

# Лекция 4

**Архитектура деңгейлері –  
командалар және деректер  
жиыны. Деректер түрлері.  
Командалар форматы.  
Регистрлер және адрестер.  
Адресациялау. Командалар  
түрлері.**

# Лекция 4

## Деректердің берілуі

Компьютердегі ақпараттың ең аз өлшем бірлігі - бит болып табылады. Бит 0 немесе 1 мәнін қабылдайды.

8 биттен тұратын топ байтты құрайды. Биттер немесе байттағы разрядтар оңнан солға қарай 0-ден 7-ге дейін нөмірленеді:

Байт таңбасыз 0-ден 255-ке дейінгі және таңбалы -128-ден +127-ге дейінгі сандардан тұруы мүмкін.

Екі байт немесе он алты байт сөз құрайды. Сөздегі биттер оңнан солға қарай 0-ден 15-ке дейін нөмірленеді: Сөз таңбасыз 0-ден 65535-ке дейінгі және таңбалы -32768-ден +32767-ге дейінгі сандардан тұруы мүмкін.

Төрт байт немесе екі сөз екілік сөз құрайды, ол 32 биттен тұрады, ал екі екілік сөз төрт сөз құрайды (64 бит).

# Лекция 4

Компьютердегі бүкіл есептеулер екілік санау жүйесінде жүргізіледі, яғни екілік код арқылы. Сандарды оналтылық түрде көрсету, сандардың екілік кодтағы қысқа түрдегі жазуы болады. Ол үшін төрт екілік разрядты бір оналтылық разрядпен жазуға болады. Сандарды көрсетудің бұл екілік форматынан басқа ондық, екілік-ондық (BCD), және ASCII - код түрлері пайдаланылады.

Сол сияқты жад көлемінің одан да ірі туынды бірліктері кең қолданылады: *Килобайт*, *Мегабайт*, *Гигабайт*, сондай-ақ соңғы кезде *Терабайт* пен *Петабайт*.

# Лекция 4

## Екілік код.

Екілік кодта екілік екі цифр бар. Ол 0 және 1. Екілік санының белгісі ретінде В әрпі қолданылады. Ол сан соңына қойылады. Мысалы: 10101001В.

Екілік санның мәні әрбір биттің орналасқан позициясы мен ондағы бірлік биттердің бар не жоғы арқылы анықталады.

Мысалы:

$$\begin{aligned} 10001010\text{b} &= 1*2^7 + 0*2^6 + 0*2^5 + 0*2^4 + 1*2^3 + 0*2^2 + 1*2^1 + \\ &0*2^0 = \\ &= 128 + 8 + 2 = 138. \end{aligned}$$

# Лекция 4

## Ондық код.

Ондық кодта 0-ден 9-ға дейінгі сандар ондық цифрлар бар. Ондық санның белгісі болып D әрпі көтсетіледі. Ол санның ең аяғына қойылады. Егер санның соңында әріп болмаса онда сан ондық деп қабылданады.

# Лекция 4

## Оналтылық код.

Ондық кодта 0-ден 9-ға дейінгі сандар және А-дан F-ке дейінгі әріптер болады. Оналтылық санның белгісі болып Н әрпі көтсетіледі. Ол санның ең аяғына қойылады. Сан міндетті түрде цифрдан басталуы керек. Байт құрамында екі оналтылық разряд, сөз құрамында төрт разряд, екілік сөз құрамында сегіз разряд бар. Мысалы:

$$1\text{VA}8\text{h} = 1 * 16^3 + 11 * 16^2 + 10 * 16^1 + 8 * 16^0 = \\ 4096 + 2816 + 160 + 8 = 708$$

# Лекция 4

## ASCII - код.

Микропроцессор мен принтер, пернетақта немесе дисплей арасындағы деректермен өзара алмасу үшін ASCII - коды пайдаланылады. ASCII - коды (информация алмасуға арналған американдық стандартты код) компьютердегі алфавитті-цифрлық информацияны кодтауға арналған.

