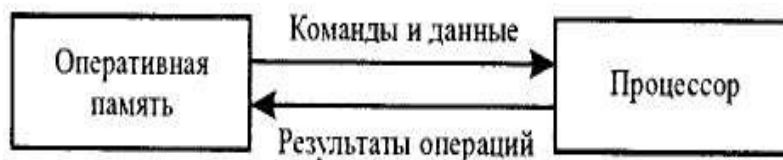


№ 3 дәріс. Микропроцессорлық архитектураның ерекшеліктері

Жад және процессор

Компьютерге кіретін құрылғылар мен түйіндердің ішінде кез-келген бағдарламаны орындау үшін ең маңыздысы-жедел жад және Орталық микропроцессор. Жедел жадта орындалатын бағдарлама өзіне тиесілі деректермен бірге сақталады; процессор бағдарламада сипатталған есептеулер мен басқа әрекеттерді орындайды. Бағдарлама қатты немесе икемді магниттік дискінің жадына жүктеледі, онда амалдық жүйе сақталады, оған жауап ретінде бағдарламаны іске қосу пәрменінің пернетақтасын енгізу. ОЖ, бағдарламаны жүктеу және қажет болған жағдайда оны жад аймағында орындау үшін конфигурациялау, процессорға жүктелген бағдарламаның бастапқы мекен-жайын айтады және оны орындау процесін бастайды. Процессор бағдарламаның бірінші пәрменін жадтан оқиды, оны орындау үшін қажетті деректерді жадтан немесе оның регистрлерінен табады және қажетті әрекетті орындағаннан кейін жадқа қайтарады немесе өз жұмысының нәтижесін регистрлерде қалдыруы мүмкін.



Сурет 1. Жедел жад пен процессордың өзара әрекеттесуі

Бірінші пәрменді орындағаннан кейін процессор келесіге ауысады және т.б. бағдарламаның соңына дейін. Бағдарламаны аяқтағаннан кейін процессор бұдан әрі не істеу керектігін білмейді, сондықтан кез-келген бағдарлама басқаруды компьютердің амалдық жүйесіне жіберетін командалармен аяқталуы керек.

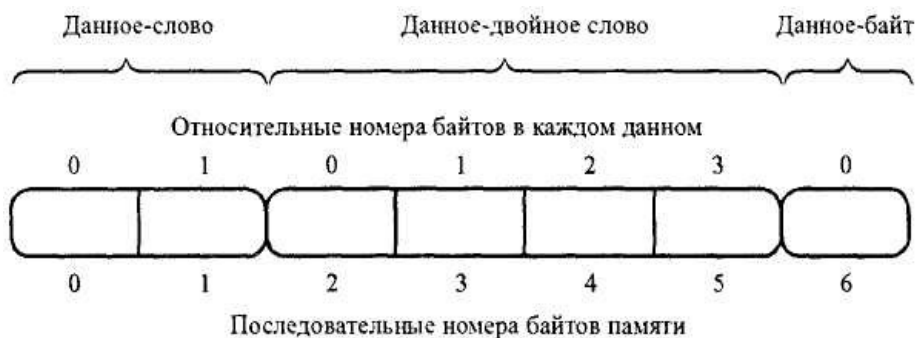
Компьютердің жедел жады-бұл көптеген екілік сақтау элементтерінен, сондай-ақ оларды басқару схемаларынан тұратын электрондық құрылғы. Жадта қол жетімді ақпараттың ең аз мөлшері-бір байт (8 екілік бит немесе бит). Барлық жедел жад байттары нөлден бастап нөмірленеді. Қажетті байттар мекен-жай функцияларын орындайтын нөмірлері бойынша жадта ізделеді.



Сурет 2. Байт, сөз және қос сөз

Кейбір деректер (мысалы, таңбалар кодтары) оларды сақтау үшін бір байтты қажет етеді: басқа деректер үшін бұл орын жетіспейді және олардың астында жадта 2,4,8 немесе одан да көп байт саны бөлінеді. Әдетте байт жұптары сөз деп аталады, ал төртеуі қос сөз деп аталады (2 – сурет), дегенмен кейде "сөз" термині машиналық ақпараттың кез келген бөлігін білдіреді.

