Введение в компьютерные науки

ЛЕКТОР К.Т.Н. МОХОВ В.А.

ГЛАВА 2. ОБРАБОТКА ДАННЫХ

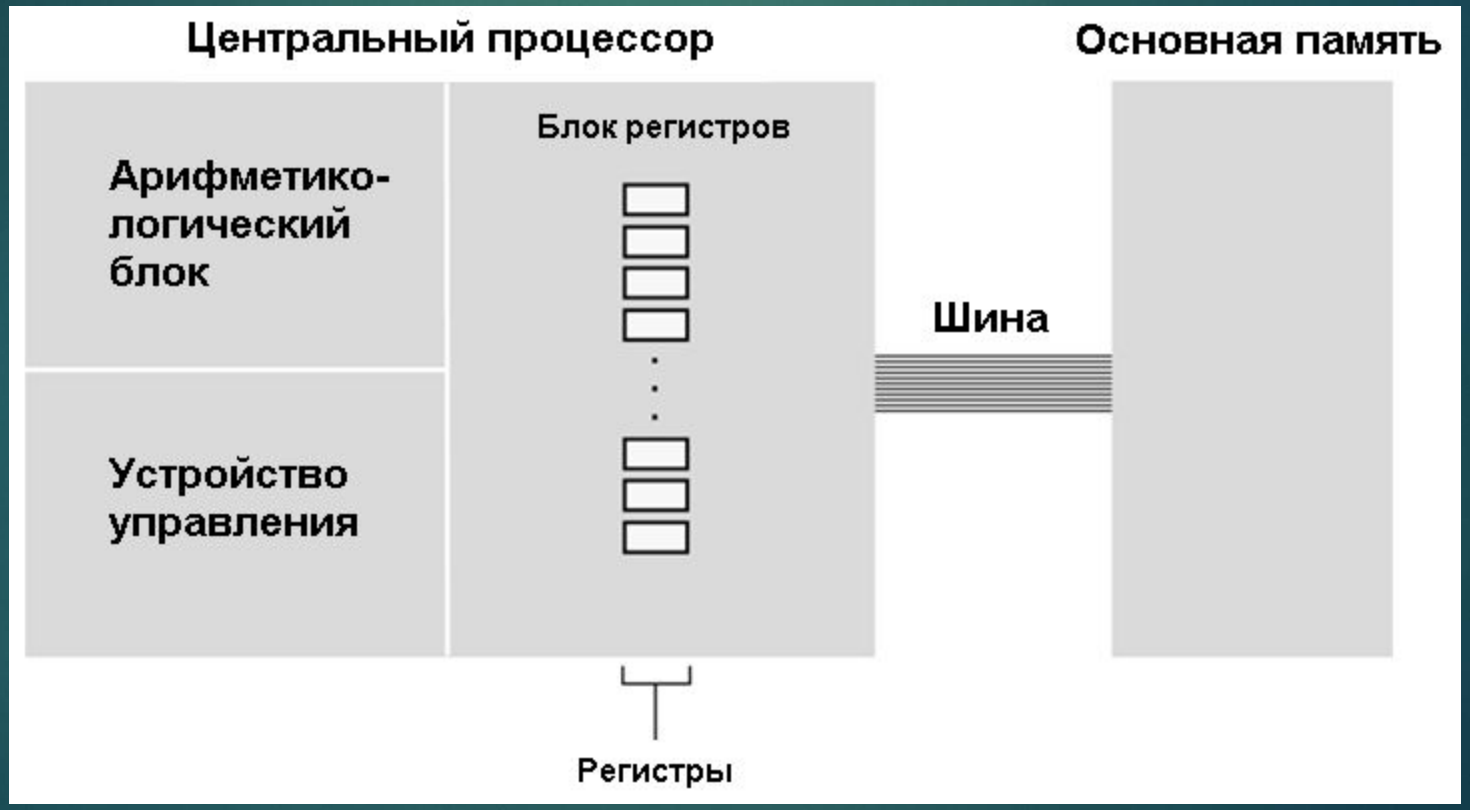
Часть 2: Обработка данных

- 2.1 Архитектура компьютера
- 2.2 Концепция хранимой программы
- 2.3 Выполнение программы
- 2.4 Арифметические и логические команды
- 2.5 Взаимодействие с другими устройствами
- 2.6 Другие типы архитектуры компьютеров

Архитектура компьютера

- Процессор или CPU (Central Processing Unit)
  - Арифметико-логический блок и устройство управления
  - Регистры
    - Общего назначения
    - Специального назначения
- Шина
- Материнская плата

Рисунок 2.1 Соединение центрального процессора и основной памяти с помощью шины



Концепция хранимой программы

Программа должна быть закодирована в виде битовых последовательностей и хранится в оперативной памяти.

Процессор может извлекать инструкции из оперативной памяти и выполнять их.

В свою очередь, программа для выполнения должна быть легко изменямой.

