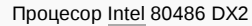


**x86** (англ. *Intel 80x86*) архітектура процесорів та ядра мікроконтролерів з однойменною системою команд. **x86** або **80x86** є загальною назвою мікропроцесорної архітектури та архітектури розробленої на її базі обчислювальної техніки. Вперше реалізована в процесорах компанії Intel. Архітектура x86 у значно вдосконаленому вигляді і нині<sup>[*колу?*]</sup> домінує на ринку настільних і мобільних комп'ютерів та малих серверів.

Крім Intel, ця архітектура також була реалізована в процесорах інших виробників: AMD, VIA, Transmeta, IDT та інші. Вдосконалений варіант цієї архітектури із розрядністю 32 біти називається IA-32 (Intel Architecture 32).

## Історія

Створені в середині 1989 і 1995 року процесори Pentium і Pentium Pro значно відрізнялися за своєю архітектурою від своїх попередників. В основу архітектури була покладена суперскалярна архітектура, яка і дала можливість отримати п'ятикратну продуктивність Pentium в порівнянні з моделлю 80486. Хоча Pentium проектувався як 32-розрядний, для зв'язку з рештою компонентів системи використовувалася зовнішня 64-розрядна шина.



<b>Розробка</b>	Intel, AMD
<b>Розрядність</b>	16, 32 і 64 біти
<b>Поява</b>	1978 (16 бітів), 1985 (32 біти), 2003 (64 біти)
<b>Тип</b>	<u>CISC</u>
<b><u>архітектури</u></b>	
<b><u>Інструкції</u></b>	змінної довжини (від 1 до 15 байтів)
<b><u>Реалізація</u></b>	умовні
<b><u>переходів</u></b>	
<b><u>Порядок</u></b>	little-endian
<b><u>байтів</u></b>	
<b>Розширення</b>	<u>x87</u> , <u>IA-32</u> , <u>x86-64</u> , <u>MMX</u> , <u>3DNow!</u> , <u>SSE</u> , <u>SSE2</u> , <u>SSE3</u> , <u>SSSE3</u> , <u>SSE4</u> , <u>SSE4.2</u> , <u>SSE5</u> , <u>AES-NI</u> , <u>CLMUL</u> , <u>RdRand</u> , <u>SHA</u> , <u>MPX</u> , <u>SGX</u> , <u>XOP</u> , <u>F16C</u> , <u>ADX</u> , <u>BMI</u> , <u>FMA</u> , <u>AVX</u> , <u>AVX2</u> , <u>AVX512</u> , <u>VT-x</u> , <u>AMD-V</u> , <u>TSX</u> , <u>ASF</u>
<b>Відкритий дизайн</b>	частково
	<b><u>Регістри</u></b>
<b><u>Загального</u></b>	
<b><u>призначення</u></b>	16 бітів: 6 частково спеціалізованих регістрів, BP і SP — спеціалізовані; 32 біти: 8 загального призначення, включно з EBP і ESP;  64 біти: 16 загального призначення, включно з RBP і RSP
<b><u>Рухома кома</u></b>	16 бітів: опційно (окремий співпроцесор); 32 біти: <u>x87</u> , опційно <u>SSE2</u> ;  64 біти: <u>x87</u> і <u>SSE2</u> , опційно <u>AVX2</u> і <u>AVX512</u>

Процесор	Розрядність шини даних	Робоча частота, МГц
i4004	4	0.75
i8008	8	0.8
i8080	8	2
i8086	16	5; 8; 10
i8088	16	5; 8
i80286	16	8; 10; 12; 16
i80386 DX	32	20; 25; 33; 40
i80386 SX	16	20; 25; 33
i80486 DX	32	25; 33; 50; 66;75;100;120
i80486 SX	32	16; 20; 25; 33
Pentium	32	60; 66; 75; 90; 100; 120; 133;
Pentium Pro	32	166; 180; 200

У таблиці нижче наведені бренди процесорів, які згруповані за поколіннями і є важливими в історії x86.

