Организация прерываний в ЭВМ

Прерывания (англ. Interrupt) – это событие, вызывающее прекращение работы основной программы и переход к выполнению процедуры или подпрограммы, предназначенной для его обработки.

Система прерываний – совокупность аппаратных и программных средств, обслуживающих прерываний.

Виды прерываний:

- **аппаратные прерываний**, которые инициируются внешними устройствами с помощью запрос на прерывания.
- **программные прерывания**, которые инициируются специальными командами (например, int21h).
- **особые случаи или исключения**, вызванные возникновением особых условий (случаев) при выполнении текущей команды.

Назначение прерываний.

- для обмена информацией между процессором и внешними устройствами.
- для устранения негативных последствий в негативных ситуациях. Например: при понижении напряжения питания.
- для обработки особых случаев.
- для реализации мультизадачности.
- для профилактики, ремонта, тестирования и отладки системы.
- для вызова программ операционной системы.

Система прерываний ЭВМ обеспечивает выполнение следующих функций:

- Обнаружение изменения состояния внешней среды (запрос на прерывание)
- Идентификация источника прерываний
- Разрешение конфликтных ситуаций в случае возникновения нескольких запросов (приоритет запросов)
- Определение возможности прерывания текущей программы
- Фиксация состояния прерываемой текущей программы
- Переход к программе, соответствующей обслуживаемому прерыванию
- Возврат к прерванной программе после окончания работы прерывающей программы

Анализ состояния внешней среды

-Программный опрос регистров состояния внешних устройств
-Формирование аппаратного сигнала запроса

Анализ запросов в большинстве процессоров осуществляется на границах машинного цикла.

Способы идентификации источников прерывания.

Радиальная система организации запросов.

Каждое внешнее устройство имеет линию, по которой передается сигнал к отдельному входу процессора. В этом случае проблемы идентификации источника не существует.

Если число источников прерывания превышает число входов процессора, то для идентификации источника используют 2 способа:

- программный опрос каждого внешнего устройства (поллинг)
- аппаратный способ, при котором внешнее устройство отправляет в процессор двоичный n-разрядный код – вектор прерывания.

Прерывания процессора 8086.

Используется векторная система прерываний.

В регистре Flags используется бит IF (Interrupt flag), который разрешает или запрещает прерывание программы.

Используется вход немаскируемых прерываний NMI.

Используется контроллер приоритетов прерываний i8259A, позволяющий передавать запросы от восьми периферийных устройств.

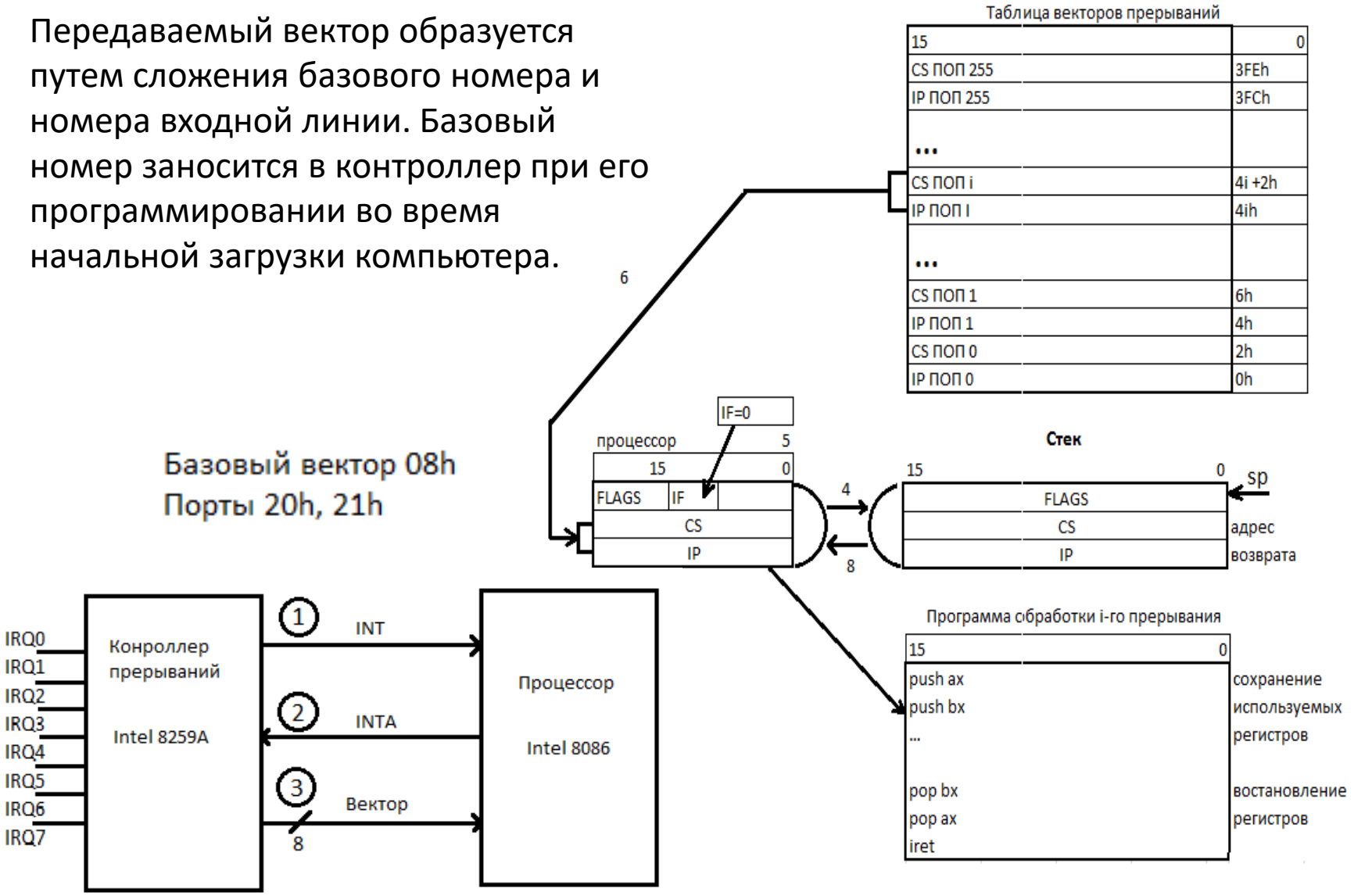
Если флаг IF=1 на запрос INT, формируемый контроллером, процессор формирует сигнал подтверждения прерывания INT A, после чего контроллер передает в процессор 8-битный код – вектор прерывания.