Терминология

- **Машинная команда:** Команда (инструкция), закодированная битовой комбинацией и распознаваемая процессором
- **Машинный язык:** Полный набор машинных инструкций с системой их кодирования

Философия машинных языков

- Компьютер с сокращенным набором команд (RISC - Reduced Instruction Set Computing)
  - Инструкций мало, они простые эффективные и быстрые
  - Достаточно низкая стоимость
  - Примеры: PowerPC от Apple/IBM/Motorola и ARM
- Компьютер с полным набором команд (CISC Complex Instruction Set Computing)
  - Инструкций много, они удобные и мощные
  - Высокая стоимость и сложность распараллеливания
  - Пример: Intel

Типы машинных команд

- **Передача данных**: копирование данных из одного места в другое
- **Арифметико-логические**: используют существующие битовые комбинации для вычисления новых битовых шаблонов
- **Управление**: управление выполнением программы

Рисунок 2.2 Сложение двух чисел, хранимых в основной памяти

Этап 1. Выбрать первое слагаемое из основной памяти и поместить его в регистр.

Этап 2. Выбрать второе слагаемое из основной памяти и поместить его в другой регистр.

Этап 3. Активизировать электронную схему суммирования, указав используемые на этапах 1 и 2 регистры в качестве входных и задав еще один регистр в качестве выходного, предназначенного для размещения результата.

Этап 4. Сохранить результат выполнения операции в основной памяти.

Этап 5. Завершить выполнение операции.

Рисунок 2.3 Деление двух чисел, хранимых в основной памяти

Этап 1. Загрузить в регистр число из основной памяти

Этап 2. Загрузить в другой регистр еще одно число из основной памяти

Этап 3. Если второе число равно нулю, перейти к этапу 6

Этап 4. Разделить содержимое первого регистра на содержимое второго и записать результат в третий регистр

Этап 5. Запомнить содержимое третьего регистра в основной памяти

Этап 6. Завершить выполнение операции

Рисунок 2.4 Архитектура машины, описанной в приложении С

Составные части машинной команды

- **Код операции:** определяет, какую операцию выполнять
- **Операнд:** дает более подробную информацию об операции
  - Интерпретация операнда меняется в зависимости от кода операции

