Режим виртуального процессора i8086

Процессор i80386 содержит средства для работы в так называемом режиме виртуального процессора i8086, называемого также для краткости режимом V86 или просто виртуальным режимом. Заметим, что до разработки фирмой Intel процессора i80386 термин "виртуальный режим" иногда использовался в литературе для обозначения защищённого режима работы процессора i80286.

Когда процессор i80386 находится в виртуальном режиме, его поведение во многом напоминает поведение i8086. В частности, для адресации памяти используется схема <сегмент:смещение>, размер сегмента составляет 64 килобайта, а размер адресуемой в этом режиме памяти - 1 мегабайт.

В режим V86 процессор может перейти из защищённого режима, если установить в регистре флагов EFLAGS бит виртуального режима (VM-бит). Номер бита VM в регистре EFLAGS - 17.

Процессор проверяет этот флаг в следующих случаях:

если загружает значение в сегментный регистр, обновляя теневую часть регистра (при этом используется модель реального адреса);

при декодировании инструкций, определяя, какие инструкции не поддерживаются в состоянии V86, а какие зависят от IOPL;

при контроле правил защиты (привилегированные инструкцииприи контроле правил защиты (привилегированные инструкции, контроль пределов, выравниванияприи контроле правил защиты (привилегированные инструкции, контроль пределов, выравнивания, контроль на уровне страниц).

Системное программное обеспечение не может непосредственно изменить состояние бита VM (например, при помощи POPFD). Вместо этого следует менять образ регистра EFLAGS в стеке (при IRET) или в TSS (при переключении задач).

Процессор начинает выполнять задачу в состоянии V86 в двух случаях:

При переключении на новую задачу, когда регистр EFLAGS, загружаемый из нового TSS, содержит бит VM=1. Следует отметить, что задача в состоянии V86 может определяться только 32-битным TSS, т.к. 16-битный TSS содержит только младшее слово регистра EFLAGS, не включающее бит VM.

При возврате из обработчика прерывания или исключения защищенного режима CPL=0 с помощью команды IRET без переключения задач (EFLAGS.NT=0), когда образ регистра EFLAGS в стеке содержит бит VM=1. (Если уровень привилегий обработчика отличен от нуля, процессор не изменит бит VM).

Процессор прекращает выполнять задачу в состоянии V86 только при возникновении прерывания или исключения в следующих случаях:

Если обработчик соответствующего прерывания / исключения представлен в IDT шлюзом задачи, при переключении на которую процессор покидает виртуальный режим (ее TSS содержит EFLAGS.VM=0).

Если обработчик прерывания / исключения находится в сегменте кода с PL=0.

Монитор V86

Для полноценного выполнения программы для МП 8086 задача в состоянии V86, описываемая 32-битным TSS, должна содержать, кроме программы МП 8086, сервисы операционной системы МП 8086 и монитор V86.

Монитор V86 - программный модуль, выполняемый в сегменте кода с PL=0 в защищенном режиме. Монитор V86 обеспечивает инициализацию, обработку прерываний и исключений, реализацию процедур ввода-вывода, эмулирующих аппаратную платформу компьютеров на основе МП 8086. Как правило, основная часть монитора V86 - это обработчик нарушения общей защиты. Именно оно позволяет эмулировать программно-аппаратную среду МП 8086 в многозадачной системе на основе микропроцессора архитектуры IA-32. Как и любая другая программа для защищенного режима, монитор V86 использует дескрипторы сегментов в GDT или локальной дескрипторной таблице задачи. Монитору могут также понадобиться дескрипторы сегментов данных, с помощью которых монитор получает доступ к IDT или другим частям программы МП 8086, находящимся в первом мегабайте линейного адресного пространства.

