



RISC- және CISC-архитектуралары,
олардың негізгі құрылу принциптары мен таратылуы
Адресация әдістері және машиналық команда типтері.

Командалары жүйесінің тиімділігі
Стектің архитектурадағы компьютерлер
Микропроцессорлар
Негізгі сипаттамалары

Базалық микропроцессор архитектурасы. Элементтерінің қарым қатынасы
Intel микропроцессорларының даму кезеңдері

Оқытушы: Жекамбаева М.Н.
«Программалық инженерия»
кафедрасының профессоры
m.zhekambayeva@satbayev.university

ЭЕМ архитектурасы

ЭЕМ Архитектурасы – бұл көпдеңгейлі аппаратты-программалық құралдар иерархиясы. Әрбір деңгей көпвариантты құру және қолданудан тұрады. Деңгейлердің нақты таратылуы ЭЕМ құрылымының ерекшеліктерін анықтайды.

Кеңінен ашып қарастыратын болсақ архитектура жүйенің ұйымдастырылуы түсінігін қамтиды, соның ішінде компьютерді өңдеудің аспектілерінің жады жүйесі, жүйелік шина құрылымы, енгізу/шығаруды ұйымдастыру және т.б.

Ал тар мағынада архитектура деп командалар жиынының архитектурасын қарастырамыз.

RISC- және CISC-архитектуралары, негізгі құру және тарату принциптері

Есептеу техникасының дамуының қазіргі заманауи кезеңінде екі негізгі командалар жиынының архитектурасы қолданылады, олар - **CISC** (**C**omplete **I**nstruction **S**et **C**omputer) және **RISC** (**R**estricted (reduced) **I**nstruction **S**et **C**omputer) архитектуралары.

CISC-архитектурасының толық командалар жиындарынан тұратын архитектура, негізін қалаушы – IBM фирмасы десек болады, оның базалық архитектурасы IBM/360, оның ядросы 1964 ж. бері біздің кезімізге дейін жетті, мысалы, заманауи IBM ES/9000 мейнфреймдері.

Ал толық командалары жиыны бар микропроцессорларды өңдеу лидерлері Intel компаниясы, ол X86 және Pentium МП. Бұл МП нарықтағы стандарты десек те болады.

RISC және CISC

RISC архитектурасы жылдам және тиімді орындалатын нұсқаулардың минималды жиынтығын пайдалануға негізделген. Ол сонымен қатар процессордың функционалдығын кеңейтуді жеңілдететін жоғары көлденең масштабтауға қолдау көрсете

CISC архитектурасы бір циклде бірнеше әрекеттерді орындай алатын күрделірек нұсқаулар жинағын пайдаланады. Бұл тапсырманы орындау үшін қажетті нұсқаулар санын азайтуға мүмкіндік береді, бірақ процессордың күрделілігіне байланысты нұсқаулардың орындалуын баяулатуы мүмкін. ді.

RISC архитектурасын жүзеге асыратын процессорлардың кейбір түрлері:

- ARM (Advanced RISC Machines) – смартфондар мен планшеттер сияқты мобильді құрылғыларда қолданылады.
- MIPS (микропроцессоры без блокированных құбырлар кезеңдері) – көптеген өнеркәсіптік және контроллерлерде қолданылады.
- PowerPC – кейбір дербес компьютерлер мен серверлерде қолданылады.
- SPARC (Scalable Processor Architecture) – кейбір суперкомпьютерлерде және өнімділігі жоғары жүйелерде қолданылады.

ARM процессорлары бар келесі құрылғылар танымал мобильді құрылғылар болып табылады:

- Apple iPhone
- Samsung Galaxy серия смартфонов
- Google Pixel
- Huawei P серия смартфонов
- Xiaomi Redmi
- iPad и другие планшеты Apple
- Samsung Galaxy Tab
- Amazon Kindle Fire
- Huawei MediaPad
- Lenovo Tab



ARM (Advanced RISC Machines) - қуатты аз тұтынуымен, жинақылығымен және жоғары өнімділігімен танымал процессор архитектурасы. Ол құрылғылардың кең ауқымында қолданылады, соның ішінде ұялы телефондар, планшеттер, интернет құрылғылары, ендірілген жүйелер және т.б.

