

poisk-ru.ru/s7465t4.html

Взаимодействие и **синхронизация процессов и потоков**. В мультипрограммных однопроцессорных системах **процессы** чередуются, обеспечивая эффективное выполнение программ.

intuit.ru/Курсы/Лекция/?page=5

5.6. Взаимодействие и **синхронизация процессов и потоков**. В мультипрограммных однопроцессорных системах **процессы** чередуются, обеспечивая эффективное выполнение программ.

mpei.ru/study/courses/oss/OSS...Multitasking.pdf

Критическая область (КО) — это примитив для **синхронизации** нескольких **потоков** одного **процесса**. Критической областью защищают участок кода, который не должен одновременно выполняться несколькими **потоками**.

knowledge.allbest.ru/Программирование/..._0.html

Обзор операционных систем, обеспечивающих взаимную **синхронизацию процессов и потоков**. Понятие критической секции и критических данных, описание приема взаимного исключения.

codenet.ru/Языки_программирования/...process-threads-sync.php

Синхронизация процессов и потоков. **Процессом** (process) называется экземпляр программы, загруженной в память. Этот экземпляр может создавать нити (thread), которые представляют собой последовательность инструкций на выполнение.

naparah.com/evm/0522964.html

В случае **синхронизации потоков** разных **процессов** операционная система должна предоставлять **потокам** системные объекты ... Приостановка и активизация **потоков** осуществляются в зависимости от состояния **синхронизирующих** объектов ОС.

StudFiles.ru/preview/4550136/ Приоритезация маскирование **прерываний**. Последовательность действий при **обработке прерываний**. Диспетчер **прерываний**. **Обработка ловушек в Windows**.

datbase.ru/windows/obrabotka...preryivaniy.html

...**прерываний** (Interrupt dispatch table, IDT) и передает управление соответствующей процедуре **обработки прерывания**. Во время загрузки системы **Windows** заполняет IDT указателями на процедуры ядра, **обрабатывающими** каждое...

life-prog.ru/vadim.php?id=101&page=5

В некоторых ОС - Linux , Microsoft **Windows** и некоторых других, **обработка прерываний** поделена на две части: обработчики **прерываний** первого уровня (англ.

<http://programming-lang.com/ru/computers/russinovich/0/j3.html>

Системные механизмы

В Microsoft Windows существует несколько базовых механизмов, которыми пользуются компоненты режима ядра: исполнительная система (executive), ядро и драйверы устройств. В этой главе описываются следующие системные механизмы (а также способы их использования):

- диспетчеризация ловушек (trap dispatching), в том числе прерываний, DPC (deferred procedure call), APC (asynchronous procedure call), исключений и системных сервисов;
- диспетчер объектов исполнительной системы;
- синхронизация, в том числе спин-блокировки, объекты диспетчера ядра (kernel dispatcher objects) и реализация механизмов ожидания;
- системные рабочие потоки;
- различные механизмы вроде поддержки глобальных флагов Windows;
- LPC (local procedure call);
- Kernel Event Tracing;
- Wow64.

Обработка прерываний и исключений

helpiks.org/7-71531.html

В Windows NT процессор передает управление обработчику ловушки ядра NT. Этот модуль играет роль коммутационной панели; он принимает исключения и **прерывания**, генерируемые процессором, и передает управление коду **обработки**...

Прерывания. Механизм обработки прерываний

pandia.ru/text/77/385/27783.php

Механизм **обработки прерываний**. Типы **прерываний**: аппаратное, программное, исключительная ситуация. ... 4. Сохранение информации о **прерванной** программе, которую не удалось спасти на шаге 2 с помощью **действий** аппаратуры.

<http://www.cyberguru.ru/programming/win32/win32-keyboard.html>

- [Win32 API. Работа с клавиатурой](#)
- [Фокус и активизация клавиатуры](#)
- [Сообщения о нажатии клавиши](#)
- [Системные и несистемные нажатия клавиш](#)
- [Описание кодов виртуальной клавиши](#)
- [Флажки сообщения о нажатии клавиши](#)
- [Символьные сообщения](#)
- [Несистемные символьные сообщения](#)
- [Сообщения о диакритическом знаке](#)
- [Состояние клавиши](#)
- [Поддержка "горячей" клавиши](#)
- [Языки, регионы и раскладки символов на клавиатуре](#)
- [Использование ввода информации с клавиатуры](#)
- [Обработка сообщений о нажатии клавиши](#)
- [Трансляция символьных сообщений](#)
- [Обработка символьных сообщений](#)

- [Отображение ввода информации с клавиатуры](#)

Win32 API. Справочник по диалоговому окну

- [Win32 API. Справочник по диалоговому окну](#)
- [Функция CreateDialog](#)
- [Функция CreateDialogIndirect](#)
- [Функция CreateDialogIndirectParam](#)
- [Функция CreateDialogParam](#)
- [Функция DefDlgProc](#)
- [Функция DialogBox](#)
- [Функция DialogBoxIndirect](#)
- [Функция DialogBoxIndirectParam](#)
- [Функция DialogBoxParam](#)
- [Функция DialogProc](#)
- [Функция EndDialog](#)
- [Функция GetDialogBaseUnits](#)
- [Функция GetDlgCtrlID](#)
- [Функция GetDlgItem](#)
- [Функция GetDlgItemInt](#)
- [Функция GetDlgItemText](#)
- [Функция GetNextDlgGroupItem](#)
- [Функция GetNextDlgTabItem](#)
- [Функция IsDialogMessage](#)
- [Функция MapDialogRect](#)
- [Функция MessageBox](#)
- [Функция MessageBoxEx](#)
- [Функция SendDlgItemMessage](#)
- [Функция SetDlgItemInt](#)
- [Функция SetDlgItemText](#)
- [Функция MessageBoxIndirect](#)
- [Структуры диалогового окна](#)
- [Структура DLGITEMTEMPLATE](#)
- [Структура DLGTEMPLATE](#)
- [Структура MSGBOXPARAMS](#)
- [Псевдоструктура DLGITEMTEMPLATEEX](#)
- [Псевдоструктура DLGTEMPLATEEX](#)
- [Сообщения диалогового окна](#)
- [Сообщение DM_GETDEFID](#)
- [Сообщение DM_REPOSITION](#)
- [Сообщение DM_SETDEFID](#)
- [Сообщение WM_CTLCOLORDLG](#)
- [Сообщение WM_CTLCOLORMSGBOX](#)
- [Сообщение WM_ENTERIDLE](#)
- [Сообщение WM_GETDLGCODE](#)
- [Сообщение WM_INITDIALOG](#)
- [Сообщение WM_NEXTDLGCTL](#)
- [Все страницы](#)

SetWindowsHookEx()

```
SetWindowsHookEx(WH_KEYBOARD_LL, (HOOKPROC) &KeyboardProc,  
GetModuleHandle(NULL), 0);
```