

**Лекція № 8 з навчальної дисципліни
“Архітектура обчислювальних систем”.
Розділ 2. Програмування низького рівня.**

Тема лекції:

**Архітектура і програмна модель мікропроцесорів
IA-32, x86-64.**

План лекції

1. Еволюція МП x86. Архітектура IA-32.
2. Режими роботи та програмна модель МП IA-32.
3. Способи (режими) адресації операндів МП IA-32.
4. Архітектура і програмна модель мікропроцесорів x86-64

1. Еволюція МП x86. Архітектура IA-32

Покоління процесорів	Характеристика	Співпроцесори	Арх.
1. МП <i>18086</i> , 1979 р.	Реальний режим, 1 Мбайт адресованої пам'яті, сегментна адресація (фіксований розмір сегмента – 64 Кбайт), 12 т/оп.	Зовнішні <i>FPU</i> та співпроцесор введення-виведення	<i>IA-16</i>
2. МП <i>180286</i> , 1982 р.	Поява захищеного режиму та віртуальної пам'яті, 4,5 т/оп.		
3. МП <i>1386</i> , 1985 р.	Безпосередня адресація 4 Гбайт пам'яті (розмір сегмента -- до 4 Гбайт), сторінкова організація пам'яті, зовнішня кеш-пам'ять, 4,5 т/оп.		
4. МП <i>1486</i> , 1989 р.	Скалярний конвеєрний МП, убудована кеш-пам'ять, множення частоти FSB, 2 т/оп.	Інтегрований <i>FPU</i> , мости (хаби)	<i>IA-32</i>
5. МП <i>Pentium (i586)</i> , 1993 р.	Суперскалярність та суперконвеєрність, передбачення розгалужень, 1 т/оп, конвеєрний <i>FPU</i>		
6. МП <i>Pentium Pro, II – IV (i686)</i> , 1995 р.	Динамічне виконання та виконання за припущенням, перейменування регістрів, RISC – ядро, дворівневий кеш, гіперконвеєрність, Hyper-threading, розширення адреси до 36 біт, 0,5 т/оп		
7. 64-розрядні x86 – сумісні (<i>Athlon-64...</i>)	Зв'язки з 3-х команд, набір ФВП різних типів та масштабованість архітектури, предикація.		

2. Режими роботи та програмна модель МП ІА-32

МП архітектури ІА-32 можуть функціонувати в одному з двох основних режимів: режимі реальної адресації (real address mode) і захищеному режимі віртуальної адресації (protected virtual address mode). Більш короткі назви – реальний режим і захищений режим.

Реальний режим є цілком сумісним з режимом роботи мікропроцесора І8086 (однозадачний, можлива адресація 1 Мбайт фізичної пам'яті, розмір сегмента – 64 Кбайт).

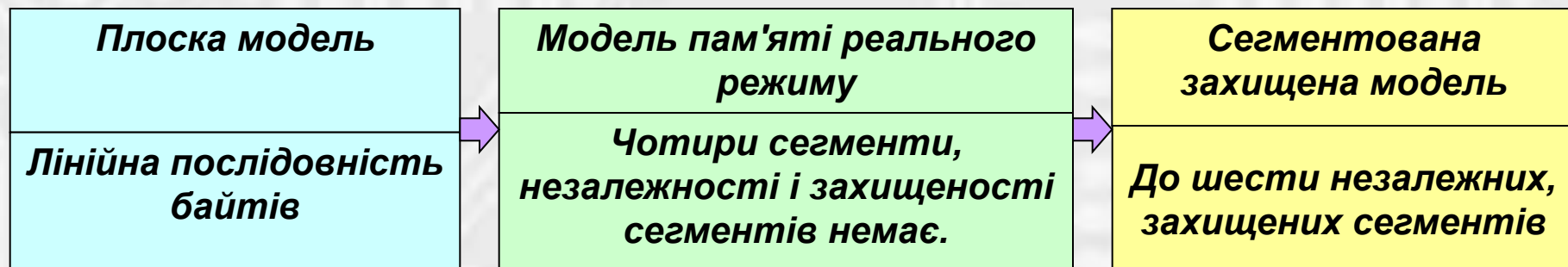
Захищений режим є багатозадачним, в цьому режимі МП може виконувати кілька задач одночасно, використовуючи механізми розподілу ресурсів обчислювальної системи між задачами, що виконуються. *Можлива адресація до 4 (64) Гбайт фізичної пам'яті, граничний розмір сегмента також складає 4 Гбайт. У цьому режимі підтримується віртуальна пам'ять обсягом до 64 Тбайт.* Вбудований блок керування пам'яттю підтримує механізми сегментації і сторінкової трансляції адрес. Тобто, через 4 Гбайт адресованої фізичної пам'яті при використанні сторінкової адресації можуть відобразитися до 64 Тбайт віртуальної пам'яті для кожної задачі.

