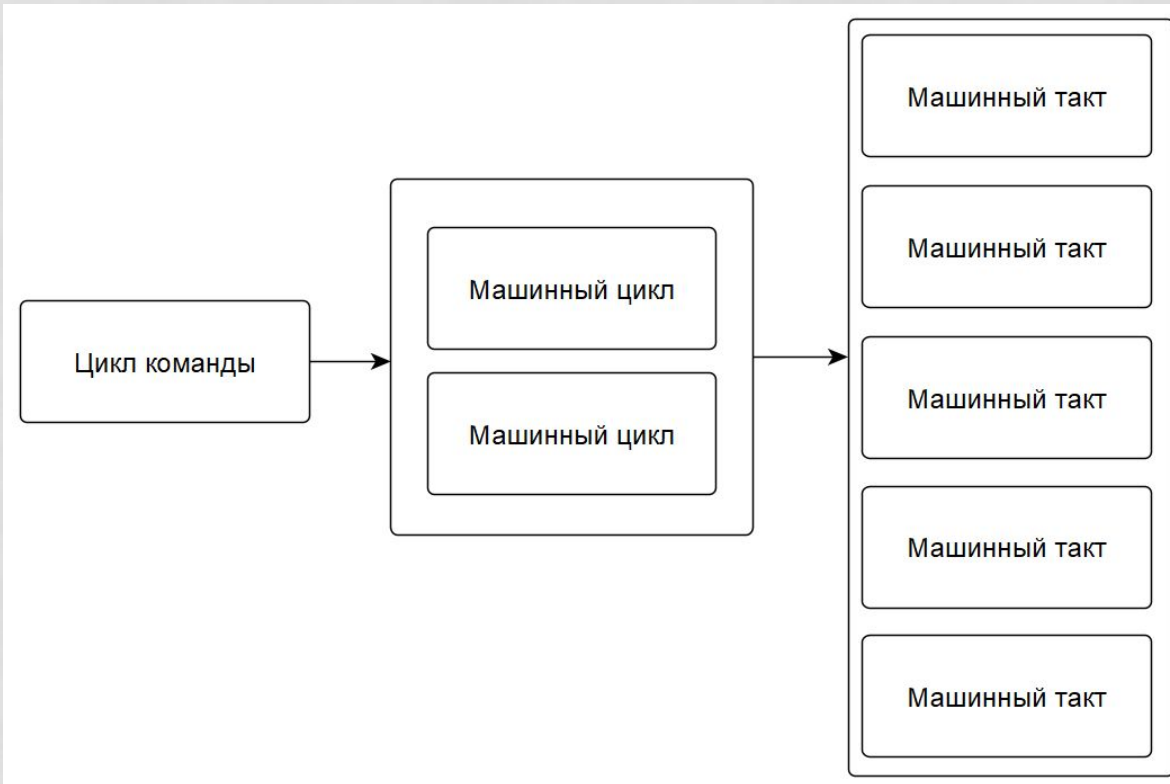


ДОПОЛНЕНИЕ К ЛЕКЦИИ 2

Москва
2018

Выполнил: Провоторов Н.В.

Цикл команды – это время, необходимое микропроцессору для выполнения отдельной команды



Цикл команды (ЦК) подразделяется на *машинные циклы (МЦ)*

Каждый машинный цикл состоит из ряда *машинных тактов (МТ)*

Машинный такт (МТ) – интервал времени, в течение которого выполняется один или одновременно несколько этапов различных *микроопераций*

