МАШИННО-ЗАВИСИМЫЕ ЯЗЫКИ ПРОГРАММИРОВАНИЯ

ИУ7, 4-й семестр, 2020 г.

Организация курса

- видео-, аудиозапись и фотосъёмка запрещены
- 2 модуля + экзамен
- 8 лекций, N лабораторных работ
- 38 часов самостоятельной подготовки к лабораторным работам

Литература

- Зубков С. В. "Assembler. Для DOS, Windows и Unix"

Цели и программа курса

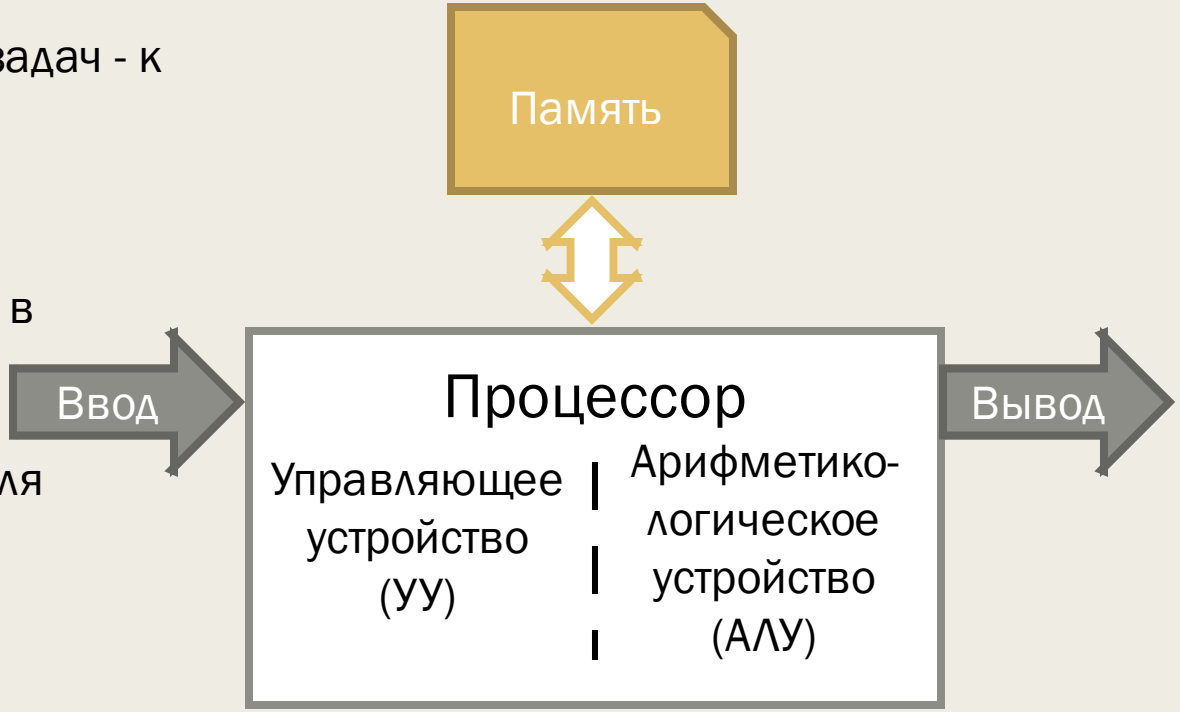
- Изучение низкоуровневого устройства ЭВМ
- Понимание исполнения программ на аппаратном уровне. Работа процессора
- Умение составлять и читать программы, включая:
  - *составление программы на низком уровне "с нуля"*
  - *взаимодействие программного кода с внешними устройствами*
  - *доп. возможности и расширения современных процессоров*
  - *отладку и реверс-инжиниринг исполняемых файлов*

История создания ЭВМ. Появление вычислителей общего назначения. Архитектура фон Неймана