2. Режими роботи та програмна модель МП ІА-32

МП ІА-32 дозволяють реалізувати різні моделі пам'яті. Найпростіша – **плоска модель пам'яті: уся пам'ять представляється єдиною лінійною послідовністю байт** (класична реалізація нейманівської архітектури). Щоб одержати плоску модель пам'яті досить зробити так, щоб усі сегментні реєстри вказували на ту саму область пам'яті.

Протилежністю плоскої моделі є **сегментована захищена модель - основна модель пам'яті, яка використовується в захищеному режимі. Кожній програмі в будь-який момент часу надаються кодовий сегмент, сегмент стека і до чотирьох сегментів даних.** Сегменти вибираються з таблиць, підготовлених операційною системою. Некоректні звернення до пам'яті блокуються системою захисту ОС.

Проміжне положення між названими моделями займає **модель пам'яті реального режиму. Тут також пам'ять організується у вигляді сегментів, але незалежності і захищеності сегментів немає.** Використання цієї моделі необхідно для забезпечення можливості адресації 1 МБайт пам'яті за допомогою шістнадцятирозрядних реєстрів. Таку модель дотепер використовують додатки для ОС реального режиму типу MS DOS.



Співвідношення моделей пам'яті

2. Режими роботи та програмна модель МП ІА-32

Програмна модель мікропроцесора ІА-32 містить 32 регістри, доступних для використання програмістом. Дані регістри можна розділити на дві великі групи:

- 16 користувальницьких регістрів;
- 16 системних регістрів.

Кількість програмно доступних регістрів у МП ІА-32, за винятком сегментних регістрів, така ж, як і в І8086, але розрядність збільшена до 32 (за винятком сегментних регістрів, які залишилися 16 - розрядними, що і відображено в їх позначеннях - вони мають приставку **e** (Extended)).

Користувальницькі регістри:

●РЗП (8 шт.):

- арифметичні регістри **eax/ax/ah/al**, **ebx/bx/bh/bl**, **ecx/cx/ch/cl**, **edx/dx/dh/dl** – призначення аналогічне арифметичним регістрам МП І8086;

eax					
			ax		
			ah	al	
31	16	15	8	7	0

Структура регістра eax

- вказівні та індексні регістри **esi/si**, **edi/di**, **esp/sp**, **ebp/bp** - призначення аналогічне вказівним та індексним регістрам МП І8086.

2. Режими роботи та програмна модель МП ІА-32

● Сегментні реєстри (6 шт.):

- **cs, ss, ds, es** – призначення аналогічне сегментним реєстрам МП І8086,
- **gs, fs** – реєстри додаткових сегментів даних. На відміну від основного сегмента даних, адреса якого утримується в сегментному реєстрі **ds**, при використанні додаткових сегментів даних їхньої адреси потрібно вказувати явно за допомогою спеціальних префіксів перевизначення сегментів у команді.