ЛЕКЦИЯ 3. ПАМЯТЬ МПС

Москва
2018

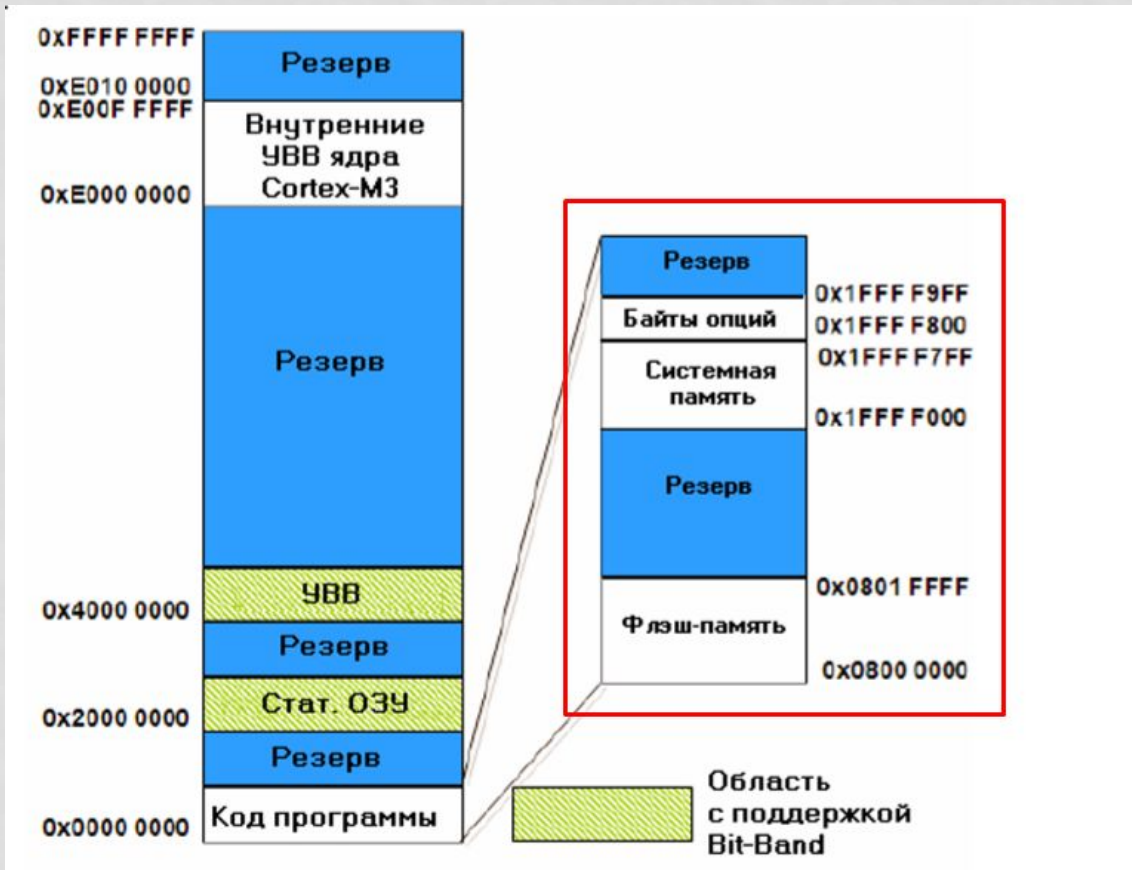
Выполнил: Провоторов Н.В.

Большинство современных микроконтроллеров имеют Гарвардскую архитектуру и содержат 3 вида памяти:

1. Память программ (ПЗУ).
2. Память данных - оперативная память (ОЗУ)
3. Регистры МК