Мысалы 0-ден 9-ға дейінгі цифрларда 48-ден 57-ге дейін (30h тан 39h) ASCII - кодтары бар. “A” дан “Z” дейінгі бас әріптер 65-тен 90-ға дейін (41h тан 5Ah) ASCII - кодтары, ал “a” дан “z” кіші әріптері 097 ден 122-ге дейінгі (61h тан 7Ah) ASCII - кодтары бар. Алған нәтижені экранға шығару үшін оны ASCII - кодында көрсету керек.

# Лекция 4

## Адрестік кеңістіктің таратылуы

Дербес компьютер модификациясына және оның құрамындағы сыртқы құрылғыларға байланысты адрестік кеңістіктің таратылуы өзгешеленуі мүмкін. Бірақ жүйенің негізгі компоненттерінің орналасуы қатал қадағаланады. Компьютердің адрестік кеңістікті қолданудың қарапайым кестесі 1 суретте келтірілген. Бұл суреттегі адрес мәндері 16-лық есептеу жүйесінде берілген. 16-лық сан екендігін, саннан кейін орналасқан  $h$  әріпіне байланысты білуге болады.



# Лекция 4

1 Кбайт	Үзу векторлары	00000h	
256 байт	BIOS деректер аймағы	00400h	
512 байт	DOS деректер аймағы	00500h	
	IO.SYS және MS DOS.SYS		
	Жүктемеленетін драйверлер		> Стандартты жады (640Кбайт)
	COMAND.COM (резиденттік бөлім)		
	Жүктемеленетін қолданбалы және жүйелік программалар үшін арналған бос жады		
64 Кбайт	Графикалық буфер EGIA	A0000h =	
32 Кбайт	UMB	B0000h	
32 Кбайт	Текстік буфер EGIA	B8000h	
64 Кбайт	ПЗУ – BIOS ұлғайтулары	C0000h	> Жоғарғы жады (384 Кбайт)
64 Кбайт	UMB	D0000h	
128 Кбайт	ПЗУ BIOS	E0000h	
64 Кбайт	HMA	100000h =	
DOS 15 Мбайт (80286) 4 Гбайт-қа дейін (80386/486)	XMS	10FFF0h	> Ұлғайтылған жады

Қосымша жады (EMS)

1 сурет. Адрестік кеңістікті тарату кестесі

# Лекция 4

## Микропроцессордың пайдаланушы регистрлері

Микропроцессордың программалық моделі программалаушы қолдануға арналған 32 регистрдан тұрады. Берілген регистрлерді екі үлкен топқа бөлуге болады: 16 пайдаланушылар регистрі және 16 пайдаланушылар регистрі және 16 жүйелік регистрлер.

# Лекция 4

## Жалпы міндетті регистрлер

	31	15	0
eax		ax = ah + al	
ebx		bx = bh + bl	
ecx		cx = ch + cl	
edx		dx = dh + dl	
esp		sp	
ebp		bp	
esi		si	
edi		di	

## Сегменттік регистрлер

	15	0
	cs	
	ds	
	ss	
	es	
	fs	
	gs	

## Команда қирсеткіші және белгілер регистрі

	31	15	0
eflags		flags	
eip		ip	

# Лекция 4

## Микропроцессордың жүйелік регистрлері

Бұл регистрлердің атының өзі айтып тұрғандай, олар жүйедегі арнайы функцияларды орындайды. Жүйелік регистрлерді қолдану қатты қадағаланады. Осы регистрлер қорғалған режимде жұмыс жасауды қамтамасыздандырады. Және де бұларды микропроцессор архитектурасы ретіндеде қарауға болады, өйткені квалификациялы жүйелік программалаушы төменгі деңгейдегі операцияларды орындай алу үшін әдейі қалдырылған.

Жүйелік регистрлерді үш топқа бөлуге болады:

- төрт басқару регистрлері; **cr0, cr1, cr2, cr3.**
- төрт жүйелік адрес регистрлері; **GDTR, LDTR, IDTR, TR**
- сегіз жөндеу регистрлері. **dr0, dr1, dr2, dr3, dr4, dr5, dr6, dr7**

# Лекция 4

## Операндтарды адресациялау әдістері

Операндтар регистрлерде, жадыда және командалардың өзінде болуы мүмкін. Операндты табу тәсілі адресация режимдерімен анықталады.