Рисунок 2.5 Формат команды для машины, описанной в приложении С

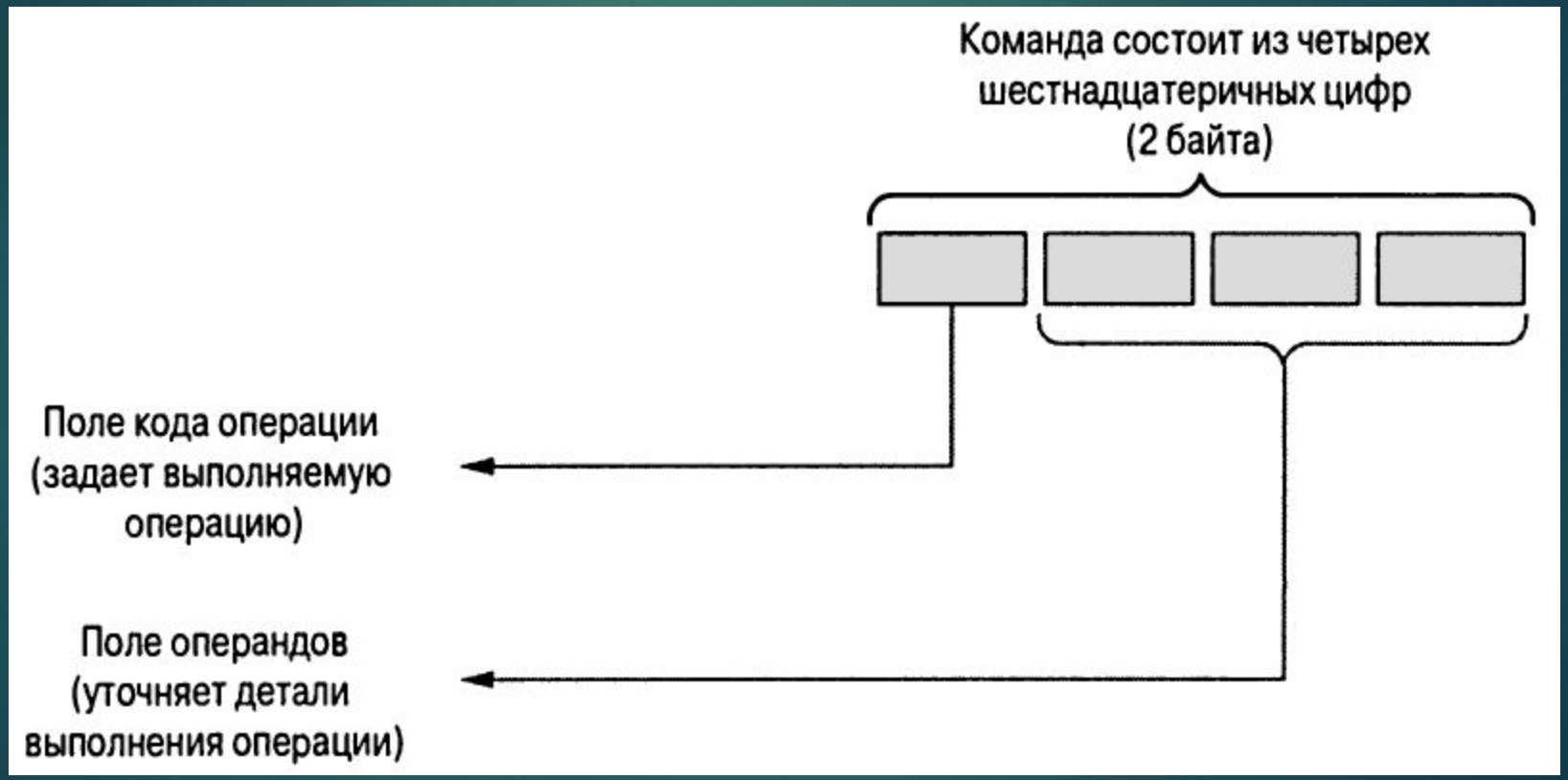


Рисунок 2.6 Расшифровка команды 35A7

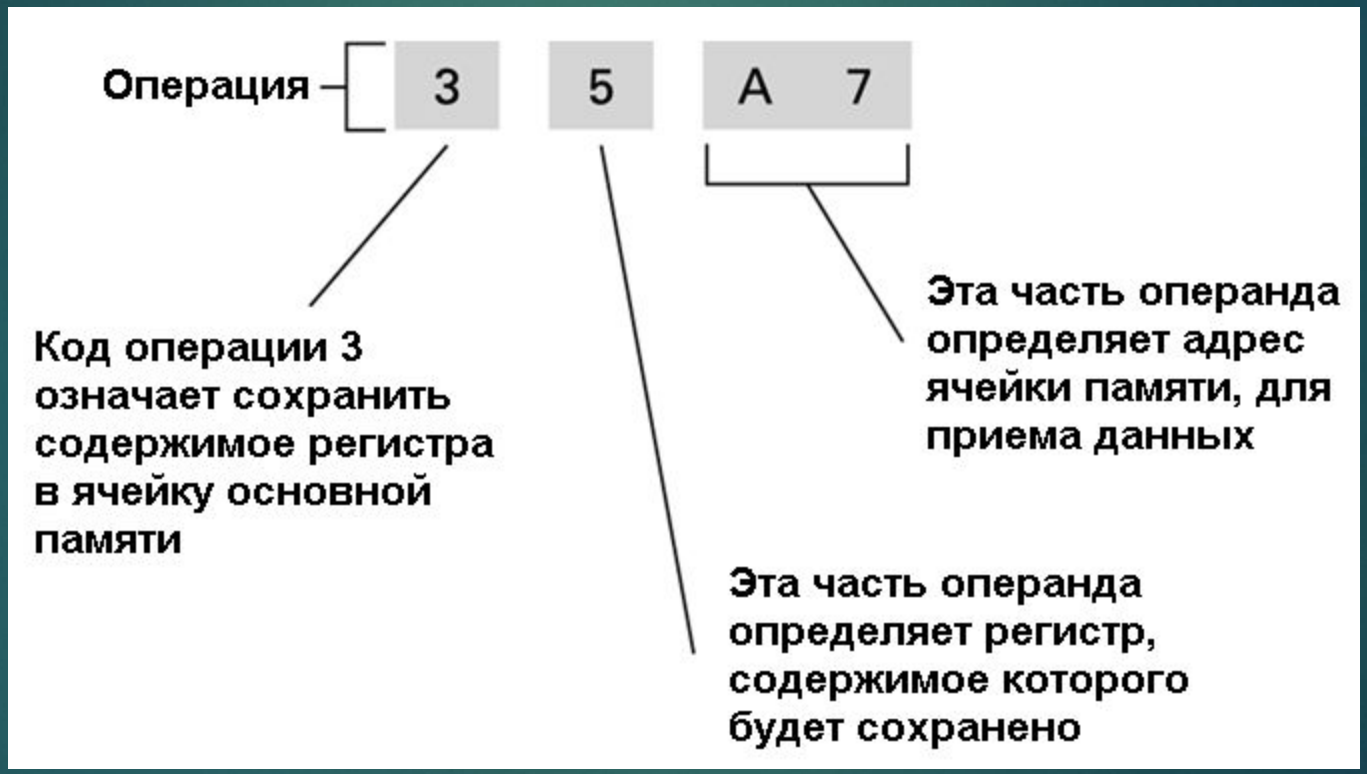


Рисунок 2.7 Расшифровка вариантов команд для рисунка 2.2

Код команды	Расшифровка
156C	Загрузка в 5 регистр бита двоичного кода из ячейки основной памяти с адресом 6C
166D	Загрузка в 6 регистр бита двоичного кода из ячейки основной памяти с адресом 6D
5056	Сложить содержимое регистров 5 и 6 в дополнительном коде и занести результат в регистр 0
306E	Сохранить содержимое регистра 0 в ячейке общей памяти с адресом 6E
C000	Останов

Выполнение программы

- Управляется посредством двух регистров специального назначения:
  - **Счётчик адреса**: содержит адрес следующей выполняемой команды
  - **Регистр команд**: содержит текущую команду
- Машинный цикл
  - Выборка
  - Декодирование
  - Выполнение

