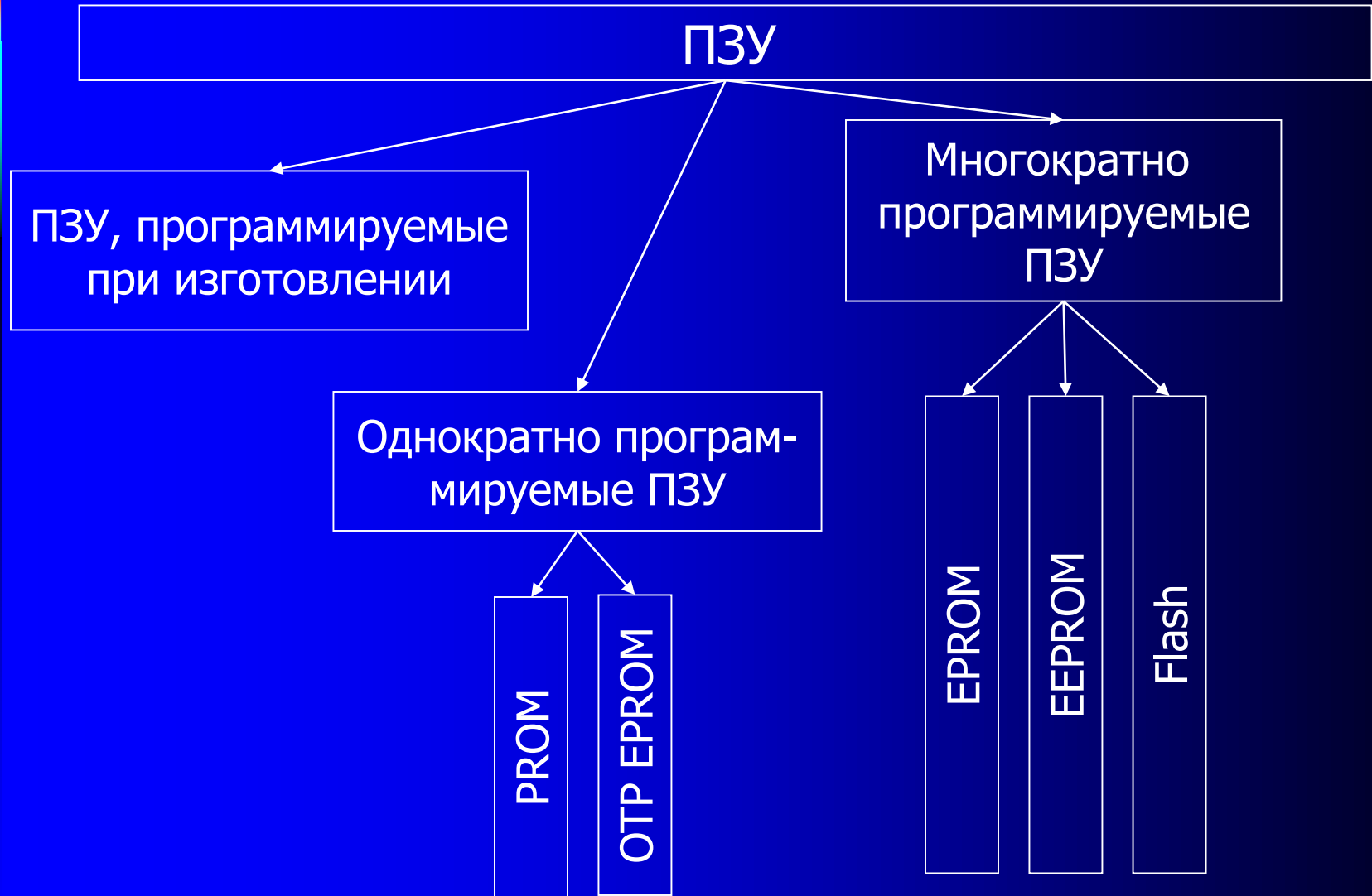
БГУИР
кафедра ЭВМ

Лекция 19
«Организация памяти-V»

План лекции

1. ПЗУ
2. Флэш-память
3. Ассоциативные ЗУ
4. Организация Кэш-памяти

Постоянные ЗУ



Постоянные ЗУ

ПЗУ – **Read Only Memory (ROM)**. Программируемые при изготовлении – MROM (Mask ROM).