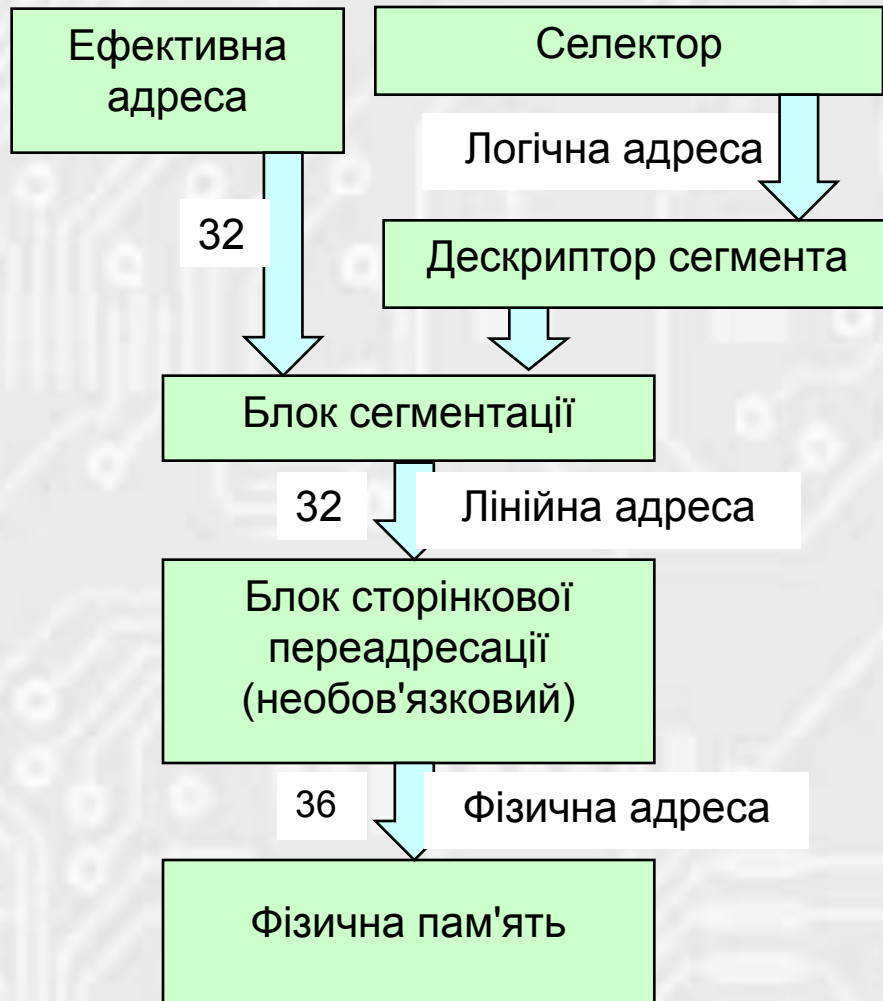
● Реєстри стану та управління (2 шт.):

- **eflags/flags, eip/ip** – призначення аналогічне відповідним реєстрам МП І8086.

Системні реєстри:

- чотири реєстри управління **cr0, cr1, cr2, cr3**. Ці реєстри призначені для загального управління системою;
- чотири реєстри системних адрес **gdtr, ldtr, idtr, tr** (реєстри управління пам'яттю). Вони призначені для захисту програм і даних у мультизадачному режимі роботи мікропроцесора.
- вісім реєстрів налагодження **dr0, dr1, dr2, dr3, dr4, dr5, dr6, dr7**. Ця група реєстрів призначена для апаратного налагодження. Апаратно мікропроцесор містить вісім реєстрів налагодження, але реально з них використовуються тільки 6. Засоби апаратного налагодження вперше з'явилися в мікропроцесорі І486. Більшість із системних реєстрів програмно доступні.

2. Режими роботи та програмна модель МП ІА-32



Формування фізичної адреси у захищеному режимі

Логічна (віртуальна) адреса, складається із селектора сегмента і ефективної адреси (зсуву).

Оскільки кожна задача може мати до 16 Кбайт селекторів (2^{14}), а зсув, обмежений розміром сегмента, може досягати 4 Гбайт, логічний адресний простір для кожної задачі може досягати 64 Тбайт.