Страничная трансляция и защита

Страничная трансляция, не являясь обязательным механизмом при выполнении одной задачи V86, тем не менее, позволяет реализовать дополнительные возможности:

перенаправлять или захватывать ссылки на устройства ввода-вывода, отображаемые в пространство памяти;

совместно использовать операционную систему МП 8086 или процедуры, записанные в ПЗУ, и общие для нескольких одновременно выполняемых задач в состоянии V86;

создавать множество задач V86, в котором каждая задача отображает нижний мегабайт линейных адресов на различные физические ячейки;

При выполнении программ МП 8086 процессор не обращается к дескрипторам и, следовательно, не использует реализуемый с помощью дескрипторов механизм защиты. В то же время, механизмы защиты, не связанные непосредственно с дескрипторами, продолжают действовать: контроль предела (все сегменты по 64Кбайт), контроль выравнивания, контроль привилегированных команд, защита на уровне страниц. Когда задача находится в состоянии V86, ей приписывается третий уровень привилегий.

В виртуальном режиме в число команд, чувствительных к уровню привилегий ввода-вывода, входят следующие команды: CLI, STI, PUSHF, POPF, INT *n* и IRET. В число этих команд не входят инструкции обращения к портам ввода-вывода (IN, INS, OUT, OUTS), т.к. в виртуальном режиме процессор вне зависимости от IOPL обращается к двоичной карте разрешения ввода-вывода;

Для защиты системных программ в задаче V86 от программ МП 8086 можно использовать два подхода. Разработчик программного обеспечения может использовать бит U/S элемента таблицы страниц для защиты монитора виртуальной машины и других системных программ, располагаемых в пространстве каждой задачи V86. Т.к. страницы монитора виртуальной машины имеют привилегии супервизора, они недоступны для программы МП 8086.

Другой способ защиты системы от прикладных программ МП 8086 заключается в резервировании в начале линейного пространства адресов каждой задачи 1 Мбайт+64 Кбайт для программ МП 8086. Задачи МП 8086 не могут генерировать адреса за пределами этого диапазона.

Обработка прерываний и исключений

При возникновении прерывания или исключения процессор вызывает соответствующий обработчик. В виртуальном режиме способ вызова обработчика зависит от типа возникшего события и от состояния системных флагов.

Для виртуального режима прерывания и исключения можно разделить на три класса:

Немаскируемые аппаратные прерывания (вход NMI#) и исключения процессора. Сюда же относятся маскируемые аппаратные прерывания, если выключено расширение VME (Pentium+).

Маскируемые аппаратные прерывания, поступающие на вход процессора INTR# или по шине APIC (Pentium+), когда включено расширение VME.

Программные прерывания, генерируемые инструкцией INT *n*.

Способ обработки прерывания/исключения зависит от:

1. *VME* (*Virtual-8086 Mode Extensions*, бит 0 регистра CR4, Pentium+) - расширения виртуального режима. Когда этот бит включен, есть возможность особой обработки аппаратных маскируемых прерываний (класс 2) с использованием виртуального флага прерываний (VIF и VIP в EFLAGS). У Intel386 и Intel486 эта возможность отсутствует, поэтому для них следует рассматривать только два класса прерываний и исключений (1 и 3).

2. *IOPL* (*I/O Privilege Level*, биты 12-13 регистра флагов) - уровень привилегий ввода-вывода. Это поле влияет на работу с виртуальным флагом прерываний (VIF) и программными прерываниями (класс 3).

3. *Битовая карта перенаправления программных прерываний* (*Software interrupt redirection bit map*, Pentium+) - опциональная структура в TSS, влияющая на обработку программных прерываний. Размещается по адресу на 32 байта меньше базового адреса битовой карты разрешения ввода-вывода и занимает 32 байта. Битовая карта перенаправления программных прерываний используется только, когда CR4.VME=1.

4. *VIF* и *VIP* (Virtual Interrupt Flag, Virtual Interrupt Pending flag - биты 19 и 20 регистра флагов, Pentium+) - виртуальный флаг прерывания и виртуальный флаг задержки прерывания. Когда включено расширение VME и IOPL<3, инструкции CLI, STI и POPF, не генерируя нарушение общей защиты, вместо бита IF меняют значение бита VIF.