ARM – смартфондар мен планшеттердің 95%-дан астамында қолданылатын мобильді құрылғылар процессорларының нарығының көшбасшысы. Ол жоғары өнімділік пен төмен қуат тұтынуды ұсынады, бұл оны шектеулі батареямен жұмыс істейтін құрылғылар үшін тамаша таңдау жасайды.



ARM RISC (Reduced Instruction Set Computing) архитектурасына негізделген, яғни оның процессорлары нұсқаулар жиыны кеңірек процессорларға қарағанда аз қуатты қажет ете отырып, жылдамырақ және тиімдірек жұмыс істеуге мүмкіндік беретін минималды нұсқаулар жинағын пайдаланады. Осы архитектураның арқасында ARM қуатты аз тұтынуға және жоғары өнімділікке байланысты смартфондар мен планшеттер сияқты мобильді құрылғыларда пайдалану үшін танымал таңдау болып табылады.

RISC- және CISC-архитектуралары, негізгі құру және тарату принциптері

Бірақ бұл жобалау соңғы жетістіктегі YIC технологиялары негізіндегі суперминикомпьютерлерді құру үшін ыңғайсыз, әрі мұндай ауысу техникалық шектеулер қойылатын кристалл ресурстарына тиімсіз болды, олар: көлемі, транзисторлар саны, кеңейтілу қуаттылығы және т. б.

Осы шектеулерді жою мақсатында Берклиде (АҚШ, Калифорния) регистрге-бағытталған RISC – архитектурасы өңделді. Мұндай компьютерлерді қысқартылған командалары жиыны бар компьютерлер деп атайды. Оның мақсаты жиы қолданылатын операцияларды жылдам тарату үшін ерекшелеу және архитектурасын құру. Бұл ресурстарды шектей отырып жоғары жіберу жылдамдықты компьютерлер құруға мүмкіндіктер берді.

RISC- және CISC-архитектуралары, негізгі құру және тарату принциптері

Осылайша компьютерлік индустрияда RISC-архитектурасының бумы пайда болды. Жұмыс станциялар және серверлер осы концепция арқылы құрыла бастады.

Негізгі төрт RISC-архитектурасының принциптері:

- Әрбір команда типіне тәуелсіз бір машиналық циклмен орындалады, оның ұзақтығы максимальды қысқа б.к.;
- Барлық командалар ұзындығы бірдей болуы қажет және минимум адрес форматтарын қолдануы керек, ол орталық процессорды басқару логикасын жеңілдетеді;
- Жадыға қатынау тек жазу және оқу операциясы кезінде ғана жүргізіледі, ал қалған өңдеулер процессордың регистрлік құрылымында орындалады;
- Командалар жүйесі жоғары деңгей тілін қолдайды (яғни басқа да программалау тілдерімен оңай қарым қатынас жасайды).

RISC- және CISC-архитектуралары, негізгі құру және тарату принциптері

RISC-процессорларының архитектурасының қарапайымдылығы оның компакттілігін, кристалды суытудың қажетінің болмауын, CISC архитектурасының даму жолдарын қатты қолдауын қамтамасыздандырады. CISC-компьютерлерінің тиімділігін арттыру, бірінші кезекте, компиляторларды жеңілдеті және екінші, орындалатын модульді минимизациялау арқылы орындалды, Қазіргі кезде CISC- процессорлары дербес компьютерлерді дайындауда кеңінен қолданылады, бірақ RISC-процессорларына сервер және жұмыс станцияларын өнімділігін арттыру кезінде қолдану дұрыс деп есептеледі.