Занесение информации в ПЗУ – часть технологического процесса изготовления микросхемы. Различные технологии, но чаще всего – отсутствие либо наличие перемычки (транзистора) на пересечении адресной и разрядной линии. Наиболее дешёвый вид ПЗУ – высокая плотность упаковки ЗЭ на кристалле и высокая скорость считывания. Пример применения – ПЗУ шрифтов в лазерных принтерах.

Постоянные ЗУ

Однократно программируемые ПЗУ – полуфабрикат, занесение информации – на этапе изготовления мелкосерийного устройства, использующего ПЗУ.

PROM – **Programmable ROM**. Первые микросхемы – на базе плавких предохранителей. Занесение информации – пережигание отдельных перемычек (пробоя отдельных диодов). Основной недостаток – необходимость спец. программатора, большой процент брака, необходимость термической постобработки схемы (- иначе низкая надёжность хранения).

ОТР EPROM – **One Time Programmable EPROM** – однократно программируемая EPROM – кристалл EPROM в пластиковом корпусе (без кварцевого окна).

Многократно программируемые ПЗУ

В многократно программируемых два этапа – стирание всего + запись новой информации.

EPROM – **Erasable Programmable ROM** – запись электрическими сигналами, но стирание – воздействием ультрафиолетового облучения через кварцевое окно корпуса микросхемы. Для исключения случайного стирания – заклеивание окна непрозрачной плёнкой. Стирание – многократное (20 мин). Иногда называют UV-EPROM - **Ultra Violet EPROM**. Время программирования – несколько сотен миллисекунд. Время считывания как у ROM и DRAM. Дороже, чем PROM, но выгоднее в итоге.

Многократно программируемые ПЗУ

EEPROM – **Electrical Erasable Programmable ROM** – стирание и запись по-байтово, причём стирание – как обязательный этап записи. Запись – намного дольше, чем чтение – несколько сотен микросекунд на байт. Не требует программатора – запись реализуется внутренней логикой микросхемы.

Два варианта – с последовательным и параллельным доступом. 90% всех микросхем – **SEEPROM – Serial EEPROM** – адреса, данные и управление – по одному входу. Преимущества – малые габариты, минимальное число линий ввода/вывода. Недостатки – большое время доступа, цена – EEPROM дороже EPROM.

FLASH ПЗУ

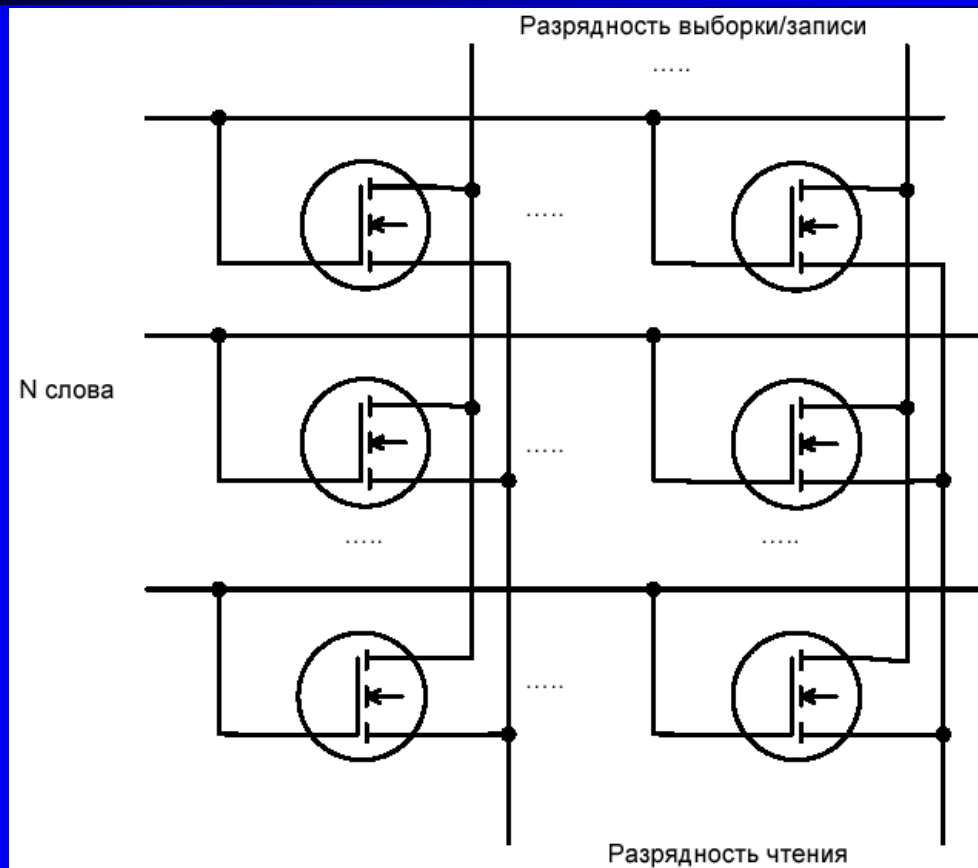
FLASH - Флэш-память появилась в конце 1980-х годов (Intel) – перепрограммируемая ПЗУ с электрическим стиранием. Стирание осуществляется сразу целой области ячеек : блока или всей микросхемы -> более быстрая запись информации (программирование ЗУ).

