42.12.2 Дати характеристику системи переривань захищеного режиму
Особливістю системи переривань в захищеному режимі є широке використання **внутрішніх апаратних переривань** - виключень. Виключення виникають як результат такого стану комп'ютерної системи, який не дозволяє виконати поточну команду. В свою чергу неможливість виконання чергової команди в багатьох випадках пов'язана з порушенням правил захисту.

За способом реакції процесора (і наступних можливостей програми обробки переривань), переривання в захищеному режимі поділяють на пастки, помилки, збої та аварії.

У випадку **пастки** в стек записується адреса команди. Тобто команда повністю виконується (якщо це команда між сегментної передачі управління, то змінюється вміст регістрів CS та EIP), а потім вміст CS та EIP записується в стек. Переривання від зовнішніх пристроїв та програмні переривання (команди INT) відносяться до пасток.

У випадку **помилки** в стек заноситься адреса команди, яка визвала виключення. Це дозволяє програмі обробки переривання внести необхідні зміни стану комп'ютерної системи і пере запустити виконання команди.

Але при виключеннях переривається виконання команди. Якщо виключення викликано командою, яка змінює вміст регістрів CS та EIP, то адреса повернення, яка записується в стек, може бути не визначеною. Крім того, виключення захисту можуть виникати при виконанні механізму переривань від зовнішніх пристроїв, коли команда-порушниця взагалі відсутня. Тому при помилках останнім в стек заноситься код помилки, який дозволяє ідентифікувати та локалізувати причину помилки в подібних ситуаціях.

У випадку **збою** повторний пере запуск команди-порушниці не можливий. Виключення типу збій переважно виникають при порушеннях під час переключення задач та реакції на привілеї, що супроводжується іншими порушеннями правил захисту.

У випадку **аварії** подальше виконання програми є неможливим і процесор переходить на процедуру початкового запуску (RESET, процесор пере запускається). Найбільш ймовірною причиною аварії є неможливість виконання стекових операцій (порушення правил захисту для стекових операцій) при реакції на інші виключення.

Для організації обслуговування переривань в захищеному режимі як і в реальному режимі використовується таблиця векторів переривань. На відміну від реального режиму вектором таблиці є не 4-х байтна логічна адреса процедури обробки переривань, а один із спеціальних 8-байтних **системних дескрипторів**, до яких відноситься шлюз переривання, шлюз пастки та шлюз задачі.

Шлюз переривання має наступну структуру

Байт 7	Байт 6	Байт 5									Байт 4	
Зміщення (адреса в сегменті) процедури обробки переривань Розряди 31-16		**P**	**DPL**	0	1	1	1	0	0	0	0	резерв
Селектор сегмента кодів, де знаходиться процедура обробки переривання		Зміщення (адреса в сегменті) процедури обробки переривань Розряди 15 - 0										
Байт 3	Байт 2	Байт 1									Байт 0	

Шлюз переривання містить дані необхідні для визначення фізичної адреси процедури обробки переривань - селектор відповідного дескриптора сегмента кодів, та зміщення процедури обробки переривань в цьому сегменті.

Шлюз пастки має наступну структуру

Байт 7	Байт 6	Байт 5	Байт 4

Зміщення (адреса в сегменті) процедури обробки переривань Розряди 31-16	**P**	**DPL**	0	1	1	1	1	0	0	0	резерв
Селектор сегмента кодів, де знаходиться процедура обробки переривання	Зміщення (адреса в сегменті) процедури обробки переривань Розряди 15 - 0										

Байт 3	Байт 2	Байт 1	Байт 0

Шлюз пастки відрізняється від шлюзу переривання лише значенням 1 в 0-му розряді 5-того байта дескриптора (молодші чотири біти 5-го байта **системного дескриптора** визначають його тип).

При використанні шлюзу переривання біт IF регістра ознак (EFLAGS) апаратурою встановлюється в 0 і попереднє значення відновлюється під час виконання команди IRET. При використанні шлюзу пастки біт IF не змінюється. В іншому шлюзи пастки та переривань не відрізняються.

Шлюз задачі використовується коли в результаті переривання необхідно переключить процесор на іншу задачу. В даній роботі переключення задач не розглядається.