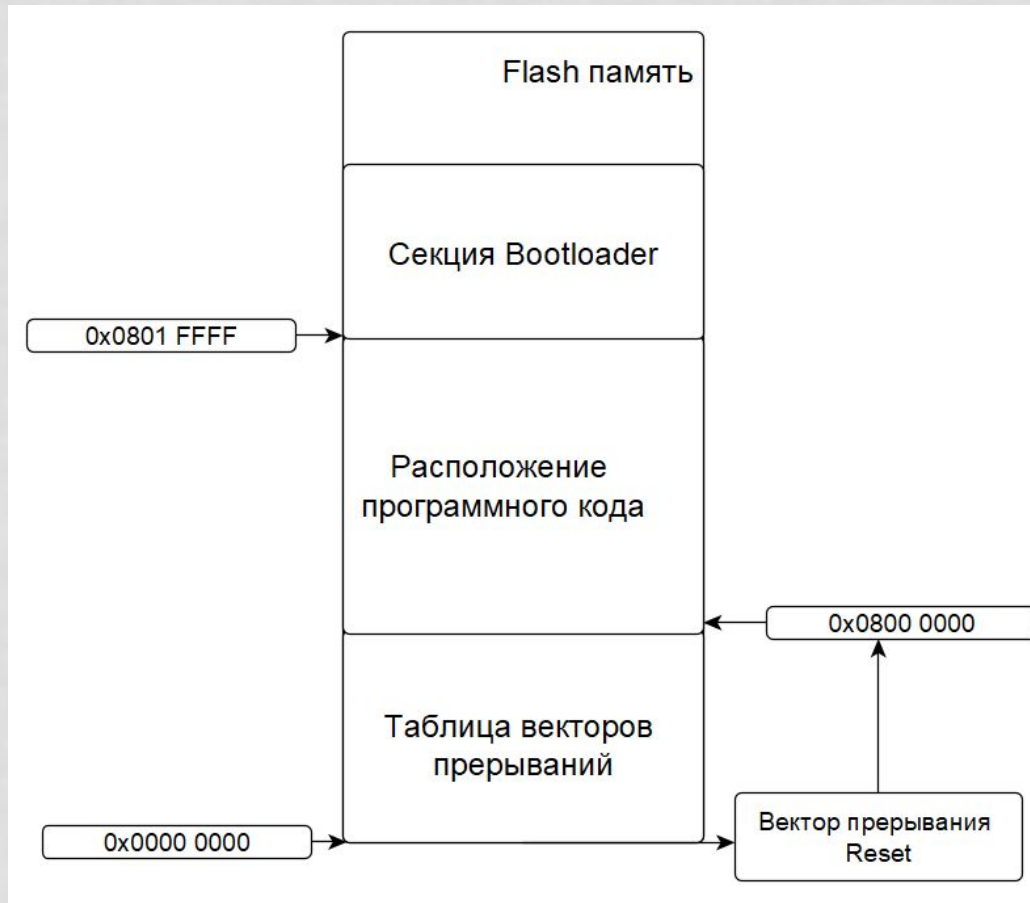
ПАМЯТЬ ПРОГРАММ

Память программ представляет собой электрически стираемое ППЗУ (FLASH)



Выводы выбора режима загрузки		Режим загрузки
BOOT1	BOOT0	
x	0	Флэш-память пользователя
0	1	Системная память

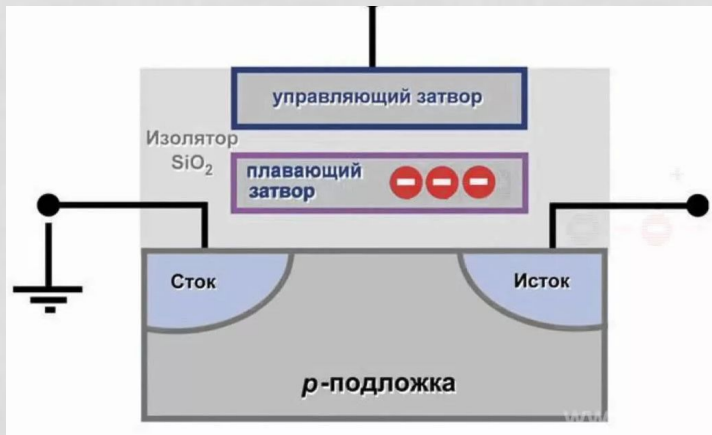
Состояния МК после сброса или включения питания



1. Проверка состояния внешних выводов BOOT0 и BOOT1
2. Переход по нулевому адресу памяти программ
3. Переход по вектору прерывания RESET, где хранится первая команда исполняемого кода

ФИЗИЧЕСКАЯ РЕАЛИЗАЦИЯ ПЗУ

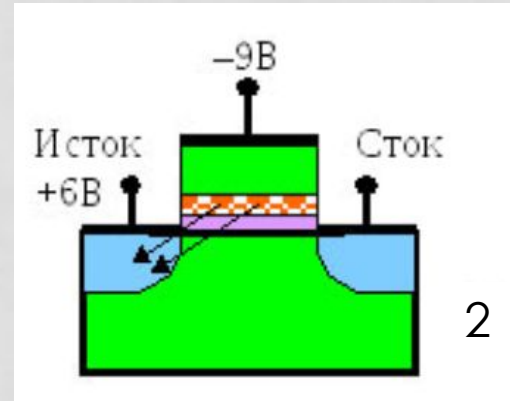
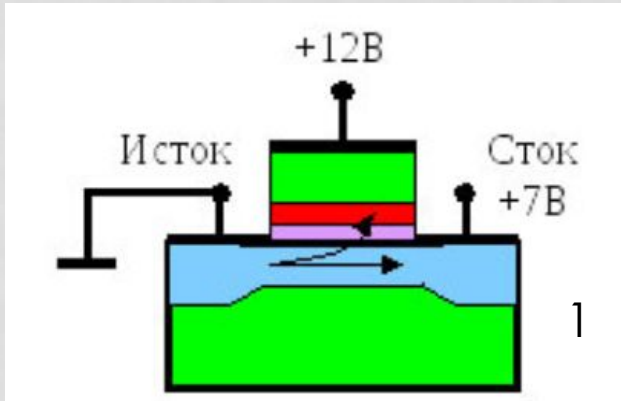
В основе флэш памяти лежит особая модификация транзистора с изолированным затвором (МОП-транзистора).