Покоління	Вперше представлений	Відомі бренди процесорів	Лінійна/фізична адреси простору
1	1978	Intel 8086, Intel 8088 та клони	Перші x86 мікропроцесори
		Intel 80186, Intel 80188 та клони, NEC V20/V30	
2	1982	Intel 80286 та клони	16-bit (30-bit віртуальні) / 24-bit (сегментовані)
3 (IA-32)	1985	Intel 80386 та клони, AMD Am386	32-bit (46-bit віртуальні) / 32-bit
4 (FPU)	1989	Intel486 та клони, AMD Am486/Am5x86	
4/5	1997	IDT/Centaur-C6, Cyrix III-Samuel, VIA C3-Samuel2 / VIA C3-Ezra (2001), VIA C7 (2005)	
5	1993	Pentium, Pentium MMX, Cyrix 5x86, Rise mP6	
5/6	1996	AMD K5, Nx586 (1994)	36-bit фізична (PAE)
6	1995	Pentium Pro, Cyrix 6x86, Cyrix MII, Cyrix III-Joshua (2000)	
	1997	AMD K6/-2/3, Pentium II/III	
	2003	Pentium M, Intel Core (2006)	
7	1999	Athlon, Athlon XP	48-bit фізична для AMD Phenom
	2000	Pentium 4	
7/8	2000	Transmeta Crusoe, Efficeon	
	2004	Pentium 4 Prescott	
	2006	Intel Core 2	
	2008	VIA Nano	
8 (x86-64)	2003	Athlon 64, Opteron	48-bit фізична для AMD Phenom
8/9	2007	AMD Phenom	
	2008	Intel Core i3/i5/i7, AMD Phenom II	
		Intel Atom	
	2011	AMD Bobcat, Llano	
9 (GPU)	2011	Intel Sandy Bridge/Ivy Bridge, AMD Bulldozer та Trinity	
	2013	Intel Larrabee	
	2014	Intel Haswell	
10 (x86-64)	20xx		
11 (x86-64)	2020	Intel Tiger Lake	

## Особливості архітектури

x86 є CISC архітектурою із змінною довжиною команд. Сумісність нових моделей процесорів із попередніми була ключовою у розвитку архітектури x86. Архітектура двічі розширювалась до більшого розміру машинного слова. Перші x86 процесори були 16-бітними. У 1985 Intel випустила 32-бітний процесор 80386, його архітектуру назвали x86-32 або IA-32 (Intel Architecture, 32 bits). У 2003, AMD випустила Athlon 64, в якому реалізовано розширення архітектури до 64 біт, яке називають AMD64 (AMD), EM64T (Intel), чи x64 (Microsoft). Нова архітектура x86-64 доповнила і поступово замінила x86 і починаючи з 2003 року стала новим індустріальним стандартом<sup>[*джерело?*]</sup>.

Дозволено невіривняний по машинному слову доступ до пам'яті. Порядок байтів у слові — little-endian (найменш значущий байт має меншу адресу). Сучасні x86 процесори перекладають набір команд x86 на RISC-подібні мікрокоманди для ефективнішого їх виконання.

## Реальний режим

Реальний режим (real mode) — класичний режим адресації, який використовувався в перших моделях. Використовує сегментну модель пам'яті, яка організована наступним чином: адресний простір в 1 MiB розбивається на 16-байтові блоки, які називаються параграфи. Всього параграфів в 1 MiB — 65536, що дозволяє пронумерувати їх 16-розрядними числами. Сегменти пам'яті мають розмір 65536 байт, і завжди починаються на межі параграфа. Адреса комірки пам'яті складається з двох частин: номер параграфа, з якого починається сегмент та зміщення всередині сегменту і зазвичай записується як SSSS:OOOO, де S та O — шістнадцяткові цифри. SSSS називається сегментною компонентою адреси, а OOOO — зміщення. Адреса комірки, яка видається на шину, являє собою сегментну компоненту помножену на 16 плюс зміщення. Сегментна компонента вміщується в спеціальний регістр, який називається сегментний, а зміщення в регістр IP(регістр інструкцій). Мікропроцесори 8086/8088, 80186/80188 та 80286 мали чотири сегментні регістри, тобто могли працювати одночасно з чотирма сегментами пам'яті.