На відміну від практики використання реального режиму в захищеному режимі таблиця векторів переривань може розміщуватись по довільній фізичній адресі, яка завантажується командою LIDT в старші 32 розряди 48-розрядного системного регістра IDTR. В молодші 16 розрядів цього регістру завантажується розмір таблиці в байтах. Команда LIDT належить до привілейованих команд і може виконуватись лише при **CPL**=0, або в реальному режимі.

При переключенні на процедуру обробки переривань значення поля **DPL** дескриптора сегмента кодів, де розташована ця процедура, не повинен перевищувати значення поточного рівня привілеїв ($\mathbf{DPL}_{\text{сегменту}} \le \mathbf{CPL}$). Оскільки переривання та виключення можуть виникнути де завгодно, то процедури обробки переривань розміщують на 0-вому рівні привілеїв, тобто в системній області. При цьому шлюзи розглядаються як дані, тобто $\mathbf{DPL}_{\text{шлюзу}} \ge \mathbf{CPL}$.

В лабораторній роботі зовнішні переривання при роботі в захищеному режимі забороняються, а обробляються лише внутрішні та програмні переривання з використанням шлюзу пастки.

В процесорах 80x86 та Pentium жорстко (на апаратному рівні) закріплено наступний розподіл векторів переривань між причинами виникнення виключень:

0-вектор. Ділення на 0. Помилка.

1-вектор. Використовується в налагоджувачах. Може бути як пасткою так і помилкою.

2-вектор. Зовнішнє переривання, яке ігнорує стан біту IF. Пастка.

3-вектор. Використовується в налагоджувачах. Пастка.

4-вектор. Контроль переповнення (виникає, коли виконується команда INTO і OF=1). Помилка.

5-вектор. Контроль переповнення масивів. Генерується командою BOUND, якщо операнд має значення за встановленою межею. Помилка

6-вектор. Неіснуючий код операції. В більшості випадків виникає в результаті помилок в передачах управління (передача управління не на перший байт команди). Коди 0d6h та 0f1h зарезервовані ф.Intel, хоча команди з такими кодами операцій поки що відсутні, але виключення при їх обробці не виникає. Помилка.

7-вектор. Співпроцесор відсутній. Виключення виникає при спробі виконати любу команду співпроцесора, а біт EM (біт 2) (емуляція співпроцесора) регістра CR0 має значення 1. Цим самим процедура обробки переривання 7 може бути використана для програмного

моделювання роботи співпроцесора. Крім того, виключення може виникнуть після переключення задач, щоб установить, при необхідності, режим роботи співпроцесора для нової задачі. Помилка.

8-вектор. **Збій.** (Подвійна помилка). Виникає, коли в результаті виконання команди необхідно одночасно формувати декілька виключень.

Якщо спроба перейти на процедуру обробки 8-го переривання також пов'язана з помилкою, то найбільш ймовірно процесор виконає процедуру RESTART.

9-вектор. В процесорах 80x86 та Pentium виключення по цьому вектору не генеруються.

10-вектор. Неправильний сегмент стану задачі (TSS). В багатозадачних системах для кожної задачі формується сегмент TSS, який містить основні системні дані для задачі. Помилка.

11-вектор. Сегмент відсутній. Виключення виникає при завантаженні сегментного регістра селектором дескриптора, біт **P** атрибутів якого має значення 0. Помилка.

12-вектор. Помилка стека. Виникає при спробі звернення до стека командами MOV та їй подібними (використання регістра BP або префікса заміни сегмента SS:) при умові виходу зміщення в сегменті за межі допустимого діапазону. При виконанні команд PUSH та CALL та виходу замежі сегмента стека найбільш ймовірно виникнення аварійної ситуації з наступним виконанням процедури RESTART. Помилка.

13-вектор. Порушення правил захисту. Виникає при всіх порушеннях правил захисту, які не охоплені попередніми виключеннями.

14-вектор. Відсутня сторінка. Генерується при віртуальній сторінковій адресації, коли відповідна сторінка віртуальної пам'яті відсутня в оперативній пам'яті. Помилка

15 - вектор. В процесорах 80x86 та Pentium виключення по цьому вектору не генеруються

16 - вектор. Помилка співпроцесора. Співпроцесор має свій значний набір виключень. Якщо розряд NE (5-тий) регістра CR0 містить 1, то це виключення центрального процесора виникає при появі команди співпроцесора, наступної за тією командою, яка сформувала виключення співпроцесора.