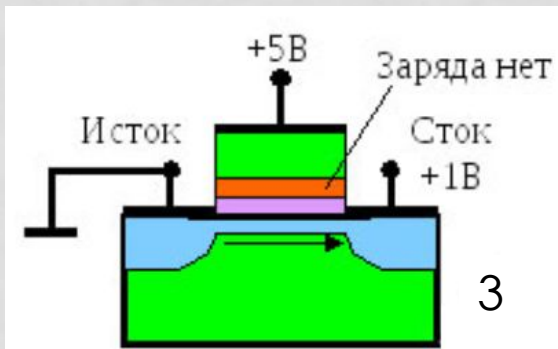
Битовая ячейка FLASH
памяти:

1. Управляющий затвор
2. Плавающий затвор
3. Изолятор
4. Сток
5. Исток

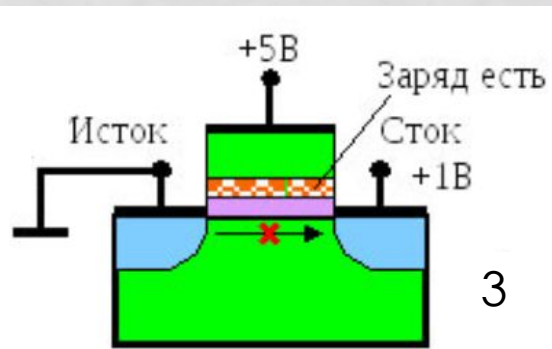
Операции записи, стирания, чтения с битовой ячейкой FLASH



1. Запись бита данных
2. Стирание бита данных



Транзистор открыт. Ток есть.
Логический «ноль»



Транзистор закрыт. Тока нет.
Логическая «единица»

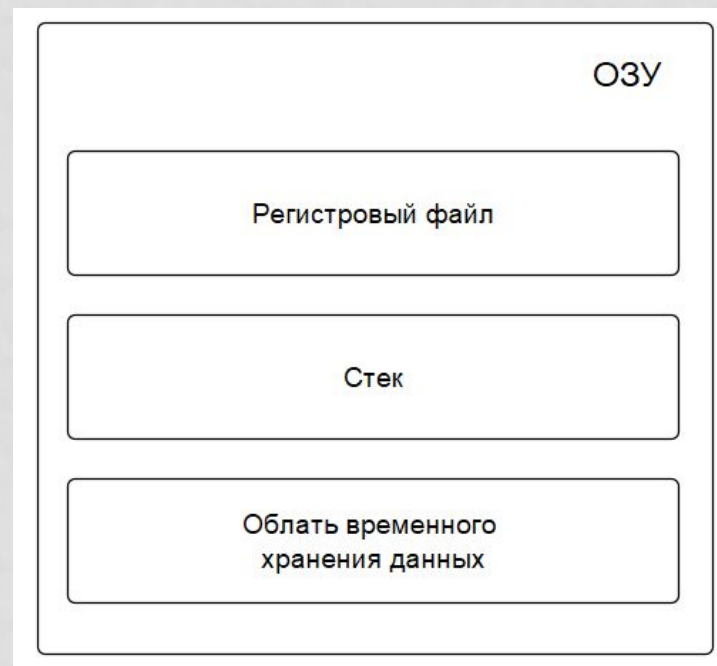
3. Чтение

ПАМЯТЬ ДАННЫХ

Память данных реализована на базе статического ОЗУ (SRAM)