Сегментні регістри та їх призначення:

- **CS** — Сегмент коду. Використовується для вибірки команд програми;
- **DS** — Сегмент даних. Використовується за замовчуванням для доступу до даних;
- **ES** — Додатковий сегмент. Є отримувачем даних в командах обробки рядків;
- **SS** — Сегмент стеку. Використовується для розміщення програмного стеку;
- **FS** — Додатковий сегментний регістр. Спеціального призначення не має. З'явився в процесорі 80386;
- **GS** — Аналогічно попередньому, але в нових процесорах с 64-бітною архітектурою має особливий статус: може використовуватись для швидкого переключення контекстів.

Незважаючи на те, що сегментні регістри мають спеціальні призначення, архітектура припускає при зверненні до даних змінити один сегмент на інший. Сегменти коду, стеку та отримувача рядків завжди використовують регістри CS, SS та ES та не можуть бути змінені. Загальний об'єм пам'яті, яка адресується в реальному режимі становить 1048576 байт (діапазон адрес 00000-FFFFFh, або у форматі сегмент:зміщення 0:0-F000h:FFFFh). Сегментний підхід дозволяє розділити всю пам'ять на 16 сегментів, які починаються з адрес, кратних 64 Кбайт.

За деякими адресами фізичної пам'яті можуть бути присутні пристрої вводу/виводу з відображенням на пам'ять, додаткові постійні запам'ятовувачі, «вікна» пам'яті EMS. Зазвичай кожен такий пристрій використовує один 64-кілобайтовий блок пам'яті; відповідно, адреса комірки в адресному просторі пристрою буде збігатися зі зміщенням у сегменті пам'яті комп'ютера.

Також в реальному режимі відсутній захист пам'яті та розмежування прав доступу, тому він вже практично вийшов з використання. Є режимом за замовчуванням для всіх моделей процесорів x86.

## Захищений режим

Захищений режим (protected mode) — режим з захистом пам'яті, який вперше з'явився в процесорі 80286 і надалі багаторазово вдосконалювався. Має більшу кількість підрежимів, за якими можна прослідкувати еволюцію сімейства ЦП. Аналогічно реальному режиму, тут також використовується сегментована модель пам'яті, але вже організована за іншим принципом: ділення на параграфи відсутнє, а розташування сегментів описується спеціальними структурами, які розташовані в оперативній пам'яті.

Існує два типи дескрипторних таблиць: глобальна та локальна. Глобальна таблиця описує сегменти операційної системи і розділяє структури даних. Локальна таблиця може бути визначена для кожного процесу. Дескриптор завантажується з пам'яті в кеш, який є прив'язаним до кожного сегментного регістра та автоматично завантажується в момент його модифікації.

Кожний програмний модуль, який виконується в захищеному режимі визначається його сегментом коду, описаним регістром CS, який і визначає його привілеї за доступом до даних та іншим модулям.

Процесор 80386, який з'явився в 1985 році, на відмінну від своїх попередників став 32-бітним. В ньому з'явилась можливість адресуватись до 4 ГіБ пам'яті, що дозволило створювати сегменти пам'яті розміром з весь адресний простір. Тому нові операційні системи використовували вироджену модель організації пам'яті, коли всі сегменти починаються з нуля. Така модель отримала назву плоскої, і адреса задається одним цілим 32-розрядним числом.

## Змішані режими

Сегментний блок керування пам'яттю сучасних процесорів, незважаючи на кардинальну різницю двох його основних режимів, в обох працює схожим способом.

Це дозволяє організувати нестандартні режими не описані в офіційній документації, але іноді корисні при написанні програм. Оскільки відомо, що внутрішні кеші дескрипторів використовуються у всіх режимах, і саме вони використовуються для адресації пам'яті, при розумінні логіки їх роботи можливе завантаження в них нестандартних значень для даного режиму. Схожі нестандартні режими отримали назву Unreal mode та активно використовуються BIOS'ами персональних комп'ютерів. Важливо помітити, що в процесорі 80286 також була можливість завантаження нестандартних значень дескриптивного кешу за допомогою недокументованої команди LOADALL, що було особливо актуально, оскільки процесор 80286 не дозволяв скидати прапорець PE (із захищеного режиму виходили за допомогою повного перезавантаження (англ. reset) процесора, яке було надзвичайно повільним і серйозно позначалося на продуктивності).