Вектор прерывания представляет собой индекс в таблице векторов прерываний (ТВП) указателя на обработчик прерывания.

ТВП размещается с нулевого адреса линейного адресного пространства и занимает 1 Кбайт = 256 векторов * 4 байта. (2 байта сегмент. 2 байта смещение.

Обработка прерывания в процессоре 8086.

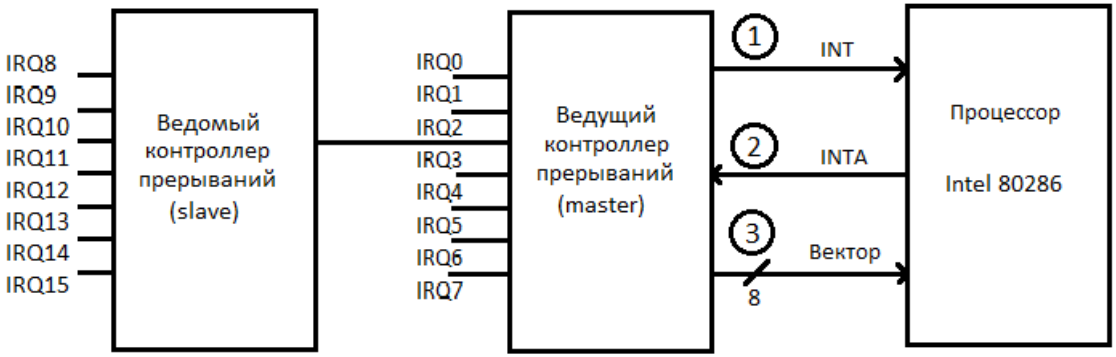
Передаваемый вектор образуется путем сложения базового номера и номера входной линии. Базовый номер заносится в контроллер при его программировании во время начальной загрузки компьютера.



Действия процессора 8086 при обработке прерываний.