Негізгі RISC-архитектурасы және CISC-архитектурасының салыстырылуы келесі кестеде келтірілген:

Архитектураның негізгі салыстырулары

CISC-архитектурасы	RISC-архитектурасы
Көпбайты командалар	Бірбайтты командалар
Аз регистрлер саны	Көп регистрлер саны
Қиын командалар	Қарапайым командалар
Бір процессор циклында бір немесе одан аз командалар	Бір процессор циклында бірнеше команда
Дәстүрлі бір орындау құрылғысы	Бірнеше орындау құрылғылары

Адресация әдістері және машиналық команда типтері. Командалар жүйесін тиімділеу

Метод адресации	Пример команды	Смысл команды	Использование команды
Регистровая	<u>Add</u> R4, R3	$R4 = R4 + R3$	Для записи требуемого значения в регистр
Непосредственная или литерная	<u>Add</u> R4, #3	$R4 = R4 + 3$	Для задания констант
Базовая со смещением	<u>Add</u> R4, 100(R1)	$R4 = R4 + M(100 + R1)$	Для обращения к локальным переменным
Косвенная регистровая	<u>Add</u> R4, (R1)	$R4 = R4 + M(R1)$	Для обращения по указателю к вычисленному адресу
Индексная	<u>Add</u> R3, (R1 + R2)	$R3 = R3 + M(R1 + R2)$	Полезна при работе с массивами: R1 – база, R3 – индекс
Прямая или абсолютная	<u>Add</u> R1, (1000)	$R1 = R1 + M(1000)$	Полезна для обращения к статическим данным

Адресация әдістері және машиналық команда типтері. Командалар жүйесін тиімділеу

Косвенная	<u>Add</u> R1, @(R3)	$R1 = R1 + M(R3)$	Если R3 – адрес указателя p, то выбирается значение по этому указателю
Автоинкрементная	<u>Add</u> R1, (R2)+	$R1 = R1 + M(R2)$ $R2 = R2 + d$	Полезна для прохода в цикле по массиву с шагом: R2 – начало массива. В каждом цикле R2 получает приращение d
Автодекрементная	<u>Add</u> R1, (R2)-	$R2 = R2 - d$ $R1 = R1 + M(R2)$	Аналогична предыдущей; обе могут использоваться для реализации стека
Базовая индексная со смещением и масштабированием	<u>Add</u> R1, 100(R2)(R3)	$R1 = R1 + M(100) + R2 +$ $R3 * d$	Для индексации массивов

Адресация әдістері және машиналық команда типтері. Командалар жүйесін тиімділеу

Тікелей деректердің және литерлік тұрақтылардың адресациясы әдетте жадыны адрестеудің әдістерінің бірі ретінде қарастырылады.

Кестеде қосу (*Add*) мысалы келтірілген, ол жиі кездесетін адрестеу түрлеріне қолданылған. Кестедегі "=" белгісі меншіктеу операциясын көрсетеді, ал *M* әріпі жадыны (*Memory*) білдіреді. Осылайшы *M(R1)* *R1* регистрінің мәнімен анықталатын жады ұяшығын көрсетеді.

Ал қиын адрестеу әдістерін қолдану программадағы командалар санын азайту үшін қолданылады, бірақ бұл кезде аппаратура қиындығы біршама артады.

Негізгі командалар типтері

Операция типтері	Мысалдары
Арифметикалық және логикалық	Бүтінсанды арифметикалық және логикалық операциялар: қосу, азайту, логикалық қосу, логикалық көбейту және т. б.
Дерекерді жіберу	Жүктеу/жазу операциялары
Командалар ағынын басқару	Шартсыз және шартты аусыулар, процедура шақыру және процедурадан қайту
Жүйелік операциялар	Жүйелік шақырулар, Виртуалды жадыны басқару командалары және т. б.
Жылжымалы нүктелі сандармен операциялар	Қосу, азайту, көбейту және бөлу операциялары

Стектік архитектурадағы компьютерлер

Компьютер құрған кезде оған сонымен бірге командалар жүйесін де қоса береді. Командалар жүйесіне байланысты операциялардың маңызды әсерлері көрсетіледі, олар:

Элементтік базасы және компьютер өндірісінің технологиялық деңгейі;

Шешілетін тапсырмалар классы, жеке командаларда орындалатын керекті операцияларды анықтайды;

Командалар жүйесі сәйкесінше компьютерлер класы үшін;

Жылдамдық талаптары деректерді өңдеуде керек болады, олар ұзындығы үлкен командаларды құру үшін қолдануылуы мүмкін (VLIW-командалар).