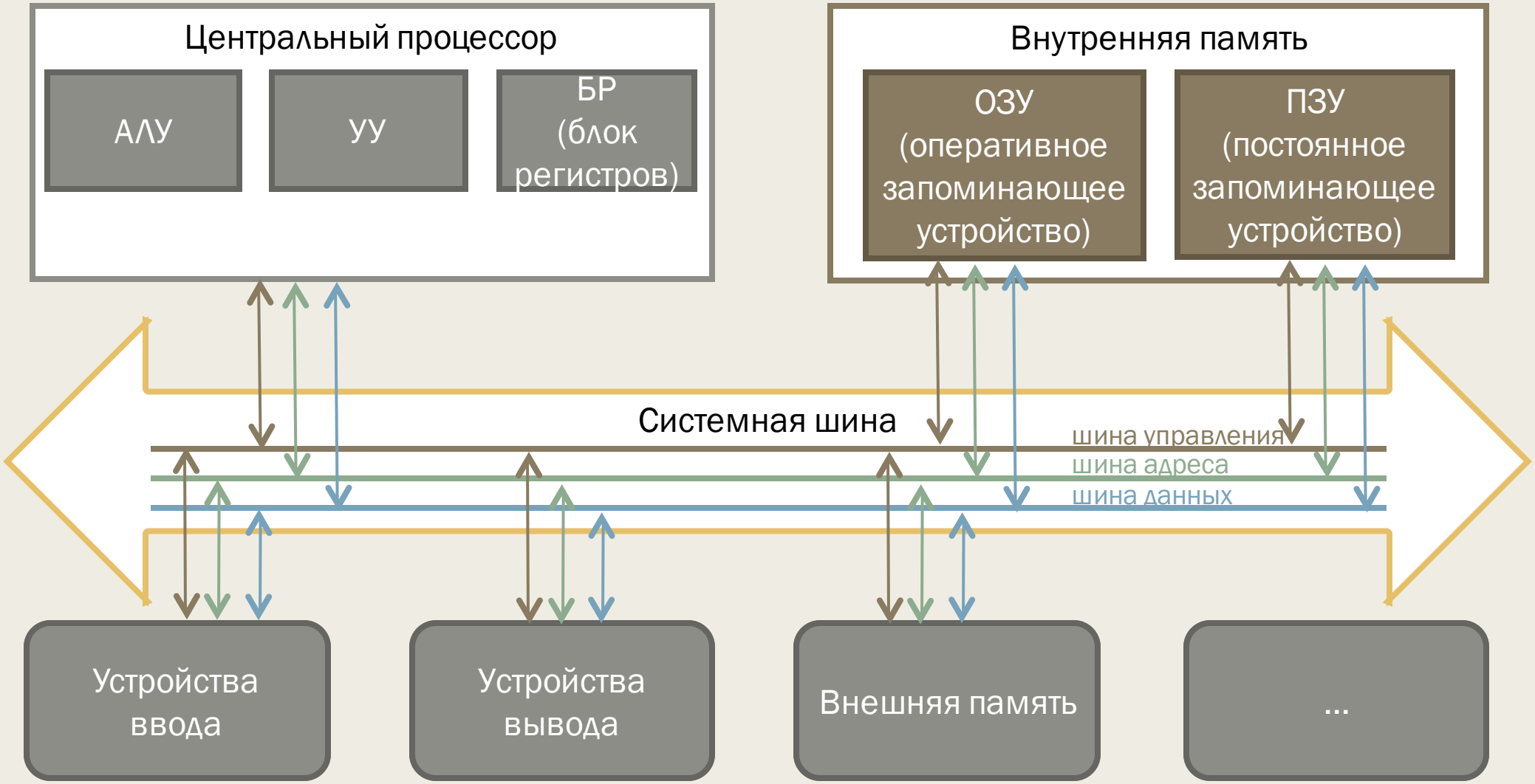
От решения частных вычислительных задач - к универсальным системам

<u>Принципы фон Неймана:</u>

1. Использование двоичной системы счисления в вычислительных машинах.
2. Программное управление ЭВМ.
3. Память компьютера используется не только для хранения данных, но и программ.
4. Ячейки памяти ЭВМ имеют адреса, которые последовательно пронумерованы.
5. Возможность условного перехода в процессе выполнения программы.



