

Работа с устройствами. РЮ. DMA.

¹Кафедра информационных технологий и систем
Национальная металлургическая академия Украины

22 сентября 2011 г.

Устройство ввода-вывода — компонент типовой архитектуры ЭВМ, предоставляющий компьютеру возможность взаимодействия с внешним миром и, в частности, с пользователями и другими компьютерами.

Все операции, не являющиеся внутренними по отношению к комплексу «CPU-Memory», рассматриваются как операции ввода/вывода.

Ввод-вывод через порты (англ. I/O ports) — схемотехническое решение, организующее взаимодействие процессора и внешних устройств (памяти, устройств ввода-вывода и т.д). Противоположность вводу-выводу через память.

Во многих моделях процессоров ввод-вывод организуется теми же функциями, что и чтение-запись в память — так называемый ”ввод-вывод через память”. Соответственно, схемотехнически устройства ввода-вывода располагаются на шине памяти, и часть адресов памяти направляются на ввод-вывод. В процессорах Intel, микроконтроллерах AVR и некоторых других существуют отдельные команды для ввода-вывода — IN и OUT — и, соответственно, отдельное адресное пространство.

Преимущества портов:

- Возможна совсем другая схемотехническая организация ввода-вывода.
- Человек, читающий ассемблерный листинг, сразу же видит, что это работа не с памятью, а с внешним устройством.
- Всё адресное пространство машины можно пустить на ОЗУ, без всяких ”дыр” наподобие UMB.
- Для машин низкой разрядности, у которых адрес в памяти задаётся регистровой парой, а на порт хватает и одного регистра, ускоряется работа с внешними устройствами.

Преимущества памяти:

- Упрощение конструкции процессора.
- Более широкий набор возможных инструкций: все инструкции, способные записать данные в память, в том числе автоинкрементные и SIMD, могут заниматься вводом-выводом.
- Удобнее работа с функциями, оперирующими большими объёмами данных (например, чтение-запись на диск, в видеоадаптер).

PIO — Программный ввод/вывод (англ. Programmed input/output, PIO), метод передачи данных между двумя устройствами, использующий процессор как часть маршрута данных (процессор выполняет команду чтения порта, считывает байт или слово данных в свой регистр, после чего переписывает его в память, затем повторяет эту процедуру до тех пор, пока вся необходимая информация не будет считана из устройства в память).

Как правило, этот термин применяется к классическому PC/AT контроллеру IDE/ATA/SATA. В этом режиме контроллер требовал исполнения драйвером или же BIOSом команды процессора REP INSW/OUTSW для передачи всех данных, обычно команда исполняется в обработчике прерывания контроллера в статусе "готов к передаче данных".

Режим имеет существенные недостатки — загрузку процессора вводом-выводом и крайне невысокую скорость передачи из-за невозможности использования ”взрывного” (burst) режима шины PCI, и устарел с появлением в середине 1990 годов поддержки DMA в IDE контроллере. Но, во-первых, если режим PIO работает одинаково во всех PC-совместимых компьютерах, то реализация DMA зависит от производителя контроллера (обычно Intel, VIA или nVidia) и требует установки драйверов/сборки ядра с поддержкой данных драйверов.

Контроллеры IDE/SATA, не совместимые со стандартным PC/AT (HighPoint, Promise, Silicon Image, AHCI и другими), не имеют и устаревшего режима PIO как метода передачи данных между процессором и контроллером. Они всегда требуют установки драйвера (драйвер обычно реализован как драйвер SCSI контроллера) и всегда используют проприетарную реализацию DMA для общения с процессором/памятью, как и контроллеры SCSI.

Контроллеры IDE/SATA, не совместимые со стандартным PC/AT (HighPoint, Promise, Silicon Image, AHCI и другими), не имеют и устаревшего режима PIO как метода передачи данных между процессором и контроллером. Они всегда требуют установки драйвера (драйвер обычно реализован как драйвер SCSI контроллера) и всегда используют проприетарную реализацию DMA для общения с процессором/памятью, как и контроллеры SCSI.

Бывают режимы PIO Mode 0, 1, 2, 3, 4. Чем больше номер режима, тем быстрее. IDE ZIP100 приводы от Iomega, например, могут обеспечить только PIO mode 0. Старые CD-ROM приводы, как правило, используют PIO mode 4, если не могут работать в режиме DMA.

- PIO Mode 0 = 3.3 Mb/s
- PIO Mode 1 = 5.2 Mb/s
- PIO Mode 2 = 8.3 Mb/s
- PIO Mode 3 = 11.1 Mb/s
- PIO Mode 4 = 16.7 Mb/s

Прямой доступ к памяти (англ. Direct Memory Access, DMA) — режим обмена данными между устройствами или же между устройством и основной памятью (RAM), без участия Центрального Процессора (ЦП). В результате скорость передачи увеличивается, так как данные не пересылаются в ЦП и обратно.