Тапсырмалар талдауы көрсетіп отырғандай, қарапайым адрестеу режимдерін және регистрлерді қолданатын алып орналастыру және процессорлық командалар аралас программалары күшті роль атқарады.

Стектік архитектурадағы компьютерлер

Қазіргі күні келесі команда құрылымдары ең көп кездесетіндері деп есептеледі: бірадресті (1А), екіадресті (2А), үшадресті (3А), адрессіз (БА), ұзындығы үлкен сөздер командалары (VLIW – БДС):

1А ~

КОП	А1
-----	----

2А ~

КОП	А1	А2
-----	----	----

3А ~

КОП	А1	А2	А3
-----	----	----	----

БА ~

КОП

БДС ~

КОП	Адреса	Теги	Дескрипторы
-----	--------	------	-------------

Стектік архитектурадағы компьютерлер

Көрсетіліп тұрғандай операнд адрес арқылы және команда құрылымында көрсетіле алады. БА-команда жағдайында операндтар стектен таңдалады, нәтиже стекке орналастырылады. Олардың негізгі сипаты стектік жадысының болуы болып табылады.

Стек – бұл деректер және операцияларды уақытша сақтауға арналған жады аймағы. Стек элементтеріне қатынау FILO (first in, last out) – бірінші кірген, соңғы шығады әдісімен орындалады.

Мысалы стектік жадыны ұйымдастыру.