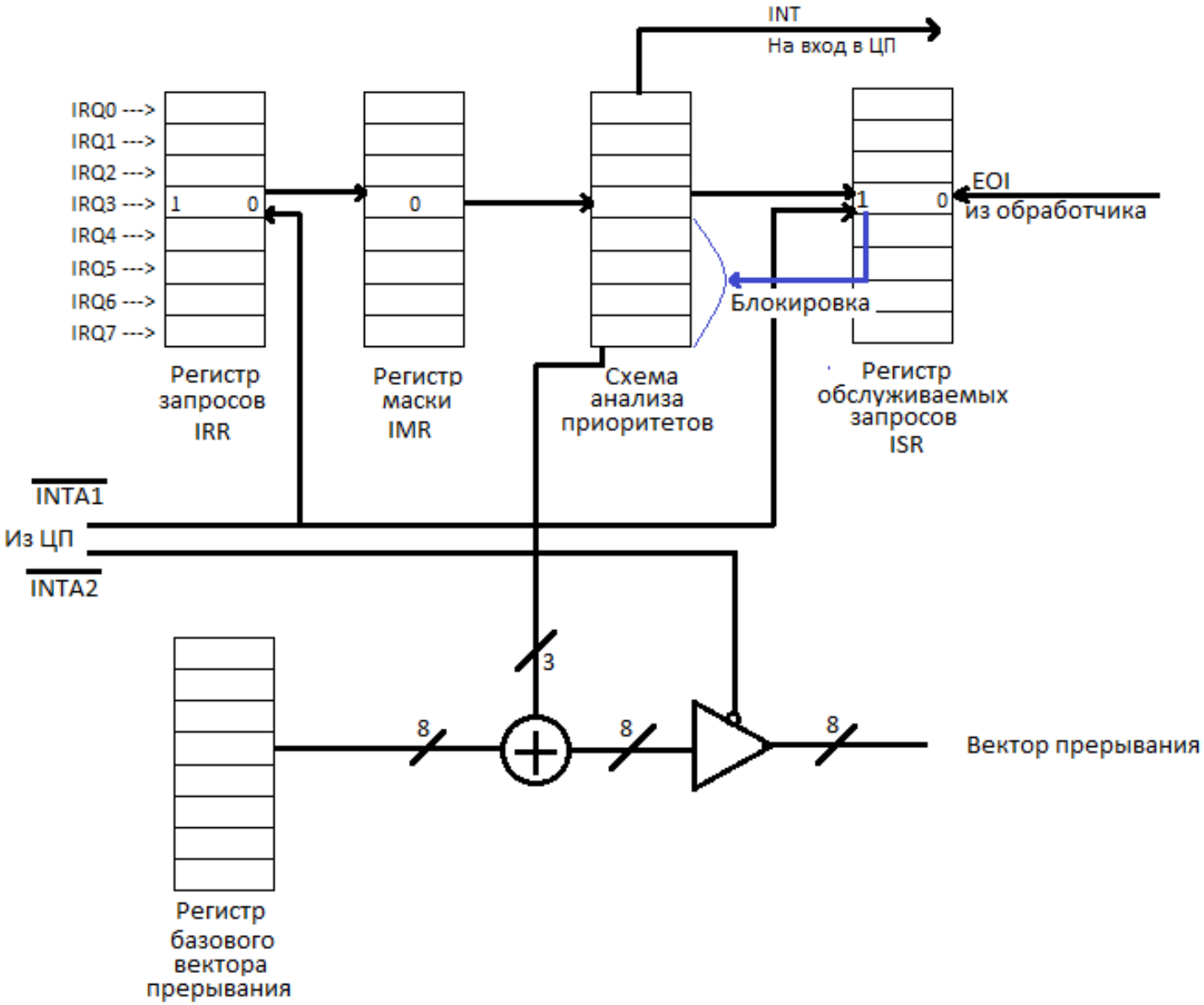
1. Анализ сигнала INT на границе команд.
2. Если INT присутствует и флаг IF=1 формирование сигнала INTA подтверждающего прерывание.
3. Ввод вектора прерывания i в цикле подтверждения прерывания.
4. Сохранение в стеке регистров FLAGS,CS и IP.
5. Сброс флага IF для запрета вложенных прерываний.
6. Извлечение из ТВП логического адреса программы обработки i-го прерывания.
7. Выполнение программы обработки i-го прерывания.
8. По команде iret, которой завершается обработчик прерывания , из стека извлекаются регистры FLAGS, CS и IP.
9. Продолжение прерванной программы со следующей команды.

Обработка внешних аппаратных прерываний для процессоров, начиная с 80286.



Запрос прерывания	Вектор прерывания	Устройство	Приоритет
IRQ0	08h	Таймер	Высший
IRQ1	09h	Клавиатура	Убывание приоритетов
IRQ2	Ah	Вход от ведомого	
IRQ8	70h	КМОП микросхема	
IRQ9	71h	резерв	
IRQ10	72h	резерв	
IRQ11	73h	резерв	
IRQ12	74h	Мышь(PS/2)	
IRQ13	75h	Исключения сопроцессора	
IRQ14	76h	жесткий диск	
IRQ15	77h	Резерв	
IRQ3	0Bh	COM2	
IRQ4	0Ch	COM1	
IRQ5	0Dh	LPT2	
IRQ6	0Eh	Гибкий диск	
IRQ7	0Fh	LPT1	Низший

Структура контроллера приоритетных прерываний.



Последовательность действий контроллера приоритетных прерываний.

Сигнал запроса прерываний (например, IRQ3), поступает в регистр запросов IRR и фиксируется в нем.

Если соответствующий разряд регистра маски IMR равен 0, сигнал проходит на схему анализа приоритетов.

В зависимости от содержимого регистра ISR (предположим, что он пуст) сигнал проходит на вход ISR и схема анализа приоритетов дает разрешение на установку соответствующего бита ISR (но не устанавливает его). Одновременно сигнал поступает на вход INT процессора. Если флаг IF=1 (прерывания разрешены), процессор формирует сигнал подтверждения прерывания ($\overline{\text{INTA1}}$). По этому сигналу в контроллере устанавливается бит регистра ISR и сбрасывается бит в регистре IRR. Запрос переводится в разряд обслуживаемых.

По сигналу ($\overline{\text{INTA2}}$) контроллер выставляет на шину данных номер прерывания.

Установленный разряд ISR блокирует формирование запросов от текущего и нижележащих уровней. Для снятия блокировки используется команда EOI (end of interrupt), которая предшествует команде iret (interrupt return).

По команде EOI сбрасывается младший из установленных разрядов ISR.

Процедура инициализации контроллеров.

Состоит из ряда команд инициализации (СКИ), посылаемых в строгой последовательности друг за другом.

СКИ2 посылается для ведущего контроллера в порт 20h, а для ведомого – в порт A0h.

Формат СКИ1

7	6	5	4	3	2	1	0
0	0	0	1	0	0		1

1- один контроллер (XT), не будет СКИ3
0 - два контроллера (AT), будет СКИ3
Идетификатор СКИ1

СКИ2 посылается для ведущего контроллера в порт 21h, а для ведомого – в порт A1h. Задает базовый вектор. В стандартной конфигурации базовый вектор ведущего – 8h, ведомого - 70h.