Селектор сегмента зберігається в старших 14 бітах сегментного регістра (CS, DS, ES, SS, FS, або GS), що беруть участь в адресації конкретного елемента пам'яті. За значенням селектора зі спеціальних таблиць дескрипторів сегментів, що зберігаються в пам'яті, витягається початкова адреса сегмента.

2. Режими роботи та програмна модель МП ІА-32

Процесор може звертатися тільки до тих сегментів пам'яті, для яких є дескриптори в таблицях. **Дескриптори являють собою 8-байтні структури даних, що визначають положення сегмента у пам'яті, розмір займаної їм області пам'яті (ліміт), його призначення і характеристики захисту.**

Блок сегментації транслює логічний адресний простір у 32-бітний простір лінійних адрес. Лінійна адреса утворюється додаванням базової адреси сегмента з ефективною адресою. **У захищеному режимі базова адреса завантажується з дескриптора, що зберігається в таблиці, по селектору, завантаженому у використований сегментний регістр.**

32-бітна фізична адреса пам'яті утворюється після перетворення лінійної адреси блоком сторінкової переадресації. Вона виводиться на зовнішню шину адреси процесора. У найпростішому випадку (при відключеному блоці сторінкової переадресації) фізична адреса збігається з лінійною. **Включений блок сторінкової переадресації здійснює трансляцію лінійної адреси у фізичний сторінками розміром 4 Кбайт, 2 або 4 Мбайт.** Блок забезпечує розширення розрядності фізичної адреси процесорів шостого покоління до 36 біт. **Блок переадресації може включатися тільки в захищеному режимі.**

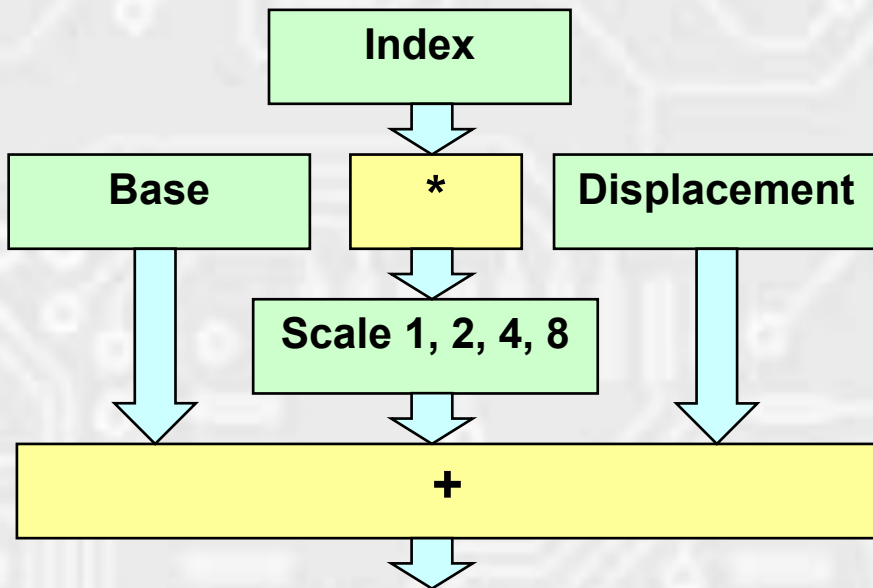
Захист пам'яті за допомогою сегментації не дозволяє: використовувати сегменти не за призначенням; порушувати права доступу; адресуватися до елементів, що виходять за ліміт сегмента; змінювати вміст таблиць дескрипторів (тобто параметри сегментів), не маючи достатніх привілеїв.

3. Способи (режими) адресації операндів МП ІА-32

Система команд МП ІА-32 передбачає 11 режимів адресації. Безпосередня адресація та пряма регістрова адресація не звертаються до пам'яті для завантаження операндів.

У дев'яти режимах виконується звернення до пам'яті, ефективна адреса ЕА обчислюється за узагальненим алгоритмом.

$$EA = Base + Index * Scale + Disp$$



- **Зсув** (Displacement, Disp) - 8, 16 або 32-бітове число, включене у команду.