Для упрощения записи – спец. блоки ИМС, делающие запись “прозрачной ” для ПО и внешних устройств.

Флэш-память строится на одностранзисторных элементах памяти (с “плавающим ” затвором), -> плотность хранения информации чуть выше, чем DRAM.

Различные технологии построения базовых элементов флэш - количество слоев, методы стирания и записи данных, структура. Наиболее широко известны **NOR** и **NAND** типы флэш-памяти , запоминающие транзисторы в которых подключены к разрядным шинам - параллельно и последовательно.

FLASH ПЗУ



Последовательный способ (NAND) организации матрицы Флэш.
NOR –

FLASH ПЗУ

Минусы – относительно невысокая скорость передачи данных.

Элементы памяти флэш -ЗУ организованы в матрицы, разрядность данных для микросхем - 1-2 байта .

Операция чтения из флэш - как в обычных ЗУ с произвольным доступом (ОЗУ или кэш). Но запись – с некоторыми особенностями, аналогичными - ПЗУ. Перед записью данных в ячейки они должны быть очищены (стерты). Стирание - перевод элементов памяти в состояние единицы и возможно только сразу для целого блока ячеек (в первых микросхемах предусматривалось стирание только для всей матрицы сразу- Bulk Erase). Выборочное - невозможно.

В процессе записи информации соответствующие элементы памяти переключаются в нулевое состояние. Также , как и в ПЗУ , без стирания можно дозаписать нули в уже запрограммированные ячейки, однако необходимость в такой операции относительно редка.

FLASH ПЗУ

Фактически при операции записи производится два действия: стирание и запись, но управление этими операциями производится внутренним автоматом и “прозрачно” для процессора.

Разбиение адресного пространства микросхемы флэш -памяти на блоки обычно двух видов: симметричное и асимметричное.

