УДК 004.454

ОБЩАЯ КОНЦЕПЦИЯ ПРОГРАММИРОВАНИЯ ДРАЙВЕРОВ ДЛЯ ОС СЕМЕЙСТВА WINDOWS NT

Морозов Д.С., Теплинский С.В.
Донецкий национальній технический университет

Главная проблема, скрытая во внутренней реализации ОС семейства Windows NT (Windows NT/2000 и выше) для системного программиста, – это строгий контроль над портами ввода-вывода. В отличие от ОС Windows 3.x/9.x, в Windows NT сгенерируется исключение привилегированной инструкции при попытке получения доступа из пользовательского приложения к порту, обращаться к которому она не привилегированна. В прочем, не столько операционная система запрещает подобные обращение, сколько любой 386-совместимый процессор (или выше), который запущен в защищенном режиме.

Получение доступа к портам ввода-вывода в защищенном режиме регулируется двумя событиями: уровнем привилегий ввода-вывода (IOPL) в регистре EFLAGS и битовой картой разрешений ввода-вывода сегмента состояния задач (Task State Segment – TSS).

В Windows NT используются только 2 уровня привилегий ввода-вывода: уровень 0 и уровень 3. Пользовательские приложения запускаются с уровнем привилегий 3, в то время как драйверы устройств и ядро – с уровнем привилегий 0, который в общем случае называется ring0. Это позволяет доверенным приложениям и драйверам операционной системы, запущенных в режиме ядра, получать доступ к портам и ограничивать менее доверенные пользовательские приложения от такого доступа во избежание конфликтов и системных сбоев. Все пользовательские приложения должны обращаться к специальному драйверу, предоставляющему гарантированный доступ к портам.

Битовая карта разрешений ввода-вывода может быть использована для непривилегированных программ (пользовательских) с целью получения доступа к определенным

портам ввода-вывода. Когда инструкция ввода-вывода начнет выполняться, процессор проверит, достаточно ли привилегий у задачи для получения доступа. Если это так, то инструкция ввода-вывода будет выполнена. В противном случае, процессор сначала проверит битовую карту разрешений ввода-вывода.

Битовая карта разрешений ввода-вывода использует один бит, представляющий определенный порт. Если бит, отвечающий за номер порта, установлен, то инструкция сгенерирует исключение. Если этот бит не установлен, операция продолжится. Такой механизм позволяет предоставить доступ определенным приложениям к определенным портам, т.е. существует одна битовая карта разрешений ввода-вывода на каждую задачу [1].

Получение доступа к портам ввода-вывода в операционной системе Windows NT

Существует два способа получения доступа к портам ввода-вывода в Windows NT. Первый – это написать драйвер устройства, который работает в ring0 (уровень 0 привилегий) для доступа к нужным портам ввода-вывода. Данные смогут перемещаться между пользовательской программой и устройством с помощью вызовов IOCTL. Драйвер будет в состоянии исполнять пользовательские инструкции ввода-вывода.

Другой возможной альтернативой является изменение битовой маски разрешений ввода вывода для разрешения определенным задачам получать доступ к портам ввода-вывода. Это гарантирует пользовательскому приложению работу в режиме ring3 и выполнение операций ввода-вывода на выбранных портах. Однако использование такого метода крайне не рекомендуется. Написание драйвера устройства для поддержки аппаратного обеспечения является более предпочтительным методом.

Однако не следует забывать о том, что при каждом вызове IOCTL для чтения или записи байта в порт процессор должен переключиться из ring3 в ring0 для выполнения операции, а

затем переключиться обратно. Т.е. при написании, например, программатора микроконтроллера, который программируется через последовательный порт, следует пересылать не каждый байт по отдельности, а указатель на буфер байт. В этом случае драйвер устройства сформирует очередь и передаст данные в порт с минимальным числом переключений процессора [2].

Общая схема работы драйвера изображена на рис. 1.

Для демонстрации возможностей работы с портами ввода-вывода были использованы свободные исходные коды проекта PORTIO, входящего в техническую документацию WinDDK (Driver Development Kit) версии 7600.16385.0 компании Microsoft. Сборка проекта из исходных кодов выполнялась в вышеуказанной среде

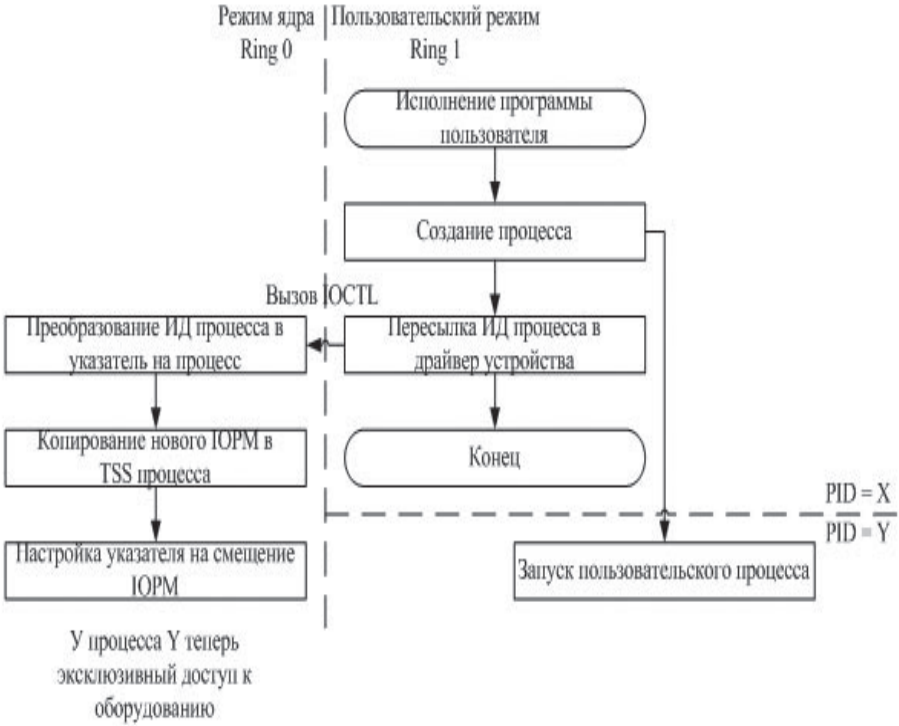

Рисунок 1 – Общая схема работы драйвера, предоставляющего доступ к портам ввода-вывода в ОС Windows NT

разработки. Результатом компиляции являются непосредственно сам файл драйвера и установочный INF-файл устройства. Также для наглядной демонстрации работы драйвера были написаны утилиты (пользовательские приложения) чтения и записи данных в порт, используя драйвер PORTIO [3].

Литература

[1] Харт, Джонсон, М. Системное программирование в среде Windows, 3-е издание: Пер. с анг. – М.: Издательский дом «Вильямс», 2005. – 592 с.: ил.
[2] PortTalk - A Windows NT I/O Port Device Driver. Electronic recourse: http://beyondlogic.org/porttalk/porttalk.htm
[3] Windows Server 2003 DDK. Electronic recourse: http://www.microsoft.com/whdc/devtools/ddk