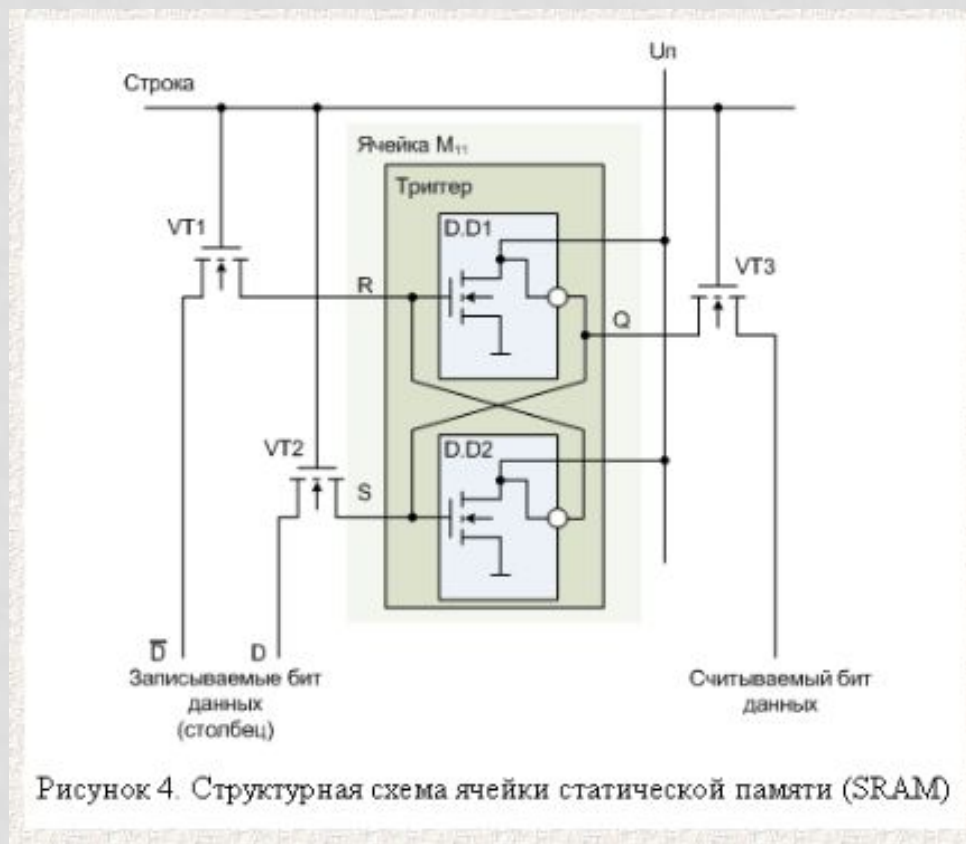
Отличительные особенности SRAM:

1. Произвольный доступ к ячейкам памяти
2. Отсутствие необходимости в регенерации ячеек
3. Простая схемотехника
4. Уровень напряжения хранения 1В



ФИЗИЧЕСКАЯ РЕАЛИЗАЦИЯ ОЗУ

Типичная ячейка статической двоичной памяти (двоичный триггер) на КМОП-технологии.



Битовая ячейка SRAM памяти:

1. VT1 – сигнал Reset
2. VT2 – сигнал Set
3. VT3 – прямой выход микросхемы триггера
4. DD1 и DD2 - инверторы

Общий вид матриц памяти SRAM

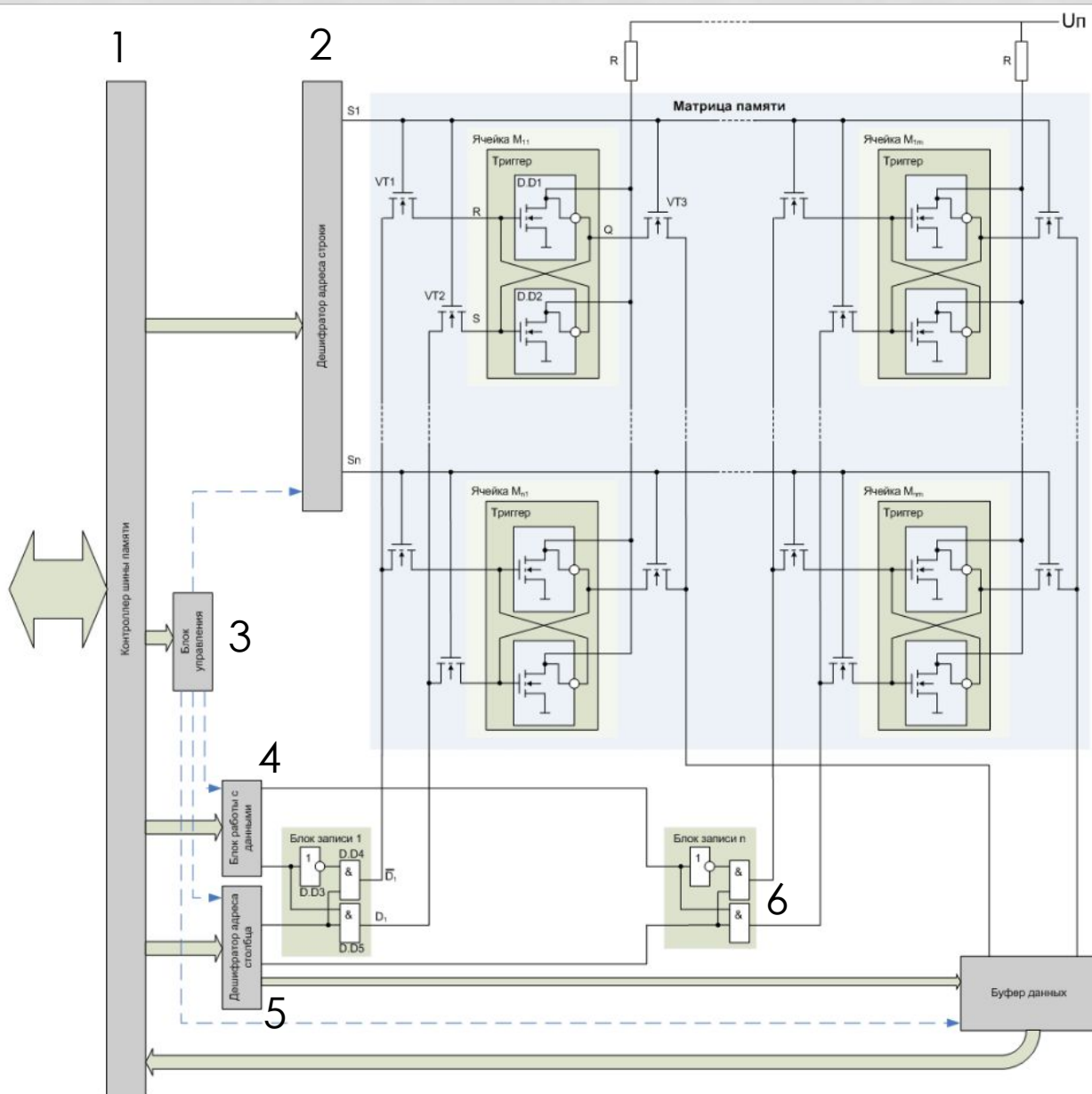
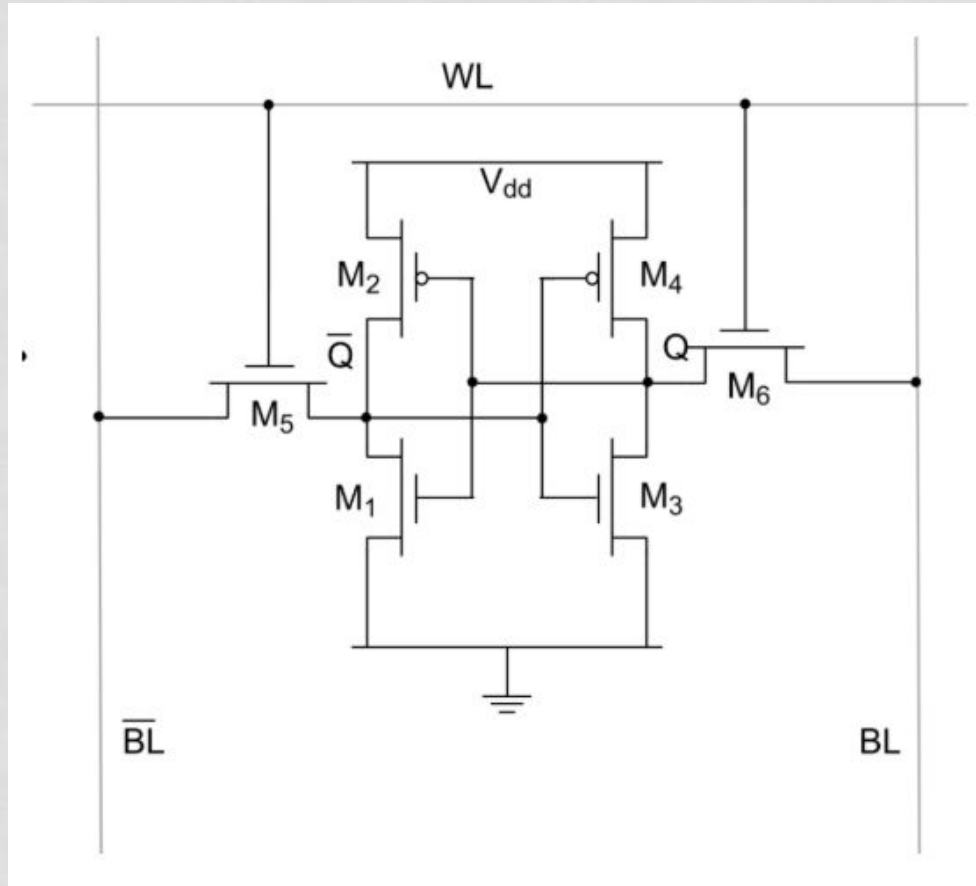