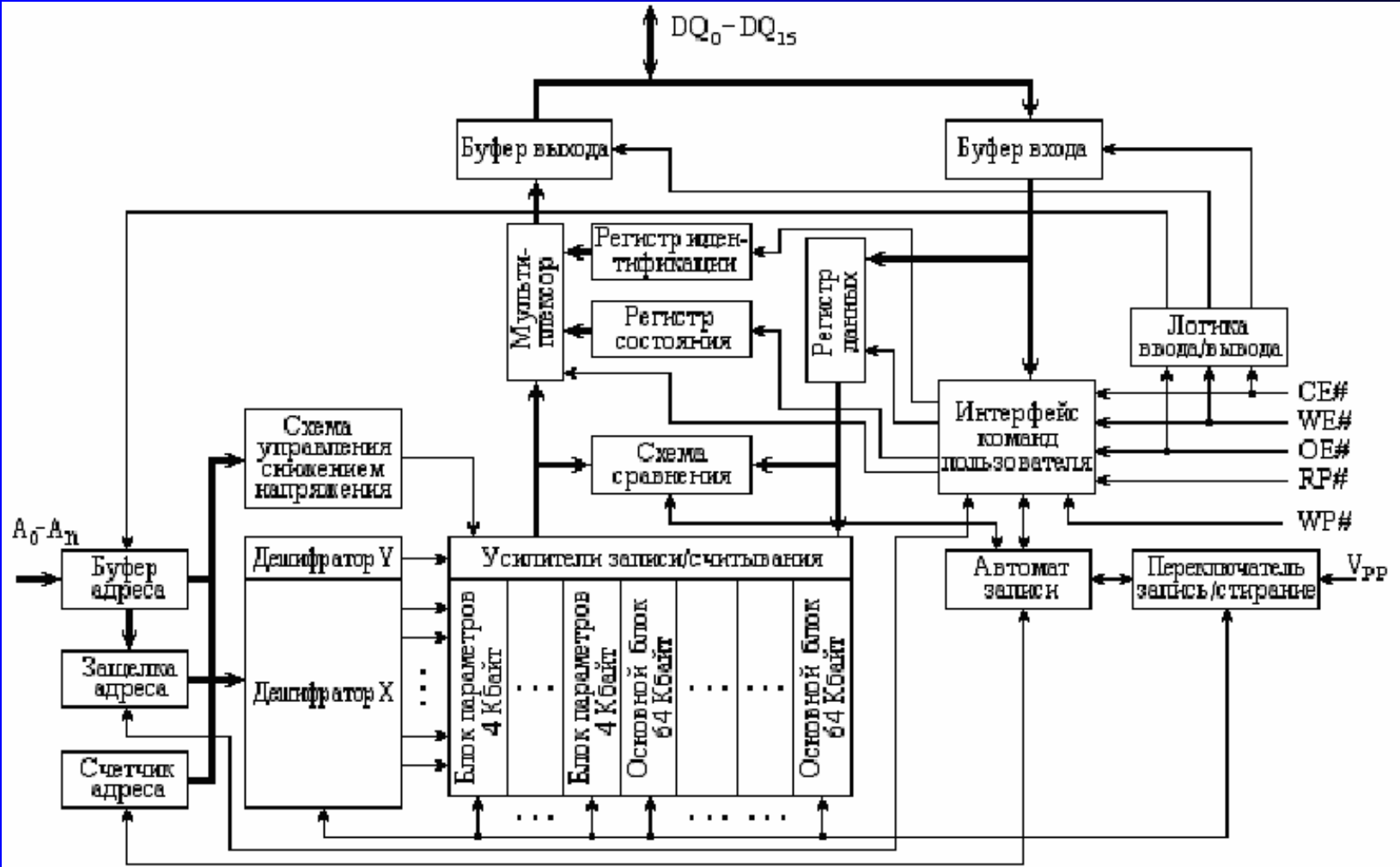
При симметричном (**Flash File**) - все блоки

Количество блоков зависит от ёмкости микросхемы. Например, в микросхеме 28F128J3 (Intel Strata Flash) емкостью 128 Мбит (16 Мбайт) имеется 128 блоков по 128 Кб.

При асимметричном (**Boot Block**) -

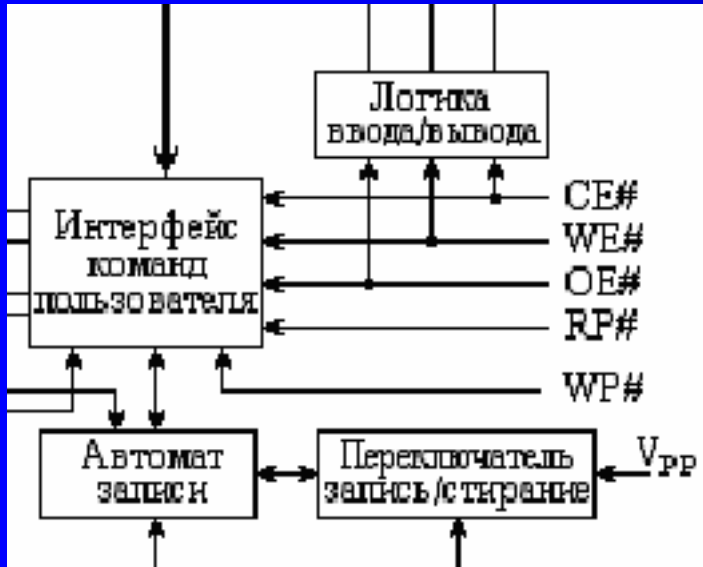
Пример - 28F640C3 (Intel Advanced+ Boot Block) ёмкостью 64 Мбит: загрузочный (Boot) блок размером 64 Кбайт, разбит на 8 блоков (parameter blocks) по 8 Кбайт, и 127 основных (main) блоков по 64 Кбайт. Загрузочный блок- либо в начале, либо в конце адресного пространства ИМС.

FLASH ПЗУ



Структурная схема Флэш-памяти (ассиметричная).

FLASH ПЗУ



WP# (Write Protect) используется для исключения возможности случайной записи по командам программы. RP# (Reset/Deep Power Down) - закрывает все блоки от записи. На вход V_{pp} подается напряжение, необходимое для ускорения операций стирания и записи данных.