Келесі мысалдың орындалуын қарастырайық. Бірадресті компьютер арқылы жасайық

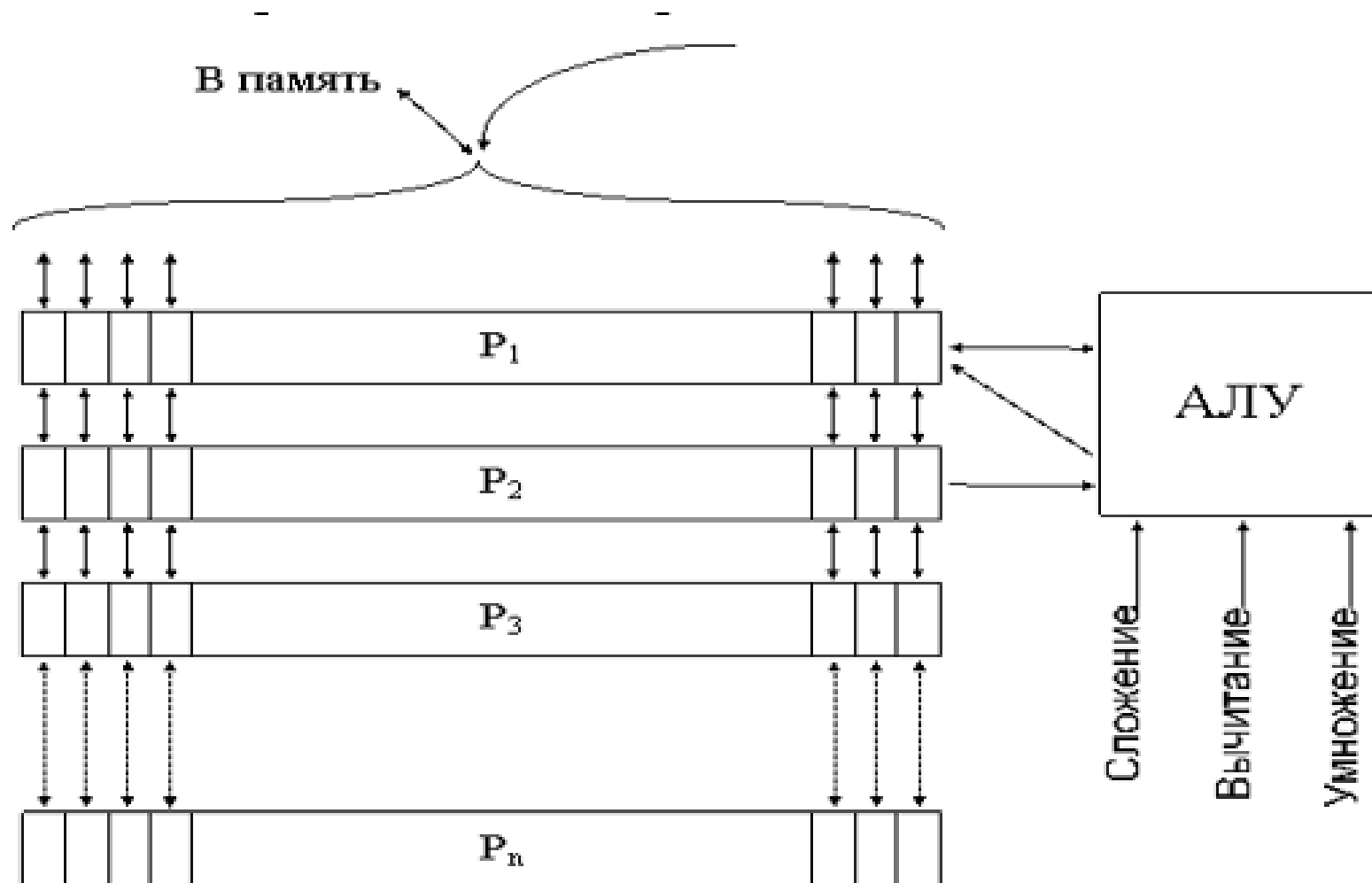
$$X = \frac{a^2 + b^2}{b + c}$$

Стектік архитектурадағы компьютерлер



Команда нөмірі	Команда	Комментарии
1	$c \rightarrow \Sigma$	Жадыдан c айнымалысын таңдау және оны Σ регистріне жүктеу
2	$(\Sigma) + b$	Жадыдан b айнымалысын таңдау Σ регистрінің құрамымен қосу операциясын орындау және нәтижені Σ регистріне сақтау
3	$(\Sigma) \rightarrow P_1$	Қосу нәтижесін P_1 – жадының жұмыс аймағына сақтау
4	$a \rightarrow \Sigma$	Жадыдан a айнымалысын таңдау және Σ регистріне жүктеу
5	$(\Sigma) \cdot a$	Көбейту операциясын нәтижесін регистрге сақтай отырып орындау
6	$(\Sigma) \rightarrow P_2$	Қосу нәтижесін P_2 – жадының жұмыс аймағына сақтау
7	$b \rightarrow \Sigma$	Жадыдан b айнымалысын таңдау және Σ регистріне жүктеу
8	$(\Sigma) \cdot b$	Көбейту операциясын нәтижесін регистрге сақтай отырып орындау
9	$(\Sigma) + (P_2)$	Және т.с.с.
10	$(\Sigma) : (\underline{P_1})$	$\Lambda = \frac{\begin{matrix} \text{ } \end{matrix}}{b+c}$

Процессордың стектік ұйымдастырылуы



Процессордың стектік ұйымдастырылуы

№	Команда	P ₁	P ₂	P ₃	P ₄
1	Вызов b	b			
2	Дублирование	b	b		
3	Вызов c	c	b	b	
4	Сложение	<u>b+c</u>	b		
5	Реверсирование	b	<u>b+c</u>		
6	Дублирование	b	b	<u>b+c</u>	
7	Умножение	b ²	<u>b+c</u>		
8	Вызов a	a	b ²	<u>b+c</u>	
9	Дублирование	a	a	b ²	<u>b+c</u>
10	Умножение	a ²	b ²	<u>b+c</u>	
11	Сложение	a ² +b ²	<u>b+c</u>		
12	Деление	$\frac{a^2+b^2}{b+c}$			

Микропроцессорлар

Біз білетіндей қазіргі ЭЕМ негізін микропроцессор құрайды— ол өңдеуші құрылғы, барлық операцияларды орындайды.