Рисунок 2.8 Схема машинного цикла

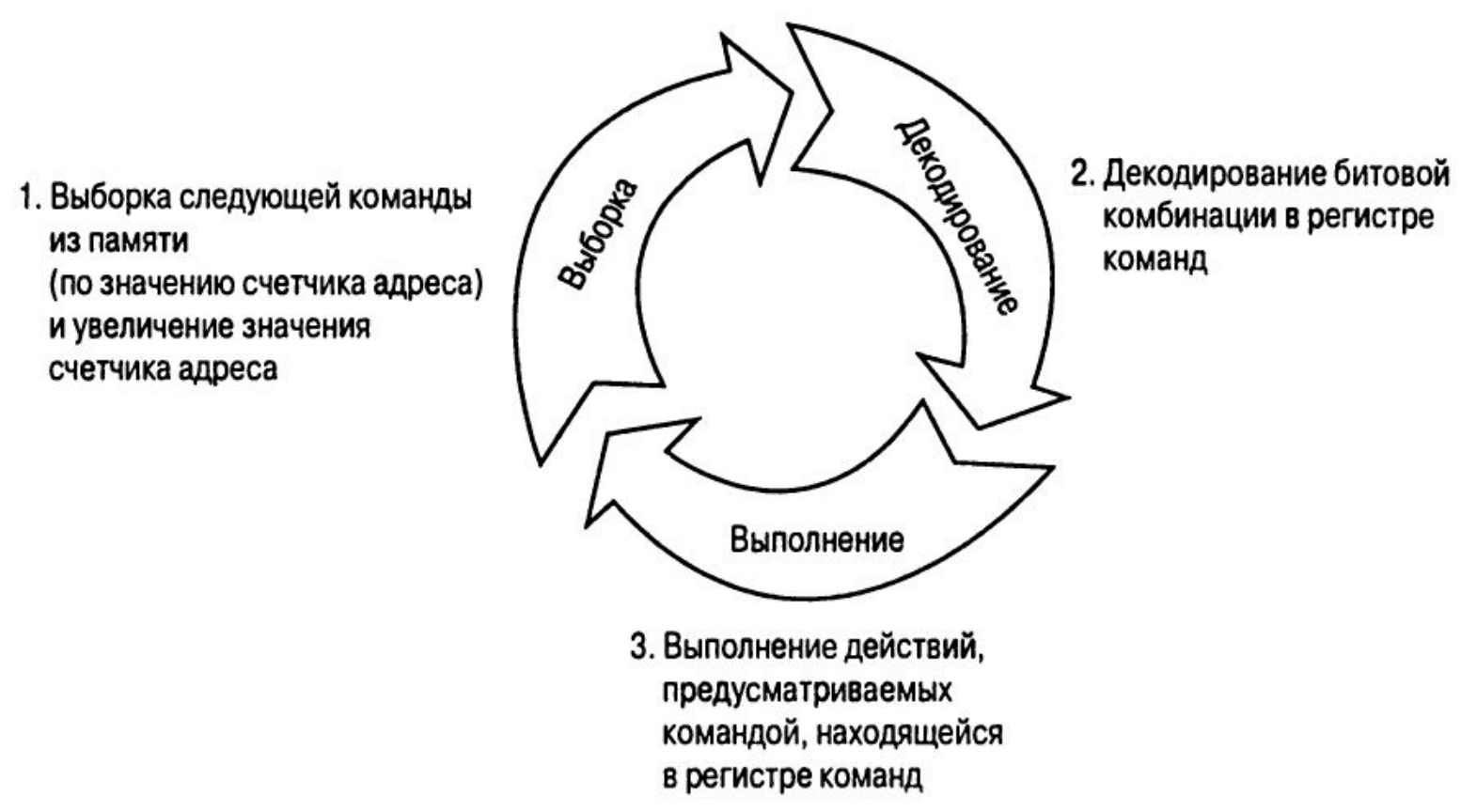


Рисунок 2.9 расшифровка команды B258