СКИ3 для ведущего содержит установленные биты, подключенные к ведомым контроллерам. Для PC AT СКИ3=4 и посылается в порт 21h. СКИ3 для ведомого контроллера содержит номер входа ведущего к которому подключен ведомый. Для PC AT СКИ3=2.

Формат СКИ4

7	6	5	4	3	2	1	0
0	0	0	0	0	0		

1 - 80x86
0 - 8080

1 - сброс бита ISR осуществляется автоматически
0 - требуется программая генерация сигнала EOI

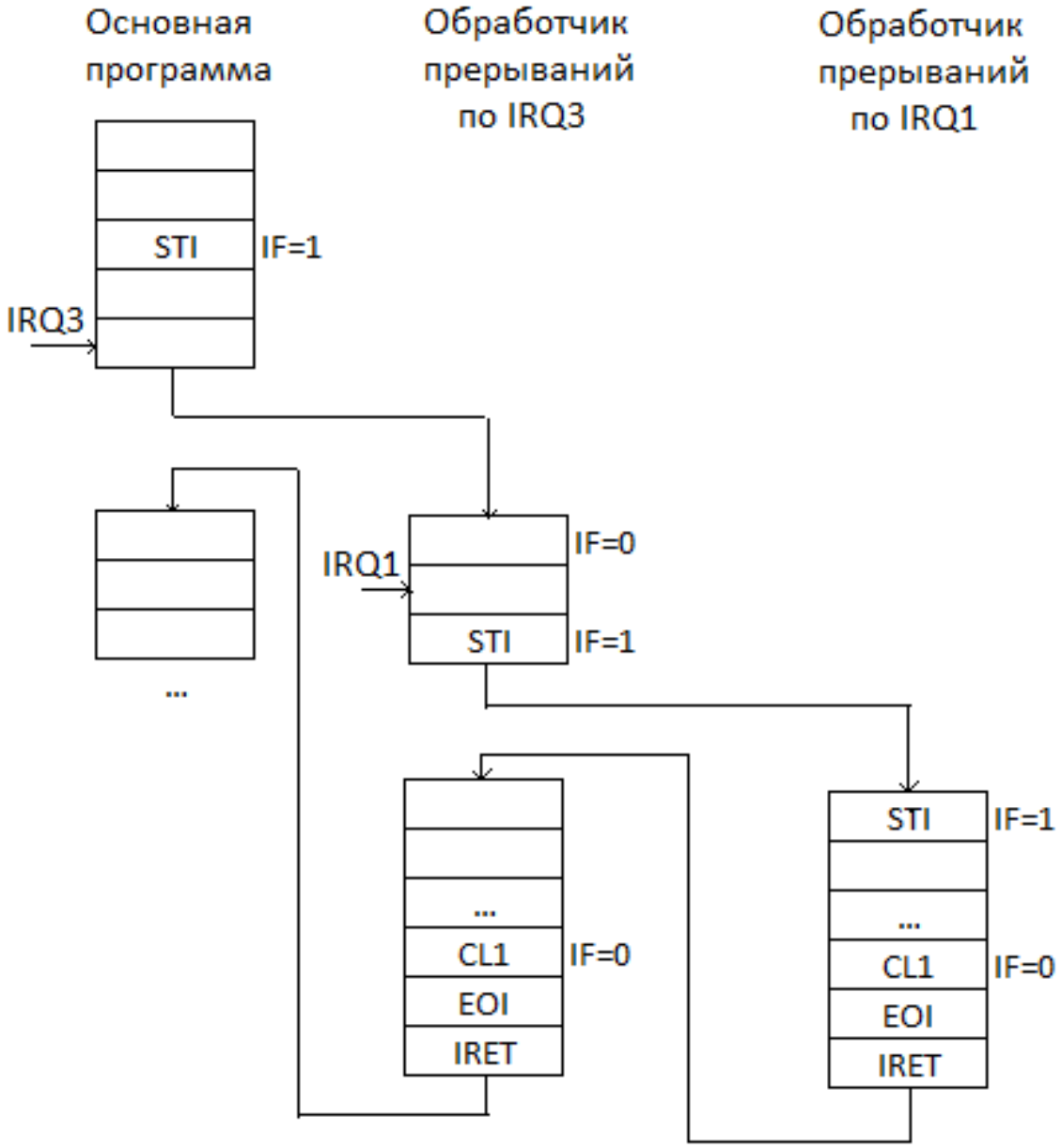
Инициализация ведущего контроллера.

```
mov dx,20h      ;первый порт контроллера
mov al,11h      ;СКИ1 означает что будет СКИ3
out dx,al       ;СКИ1 в порт
jmp $+2         ;задержка
inc dx          ;второй порт контроллера
mov al,bl       ;СКИ2 базовый адрес в bl
out dx,al       ;СКИ2 в порт
jmp $+2         ;задержка
mov al,4        ;СКИ3 ведомый подключен к уровню 2
out dx,al       ;СКИ3 в порт
jmp $+2         ;задержка
mov al,11h      ;СКИ4 80x86, требуется EOI
out dx,al       ;СКИ4 в порт
```

Инициализация ведомого контроллера.

```
mov dx,20h      ;первый порт контроллера
mov al,11h      ;СКИ1 означает что будет СКИ3
out dx,al       ;СКИ1 в порт
jmp $+2         ;задержка
inc dx          ;второй порт контроллера
mov al,bl       ;СКИ2 базовый адрес в bl
out dx,al       ;СКИ2 в порт
jmp $+2         ;задержка
mov al,2        ;СКИ3 ведомый подключен к уровню 2
out dx,al       ;СКИ3 в порт
jmp $+2         ;задержка
mov al,1        ;СКИ4 80x86, требуется EOI
out dx,al       ;СКИ4 в порт
```

Вложенные аппаратные прерывания.