Қаізіргі кезде олардың түрі өте көп, олар, тағайындалуымен, функциональды мүмкіндіктерімен, құрылымымен, орындалуымен ерекшелінеді. Негізінен олар өңделетін ақпараттық бірліктегі разрядтылық санымен ерекшеленеді: 8-биттік, 16-биттік, 32-биттік, 64-биттік, 128-биттік және т.с.с. 8-биттік МП: i8080, i8085, Z80 (Z: МП фирмы Zilog) және басқалар.

16-биттік МП: i8086, i8088.

32-биттік МП: i80386, i80486.

64-биттік МП:

128-биттік МП:

МП негізгі сипаттамалары

Тактілік жиілік - бұл бір секундтағы электрлік импульстар саны, ЭЕМ элементтерінің ауысуының максимальды уақытын анықтайды;

Разрядтылық – біруақытта өңделетін екілік разрядтардың максимальды саны.

МП разрядтылығы $m/n/k$ деп белгіленеді, олар:

m – ішкі регистрлердің разрядтылығы, қандай да бір процессор класына жататындығын көрсетеді;

n – деректер шинасының разрядтылығы, ақпараттарды беру жылдамдығын анықтайды;

k – адрес шинасының разрядтылығы, адрестік кеңістіктің өлшемін анықтайды. Мысалы, МП i8088 келесі мәндермен сипатталады $m/n/k=16/8/20$;

Архитектура

МП архитектурасы түсінігі командалар жиыны мен адрестеу әдістерінен, уақыт бойынша команданы орындау мүмкіндіктерін біріктіруден, МП құрамында қосымшы құралдардың болуынан және олардың жұмыс режимдеріне байланысты болады. Осы себептерге байланысты **микро** және **макро архитектуралар бар**.

МП Микро архитектурасы – бұл аппараттық ұйымдастырылуы, яғни МП логикалық құрылымы, регистрлері, басқарушы сұлбалары, АЛҚ, сақтау құрылғылары, әрі оларды байланыстыратын ақпараттық магистралдар.

МП Макро архитектурасы – бұл командалар жүйесі, өңделетін деректер типі, адрестеу режимі және МП жұмыс принциптері.