Берілген көп байттың мазмұнын талқылау кезінде оның құрамдас байттарына сілтеме жасау керек; бұл байттар шартты түрде нөлден нөмірленеді және санның оңнан солға қарай өсу ретімен орналастырылады (қағазда бейнеленген кезде), сол жақта үлкен сандары бар байттар, ал оң жақта Кіші сандары бар байттар болады. Сол жақтағы байт әдетте аға, ал оң жақтағы байт кіші деп аталады. Байттарды орналастырудың бұл тәртібі біз үшін сандарды жазудың әдеттегі формасымен байланысты: көп биттік санда сол жақта жоғары разрядтар, ал оң жақта Кіші разрядтар көрсетіледі. Келесі Сан, егер алдыңғы саннан кейін жазылса, қайтадан аға дәрежеден басталып, кіші санмен аяқталады. Дегенмен, компьютер жадында деректер байт нөмірлерінің үздіксіз өсуінің табиғи тәртібінде орналасады және осылайша жадтағы әрбір сөз немесе қос сөз оның кіші байтынан басталып, үлкенімен аяқталады (3-сурет).



Сурет 3. Көп байтты деректердегі байттарды нөмірлеу

Қатаң айтқанда, компьютердің жадында тек бүтін екілік сандарды сақтауға болады, өйткені жад екілік сақтау элементтерінен тұрады. Басқа деректерді жазу үшін, мысалы, таңбалар немесе Бөлшек сандар, олар үшін кодтау ережелері қарастырылған, яғни.сол немесе ұзындықтағы биттер тізбегі түрінде ұсыну. Сонымен, бір дәлдіктің нақты саны жадта қос сөзді (32 бит) алады, онда 23 бит мантиссаға, 8 бит ретке және тағы бір бит сан белгісіне бөлінеді. Мұндай деректермен жұмыс істейтін бағдарламалар, әрине, оларды жазу ережелерін білуі керек және оларды өңдеу және ұсыну кезінде басшылыққа алуы керек.

Барлық сандық электронды құрылғылар жұмыс істейтін екілік санау жүйесі адамға ыңғайсыз. Жад ұяшықтарының немесе процессор регистрлерінің екілік мазмұнын ұсынудың қарапайымдылығы үшін кейде сегіздік, көбінесе он алтылық Сандық жүйелер қолданылады. Intel процессорлары үшін он алтылық жүйе қолданылады. Он алтылық санның әр разряды 16 мәнді қабылдай алады, оның алғашқы 10 – ы қарапайым ондық сандармен, ал соңғы 6-ы латын әліпбиінің А-дан F - ге дейінгі әріптерімен белгіленеді, мұндағы А 10, В - 11, С - 12, D-13, E-14, аF-15 білдіреді. Ассемблер тілінде ондық бөлшектерден ажырату үшін он алтылық сандар h (немесе H) әрпімен аяқталады. Осылайша, 100-ондық сан, ал 100h – он алтылық (256-ға тең). Бір он алтылық Сан оны компьютердің жадына жазу үшін төрт екілік разрядты қажет ететіндіктен, байт мазмұны екі он алтылық цифрмен сипатталады (00h – ден FFh-ге дейін немесе 0-ден 255-ке дейін), ал сөздің мазмұны төртеуімен (0000h-ден doffffh, немесе 0-ден 65535-ке дейін).

Жедел жад ұяшықтарынан басқа, деректерді сақтау үшін процессорда орналасқан және регистрлер деп аталатын сақтау ұяшықтары да қолданылады. Регистрлердің артықшылығы олардың жоғары жылдамдығында, жедел жадқа қарағанда әлдеқайда көп, ал кемшілігі-олардың саны өте аз – тек он шақты. Сондықтан регистрлер деректерді қысқа мерзімді сақтау үшін ғана қолданылады.

Сегменттік адрестеудің мәні келесідей. Жадқа жүгіну тек сегменттердің көмегімен жүзеге асырылады-Nate немесе физикалық жадтың басқа бөлімдері салған логикалық құрылымдар. Кез-келген жад ұяшығының атқарушы мекен-жайын процессор осы ұяшық орналасқан сегменттің бастапқы мекен-жайын қосу арқылы есептейді, оған сегменттің басынан (байт)мещысады. Бұл орын ауыстыру кейде салыстырмалы мекен-жай деп аталады.