Для улучшения характеристик в флэш-памяти предусматриваются:

1)

FLASH ПЗУ

Для улучшения характеристик в флэш-памяти:

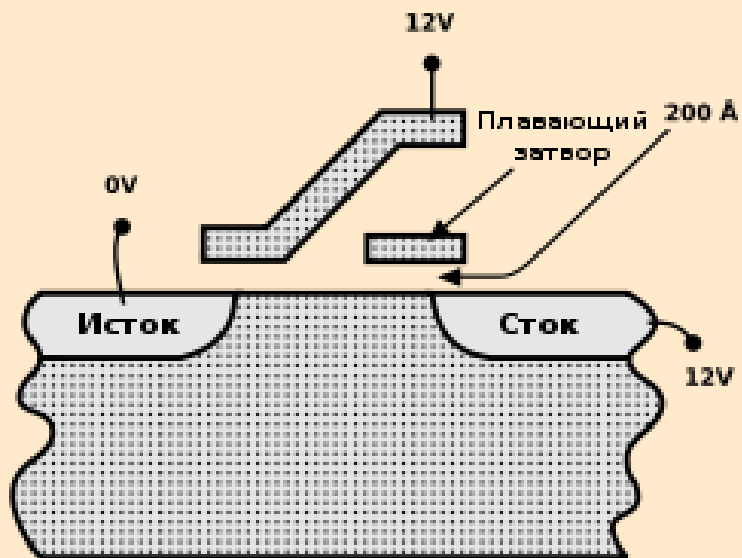
- 4) использование режимов понижения мощности когда к ЗУ нет обращения (важно для мобильных устройств);
- 5) приспособленность к работе с различными питающими напряжениями;
- 6)

Применение – BIOS, «твердотельные диски» (**solid-state disks**), карты памяти мобильных устройств.

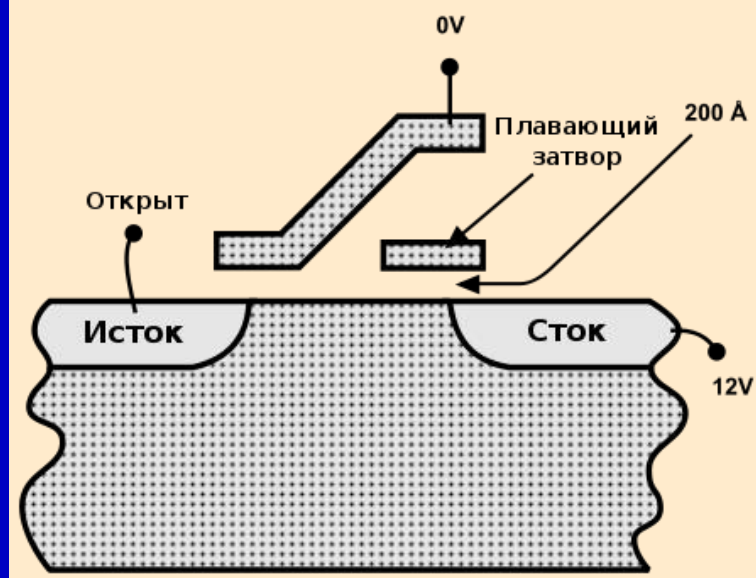
Время записи байта – 10 мкс, время доступа при чтении – 35-200нс, полное стирание 1-2 с, высокая плотность размещения на кристалле – на 30% больше чем у DRAM.