В таблице IDT соответствующий обработчик прерывания / исключения может быть представлен шлюзом ловушки или прерывания, ссылаясь на неподчиняемый сегмент кода с PL=0. В таком случае процессор переключается в защищенный режим на нулевой уровень привилегий и сохраняет в стеке обработчика код ошибки (если есть), адрес возврата (CS:EIP), регистр флагов (EFLAGS), указатель стека в прерванной программе (ES:ESP) и значения сегментных регистров (ES, DS, FS, GS). При этом для обработчика очищаются значения сегментных регистров (т.к. они содержат селекторы задачи V86, которые нельзя использовать в защищенном режиме) и сбрасывается бит VM в регистре флагов. При возврате из обработчика по инструкции IRET процессор должен находиться на нулевом уровне привилегий, иначе он не сможет восстановить значение бита VM для прерванной программы. Монитор V86 получает управление именно таким образом (как обработчик исключения #13).

Обработчик прерывания или исключения может быть также вызван через шлюз задачи. Тогда обработчик выполняется в отдельной задаче. При переключении задач контекст процессора (в т.ч. бит EFLAGS.VM=1) сохраняется в TSS прерванной задачи. В поле "Связь TSS" вызванной задачи заносится селектор TSS прерванной задачи. Для новой задачи сбрасывается бит VM и устанавливается бит NT. При возврате из обработчика по IRET и обратном переключении задач процессор должен находиться на нулевом уровне привилегий, чтобы изменить бит VM.

При обработке прерываний класса 2 дополнительные возможности предоставляет механизм VME. Задача в состоянии V86 при помощи флага VIF, который она может менять при помощи обычных инструкций STI и CLI, сообщает монитору V86 о разрешении или запрещении прерывать программу для обработки маскируемых прерываний, не влияя на общесистемный флаг разрешения прерываний IF. Флаг VIP предоставляет монитору возможность фиксировать отложенные и не обработанные прерывания (возникшие, когда флаг VIF=0).

При возникновении прерывания обработчик может проанализировать флаг VM в стеке и при необходимости передать управление монитору V86, который в зависимости от значения флага VIF решает, обрабатывать прерывание или нет. Если VIF=0, то прерывание следует отложить, тогда монитор выставляет бит VIP и возвращает управление прерванной программе. Если VIF=1, то монитор вызывает обработчик программы 8086 или обработчик защищенного режима.

Если в дальнейшем программа 8086 разрешает "виртуальные" прерывания при помощи инструкции STI, а при этом флаг VIP=1, то процессор генерирует нарушение общей защиты, так что монитор V86 получает возможность обработать отложенное прерывание.

Обработка программных прерываний (класс 3) зависит от возможности использования битовой карты перенаправления программных прерываний. Каждый бит в этой карте указывает на необходимость перенаправления того или иного программного прерывания (бит 0 соответствует прерыванию 0 и т.д.) Если бит выставлен, то соответствующее программное прерывание пененаправляется обработчику защищенного режима. Если бит сброшен, то процессор обрабатывает прерывание так, как если бы оно генерировалось в реальном режиме (или МП 8086), т.е. передает его обработчику в задаче V86, вычисляя адрес обработчика по таблице векторов по линейному адресу 0. Следует отметить, что битовая карта перенаправления прерываний не влияет на обработку исключений (0-32) и аппаратных прерываний.

Выводы

Виртуальный режим предназначен для работы программ, ориентированных на процессор i8086 (или i8088). Но виртуальный режим - это не реальный режим процессора i8086, имеются существенные отличия. Процессор фактически продолжает использовать схему преобразования адресов памяти и средства мультизадачности защищённого режима.
В виртуальном режиме используется трансляция страниц памяти. Это позволяет в мультизадачной операционной системе создавать несколько задач, работающих в виртуальном режиме. Каждая из этих задач может иметь собственное адресное пространство, каждое размером в 1 мегабайт. Все задачи виртуального режима обычно выполняются в третьем, наименее привилегированном кольце защиты. Когда в такой задаче возникает прерывание, процессор автоматически переключается из виртуального режима в защищённый. Поэтому все прерывания отображаются в операционную систему, работающую в защищённом режиме.
Обработчики прерываний защищённого режима могут моделировать функции соответствующих прерываний реального режима, что необходимо для правильной работы программ, ориентированных на реальный режим операционной системы MS-DOS.