

удовлетворен, он перемещается в очередь **В**. Квант обслуживания запросов из очереди **В** равен $t_B >> t_A$. Если за число квантов, равное n_B , запрос не удовлетворен, он перемещается в очередь **С**. Квант в очереди **С** равен $t_C >> t_B$. При выборе запросов из очередей ОС руководствуется следующими правилами. Если очередь **А** не пуста, запрос выбирается из нее. Если очередь **А** пуста и очередь **В** не пуста, запрос выбирается из очереди **В**. Если обе очереди **А** и **В** пусты, запрос выбирается из очереди **С**.

Совокупность приведённых выше правил позволяет обеспечить предпочтительную обработку диалоговых запросов, так как они по определению не являются длительными. В то же время длительные с вычислительной точки зрения запросы будут обрабатываться квантами существенно большей величины, что приведет к снижению затрат времени на переключение контекста при их обслуживании.

Конечным результатом использования рассмотренной трёхочередной дисциплины является снижение времени реакции системы с одновременным сокращением непроизводительных затрат времени ОС.

3.6. Прерывания

Прерывание – временное изменение нормальной последовательности выбора процессором команд программы.

В результате прерывания управление передается другой программе, называемой *обработчиком прерываний*.

Прерывания позволяют оповестить выполняющуюся программу об изменениях вне её. Прерывания являются важным аппаратным средством, положенным в основу многозадачности современных ОС.

Каждое прерывание имеет свой источник.

Источник прерывания — аппаратное средство, о событиях которого оповещает программу данное прерывание.

Обычно в архитектуре ЭВМ прерывания делятся на классы (виды, типы и т. д.), каждому из которых назначен свой приоритет. Классификация прерываний происходит в соответствии с их источниками. Например, в IBM System/370 существует шесть классов прерываний, названия и источники которых приведены в следующей таблице.

Класс прерываний	Источники прерываний
Внешние прерывания	Средства отсчета времени, другие ЦП, другие вычислительные машины
Прерывания ввода/вывода	Внешние устройства, устройства управления, устройства ввода/вывода
Программные прерывания	Особые случаи при выполнении программ
Прерывание по обращению к супервизору	Команда SVC «Обращение к супервизору»
Прерывание от схем контроля	Схемы контроля вычислительной машины
Прерывание повторного пуска	Кнопка повторного пуска, другие ЦП многопроцессорного комплекса

Выделяют две следующие группы прерываний.

- Синхронные прерывания — прерывания, связанные с управлением ресурсами (получение, освобождение).

- Асинхронные прерывания — прерывания, связанные с параллельным выполнением процессов. Например, прерывания ввода/вывода позволяют синхронизировать работу устройств ввода/вывода и выполнение программы ЦП.

Разрешение прерываний

Чтобы прерывание могло произойти, формирования в источнике прерываний определенных условий не достаточно; прерывание должно быть *разрешено*. Разрешение или запрет тех или иных классов прерываний выполняется при помощи установки необходимых значений в соответствующих разрядах регистров управления аппаратурой (управляющих регистрах). Если прерывания запрещены, то говорят, что они *замаскированы*.

Архитектура ЭВМ может предусматривать общее или индивидуальное разрешение прерываний. При общем разрешении прерываний разрешаются или запрещаются прерывания всех классов одновременно. При индивидуальном разрешении доступность прерываний различных классов может быть настроена отдельно. Кроме того, при индивидуальном разрешении прерываний ЦП может предоставлять инструмент для временного запрета всех прерываний без изменения их индивидуальных настроек. Возможность маскирования некоторых классов прерываний может отсутствовать.

Если прерывание замаскировано, ЦП запоминает наличие отложенного (хранящегося) прерывания. В некоторых случаях наличие отложенного прерывания может быть проверено и сброшено программно. Например, наличие отложенных прерываний в Atmel AVR может быть выявлено в результате анализа флагов аппаратных средств, а хранящиеся запрещенные прерывания ввода/вывода в IBM System 370 могут быть обнару-

жены и обработаны при помощи инструкции ТЮ «Проверить ввод/вывод».

Передача управления обработчику прерываний

ЦП передает управление обработчику (т.е. аппаратно выполняет прерывание) только при формировании условий разрешенных прерываний. Если условия прерывания формируются во время выполнения некоторой команды, и эти условия не препятствуют нормальному завершению текущей команды, то обычно ЦП обрабатывает текущую команду до конца. В противном случае, например, если исполнительный адрес выходит за пределы адресного пространства, текущая команда подавляется.

Нормальная последовательность выбора команд при прерывании временно приостанавливается. Вместо этого ЦП: 1) сохраняет адрес следующей команды и 2) выполняет команду, расположенную в ОП по адресу, соответствующему классу происходящего прерывания. В дополнение к названным могут быть аппаратно выполнены и другие действия, например, могут быть запрещены все прерывания или на основании некоторых внешних данных установлен *код прерывания*, уточняющий его причину.