FLASH ПЗУ

Программирование инжекцией электронов

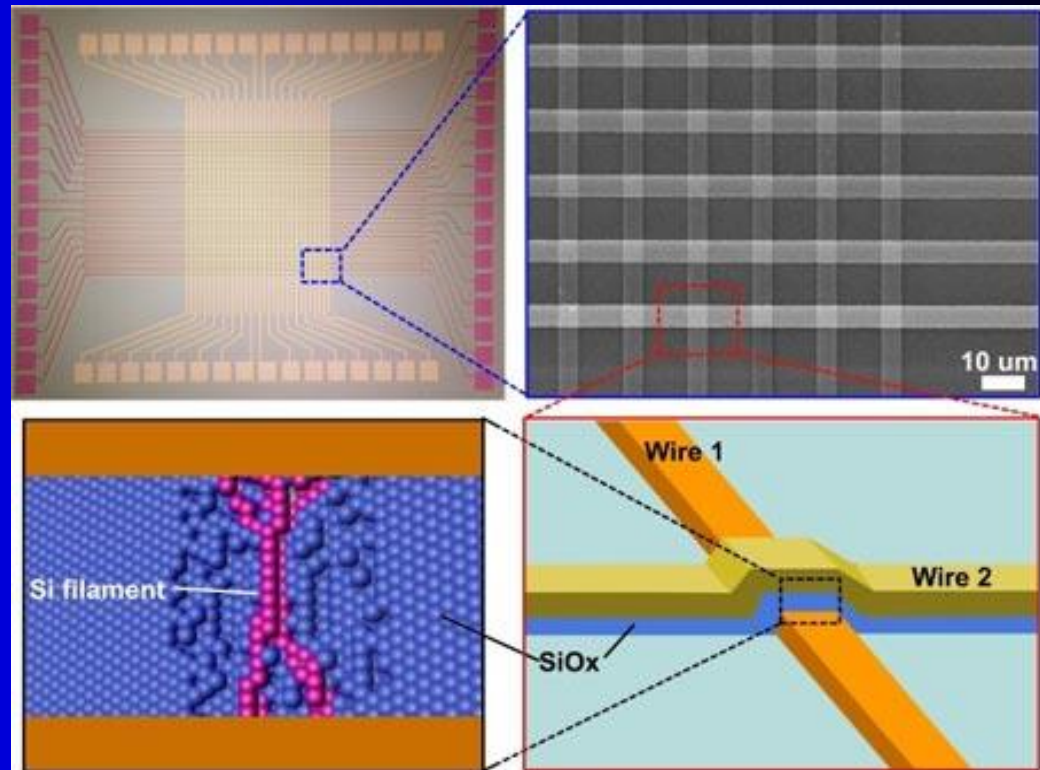


Стирание через туннельный эффект



Наследник FLASH ПЗУ – Память из оксида кремния

- В прошлом году, экспериментируя с полосками графита 10-нанометровой ширины, исследователи убедились, что их можно разрывать и повторно соединять, воздействуя электрическим током. Поскольку полоски принимают два хорошо различимых состояния, этот эффект можно использовать для создания памяти нового типа.
- В результате нового исследования ученые пришли к выводу, что вовсе не обязательно использовать графит. Заклучив слой вышеупомянутого оксида кремния между слоями поликристаллического кремния, которые играли роль электродов, участники проекта смогли добиться того же эффекта. Они обнаружили, что нанопроводники из оксида кремния шириной 5 нм могут соединяться и разъединяться, подобно микроскопическим переключателям, под действием приложенного к ним напряжения.