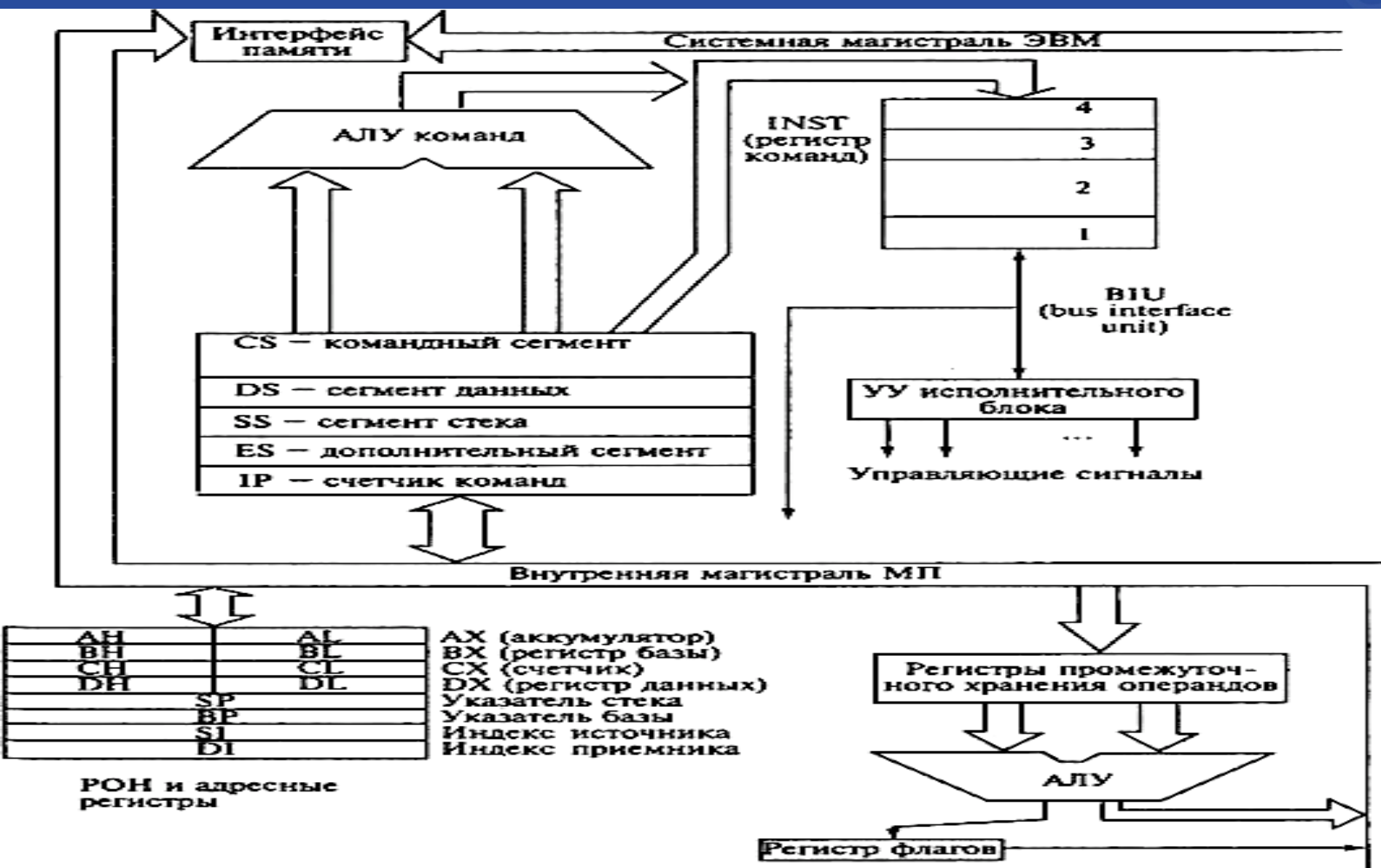
МП өнімділігіне әсер етушілер

Тактілік жиілік (ядро жиілігі) – секунд ішіндегі электрлік импульстардың саны. Әрбір импульс ақпараттардан тұрады – ол процессорға берілген команда немесе жады деректері болуы мүмкін. Тактілік жиілік аналық плата орналасқан блоктардың бірі кварцтық генератормен беріледі. Оның нақтылығы өте жоғары болады мега немесе гигагерц диапазонында беріледі. Бір герц - бір импульс, бір мегагерц - 10^6 импульс, бір гигагерц - 10^3 мегагерц. Мысалы 800 МГц тактілік жиілігінде жұмыс жасайтын МП, секундына 800 миллион жұмыс тактісін орындайды. Ядроның тактілік жиілігі көп болған сайын оның деректерді өңдеу жылдамдығы да жоғары болады. Қазіргі кезде МП 300 МГц тен *2,5 ГГц ке дейін жұмыс жасайды.*

Жүйелік шина жиілігі – ол МП басқа құралдармен байланыстырушы. МП екі жиілігі болады: ядро және жүйелік шина жиілігі. Бұл жиілік деректер алмасу жылдамдығына жауапты. Қазіргі кезде 66 МГц тен 266МГц ке дейін.

Кэш-жады көлемі – кэш-жады аз көлемдегі жылдам жады, жадыға қатынауды талап ететін операциялардың орындалуын жылдамдату үшін қолданылады. Кештің деңгейлері бар: (L1) – команда кэшi (инструкция), МП бір кристалда орналасады. (L2) – деректер кэшi – деректермен орындалатын операцияларды жылдамдатады. Жалпы өнімділік L2 өлшеміне байланысты. L2 үлкен болған сайын процессор қымбат болады, себебі кэшке арналған жады қымбат. Сол себепті кэш иілігін арттырған тиімді. Қазір 8 Кб ден 5Мб дейін кэш бар

МП базалық құрылымы



Intel МП кезеңдері

Intel МП 16-ты разрядты 8086 процессорларынан 1978 жылы маусым айында шығарылып басталды. Қазір кезеңдер әр фирмадағы процессорлардың негізінде ерекшеленеді.