## Сторінкова організація пам'яті

У процесорах починаючи з 80386 з'явився модуль керування пам'яттю, яке дозволяє організувати відображення сторінок пам'яті, що було ще одним приводом переходу на плоску модель з появою 32-розрядних обчислень. Використовуючи трансляцію сторінок операційна система може створювати власне лінійний адресний простір для кожного процесу. Слід зазначити, що сторінковий блок управління пам'яттю

доступний тільки в захищеному режимі.

Регістри процесора

**IP** (англ. *Instruction Pointer*) — регістр, який вказує на зміщення (адреса) інструкцій в сегменті коду (1234:0100h сегмент/зміщення).

IP — 16-бітний (молодша частина EIP)

EIP — 32- бітний аналог (молодша частина RIP)

RIP — 64-бітний аналог

**Сегментні регістри** — регістри, що вказують на сегменти.

CS (англ. *Code Segment*), DS (англ. *Data Segment*), SS (англ. *Stack Segment*), ES (англ. *Extra Segment*), FS, GS

В реальному режимі роботи процесора сегментні регістри містять адресу початку 64Kb сегмента, зміщений управо на 4 біта. В захищеному режимі роботи процесора сегментні регістри містять селектор сегмента пам'яті, виділеного ОС.

CS — показчик на кодовий сегмент. Зв'язка CS: IP (CS: EIP / CS: RIP — в захищеному / 64-бітному режимі) вказує на адресу в пам'яті наступної команди.

**Регістри даних** — служать для зберігання проміжних обчислень.

RAX, RCX, RDX, RBX, RSP, RBP, RSI, RDI, R8 — R15 — 64-бітні

EAX, ECX, EDX, EBX, ESP, EBP, ESI, EDI, R8D — R15D — 32-бітні (extended AX)

AX (англ. *Accumulator*), CX (англ. *Count Register*), DX (англ. *Data Register*), BX (англ. *Base Register*), SP (англ. *Stack Pointer*), BP (англ. *Base Pointer*), SI (англ. *Source Index*), DI (англ. *Destination Index*), R8W — R15W — 16-бітні

AH, AL, CH, CL, DH, DL, BH, BL, SPL, BPL, SIL, DIL, R8B — R15B — 8-бітні (половинки 16-бітних регістрів)

наприклад, AH — high AX — старша половина 8 біт

AL — low AX — молодша половина 8 біт

RAX		RCX		RDX		RBX	
	EAX		ECX		EDX		EBX
	AX		CX		DX		BX
	AH   AL		CH   CL		DH   DL		BH   BL

RSP		RBP		RSI		RDI		Rx	
	ESP		EBP		ESI		EDI		RxD
	SP		BP		SI		DI		RxW
	SPL		BPL		SIL		DIL		RxB

де x — 8..15.

**Регістр прапорів** FLAGS (16 біт) / EFLAGS (32 біта) / RFLAGS (64 біта) — містить поточний стан процесора.

**Системні регістри** GDTR, LDTR і IDTR введені в процесорах починаючи з Intel286 і призначені для зберігання базових адрес *таблиць дескрипторів* — найважливіших складових системної архітектури при роботі в захищеному режимі.

Регістр GDTR містить 32-бітний (24-бітний для Intel 286) базовий адресу і 16-бітний межа *глобальної таблиці дескрипторів (GDT)*.

Видима частина регістра LDTR містить тільки селектор дескриптора *локальної таблиці дескрипторів (LDT)*. Сам дескриптор LDT автоматично завантажується в приховану частину LDTR з глобальної таблиці дескрипторів.

Регістр IDTR містить 32-бітний (24-бітний для Intel286) базовий адресу і 16-бітний межа *таблиці дескрипторів переривань (IDT)*. У реальному режимі може бути використаний для зміни місця розташування *таблиці векторів переривань*.