Рисунок 2.10 Программа с рисунка 2.7 хранится в памяти и готова к выполнению

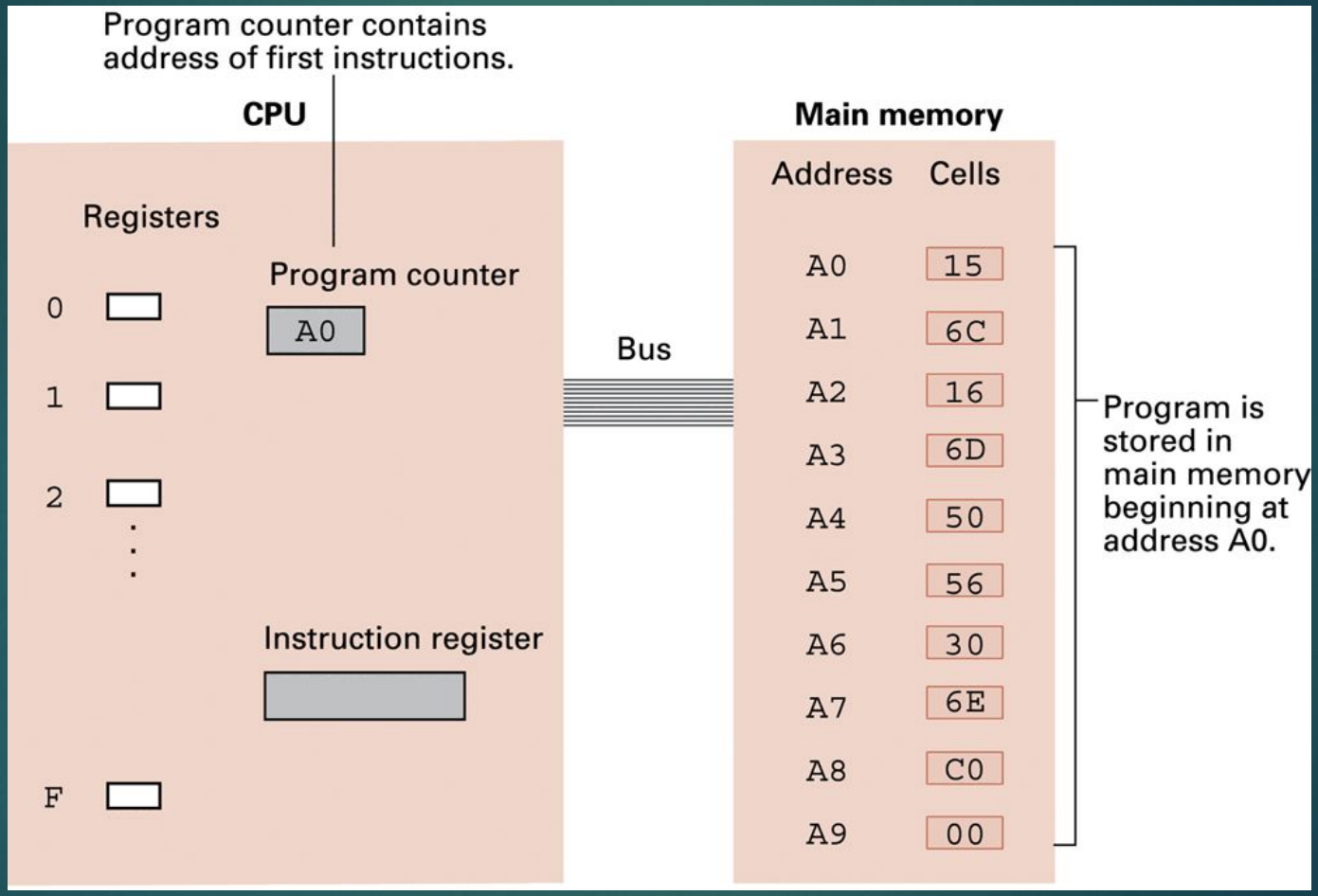
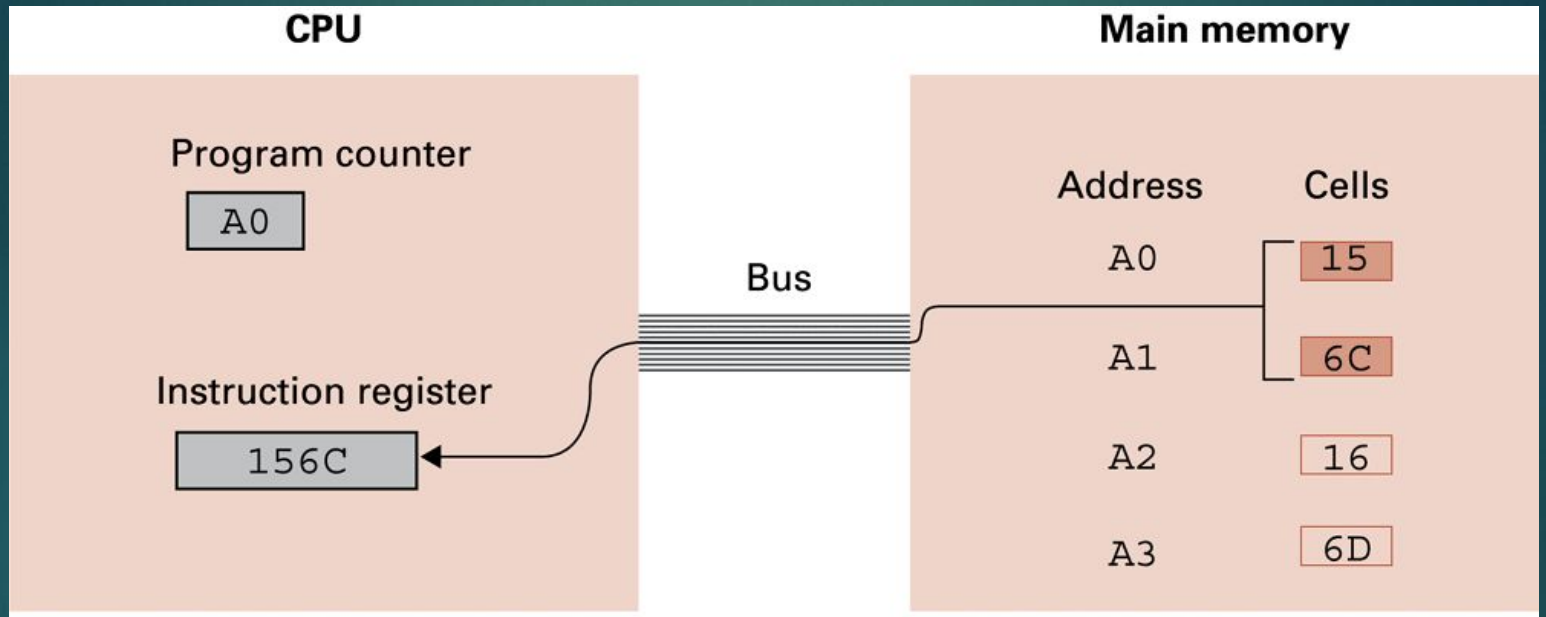
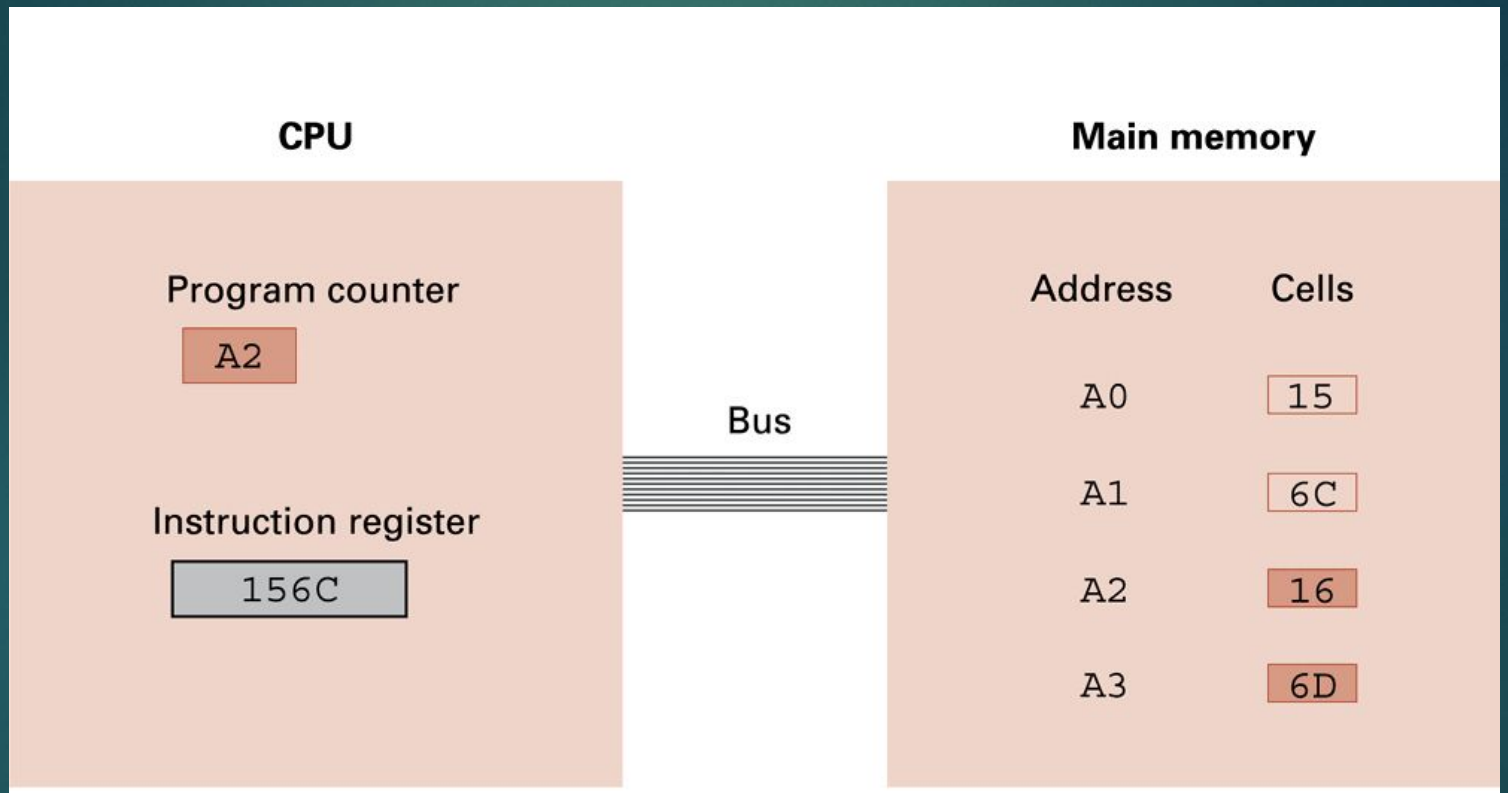


