**6. Назначение и структура процессора. Система команд. Динамическая и статическая память.**

**Центральный процессор** - исполнитель машинных инструкций, часть аппаратного обеспечения компьютера или программируемого логического контроллера, отвечающий за выполнение операций, заданных программами. В большинстве ЭВМ принципы Фон-Неймана реализованы в следующем виде:

* 1) ОП организована как совокупность машинных слов фиксированной длины, например первые ЭВМ имели разрядность 8, 16, 32, 64. Существовали 45-разрядные, 35 – разрядные ЭВМ
* 2) ОП образует единое адресное пространство. Адреса машин. слов возрастают от младших к старшим.
* 3) В ОП размещаются как данные, так и программы
* 4) Команды выполняются в естественной последовательности по возрастанию адресов в ОП, пока не встретится команда ветвления
* 5) ЦП может произвольно обращаться к любым адресам в ОП для выборки или записи в машин. слова чисел и команд

**Команда** – описание операции, кот необходимо выполнить.

Каждая команда начинается с кода операции(КОП), содержит необходимые адреса, характеризующиеся форматом, который определяет структуру команды, её организацию, код, длину, метод расположения адресов. Длина различн. команд может быть одинаковой или разной. Команды подразделяются на арифметические, логические, ввода-вывода, передачи данных.

Классы команд:

1. Класс обработки данных:
   1. короткие операции
      1. логические: логическое сложение, лог. умножение, инверсия, лог. сравнение
      2. арифм.: сложение операндов, вычитание, арифм. сравнение
   2. длинные операции: сложение, умножение
2. операции управления: безусловный переход, условный переход
3. операции обращения к внешним устройствам: требование на запись или чтение

**Типовая структура 3-адресной команды:** КОП-А1-А2-А3. А2 и А3 – адреса ячеек где расположены 1 и 2 числа, А1 – адрес ячейки регистра, куда необходимо поместить результат, КОП-код оп-ции

**Типовая структура 2-адресной команды:** А1 – ад-с ячейки, где хр-ся первое из чисел, участ-х в операции и куда будет записан рез-т, А2- 2 число

**Типовая структура 1-адресной команды:** А1 – может обозначать адрес ячейки где хранится 1 из чисел, участвующих в операции либо адрес ячейки для результата

безадресная команда содержит КОП

Каждая команда сод-т эл-ты, определяющие:

* 1) Что делать(код)
* 2) Объекты, над которыми надо что-то сделать
* 3) Как делать(типы операндов)

Максим.длина команды 15байт

Префиксы(необязат поле – 1 байт)

* префикс замены сегмента
* префикс разрядности адреса
* префикс разрядности операнда
* пр-с повторения

Каждая команда вып-ся за 1 или неск тактов.

Послед-ть взаимосвяз-х команд именуется **макрокомандой**. Исп-е макрокоманд упрощает программир-е и обеспеч-ет механизм вставки добавления пр-мы.

**Цикл процессора** – это период времени, за кот осущ-ся вып-е команды исход. пр-мы в машинном виде.

Состоит из неск. тактов.

**Такт работы процессора** – период времени между соседними импульсами генер-ра тактовых импульсов.

Процедура, соответ. такту реализуется определенной логич. цепью проц-ра и наз-ся - **микропрограммой**.

Существуют две стратегии распределения оперативной памяти, как и любого ресурса: статическое и динамическое распределение.

При ***статическом распределении*** вся необходимая оперативная память выделяется процессу в момент его порождения. При этом память выделяется единым блоком необходимой длины, начало которого определяется базовым адресом. Программа пишется в адресах относительно начала блока, а физический адрес команды или операнда при выполнении программы формируется как сумма базового адреса блока и относительного адреса в блоке. Значение базового адреса устанавливается при загрузке программы в оперативную память. Так как в разных программах используются блоки разной длины, то при таком подходе возникает проблема фрагментации памяти, то есть возникают свободные участки памяти, которые невозможно без предварительного преобразования использовать для вычислительного процесса.

В кч-ве запоминающ эл-та исп-ся статич триггер, состоящий из 4-6 транзисторов

+ обладает большим быстродействием

– Плотность упаковки таких микросхем меньше пл-ти уп-ки динамич памяти, стоимость выше

При ***динамическом распределении*** памяти каждой программе в начальный момент выделяется лишь часть от всей необходимой ей памяти, а остальная часть выделяется по мере возникновения реальной потребности в ней. Такой подход базируется на следующих предпосылках.

В кач запомин эл-та исп-ся простейшая сборка, состоящая из 1 транзистора и 1 конденсатора

+ высокая плотность интеграции(кол-во числа запом.эл-в на ед-цупов-ти), малое потр-е энергии

– для того, чтобы сохранить записанную инф-ю запом. эл-т должен постоянно регенерироваться