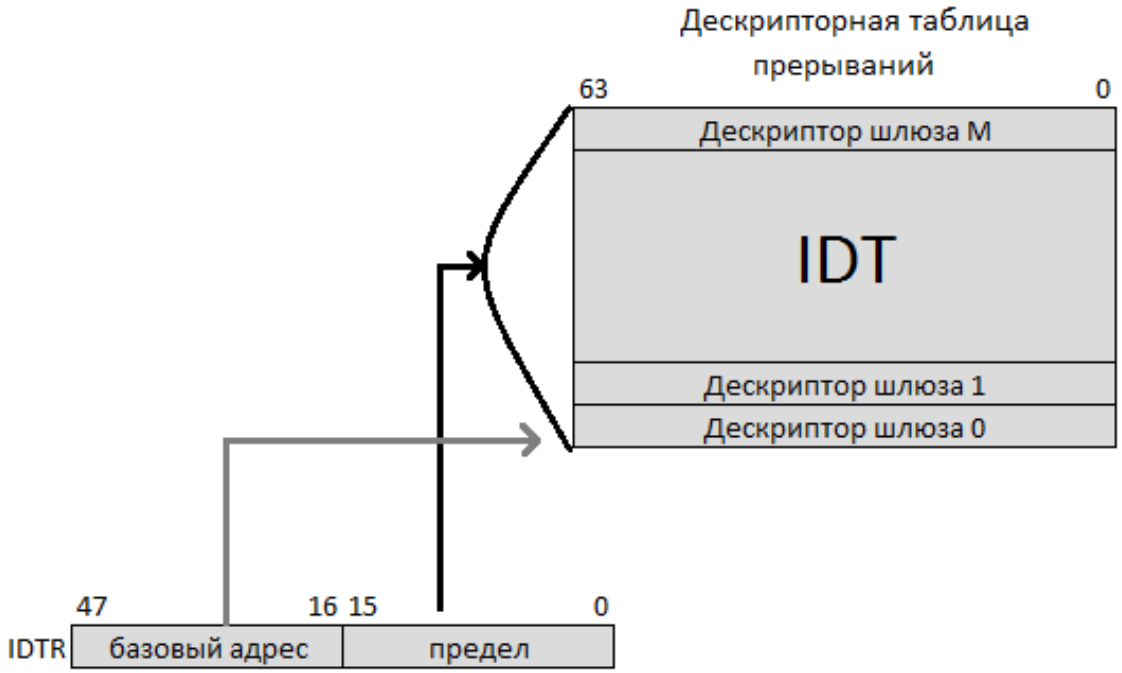


Особенности обработки прерываний в защищенном режиме работы процессора.

1. Вместо таблицы векторов прерываний используется дескрипторная таблица прерываний.
2. Большее количество особых случаев (в процессоре Intel 486 из 32 зарезервированных особых случаев используется 16).
3. Более сложный процесс перехода к обработчику прерываний.
4. Передача обработчику дополнительной информации о причине возникновения особого случая.

IDTR (Interrupt Descriptor Table Register).
Определяет местоположения в ОП дескрипторной таблицы прерываний.

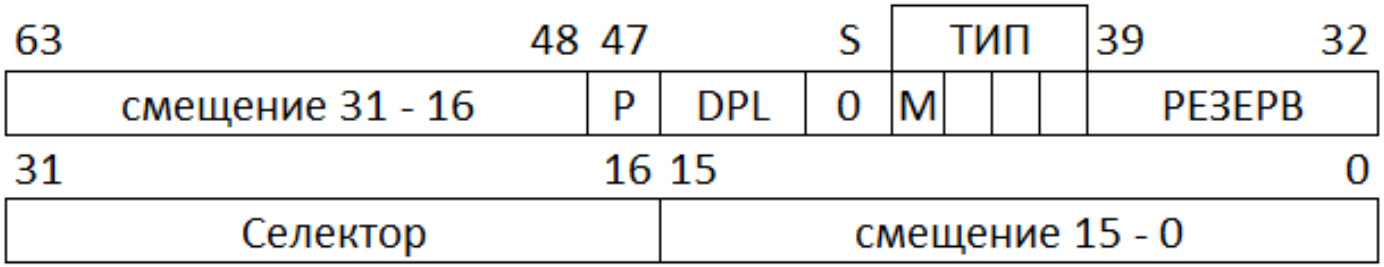
Таблица IDT содержит дескрипторы шлюзов, необходимые для выполнения процедур обработки прерываний (исключений).



В дескрипторной таблице прерываний IDT могут находится дескрипторы трех типов:

- шлюз прерывания;
- шлюз ловушки;
- шлюз задачи.

Формат шлюзов прерывания и ловушки.

