Тема. Память

В память можно сохранять все что угодно - текст, изображения, видео в виде двоичных чисел.

Типы памяти:

* Регистры
* Кэш - память
* Оперативная память
* Внешняя память

Память - это множество ячеек - битов объеденных по 8 (в байты). За раз процессор может обратиться к биту, но не к байту.

**Бит - минимальная единица хранения информации.**

**Байт - минимальная ячейка адресации.**

Виды адресации:

* Байтовая адресация

- каждый байт имеет отдельный адрес

* Словесная адресация

- размер ячейки памяти равен машинному слову (32, 64 бита)

Размер ОЗУ. Адреса

Размер оперативной памяти должен быть кратен размеру ячейки (кратен 8ми).

Размер регистра процессора определяет максимальный размер памяти, который он может адресовать.

Хранение типов данных в памяти.

byte - 1 байт

short - 2 байта

int - 4 байта

long - 8 байт

unsigned byte - один байт может хранить от 0 до 25510. ( 25510 = 111111112)

signed byte - один бай может хранить от -12810 до 12710 (12710 = 11111112)

Хранение отрицательных чисел.

Пример: x = -510

510 = 1012

0101 -> 1101 -> 1010 -> 1011

Значимые и ссылочные типы данных:

* Значимые типы данных копируются по значению, а не по ссылке. Изменяя одну переменную мы никак не влияем на вторую.

byte x = 5;

byte y = x;

* Ссылочные типы данных ссылаются на одно и тоже значение в памяти. Поэтому изменяя один объект - мы изменяем и другой.

В качестве примера может быть ссылка в памяти на массив.

byte[] arr = {2,4,5,7,9};

byte[] copy = arr; //Скопировалась лишь ссылка

Проблема архитектур и обратной совместимости.

Intel vs AMD

У каждой архитектуры свой набор команд. Если каждый создатель ЯП будет транслировать свой язык в определённый набор команд, то тогда процессор должен уметь запускать программу на каждом ЯП,а не хотя бы на одном.

Необходимо, чтобы ПО для старых процессоров запускалось на новых, при этом новые имели бы какие-то нововведения.

Единая архитектура наборов программ:

Архитектуры:

* Х86 - ПК
* ARM - мобильный устройства
* AVR - микроконтроллеры

У каждой свой набор команд.

Х86. Разрядность

intel 8086 … Core i7 - 32 битный процессор

Разрядность определяет размер машинного слова. (16, 32, 64 бита)

Регистры.

Маленькие ячейки памяти расположенные на процессоре.

Размер регистра = разрядность процессора.

* *Регистры общего назначения* (для программистов): переменные, параметры, результаты вычислений
* *Регистры специального назначения* (для конкретного содержимого) : сегментные, для работы со стеком, флаговые регистры, указатель команд

Прерывания.

Процессор постоянно чем-то занят.

Аппаратные устройства передают сигнал на специальный контроллер, а от него на процессор передается сигнал на прерывание.

**Прерывание** - сигнал процессору о том, чтобы он прервал выполнение текущей программы и передал управление специальной функции-обработчику прерывания.

* Исключения(генерируются ЦП при попытке программы нарушить уровни защиты)
* Аппаратные(генерируются контроллером прерываний)
* Программные прерывания(создаёт программист)

Многозадачность

* Программа разбивается на процессы, а те в свою очередь на потоки или задачи.
* Процессор делает все по очереди, но благодаря прерыванием он может создать иллюзию одновременной работы нескольких приложений, быстро переключаясь между ними и передавая контроль.

**Ассемблер.**

Язык ассемблера - это язык программирования, состоящий из команд процессора, представленных в виде символических обозначений.

Ассемблер - компилятор языка ассемблера в машинный код.

Команды ассемблера.

command op1, op2

command op1

command

Примеры:

mov ax,5 - поместить в регистр AX значение 5.

mov ax,bx - поместить в регистр AX значение из регистра BX.

Сегменты программы.

* Сегмент кода - наш код
* Сегмент данных - глобальные переменные
* Сегмент стека - стек вызовов

Сегменты располагаются в ОЗУ и имеют собственные адреса: CS(code),DS(data),SS(stack),ES,GS,FS

Логический адрес: DS:var = адрес сегмента : смещение

Минимальный размер - 16 байт. Максимальный - 64 Кбайт