Соответствие между классом прерывания и адресом команды, на которую передается управление, устанавливается разработчиком архитектуры ЭВМ. Во многих случаях такие адреса группируются в таблицы прерываний, например, Interrupt Descriptor Table для Intel x86. Таблица прерываний может располагаться в ОП следующим образом:

- в одном и том же неизменном месте (например, в постоянно распределенных областях ОП в однопроцессорных IBM System 370);

- располагаться в одном из фиксированных мест (например, в Atmel AVR);
- в произвольном месте (например, в Intel x86 адрес IDT задается регистром IDTR).

Для обеспечения возможности возобновления прерываемой программы должна быть сохранена вся необходимая информация: адрес команды, значения регистров, виртуальное адресное пространство, служебные данные ОС. Иными словами, при прерывании должен быть сохранен контекст прерываемого процесса. При сохранении контекста комбинируются аппаратные и программные действия.

Обычно аппаратно сохраняется адрес возобновления прерванной программы. Этот адрес может сохраняться как в стеке, так в фиксированной области ОП. Использование стека представляется более гибким.

Другие составляющие контекста (регистры, виртуальное адресное пространство и т.п.) сохраняются программно.

Часто на время выполнения обработчика прерываний запрещаются если не все прерывания, то как минимум прерываний обрабатываемого класса. Этим обеспечивается последовательная (FIFO) обработка прерываний.

По окончании обработки прерывания управление может быть возвращено прерванной программе. Для этого обработчик прерываний должен восстановить контекст этой программы и передать управление по сохраненному в момент прерывания адресу.

Обработка прерываний в ОС

Обработка прерываний является одной из важнейших функций ОС по обслуживанию процессов. Покажем невозможность обработки прерываний пользовательскими процессами.

- 1) Даже относящееся к процессу пользователя прерывание не может быть им обработано самостоятельно, т.к. сопряжено с целым рядом действий, на которые у процесса пользователя нет полномочий: проверка значений управляющих регистров, опрос аппаратуры и т.д.
- 2) Корректная обработка прерываний невозможна без соблюдения определенного протокола взаимодействия с ЦП и другими компонентами ОС, поэтому требует от разработчика обладания достаточно высокой квалификацией.

Основываясь на приведенной ранее классификации прерываний для архитектуры IBM System 370, рассмотрим, как эти классы прерываний используются, и в чем заключается их обработка.

Внешние прерывания обычно сигнализируют об окончании отсчитываемого интервала времени. В ОС такими интервалами являются, например, кванты предоставления ЦП при использовании кругового циклического алгоритма или многоочередной дисциплины обслуживания. Реальная ЭВМ содержит ограниченный набор таймеров, на основе которого при помощи виртуализации создается большое количество виртуальных таймеров. При этом используется следующий механизм.

Все запросы к виртуальным таймерам на прерывание образуют очередь (рис. 11), упорядоченную по возрастанию моментов времени срабатывания таймера. Значение момента времени прерывания из самого первого запроса очереди заносится в реальный таймер. При наступлении этого момента времени происходит прерывание, в ходе обработки которого:

- 1) оповещается обработчик события, соответствующего данному виртуальному таймеру;

- 2) если таймер был создан для однократного срабатывания, он удаляется, иначе обновляется значение и очередь упорядочивается;
- 3) очередь обновляется.

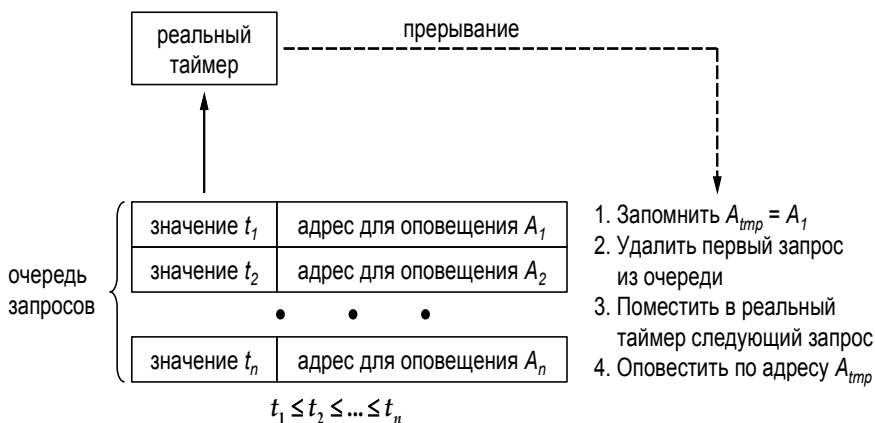


Рис. 11. Создание множественных виртуальных таймеров на основе одного реального.

Если виртуальный таймер был создан программой-диспетчером, то в качестве адреса для оповещения будет указана одна из функций этой программы. Таким образом обеспечивается выполнение обработки истечения кванта программой-диспетчером.