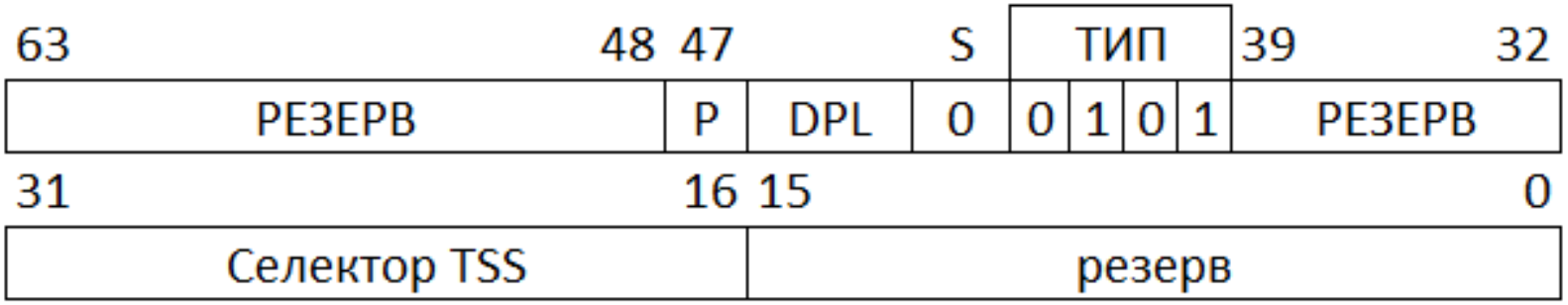


M – определяет разрядность процессора.
M = 0 – процессор 286 (16-разрядный).
M = 1 процессор 386 (32-разрядный).

ТИПЫ:
0110B = 06h – шлюз прерываний процессора 286.
0111B = 07h – шлюз ловушки процессора 286.
1110B = 0Eh – шлюз прерываний процессора 386.
1111B = 0Fh – шлюз ловушки процессора 386.

Шлюз прерываний отличается от шлюза ловушки тем, что при входе в обработчик через шлюз прерываний сбрасывается флаг IF и запрещаются вложенные аппаратные прерывания.

Формат шлюза задачи.



Тип = 0101B = 05h – определяет шлюз задачи.

В случае перехода к обработчику через шлюз задачи происходит смена контекста задачи. В TSS старой задачи сохраняется содержимое регистров процессора. В регистры процессора записывается контекст новой задачи из TSS, на который указывает селектор TSS.

По команде iret, которой завершается обработчик, происходит обратное переключение задачи.

Преимущества обработки прерываний через шлю задачи:

- Автоматически сохраняется весь контекст прерванной задачи;
- Обработчик не может исказить прерванную задачу, т.к. он полностью изолирован от неё.
- Обработчик прерываний может работать на любом уровне привилегий и в заведомо правильной среде; он может иметь своё локальное адресное пространство благодаря наличию отдельной LDT.

Недостатки обработки прерываний через шлюз задачи:

- Увеличивается время переключения на обработчик;
- В шлюзе задачи невозможно определить начальную точку выполнения задачи;
- Сложность получения информации о прерванной задаче.

Классификация особых случаев.

Особый случай или исключение (exception) – невозможность процессора по тем или иным причинам выполнить текущую команду. Особые случаи бывают трех видов:

Нарушение (fault) – такой особый случай, который обнаруживается процессором до возникновения фактической ошибки. После устранения обработчиком причины должен быть выполнен рестарт текущей команды.

Ловушка (trap) – такой особый случай, который обнаруживается процессором после выполнения виновной команды. То есть команда сама инициирует переход к обработчику и послеего завершения выполнение прерванной программы продолжается со следующей команды.

Авария (abort) – такой особый случай, при котором нельзя установить его причину и выполнить рестарт программы.

Особые случаи процессора 8086:

0 – деление на ноль;
1 – пошаговая работа (если TF = 1);
2 – немаскируемое прерывание NMI;
3 – контрольная точка, формируется командой INT3 (код 0cch);
4 – прерывание при переполнении. Возникает, если при выполнении INT0 флаг OF-1.

Особые случаи защищенного режима процессоров INTEL.

Номер	Причина	ТИП	Код ошибки
0	Ошибка деления	нарушение	нет
1	Отладка	нарушение/ловушка	нет
2	Немаскируемое прерывание	ловушка	нет
3	Контрольная точка INT3	ловушка	нет
4	Переполнение	ловушка	нет
5	Нарушение границы массива	нарушение	нет
6	Недействительный код операции	нарушение	нет
7	Устройство FPU недопустимо	нарушение	нет
8	Двойное нарушение	авария	нет
9	Не используется	нарушение	
10	Недействительный TSS	нарушение	да
11	Неприсутствие сегмента	нарушение	да
12	Нарушение стека	нарушение	да
13	Нарушение общей защиты	нарушение/ловушка	да
14	Страничное нарушение	нарушение	да
15	Зарезервирован		
16	Ошибка операции с плавающей точкой	нарушение	нет
17	Контроль выравнивания	нарушение	да
18 - 31	Зарезервирован		

Описание особых случаев.

Ошибка деления (0) – автоматически формируется, когда в команде DIV или IDIV делитель равен нулю или частное слишком велико для получателя (AL, AX, EAX).