1. Регистрлі адресация
2. Тікелей адресация
3. Төте адресация
4. Жанама адресация
5. Ығысумен база бойынша адресация
6. Масштабтаумен жанама адресация
7. Индекстелумен база бойынша адресация

# Регистрлі адресация

- Операнд кез – келген жалпы міндетті регистрде немесе сегменттік регистрде орналасуы мүмкін:
- `mov AX, BX`

# Тікелей адресация

- Операнд командада орналасады:
- а) `mov AX,10` ; 10 - тікелей операнд
- б) `mov BX,OFFSET A` ; `OFFSET A` - тікелей операнд
- в) `mov AX,K` ; `K` – тікелей операнд, егер `K EQU немесе = директивалары` арқылы анықталған болса , онда `K EQU 10` деп беріледі.

# Төте адресация

- Операнд жадыда орналасады.  
Командада операнд адресі көрсетіледі  
`mov AX,A ;`
- Деректер сегментінде келесі оператор түрінде берілген `A DW 10,20,30`, `A` – операнд адресі
- `mov AX,A+2 ; AX = 20`



# Жанама адресация

- Операнд жадыда орналасады, ал регистрде операнд адресі орналасады
- `mov AX,[BX]` ; BX-те операнд адресі орналасқан
- Квадраттық жақшалар регистрде адрес орналасқандығын білдіреді. 80386 процессорларына дейін операнд адрестерін BX, BP, SI, DI регистрлерінде ғана көрсетуге болатын, бірақ ол шектеулер кейіннен ұлғайтылып EAX, EBX, ECX, EDX, ESI, EDI, EBP және ESP регистрлерін қолдануға да болатын болды.

# Ығысумен база бойынша адресация

- Операнд жадыда орналасады, операнд адресі ығысу мен базалық регистр мәнінің қосындысы арқылы есептелінеді.
- `mov AX,[BX+2]`
- `mov AX,[BP] +2`
- `mov AX, 2[BX]`
- 80386 процессорларына дейін базалық регистр ретінде BX, BP, SI немесе DI және ығысу ретінде байт немесе сөзді ғана алуға болатын. 80386 процессорларынан және одан жоғарғыларында EAX, EBX, ECX, EDX, ESI, EDI, EBP және ESP регистрларын қолдануға болады. Бұл әдістің көмегімен бір өлшемді массив байттарына қатынау жасауға болады.

# Масштабтаумен жанама адресация

- Бұл әдіс алдыңғы әдіске ұқсас, бірақ бұның көмегімен регистрге элемент нөмірін жіберу арқылы сөз, екілік сөз және төрттік сөз массивтерінің элементтерін оқуға болады:
- `mov AX, [ESI*2] +2`
- 1, 2, 4 немесе 8 тең көбейткіш массив элементінің өлшеміне сәйкес – байт, сөз, екілік сөз немесе төрттік сөз. Регистрлердің ішінен тек EAX, EBX, ECX, EDX, ESI, EDI, EBP, ESP қолдануға болады, SI, DI, BP немес SP регистрлері қолдануға келмейді.

# Индекстелумен база бойынша адресация

- Бұл адресациялау әдісінде жадыдағы операндтың ығысуы, екі регистрдегі мәннің қосындысы арқылы және егер ол көрсетілсе, ығысуының қосындысы арқылы есептелінеді. Келесі формадағы жазбаларды қолдануға болады:
  - `mov AX,2[BX][SI]`
  - `mov AX, [BX+2][SI]`
  - `mov AX,[BX][SI]+2`
  - `mov AX,2[BX][SI+2]`
  - `mov AX,[BX+SI+2]`

## Лекция 4

**микропроцессор командалар жүйесіне шолу**  
микропроцессорлар тобының барлығы i8086  
командалар жиынымен жұмыс істейді. Командалар  
жиынын келесі топқа бөлуге болады:

1. деректерді алып-орналастыру командалары,
2. басқару тізгінін ауыстыру командалары,
3. арифметикалық командалар,
4. екілік разрядтармен жұмыс командалары,
5. тізбекті командалар,
6. ұзу командалары,
7. микропроцессорды басқару командалары.

## Лекция 4

СӨЖ тапсырмалары:

- 1 Екілік-ондық код (BCD)?
- 2 Командалар жүйесімен жұмыс?