Бірінші (8086 және 8088 процессорлары және 8087 математикалық сопроцессоры) барлық кезеңдерге негіз болды, 16-разрядты регистр, 1 Мб аймағындағы сегменттік адрестеу, команда және ұзу жүйелері және т.б. атрибуттары болды. Аз конвейеризацияны қолданды.

Екінші (80286 және сопроцессор 80287) «қорғалған режим» қосылды, виртуальды жадыны пайдаланды, әрбір тапсырма үшін 1Гб. Көптапсырмалы операциялық жүйелерде жұмыстар жүргізді. Тактілік жиілігі өсті, конвейеризация жақсартылды.

Intel МП кезеңдері

Үшінші (80386/80387 сыртқы шина разрядтылығын анықтайтын DX және SX «суффикстарымен» белгіленген 32-разрядты архитектура. Өлшемдердің кеңейтілуімен бірге (16 бит 0 ден 65535 немесе –32768 ден +32767 дейін болса, ал 32 бит – төрт миллиардтан астам), адрестелетін жады көлемі де артты. Бұл жүйелерде Microsoft Windows жүйесі кеңінен қолданыла бастады.

Төртінші (80486 сондай ақ DX және SX) үшінші кезеңге онша архитектураға өзгерістер енгізілмеді бірақ өнімділігін арттыру бойынша жұмыстар жүргізілді. Орындалатын конвейер жұмысы қиындатылды. Сыртқы сопроцессордан бас тартты – ол орталық процессормен бірге бір кристалға орналастырыла бастады (немесе қолданылмады).

Intel МП кезеңдері

Бесінші (Intel фирмасының Pentium процессорлары және AMD фирмасының K5) суперскалярлы архитектураны ұсынды. Конвейерлерді жадыдан командалар және деректермен жылдам қамтамасыз ету үшін бұл процессорлардың деректер шинасы 64-разрядты болды. MMX (Matrics Math Extensions – кеңейтілген матрицалық математикалық операцияларға арналған командалар жиыны) пайда болды. Дәстүрлі 32-разрядты процессорлар тек 8-разрядтық сандармен операциялар жүргізсе MMX 64 разрядпен жұмыс жасады. Олар SIMD (Single Instruction/Multiple Data - «один поток команд/много потоков данных») жұмысын жүргізді. Негізінен жаңа командалар мультимедиа программалардың жұмысын жақсарту үшін жасалды.

Intel МП кезеңдері

Алтыншы Pentium Pro, Pentium III, Celeron және Xeon (және AMD үшін K6, K6-2, K6-2+, K6-III процессорлары). Динамикалық орындауды қолдады, яғни командаларды тәртіп бойынша емес ыңғайлылық бойынша қолдану әдісі. Сонымен қатар MMX жақсартылды, AMD үшін 3dNow!, ал Intel үшін- SSE (Streaming SIMD Extensions – потоковые расширения SIMD) қосылды.

Жетінші Athlon (AMD фирмасының) процессорларымен басталды. Суперскалярлық және суперконвейерлік қосылды.

Одан кейінгі кезеңдер Әр фирманың шығарулары бойынша ерекшеленді. Мысалы: Intel қазір 12 кезең болса, AMD 9 кезең деп жатады.

Intel МП кезеңдері

Конвейерлік өңдеу әрбір команданың орындалуын бірнеше этапқа бөліп орындауды болжайды, бұл кезде әрбір этап процессор конвейерінің өзінің қадамында орындалады. Процессорда команданы орындау кезінде конвейерде біруақытта бірнеше команда орындала алады. Мысалы классикалық Pentium процессорының конвейерінде бес қадам бар. Ал суперконвейерлі архитектураның конвейерлерінің қадамдарының саны көбірек болады.

Ал скалярлы деп бір конвейерлі процессорды атайды. Оған Intel процессорларының 80486-дейінгілері кіреді. Cyrix және AMD фирмалары да осы МП жұмыс істейді.

Мысалы Cyrix М-1 және М-2 МП, ал AMD К-5 және К-6 МП жатқызуға болады, олар Pentium және Pentium Pro аналогтары болып табылады.