Рисунок 5. Упрощенная структурная схема статической оперативной памяти (SRAM)

Битовая ячейка SRAM памяти:

1. Контроллер шины памяти
2. Дешифратор адреса строки
3. Блок управления
4. Блок работы с данными
5. Дешифратор адреса столбца
6. Блок записи

Более подробная реализация ячейки статической двоичной памяти (двоичный триггер) на КМОП-технологии состоит из двух перекрёстно (кольцом) включённых инверторов и ключевых транзисторов для обеспечения доступа к ячейке



РЕГИСТРЫ МК

Регистры МК также выступают в качестве одного из видов памяти

В число этих регистров входят обычно:

1. регистры процессора
2. регистры управления
3. регистры, обеспечивающие ввод/вывод данных (регистры данных портов, регистры управления параллельным, последовательным или аналоговым вводом/выводом).

НЕПОСРЕДСТВЕННАЯ АДРЕСАЦИЯ



Рис. 3.1. Непосредственная адресация.

Предполагается, что операнд (входной) находится в памяти непосредственно за кодом команды.

```
mov al, 5  
add bx, 1234h
```

ПРЯМАЯ АДРЕСАЦИЯ



Предполагается, что операнд (входной или выходной) находится в памяти по адресу, код которого находится внутри программы сразу же за кодом команды.

```
mov dx,[a]
```

РЕГИСТРОВАЯ АДРЕСАЦИЯ

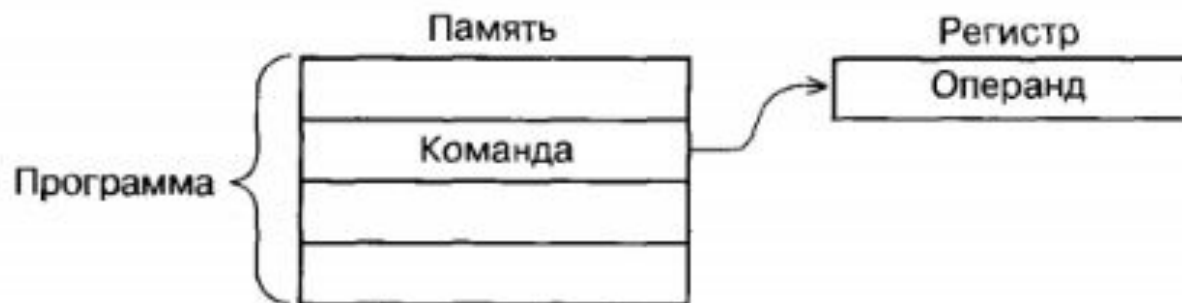


Рис. 3.3. Регистровая адресация.

предполагается, что операнд (входной или выходной) находится во внутреннем регистре процессора.