Төрт кіші бит жоқ сегменттің бастапқы мекен-жайы, яғни 16-ға бөлінген сегменттік регистрлердің біріне орналастырылған және сегменттік мекен-жай деп аталады. Бастапқы мекен-жайдың өзі көлеңкелі регистр деп аталатын процессордың арнайы ішкі регистрінде сақталады. Әрбір сегменттік регистрдің өзіндік көлеңкелі регистрі бар; сегменттің бастапқы мекен-жайын бағдарлама тиісті сегменттік регистрге сегменттік мекен-жайдың жаңа мәнін енгізген кезде процессор жүктейді.

Сегменттік мекен-жайды 16-ға көбейту процедурасы (немесе сол сияқты, 10 h) нақты режимнің негізгі ерекшелігі болып табылады, ол нақты режимде қол жетімді мекен-жайлардың ауқымын 1 Мб-қа шектейді. Шынында да, сегменттік мекен-жайдың максималды мәні – FFFFh немесе 64К-1, бұл жадтағы сегменттің бастапқы мекен-жайының максималды мәні FFFF0h немесе 1 Мбайт-16. Егер сегменттің бастапқы мекен-жайына 0-ден FFFFh-ге дейінгі кез-келген орын ауыстыруды қосуға болатындығын ескерсек, онда соңғы адрестік байттың мекен-жайы 10-ға тең болады.Мбайт+64 Кбайт-17.

Процессор құрған адрестер ауқымы процессордың адрестік кеңістігі деп аталады; Нақты режимде ол 1 Мб-тан сәл асады. 1 Мбайт шегінде мекенжайды сипаттау үшін 20 екілік разряд немесе 5 он алтылық қажет. 8086 процессорында тек 20 мекен - жай желісі бар, сондықтан 1 Мбайттан асып кете алмайды; қазіргі заманғы 32 биттік процессорлар, егер олар нақты режимде жұмыс істесе, сәл үлкенірек (шамамен 64 Кбайт) мекен-жай кеңістігі бар. Егер процессор қорғалған режимде жұмыс істесе (32 биттік регистрлерді қолдана отырып), онда оның адрестік кеңістігі 232=4 Гбайтқа дейін артады

Процессор регистрлері, жалауша регистрлері

Мысалы, Pentium сияқты заманауи микропроцессорларда нақты режимде пайдалануға арналған және іс жүзінде 8086 процессорына сәйкес келетін бөлікті (біз оны MP 86 деп атадық) ажыратуға болады. Процессорда он екі 16 биттік бағдарламалық адрестік регистрлер бар, олар әдетте үш топқа біріктіріледі: деректер регистрлері, көрсеткіштер регистрлері және сегменттік регистрлер. Деректер регистрлері мен көрсеткіштер регистрлері көбінесе "жалпы мақсаттағы регистрлер"жалпы атауымен біріктіріледі. Сонымен қатар, процессордың құрамына командалық Индекс және жалаулар тізілімі кіреді.

Басқа есептеу жүйелерінің процессорының күй регистріне баламалы жалаулар регистрінде процессордың ағымдағы күйі туралы ақпарат бар (сурет. 3.2). Оған 6 күй жалаушасы және 3 процессордың күйін басқару биттері кіреді, бірақ оларды әдетте жалаулар деп те атайды.

Биты															
15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
				OF	DF	IF	TF	SF	ZF		AF		PF		CF

CF тасымалдау жалауы (Carry Flag) арифметикалық амалдарды орындау кезінде тасымалдауды немесе несиені көрсетеді, сонымен қатар Жүйелік функцияларға сілтеме жасау кезінде қате көрсеткіші ретінде қызмет етеді.

PF Паритет Туы (Parity Flag) 1-ге орнатылады, егер операция нәтижесінің төменгі 8 биті екілік бірліктердің жұп санын қамтыса.

Af көмекші тасымалдау жалауы (Auxiliary Flag) оралған екілік ондық сандарға қатысты операцияларда қолданылады. Ол аға дәптерден (4-бит) тасымалдауды немесе несиені көрсетеді.

Егер операцияның нәтижесі нөлге тең болса, ZF нөлдік жалауы (нөлдік жалау) 1-ге орнатылады.

SF белгісінің жалауы (sign Flag) теріс нәтиже кезінде 1-ге орнатылып, операция нәтижесінің белгісін көрсетеді.

Of (Overflow Flag) толып кету жалаушасы толып кетуді тіркейді, яғни операция нәтижесінің берілген процессор үшін рұқсат етілген мәндер ауқымынан тыс шығуы.