Прерывания ввода/вывода возникают в двух случаях: когда компоненты системы ввода-вывода заканчивают начатую ранее работу и когда эти компоненты хотят обратить на себя внимание ЦП. Прерывания первого типа используются в ОС для синхронизации процессов ЦП с процессами, развивающимися в системе ввода-вывода. Прерывания второго типа позволяют ЦП вовремя инициировать операцию чтения для обработки новых данных.

Начиная операцию ввода-вывода по инициативе некоторого процесса, ОС выполняет следующие действия:

- 1) проверяется возможность выполнения операции ввода/вывода;
- 2) если процесс запрашивает выполнение ввода/вывода в синхронном режиме¹, ОС переводит процесс в необслуживаемое состояние, исключая его из очереди претендентов на ЦП;
- 3) создается дескриптор операции ввода/вывода;
- 4) дескриптор заполняется информацией о запрошенной операции ввода/вывода;
- 5) в дескрипторе запоминается информация о процессе;
- 6) дескриптор ставится в очередь к требуемому устройству ввода/вывода.

Когда дескриптор окажется первым в очереди к запрошенному устройству ввода/вывода, ОС запустит требуемую операцию. По завершении операции ввода/вывода произойдет прерывание, обрабатывая которое ОС извлекает из первого дескриптора информацию о запросившем ввод/вывод процессе. При необходимости ОС выполняет регистрацию и исправление ошибок ввода/вывода. Если завершается операция, выполнявшаяся синхронно, ОС вновь включает процесс в очереди претендентов на ЦП.

Программные прерывания происходят в случаях ошибок в программе и тогда, когда выполнение запрошенного дей-

1 При запуске операции ввода/вывода в *синхронном* режиме возврат к выполнению основной программы происходит только по завершении ввода/вывода, т. е. совмещения во времени работы ЦП и системы ввода/вывода не происходит. Выполнение операции ввода/вывода в *асинхронном* режиме не приостанавливает выполнения программы ЦП, но требует от прикладной программы дополнительных действий по синхронизации процессов.

ствия аппаратно не возможно. Примером ошибки в программе является использование адреса, выходящего за пределы адресного пространства. Примером действия, которое не может быть завершено аппаратно, и требующего программного вмешательства, является обращение к данным, существующим в виртуальном адресном пространстве, но не представленным в реальном².

По этим причинам в некоторых архитектурах ЭВМ программные прерывания называются *исключениями* (Intel x86) или *особыми случаями* (IBM System 370).

Обработка программных прерываний может заключаться в выполнении корректирующего действия, в запуске отладчика или в аварийном завершении процесса.

Прерывания по обращению к супервизору запрашивают стандартные действия, идентифицируемые кодом прерывания. Это позволяет прикладной программе обратиться к ОС (супервизору) для выполнения требуемых действий. Примерами таких запросов являются предоставление реальной памяти, открытие файла, выполнение операции ввода-вывода. Обработка прерываний по обращению к супервизору заключается в выполнении запрошенных действий.

Прерывания от схем контроля сообщают ОС об ошибках в функционировании аппаратуры. Все произошедшие ошибки должны быть занесены в специальный журнал, используемый инженерами по обслуживанию ЭВМ для диагностики и устранения неисправностей. По возможности ОС пытается предпринять корректирующие действия.

Прерывания повторного пуска инициируют повторный пуск ЦП, обычно используемый для вывода ОС из некоторой тупиковой ситуации, выйти самостоятельно из которой она

2 Такая ситуация постоянно возникает при работе виртуальной памяти, она рассмотрена в разд. «Организация виртуальной памяти».

не может. Причинами попадания ОС в тупиковую ситуацию могут быть как отказы аппаратуры, так и ошибочное функционирование ОС. Обработка прерывания повторного пуска заключается в сбросе ЦП и начальной загрузке программы.

Отметим, что в мультипроцессорных системах повторный пуск может использоваться для включения в работу дополнительных процессоров. В этом случае ОС на ведущем процессоре 1) задает в таблице прерываний для ведомого процессора требуемый адрес обработчика прерываний повторного пуска и 2) отправляет адресуемому ведомому процессору запрос на прерывание повторного пуска.

4. Подсистемы ОС

Мы будем рассматривать подсистемы:

- Подсистема управления процессами.
- Подсистема распределения ресурсов.
- Подсистема управления памятью.
- Подсистема управления вводом/выводом.
- Подсистема управления файлами.

4.1. Подсистема управления процессами

В большинстве ОС принята следующая двухуровневая схема:

- Планирование процессов (Долгосрочное планирование).
- Диспетчеризация процессов (Краткосрочное планирование).

Такое разделение построено на частотном принципе ОС.

По концепции виртуализации планированию подвергается не виртуальный процесс, а виртуальная машина в целом.