Структурная схема ЭВМ



Память. Единица адресации. Представление символов

Байт - минимальная адресуемая единица памяти

- 8 бит
- диапазон значений - 0..255
- $8 = 2^3 = 10_{16}{}^2$

Машинное слово — машинно-зависимая величина, измеряемая в битах, равная разрядности регистров/шины данных

Параграф - 16 байт

ASCII (*аски́*) - ***A**merican **s**tandard **c**ode for **i**nformation **i**nterchange, США, 1963.*

ASCII Code Chart

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	HT	LF	VT	FF	CR	SO	SI
1	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
2		!	"	#	$	%	&	'	(	)	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{	|	}	~	DEL

- *7-битная кодировка*
- *первые 32 символа - служебные*
- *старшие 128 символов 8-битной кодировки - национальные языки*

Системы счисления

Двоичная (binary)

- 0, 1, 10, 11, 100, 101…
- $2^8 = 256$
- $2^{10} = 1024$
- $2^{16}=65536$

Шестнадцатеричная (hexadecimal)

- 0, 1, …, 8, 9, A, B, C, D, E, F, 10, 11, 12, …, 19, 1A, 1B…
- $2^4=10$
- $2^8=100$
- $2^{16}=100000$

$$1011011011111000_2 = B6F8_{16}$$

Представление отрицательных чисел. Дополнительный код

-00101101 => инверсия и прибавление единицы:

- 11010010
- 11010011

-1 => 11111111

-1+1 = 0

11111111 + 1 = (1)00000000

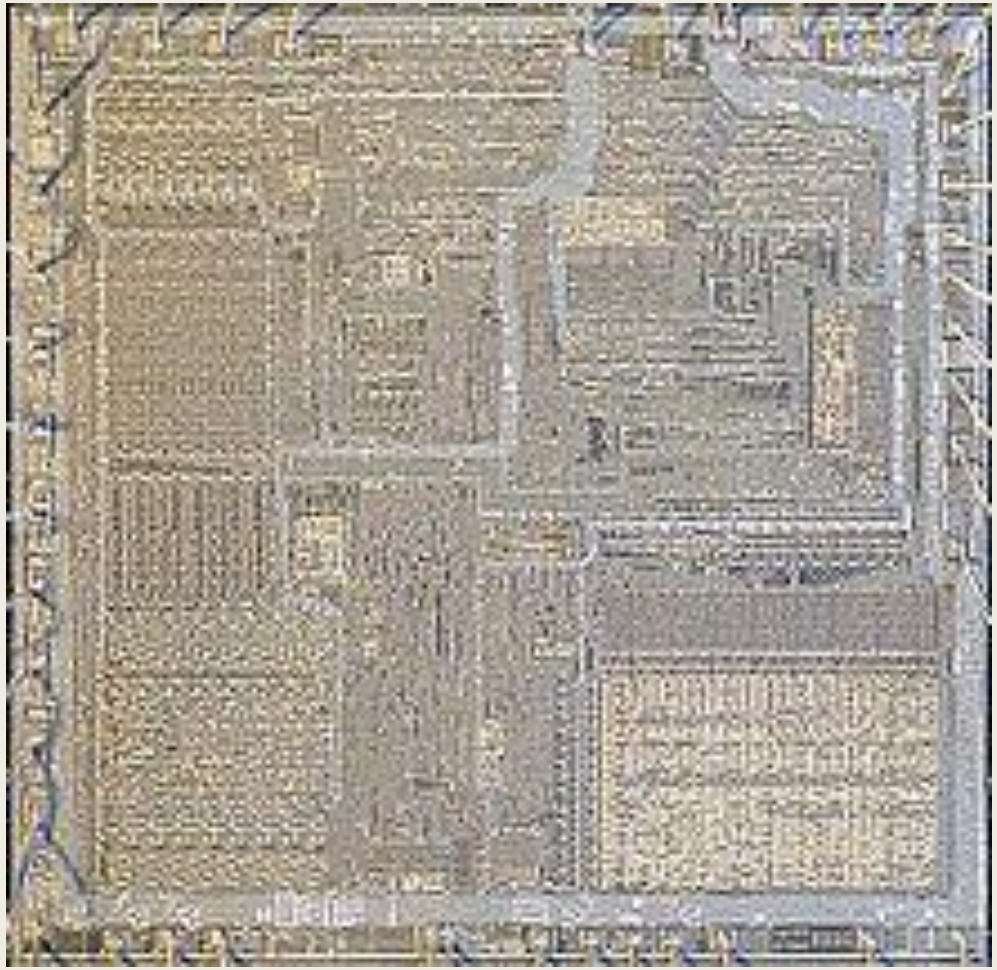
Виды современных архитектур ЭВМ

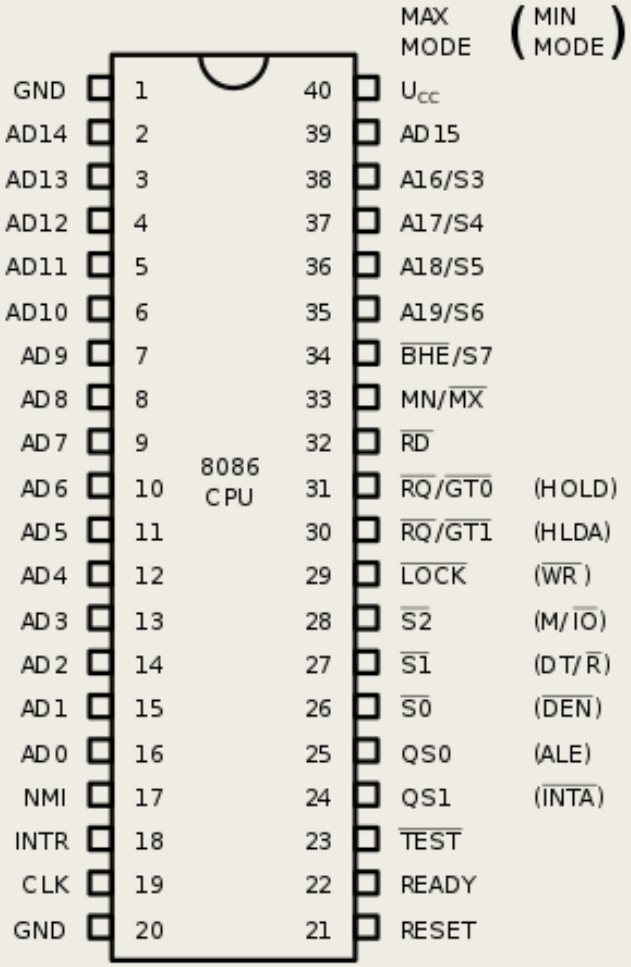
- x86
- x86-64
- IA64
- ARM
- MIPS (в т. ч. Байкал)
- Эльбрус