Видима частина регістра TR містить селектор дескриптора сегмента стану задачі (TSS). Сам дескриптор TSS автоматично завантажується в приховану частину TR з глобальної таблиці дескрипторів.

Зсувні регістри представляють собою послідовно з'єднану ланцюжок тригерів. Основний режим роботи — зрушення розрядів коду від одного тригера до іншого на кожен імпульс тактового сигналу. За призначенням регістри розрізняються на:

- Акумулятор — використовується для зберігання проміжних результатів арифметичних і логічних операцій та інструкцій введення-виведення;
- Флагової — зберігають ознаки результатів арифметичних і логічних операцій;
- Загального призначення — зберігають операнди арифметичних і логічних виразів, індекси й адреси;
- Індексні — зберігають індекси вихідних і цільових елементів масиву;
- Вказівні — зберігають показники на спеціальні області пам'яті (показчик поточної операції, показчик бази, показчик стека);
- Сегментні — зберігають адреси та селектори сегментів пам'яті;

- Керуючі — зберігають інформацію, керуючу станом процесора, а також адреси системних таблиць.

### Лічильник команд

**IP** (англ. *Instruction Pointer*) — регістр, що містить адресу-зміщення наступної команди, що підлягає виконанню, щодо кодового сегмента CS.

Регістр IP пов'язаний з CS у вигляді CS: IP, де CS є поточним кодовою сегментом, а IP — поточним зсувом щодо цього сегмента.

Регістр IP є 16-розрядним регістром-показчиком. Крім нього, до складу регістрів цього типу входять SP (англ. *Stack Pointer* — показчик стека) і BP (англ. *Base Pointer* — базовий показчик).

### Принцип роботи

Наприклад, CS містить значення 2CB5 [ 0 ]H, в регістрі IP зберігається зсув 123H.

Адреса наступної інструкції, що підлягає виконанню, обчислюється шляхом підсумовування адреси в CS (сегменті коду) зі зміщенням в регістрі IP:

$$2CB50H + 123H = 2CC73H$$

Таким чином, адреса наступної інструкції для виконання дорівнює 2CC73H .

При виконанні поточної інструкції процесор автоматично змінює значення в регістрі IP, в результаті чого реєстрова пара CS: IP завжди вказує на наступну інструкцію, що підлягає виконанню.

### Див. також

- [Список мікропроцесорів Intel](#)
- [Комп'ютерна платформа](#)

### Джерела

### Посилання

1. AMD64 Technology: AMD64 Architecture Programmer's Manual Volume 3: General-Purpose and System Instructions ([https://web.archive.org/web/20030802032632/http://www.amd.com/us-en/assets/content\\_type/white\\_papers\\_and\\_tech\\_docs/24594.pdf](https://web.archive.org/web/20030802032632/http://www.amd.com/us-en/assets/content_type/white_papers_and_tech_docs/24594.pdf))
2. Інші документи на сайті AMD щодо AMD64-архітектури (<https://web.archive.org/web/20080725164618/http://developer.amd.com/documentation/guides/Pages/default.aspx#manuals>)
3. Instruction tables. Lists of instruction latencies, throughputs and micro-operation breakdowns for Intel and AMD CPU's. Agner Fog, Copenhagen University College of Engineering ([http://www.agner.org/optimize/instruction\\_tables.pdf](http://www.agner.org/optimize/instruction_tables.pdf)) [Архівовано ([https://web.archive.org/web/20140730030310/http://www.agner.org/optimize/instruction\\_tables.pdf](https://web.archive.org/web/20140730030310/http://www.agner.org/optimize/instruction_tables.pdf)) 30 липня 2014 у Wayback Machine.]

Отримано з <https://uk.wikipedia.org/w/index.php?title=X86&oldid=38827820>

Цю сторінку востаннє відредаговано о 04:08, 12 березня 2023.

Текст доступний на умовах ліцензії Creative Commons Attribution-ShareAlike; також можуть діяти додаткові умови. Детальніше див. Умови використання.