Рисунок 2.11 Выполнение шага выборки машинного цикла



a. At the beginning of the fetch step the instruction starting at address A0 is retrieved from memory and placed in the instruction register.

Figure 2.11 Выполнение выборки машинного цикла (продолжение)

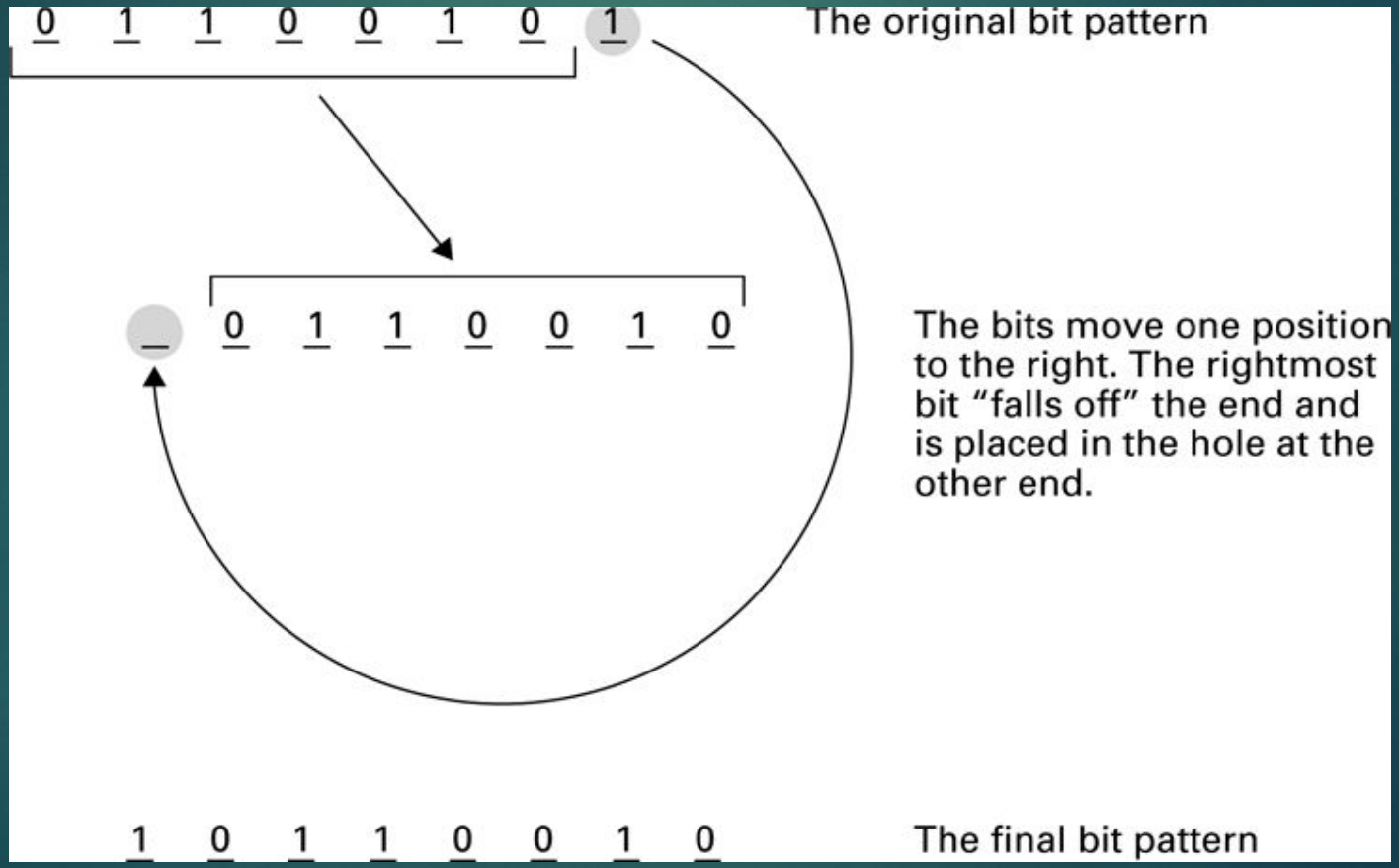


b. Then the program counter is incremented so that it points to the next instruction.

Арифметико-логические операции

- Логические: AND, OR, XOR
  - Маскирование
- Вращение и сдвиг: циклический сдвиг, логический сдвиг, арифметический сдвиг
- Арифметические: сложение, вычитание, умножение, деление
  - Точность выполнения операций зависит от того, как значения кодируются (двоичный дополнительный код или код с плавающей точкой).

Рисунок 2.12 Циклическое вращение битового шаблона 65 (в шестнадцатеричном коде) – сдвиг на 1 разряд вправо



Взаимодействие с другими устройствами

- **Контроллер:** Промежуточное устройство, которое управляет взаимодействием между компьютером и устройством
  - Специализированные контроллеры для каждого типа устройств
  - Контроллеры общего назначения (USB и FireWire)
- **Порт:** Точка, в которой устройство подключается к компьютеру
- **Ввод\вывод с отображением на память:** Процессор взаимодействует с периферийными устройствами, как если бы они были ячейки памяти