Семейство процессоров x86

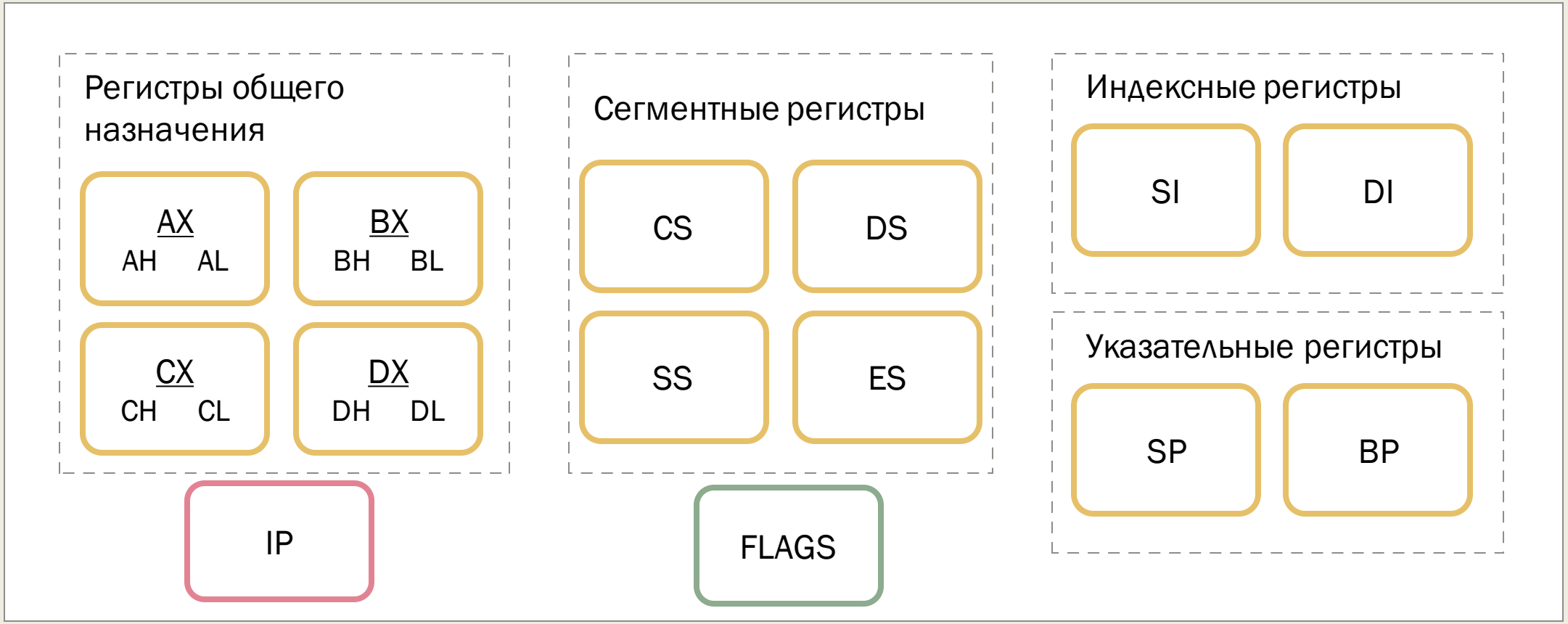
- Микропроцессор 8086: 16-разрядный, 1978 г., 5-10 МГц, 3000 нм
- Предшественники: 4004 – 4-битный, 1971 г.; 8008 – 8-битный, 1972 г.; 8080 – 1974 г.
- Требует микросхем поддержки
- 80186 – 1982 г., некоторое развитие, интегрированы микросхемы поддержки
- 80286 – 1982 г., 16-разрядный, добавлен защищённый режим
- 80386, 80486, Pentium, Celeron, AMD … - 32-разрядные, повышение быстродействия и расширение аппаратного функционала (системы команд)
- x86-64 (x64) - семейства с 64-разрядной архитектурой
- Советский аналог - К1810ВМ86, 1985 г.

Устройство 8086





Архитектура 8086 с точки зрения программиста



Язык ассемблера

Язык **ассемблера** - машинно-зависимый язык программирования низкого уровня, команды которого прямо соответствуют машинным командам.

Исполняемые файлы. Компиляция. Линковка

- **Исполняемый файл** — файл, содержащий программу в виде, в котором она может быть исполнена компьютером.
- Получение исполняемых файлов: компиляция + линковка.
- Компилятор - программа для преобразования исходного текста другой программы на определённом языке в объектный модуль.
- Компоновщик (линковщик, линкер) - программа для связывания нескольких объектных файлов в исполняемый.

Исполняемые файлы. Запуск программы. Отладчик

- .EXE, .COM. Запуск новой программы операционной системой:

1. Определение формата файла.
2. Чтение и разбор заголовка.
3. Считывание разделов исполняемого модуля (файла) в ОЗУ по необходимым адресам.
4. Подготовка к запуску, если требуется.
5. Передача управления на точку входа.

- Отладчик — программа для автоматизации процесса отладки. Может выполнять трассировку, отслеживать, устанавливать или изменять значения переменных в процессе выполнения кода, устанавливать и удалять контрольные точки или условия остановки.

"Простейший" формат исполняемого файла

.COM (command) - простейший формат исполняемых файлов DOS и ранних версий Windows.

- C< 64 Кб

Запуск COM-программы:

- Система выделяет свободный сегмент памяти и заносит его адрес во все сегментные регистры (CS, DS, ES и SS).
- В первые 256 байт этого сегмента записывается PSP.
- Непосредственно за ним загружается содержимое COM-файла без изменений.
- *Указатель стека (регистр SP) устанавливается на конец сегмента.*
- *В стек записывается 0000h (адрес возврата для команды ret).*
- Управление передаётся по адресу CS:0100h, где находится первый байт исполняемого файла.