- **База** (Base) - вміст базового регістра. Звичайно використовується для вказівки на початок деякого масиву.

- **Індекс** (Index) - вміст індексного регістра. Звичайно використовується для вибору елемента масиву.

- **Масштаб** (Scale) - множник (1, 2, 4 або 8), зазначений у коді інструкції. Для вказівки розміру елемента масиву, доступний тільки при 32-бітій адресації.

3. Способи (режими) адресації операндів МП ІА-32

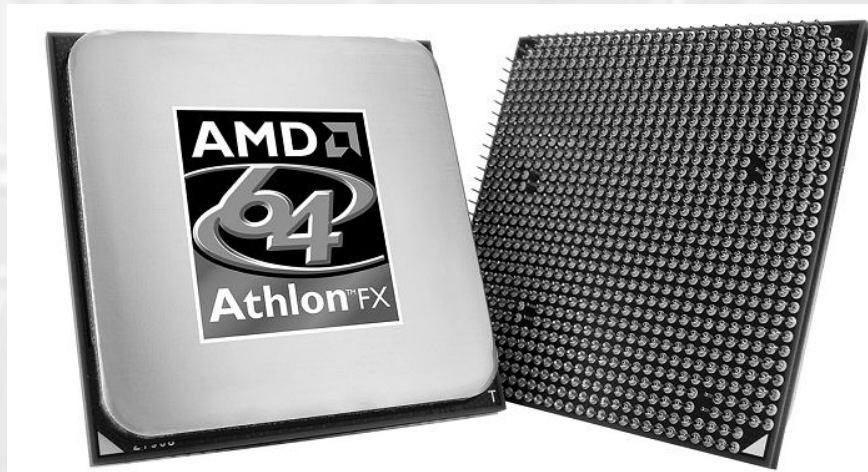
<i>Режим</i>	<i>Адреса</i>
Пряма адресація	$EA = Disp$
Непряма регістрова адресація	$EA = Base$
Базова адресація	$EA = Base + Disp$
Індексна адресація	$EA = Index + Disp$
Масштабована індексна адресація	$EA = Index * Scale + Disp$
Базово-індексна адресація	$EA = Base + Index$
Масштабована базово-індексна адресація	$EA = Base + Index * Scale$
Базово-індексна адресація зі зсувом	$EA = Base + Index + Disp$
Масштабована базово-індексна адресація зі зсувом	$EA = Base + Index * Scale + Disp$

4. Архітектура мікропроцесорів x86-64

Архітектура x86-64 була запропонована фірмою AMD та вперше реалізована в МП Opteron та Athlon 64 (Hummer). Вона є сумісною з архітектурою IA-32.

Особливості Hummer:

- 64-розрядна віртуальна адресація, “плоский” адресний простір;
- 64-розрядний лічильник команд; можливість адресації відносно нього;
- підвищена точність передбачення переходів (за рахунок збільшення масиву адрес переходів та таблиці глобальної історії переходів);
- інтегрований контролер оперативної пам'яті.
- застосування **HiperTransport** (технологія організації високопродуктивних дуплексних з'єднань типу “точка-точка”, інтеграція в МП основних функцій північного мосту).
- збільшена довжина обчислювального конвеєра.



4. Архітектура мікропроцесорів x86-64

Режими функціонування МП архітектури x86-64

Operating Mode		Operating System Required	Application Recompile Required	Defaults		Register Extensions	Typical
				Address Size (bits)	Operand Size (bits)		GPR Width (bits)
Long Mode	64-Bit Mode	New 64-bit OS	yes	64	32	yes	64
	Compatibility Mode		no	32	16	no	32
				16			16
Legacy Mode	Protected Mode	Legacy 32-bit OS	no	32	32	no	32
				16	16		
	Virtual-8086 Mode			16	16		16
	Real Mode			Legacy 16-bit OS	16		16