Рисунок 2.13 Подключение контроллеров к шине компьютера

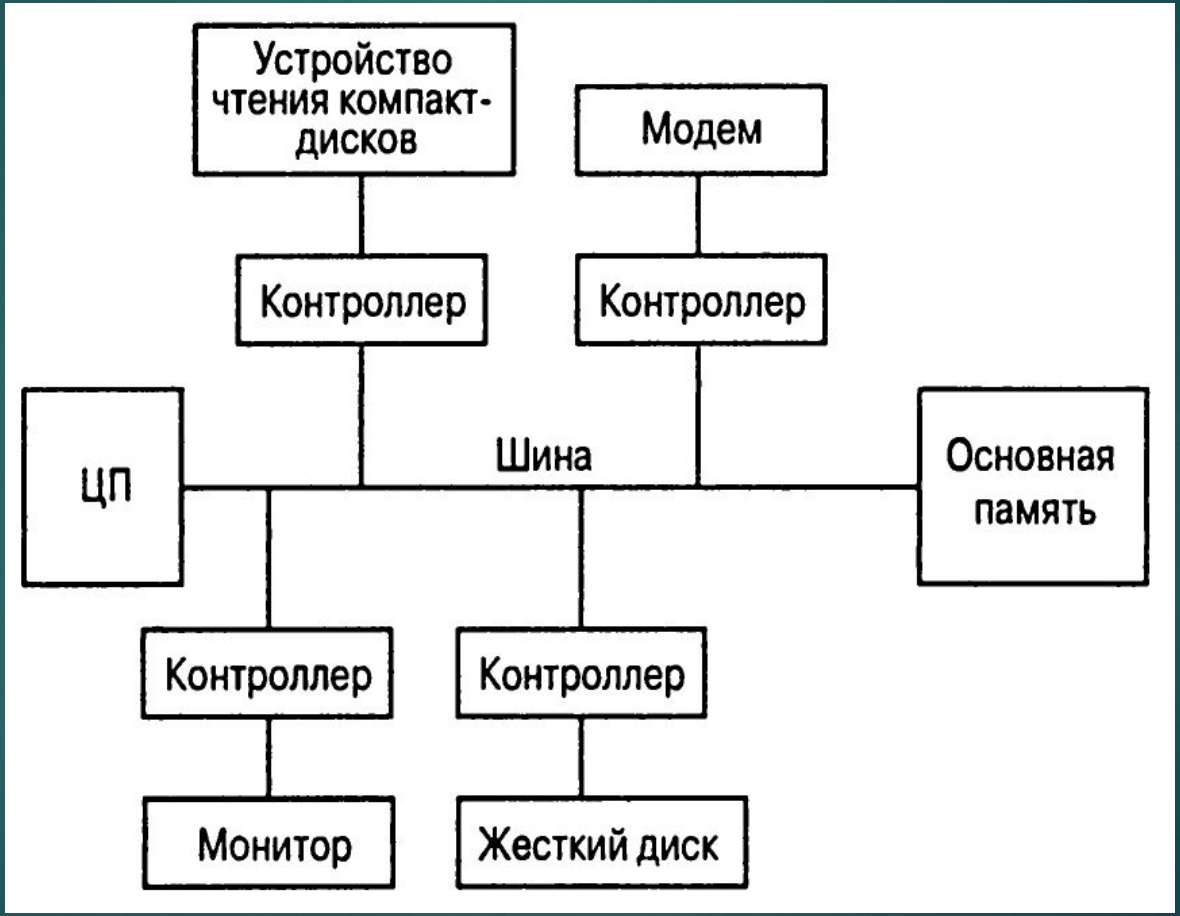
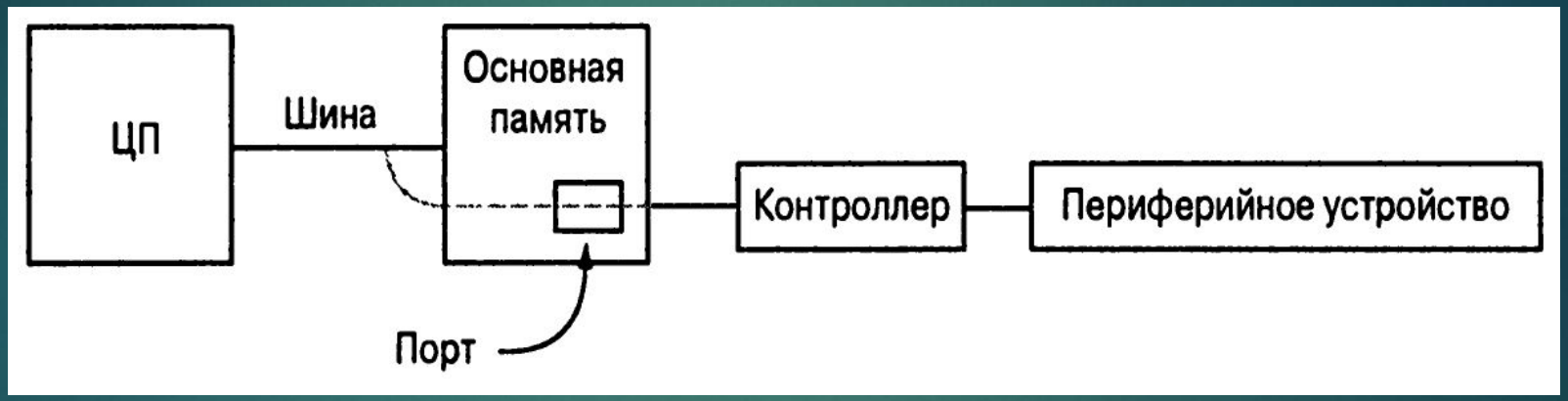


Рисунок 2.14 Концептуальная схема метода отображения ввода/вывода на память



Взаимодействие с другими устройствами (продолжение)

- **Прямой доступ к памяти (DMF - Direct memory access):** Основная память доступна с помощью контроллера на шине
- **Ограничение Фон Неймана:** Недостаточная скорости шины препятствует производительности
- **Подтверждение связи:** Процесс координации передачи данных между компонентами

Взаимодействие с другими устройствами (продолжение)

- **Параллельная передача:** Нескольких битов одновременно передаются по каналам связи.
- **Последовательная передача:** Биты передаются друг за другом по одному каналу связи.

Скорости передачи данных

- Единицы измерения
  - Bps: биты в секунду
  - Kbps: Kilo-bps (1,000 bps)
  - Mbps: Mega-bps (1,000,000 bps)
  - Gbps: Giga-bps (1,000,000,000 bps)
- Пропускная способность: Максимально допустимая скорость

Другие архитектуры

- Технологии для увеличения пропускной способности:
  - Конвейеризация: Перекрытие шагов машинного цикла
  - Параллельная обработка: Использование нескольких процессоров одновременно
- Классификация Флинна (о признакам наличия параллелизма в потоках команд и данных)
- SISD: Одиночный поток Команд, Одиночный поток Данных
- MIMD: Множественный поток команд, множественный поток данных
- SIMD: Одиночный поток команд, множественный поток данных