Отладка (1) – формируется в следующих случаях:

- Нарушение к.т. по адресу команд;
- Нарушение к.т. по адресу данных;
- Ловушка покомандной работы (TF = 1)
- Ловушка контрольной точки по переключению задачи (бит T=1 в сегменте TSS).

Немаскируемое прерывание NMI (2) – единственное внешнее радиальное прерывание.

Контрольная точка (3) – формируется при выполнении однобайтной команды INT3 (код 0cch).

Переполнение (4) – возникает при выполнении команды INT0 при условии, что OF=1.

Нарушение границы массива (5) – возникает при выполнении команды BOUND, если контрольная проверка дает отрицательный результат, т.е. проверяемый (первый) операнд не попадает в диапазон значений, определенных вторым и третьим операндами команды.

Недействительный код операции (6) – генерируется, когда операционное устройство процессора обнаруживает неверный код операции.

Устройство недоступно(7) – возникает в двух случаях:

- Процессор выполняет команду ESC и бит EM=1 в CR0.
- Процессор выполняет команду wait или ESC и бит TS=1 в CR1.

Двойное нарушение (8) – когда процессор обнаруживает особый случай при попытке вызвать обработчик предыдущего особого случая, два особых случая обрабатываются последовательно. Если процессор не может их обработать последовательно, формируется особый случай.

Недействительный сегмент TSS (10) – возникает при попытке переключения на задачу с неверным TSS.

Неприсутствие сегмента (11) – формируется, когда бит P=0 в дескрипторе сегмента.

Нарушение стека (12) – возникает в двух ситуациях:

- Нарушение предела сегмента при обращении к регистру SS (POP, PUSH, ENTER, MOV AX, [BP+6]).
- При попытке загрузить в регистр SS дескриптор, который отмечен как неприсутствующий.

Нарушение общей защиты (13) – все нарушения защиты, которые не служат причиной конкретного особого случая:

- Превышение предела сегмента (кроме стека);
- Передача управления сегменту, который не является выполняемым;
- Запись в защищенный от записи сегмент;
- Считывание из выполняемого сегмента;
- Загрузка в SS селектора сегмента, защищенного от записи;
- Загрузка в SS, DS, ES, FS, GS селектора системного сегмента;
- Загрузка в SS, DS, ES, FS, GS селектора выполняемого сегмента;
- Переключение на занятую задачу;
- Нарушение правила привилегий и др.

Страничное нарушение (14) – возникает, когда разрешено страничное преобразование и имеет место одна из ситуаций:

- В элементе PDE или PTE сброшен бит P;
- Процедура не имеет достаточного уровня привилегий для доступа к адресуемой странице.

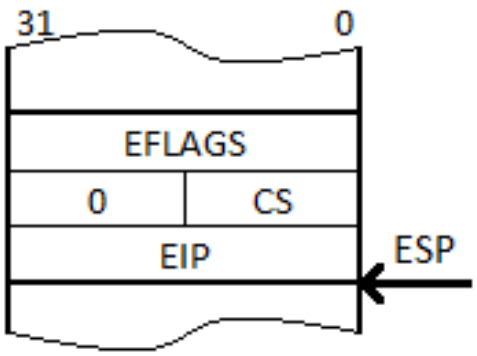
Ошибка операции с плавающей точкой (16) – сигнализирует об ошибке, возникшей при выполнении команды FPU.

Контроль выравнивания (17) – возникает при нарушении выравнивания операндов. Для реализации контроля выравнивания должны выполняться условия:

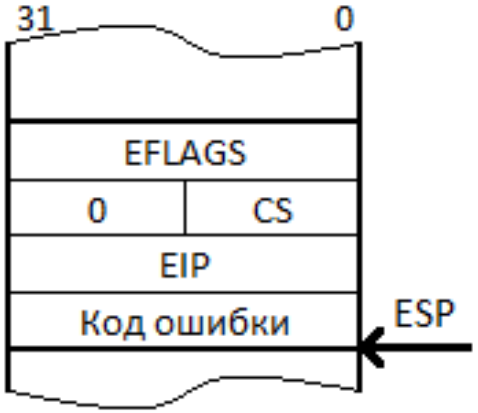
- AM=1 в CR3;
- AC=1 в EFLAGS;
- PL=3 для выполняемой программы.

Передаваемая через стек информация обработчику особых случаев.

Без смены привилегий.

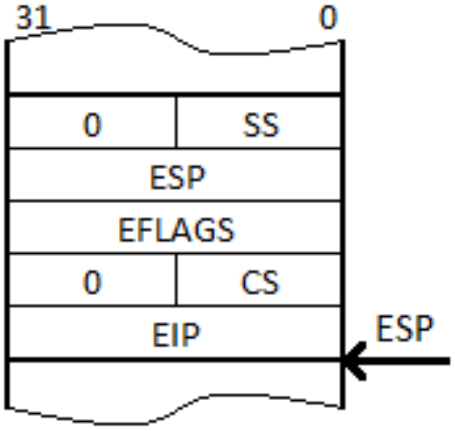


Без кода ошибки

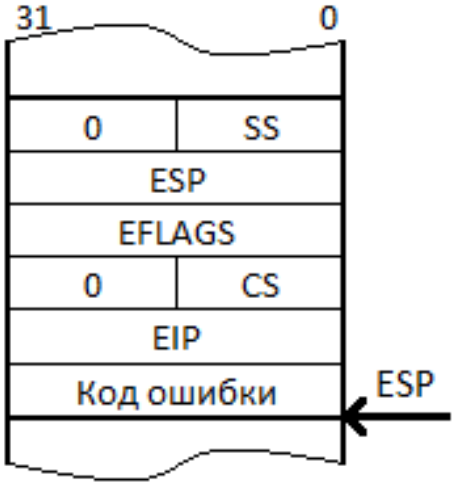


С кодом ошибки

Со сменой привилегий.