Күй жалаушаларын процессор әр пәрменді орындағаннан кейін автоматты түрде орнатады. Сонымен, егер AX регистрінде 1 Саны болса, онда декремент пәрменін орындағаннан кейін (1-ге азайту)

Деректер регистрлері тобына AX, VH, SX және DX төрт регистрі кіреді. Бағдарламашы оларды кез-келген нысанды (деректерді немесе мекен-жайларды) уақытша сақтау және оларға қажетті операцияларды орындау үшін өз қалауы бойынша қолдана алады. Бұл ретте регистрлер аға (AH, BH, CH және DH) және кіші (al, BL, CL және DL) жартыларына тәуелсіз жүгінуге мүмкіндік береді.

Деректер регистрлері тобына төрт AX, BX, CX және DX регистрлері кіреді. Сіз оларды кез-келген нысанды (деректерді немесе мекен-жайларды) уақытша сақтау және оларға қажетті операцияларды орындау үшін өз қалауыңыз бойынша пайдалана аласыз. Бұл жағдайда регистрлер үлкен (AH, BH, CH және DH) және кіші (al, BL, CL және DL) жартысына тәуелсіз қол жеткізуге мүмкіндік береді. Сонымен, команда EO регистрінің аға байтын ШҚ регистрінің кіші байтына жібереді, бұл ретте регистрлердің екінші байтына әсер етпейді. Алдымен операнд қабылдағышы, ал үтірден кейін операнд көзі көрсетіледі, яғни команда оңнан солға қарай орындалады.

Көптеген жағдайларда деректер регистрлері өте эквивалентті, бірақ AX регистрін қолданған жөн, өйткені көптеген командалар жадта аз орын алады және егер олардың ОПЕРАНДТАРЫ AX регистрі (немесе оның жартысы AL) болса, тезірек орындалады. Екінші жағынан, бірқатар командалар регистрлерді жасырын түрде пайдаланады. Сонымен, барлық цикл командалары CX регистрін қайталау санының есептегіші ретінде пайдаланады; көбейту және бөлу командаларында AX және DX регистрлері жасырын Операндтар ретінде әрекет етеді; енгізу-шығару операцияларын тек AX немесе AL регистрлері арқылы жүзеге асыруға болады және т. б.

Көрсеткіш регистрлер тобының соңғысы, sp стек көрсеткіші, басқалардан бөлек тұрады, бұл тек стек шыңының көрсеткіші ретінде қолданылады-арнайы құрылым. Si, DI, BP және SP регистрлері, деректер регистрлерінен айырмашылығы, байтты адрестеуге жол бермейді. Төрт сегменттік CS, DS, ES және SS регистрлері процессор архитектурасының маңызды элементі болып табылады, 16 биттік Операндтар арқылы 20 биттік мекенжай

кеңістігін адрестеуге мүмкіндік береді. IP командалық индексі әр сәтте орындалатын команданың салыстырмалы мекен-жайын көрсете отырып, бағдарламаның орындалу барысын "бақылайды". IP регистрі бағдарламалық түрде қол жетімді емес (IP-бұл бағдарламалау тілінде қолданылатын мнемоникалық белгілеу емес, оның қысқартылған атауы); ондағы мекен-жайды құру Ағымдағы команданың ұзындығын ескере отырып, микропроцессорды орындайды. Ауысулар, үзілістер, ішкі бағдарламаларды шақыру және олардан қайтару командалары IP мазмұнын өзгертеді, осылайша бағдарламаның қажетті нүктелеріне ауысады. Басқа есептеу жүйелерінің процессор күйінің регистріне баламалы жалаулар регистрі (көбінесе жалаулар деп аталады) процессордың ағымдағы күйі туралы ақпаратты қамтиды.

Негізгі әдебиеттер: [1,2,3].

Қосымша әдебиеттер: [8,11].

Бақылау сұрақтары:

1. Жедел жад пен процессордың өзара әрекеттесу принципін сипаттаңыз.
2. Компьютердің жедел жады дегеніміз не?
3. Жадқа қол жеткізу үшін ақпараттың ең аз мөлшері қандай?
4. Көп байтты деректерде байттарды нөмірлеу қалай жүзеге асырылады?
5. Процессор регистрлері мен жалауша регистрлерін тізімдеңіз.