64-розрядна адресація застосовується у розширеному режимі (Long Mode). Режим задається встановленням керуючого біта LMA (Long Mode Active). В ньому регістри сегментів ES, DS, FS, GS, SS ігноруються. Біти CS уточнюють режим роботи МП. Розширений режим має два «підрежими»: 64-розрядний режим і режим сумісності. В останньому забезпечується сумісність з 16-ти і 32-розрядними режимами x86. Вибором підрежиму керує біт CS.L. Якщо він скинутий, 64-розрядна ОС може виконувати 16-ти і 32-розрядні x86-доданки. За вибір розміру операнда відповідає біт CS.D. За умовчанням в 64-розрядному режимі (CS.L = 1, CS.D = 0) адреси 64-розрядні, операнди – 32 розрядні.

4. Архітектура мікропроцесорів x86-64

□ *Long Mode*

Основний режим МП AMD64. Для використання цього режиму необхідна 64-бітова ОС. Режим дозволяє виконувати 64-бітові програми; також надається підтримка виконання 32-бітового кода, хоча 32-бітові програми не можуть використовувати 64-бітові системні бібліотеки та навпаки. Тому більшість 64-розрядних ОС надають два набори необхідних системних файлів.

В цьому режимі відсутні: підрежим віртуального МП 8086, сегментована модель пам'яті (але залишилася можливість використання сегментів FS и GS, що може бути корисним для швидкого знаходження важливих даних при переключенні завдань), апаратна багатозадачність, а також декілька команд, в тому числі і для оброблення двійково-десяткових чисел.

Режим активується встановленням прапора CR0.PG, який вмикає сторінковий блок управління пам'яттю MMU (англ. *memory management unit*) за умови, що таке переключення дозволено. Таким чином, виконання 64-бітового кода із забороненим сторінковим перетворенням неможливе.

□ *Legacy Mode*

Дозволяє МП AMD64 виконувати інструкції процесорів x86 та надає повну сумісність із 32-бітовим кодом та ОС. Додаткові функції, що надаються архітектурою AMD64 (наприклад, додаткові регістри) недоступні. В цьому режимі 64-бітові програми та ОС не функціонують.