`xchg BX, BP`

КОСВЕННО-РЕГИСТРОВАЯ АДРЕСАЦИЯ

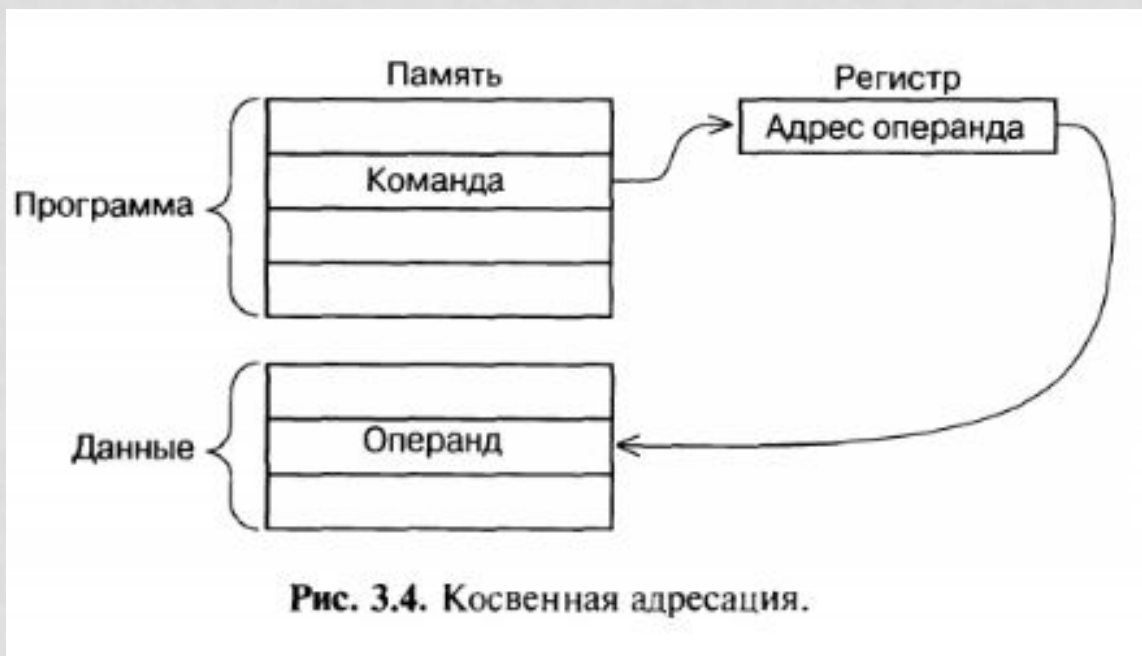


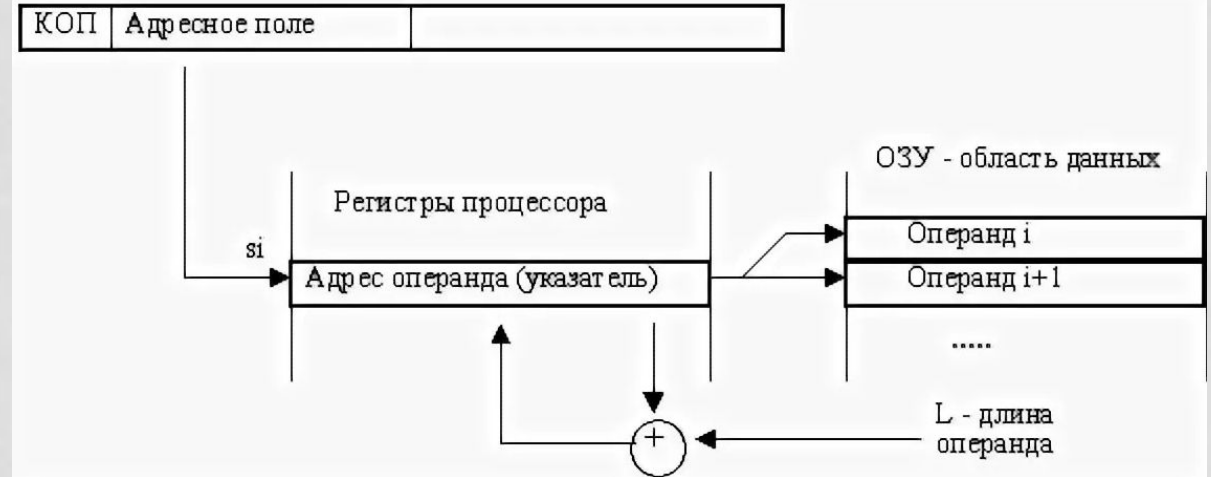
Рис. 3.4. Косвенная адресация.

Предполагается, что во внутреннем регистре процессора находится не сам операнд, а его адрес в памяти

```
add ax,[bx]  
mov dl,[si]
```

Реже встречаются еще два метода адресации:

1. Автоинкрементная адресация
2. Автодекрементная адресация



1. Близка к косвенной адресации, но отличается от нее тем, что после выполнения команды содержимое используемого регистра увеличивается на единицу или на два.
2. Похоже на автоинкрементную, но только содержимое выбранного регистра уменьшается на единицу или на два перед выполнением команды.