Кроме того, данные пересылаются сразу для многих слов, расположенных по подряд идущим адресам, что позволяет использовать т. н. ”пакетного” (burst) режима работы шины — 1 цикл адреса и следующие за ним многочисленные циклы данных. Аналогичная оптимизация работы ЦП с памятью крайне затруднена.

DMA-контроллер может получать доступ к системной шине независимо от центрального процессора. Контроллер содержит несколько регистров, доступных центральному процессору для чтения и записи. Регистры контроллера задают порт (который должен быть использован), направление переноса данных (чтение/запись), единицу переноса (побайтно/пословно), число байтов, которое следует перенести.

ЦП программирует контроллер DMA, устанавливая его регистры. Затем процессор даёт команду устройству (например, диску) прочитать данные во внутренний буфер. DMA-контроллер начинает работу, посылая устройству запрос чтения (при этом устройство даже не знает, пришёл ли запрос от процессора или от контроллера DMA). Адрес памяти уже находится на адресной шине, так что устройство знает, куда следует переслать следующее слово из своего внутреннего буфера. Когда запись закончена, устройство посылает сигнал подтверждения контроллеру DMA. Затем контроллер увеличивает используемый адрес памяти и уменьшает значение своего счётчика байтов. После чего запрос чтения повторяется, пока значение счётчика не станет равно нулю. По завершении цикла копирования устройство инициирует прерывание процессора, означающее завершение переноса данных. Контроллер может быть многоканальным, способным параллельно выполнять несколько операций.

В шинах MicroChannel, SBus, разработанной под их большим влиянием PCI и её концептуальных производных AGP и PCI-X, используется иная реализация DMA. Эти шины позволяют любому устройству заявить о возникновении потребности к захвату шины, такая потребность удовлетворяется т. н. арбитром при первой возможности. Устройство, успешно осуществившее захват шины, самостоятельно выставляет на шину сигналы адреса и управления и исполняет в течение какого-то времени ту же ведущую роль на шине, что и ЦП. Доступ ЦП к шине при этом кратковременно блокируется. В такой реализации DMA не существует DMA-контроллера, а также номера входа DMA-контроллера.

Процессор позволяет осуществлять ввод-вывод как через память, так и через порты. Доступ к портам осуществляется особыми командами IN, INS, INSB, INSW, INSD, OUT, OUTS, OUTSB, OUTSW и OUTSD; доступ к памяти — обычными функциями работы с памятью (MOV, ADD, LODSB и прочими). Два последовательных порта могут объединяться в один двухбайтовый порт, четыре — в один четырёхбайтовый. Как и с памятью, для максимальной производительности номера портов должны быть выровнены соответственно по 2- и 4-байтовой границе. Если требуется записывать данные в два соседних порта в определённом порядке, этого нельзя делать многобайтовыми командами (на выровненных портах ввод-вывод происходит параллельно, на невыровненных — порядок вызова не определён).

Процессор умеет проецировать порты в память; при этом процессор гарантирует, что операция ввода-вывода через порт завершится до того, как начнёт выполняться следующая команда. С вводом-выводом через память такой гарантии нет. Впрочем, процессор не проверяет ошибок чётности на портах, поэтому в системах высокой надёжности программист должен быть готов к ошибкам чётности.

Физически адрес порта подаётся через ту же адресную шину, что и адрес памяти. В процессорах до Pentium был штырь MI/O# (1 — память, 0 — порт); в более поздних эту функцию выполняют линии запроса команды.

Для работы с шиной PCI в PC-AT и совместимых машинах выделено два основных порта.
0CF8h -W порт адреса Address
0CFCh RW порт данных Data
Оба порта являются 32 битными.

Порт адреса представляет из себя следующую 32 битную структуру

31	30 24	23 16	15 11	10 8	7 2	1	0
с	резерв	шина	устройство	функция	Индекс регистра	0	0

Порт адреса задает шину, устройства, и адрес регистра в конфигурационном пространстве устройства. Грубо говоря, устройство - это физически присутствующее устройство. А функция - это логическое устройство.

C - флаг доступа к устройству.

Младшие два бита в порту адреса всегда 0.

По окончании работы с устройством следует сбросить адрес в 0.

Если в ответ на запрос нулевого регистра возвращается 0FFFFh, то устройства не существует. Vendor ID (ID производителя) - для Intel это 8086h, но не может принимать значение 0FFFFh.

Device ID (ID устройства) - принимает различные значения
Revision ID (ID модификации) - обозначает номер модификации устройства, назначается производителем.

Class Code (Код класса) - состоит из трех частей

23	16	15	8	7	0
Base Class Code		Sub Class Code		Programming Interface	

Base Class Code - базовый класс, сокращенно BCC.

Sub Class Code - подкласс, сокращенно SCC.

Programming Interface - интерфейс, сокращенно PI.