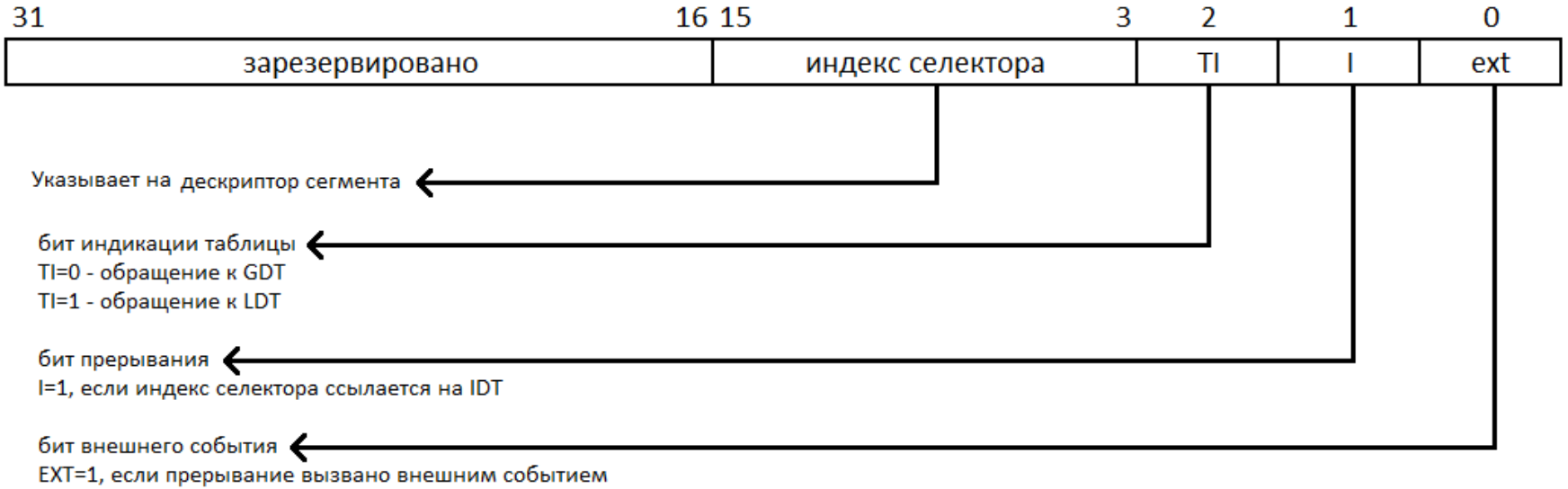


Без кода ошибки

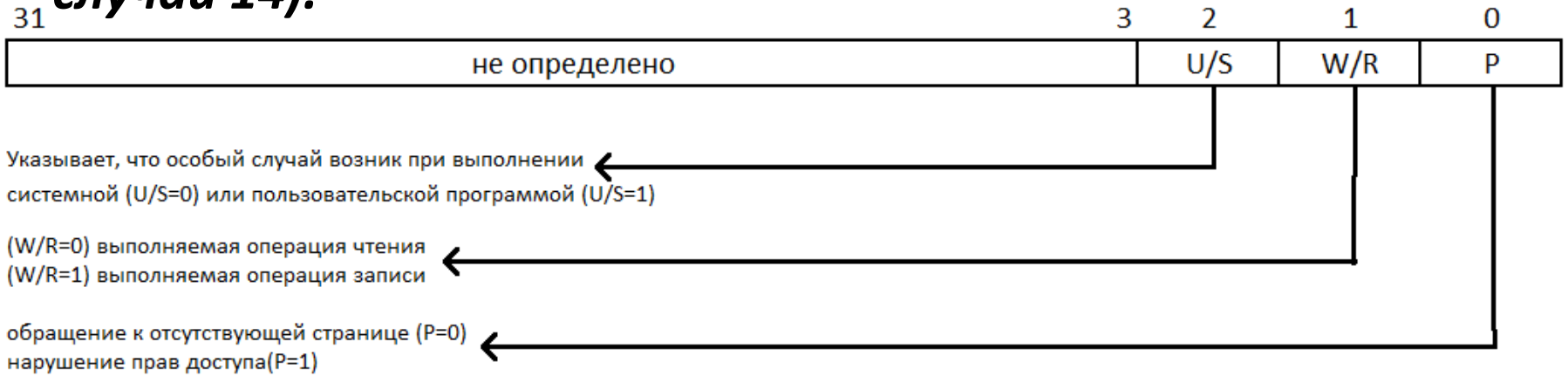


С кодом ошибки

Типовой формат 32-разрядного кода ошибки для особых случаев 8, 10-13, 17.



Формат кода ошибки для страничного нарушения (особый случай 14).



CR2 содержит линейный адрес, преобразование которого вызвало особый случай.