

Лабораторная работа 5. Обработка прерываний и регулировщик.

Цель.

Реализовать метод обработки прерываний и регулировщик применительно к ПО «Модель ОС».

Теоретическая часть.

В реальной ОС порядок выполнения подпрограмм не известен заранее и определяется внешними факторами по сигналам прерывания.

Регулировщик - часть ОС, отвечающая за состояния процессов.

Только регулировщик имеет право изменять состояния процессов, и тем более в мультипроцессорной системе.

Для упрощения программы модель ОС разрабатывается с допущением, когда порядок выполнения подпрограмм заранее определен и происходит последовательно. В этом случае не может быть ситуации (при условии правильного программирования), при которой изменение состояния процессов выполняется несколькими подпрограммами одновременно (или несколькими процессами одновременно).

Поэтому основную функцию регулировщика по изменению состояния процессов и переводу их из одного списка (например, заблокированных) в другой (готовых) можно реализовать не только отдельной подпрограммой, но и без нее, операторами, расположенными в разных местах программы моделирования.

Регулировщик реализовать в виде подпрограммы Process.

Если модель ОС имеет несколько процессоров, это означает, что одновременно (в цикле) могут находиться в активном состоянии столько же процессов. В этом случае обращение к общим данным (например, выбор номера процесса для выполнения на процессоре или выделение дополнительной памяти) должно быть монополено (один процессор занял и обрабатывает общие данные, другие ожидают доступа). Иначе говоря, если один процессор обращается к общим данным (например, к списку готовности, таблицам распределения памяти при загрузке и выгрузке заданий) и это происходит в несколько тактов, то другие процессоры, если хотят получить доступ к общим данным, должны ждать оставшиеся такты от времени T_{Globl} .

Алгоритм выполнения одного такта моделирования в главном цикле модели ОС.

А. Инициализация цикла: выбрать процесс, сделать его активным и восстановить слово состояния процесса (в этой ситуации оно равно начальному значению).

Б. Действия в цикле.

1. Моделирование ЦПр: для всех активных процессов выполнить очередную команду на своих процессорах, уменьшить остаток кванта времени каждого активного процесса (для алгоритма RR).

2. Моделирование процессоров ввода (вывода): для всех процессоров ввода (вывода) выполнить один такт команды ввода (вывода) (уменьшить счетчик тактов процессора ввода (вывода) и обработать прерывания от процессоров ввода (вывода) (проверить счетчики тактов на нуль и изменить состояния процессов, завершивших ввод (вывод), путем вызова регулировщика).

3. Моделирование прерываний по времени (для алгоритма RR): по окончании квантов времени активных процессов перевести процессы в состояние готовности с сохранением слова состояния процесса (вызов регулировщика).

4. Проверка достигнутого состояния системы: для всех процессоров: если резидентный процесс перестал быть активным или если процессор находится в состоянии «Ожидание» и список готовности не пуст, то вызов планировщика для выбора следующего процесса для выполнения на процессоре, если таковых процессов нет, то процессор переходит в состояние « Ожидание», а ОС может породить системный процесс для загрузки нового задания, если есть ресурсы (место в таблице слов состояний процессов, достаточная память и свободная запись в таблице распределения памяти).

5. Обновление баз данных модели ОС (это часто происходит по ходу выполнения предыдущих пунктов).

6. Отображение изменений в состоянии системы: вывод всех данных на экран или только тех, которые изменили свое значение (это сложнее, но предпочтительнее, т.к. работает быстрее).

Практическая часть

1. Доделать подпрограмму «Выполнение операции ввода (вывода)».
2. Реализовать индивидуальный алгоритм планировщика в виде процедуры.
3. Реализовать подпрограмму работы процессоров ввода (вывода).
4. Реализовать моделирование сигнала прерывания по окончании операции ввода (вывода).
5. Реализовать проверку состояния системы.
6. Обновить подпрограмму «Выполнение одного такта моделирования».
7. Обновить подпрограмму индикации.

Получить результаты:

- формулы выбора следующего процесса для обслуживания на ЦПр,
- индикацию параметров в соответствии с индивидуальным алгоритмом работы планировщика.

В отчете представить:

1. Блок схемы алгоритмов
2. Анализ программного кода
3. Листинг программы
4. Экранные формы