Классификация команд 8086

- Команды пересылки данных
- Арифметические и логические команды
- Команды переходов
- Команды работы с подпрограммами
- Команды управления процессором

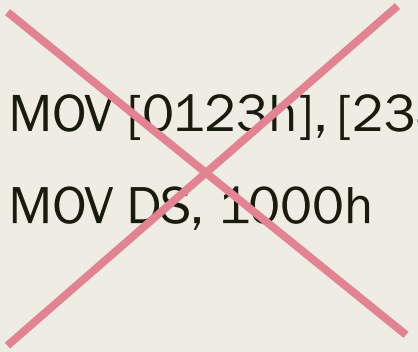
Команда пересылки данных MOV

MOV <приёмник>, <источник>

Источники: непосредственный операнд, РОН, сегментный регистр, переменная (ячейка памяти).

Приёмник: РОН, сегментный регистр, переменная (ячейка памяти).

- MOV AX, 5
- MOV BX, DX
- MOV [1234h], CH
- MOV DS, AX



- MOV [0123h], [2345h]
- MOV DS, 1000h

Целочисленная арифметика, базовые команды

- ADD <приёмник>, <источник> - выполняет арифметическое сложение приёмника и источника. Сумма помещается в приёмник, источник не изменяется.
- SUB <приёмник>, <источник> - вычитание. Аналогично ADD.
- MUL <источник> - умножение (без знака). Умножаются источник и AL/AX, в зависимости от размера источника. Результат помещается в AX либо DX:AX.
- DIV <источник> - деление (без знака). Деление AL/AX на источник. Результат помещается в AL/AX, остаток - в AH/DX.

Побитовая арифметика

- AND <приёмник, источник> - побитовое "И". AND al, 00001111b
- OR <приёмник, источник> - побитовое "ИЛИ". OR al, 00001111b
- XOR <приёмник, источник> - побитовое исключающее "ИЛИ". XOR AX, AX
- NOT <приёмник> - инверсия
- SHL <приёмник>, <счётчик> - сдвиг влево
- SHR <приёмник>, <счётчик> - сдвиг вправо

Команда безусловной передачи управления JMP

JMP <операнд>

- Передаёт управление в другую точку программы, не сохраняя какой-либо информации для возврата.
- Операнд - непосредственный адрес, регистр или переменная.

Пример

```
...
   XOR    AX, AX
   MOV    BX, 5
label1:
   INC    AX
   ADD    BX, AX
   JMP    label1
...
```

```
AX 0000   SI 0000   CS 19F5   IP 0100
BX 0000   DI 0000   DS 19F5
CX 0024   BP 0000   ES 19F5   HS 19F5
DX 0000   SP FFFE   SS 19F5   FS 19F5

CMD >

0100 33C0          XOR    AX,AX
0102 BB0500        MOV    BX,0005
0105 40            INC    AX
0106 03D8          ADD    BX,AX
0108 EBFB          JMP    0105
010A BA1401        MOV    DX,0114
010D CD21          INT    21
010F B44C          MOV    AH,4C
```

Взаимодействие программы с внешним миром (ОС, пользователь)

Прерывания:

- аппаратные
- программные

int - генерация программного прерывания.

21h - прерывание DOS, предоставляет примерно 70 функций.

Номер функции передаётся через ah, параметры каждой функции и возвращаемый результат описаны в документации.

Память в реальном режиме работы процессора (для 8086)

1 Мб памяти = 2^{20}

Номер параграфа начала сегмента (сегментная часть адреса, сегмент)

19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

Смещение

CS:IP, DS:BX, SS:SP…

[SEG]:[OFFSET] => физический адрес:
SEG*16 + OFFSET

5678h:7890h =>

```
  56780
+  7890
-------
  5E010
```

Логическая структура памяти. Сегменты

- Сегмент кода (CS)
- Сегменты данных (**DS**, ES, FS, GS)
- Сегмент стека (SS)