4. Архітектура мікропроцесорів x86-64

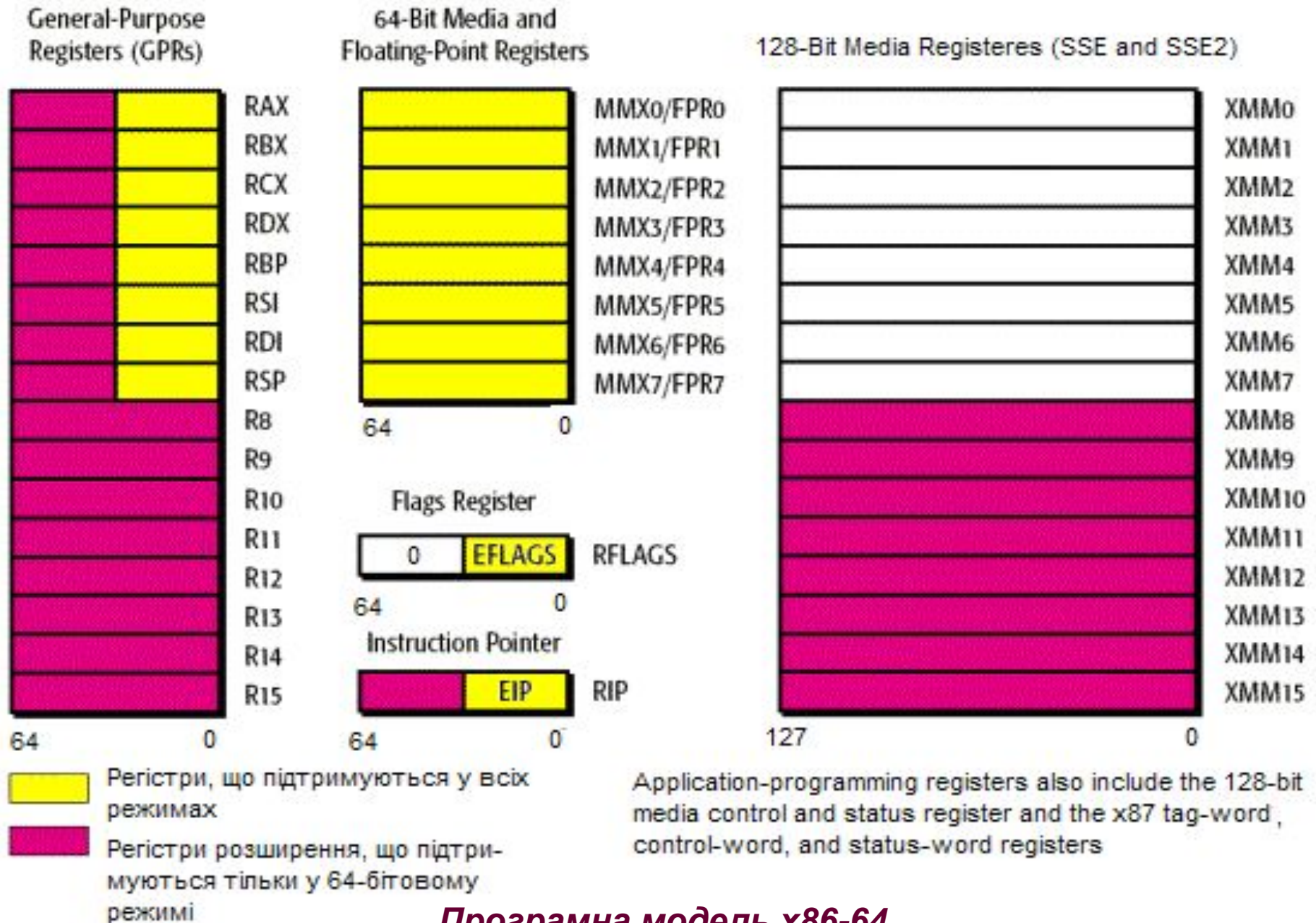
Програмна модель x86-64

Набір інструкцій x86-64 (AMD64) -- розширення архітектури Intel IA-32 (x86-32). Головна відмінна риса AMD64 – підтримка 64-бітових РЗП, 64-бітових арифметичних та логічних операцій над цілими числами та 64-бітових віртуальних адрес. Для адресації нових регістрів в формати команд введені так звані «префікси розширення регістра», для яких був вибраний діапазон кодів 40h-4Fh.

Програмна модель x86-64 має:

- 16 цілочисельних 64-бітових регістрів загального призначення (RAX, RBX, RCX, RDX, RBP, RSI, RDI, RSP, R8 -- R15);
- 8 80-бітових регістрів з рухомою комою (ST0 -- ST7);
- 8 64-бітових регістрів Multimedia Extensions (MM0 -- MM7, мають спільний адресний простір із регістрами ST0 -- ST7);
- 16 128-бітових регістрів SSE (XMM0 — XMM15);
- 64-бітові покажчик RIP та регістр прапорів RFLAGS.

4. Архітектура мікропроцесорів x86-64



Програмна модель x86-64

4. Архітектура мікропроцесорів x86-64

РЗП (16 шт, 64-розрядні):

- арифметичні регістри **rax/eax/ax/ah/al**, **rbx/ebx/bx/bh/bl**, **rcx/ecx/cx/ch/cl**, **rdx/edx/dx/dh/dl**; вказівні та індексні регістри **rsi/esi/si**, **rdi/edi/di**, **rsp/esp/sp**, **rbp/ebp/bp** - призначення аналогічне відповідним регістрам МП І8086.

RAX							
		eax					
		ax					
		ah			al		
63	32	31	16	15	8	7	0

**Структура
регістра rax**

- регістри **R8 – R15**. Для звернення до молодших 8, 16, 32 бітів цих регістрів можна застосовувати суфікси b, w, d відповідно.

R9							
		R9d					
		R9w					
					R9b		
63	32	31	16	15	8	7	0

**Структура
регістра R9**