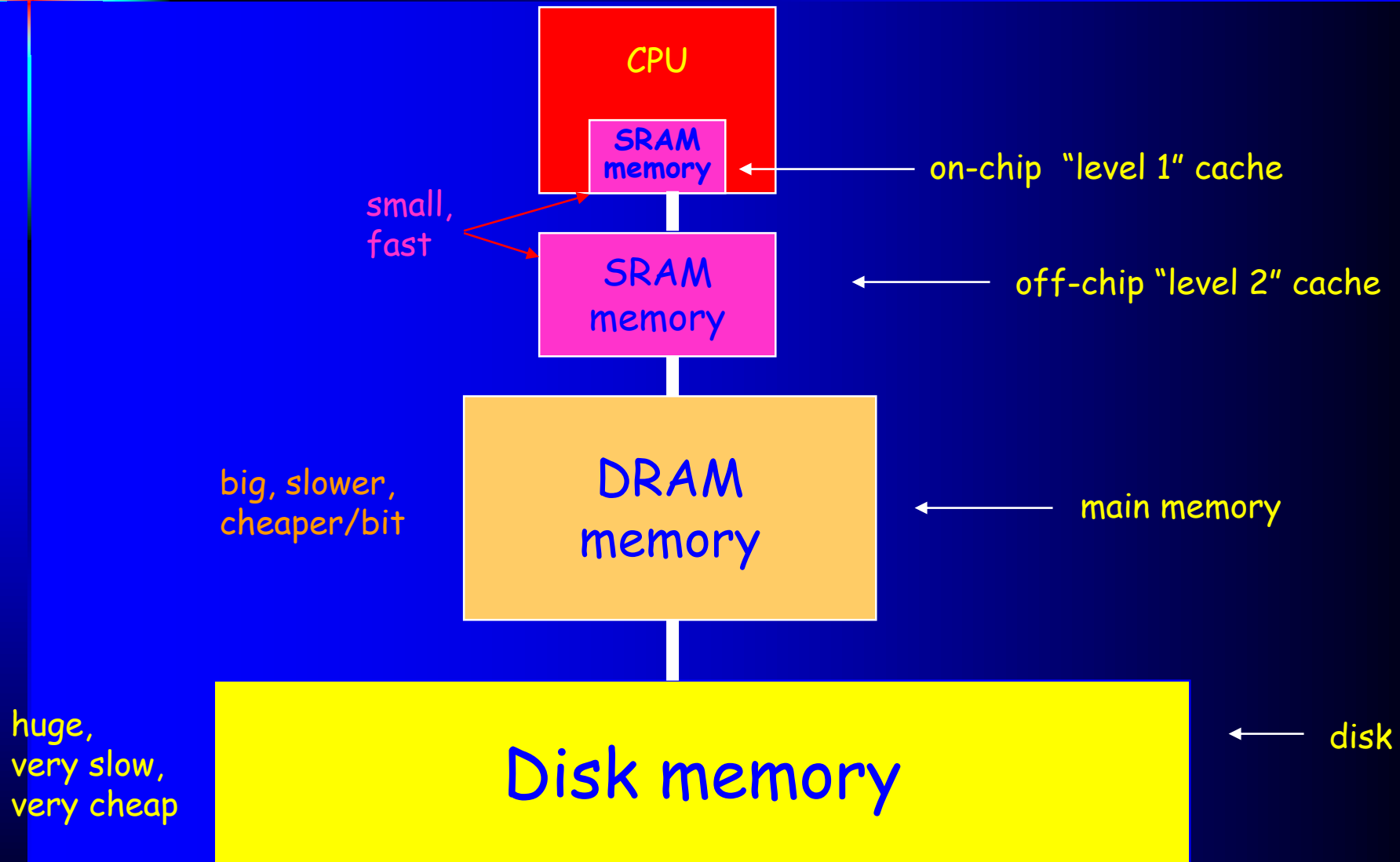
От флэш-памяти новая разработка выгодно отличается

Memory technologies

- SRAM
 - время доступа: 3-10 нс (on-processor SRAM - 1-2нс.)
 - стоимость: \$100 за МБайт (??).
- DRAM
 - время доступа : 30 - 60 нс
 - стоимость: \$0.50 за МБайт.
- Disk
 - время доступа : 5 to 20 МИЛЛИОНОВ нс
 - стоимость от \$0.01 за МБайт.

Желательно иметь память со скоростью SRAM и объёмом жёстких дисков.

A typical (cache) memory hierarchy



Производительность иерархии памяти

Основная формула:

- $T_{avg} = P_{hit} * T_{hit} + P_{miss} * T_{miss}$

T_{hit} = время выполнения запроса данных из памяти в случае, если данные присутствуют на текущем уровне иерархии

T_{miss} = время выполнения запроса данных в случае если данные на текущем уровне не присутствуют и необходимо обращение к нижестоящему уровню

P_{hit}, P_{miss} = вероятности попадания (hit) или промаха (miss) для текущего уровня

– P_{hit} всегда 100% для самого нижнего уровня иерархии

Пример 1

Память состоит из кэш-памяти и основной памяти. Если обращение к кэш выполняется за 1 такт, и 100 тактов –

$T_{hit} = 1 \text{ cycle}$

$T_{miss} = 100 \text{ cycles}$

$P_{hit} = .97$

$P_{miss} = .03$

$T_{avg} =$

Пример 2

- Иерархия памяти состоит из кэш, основной и виртуальной памяти. В случае если вероятность попадания кэш – 98%, а основной памяти – 99%, и среднее время доступа к кэш – 2 такта, основной – 150 тактов и 100 000 тактов

Пример 2

Подсчёт снизу вверх:

$$T_{\text{avg, main}} =$$

$$T_{\text{avg, cache}} =$$

Вывод: даже несмотря на то, что промах кэша составит только 2% от всех запросов, среднее время доступа к данным может возрасти более чем

Вопросы кэш памяти (Cache Issues)

Доступ к памяти -

- Каким образом аппаратура узнаёт – есть попадание или промах требуемых данных?

В случае промаха кэш-памяти -

Ассоциативная память



Структура ассоциативного ЗУ