

Учебник для экзамена LPI 101: Аппаратные средства и архитектура

Администрирование Linux для начинающих (LPIC-1), тема 101

Параметры BIOS

Мы начнем с общего обзора современных персональных компьютеров, а затем обсудим вопросы настройки компьютера. Мы сосредоточимся на компьютерах, использующих процессоры семейства x86, такие как Intel® Pentium® или AMD Athlon, и шину PCI, поскольку они наиболее распространены в настоящее время.

Многие затронутые здесь темы имеют множество пересечений с рабочими программами LPI для специфической периферии. В следующих разделах этого учебника мы будем ссылаться на этот раздел, как на базовый материал.

Обзор компьютеров и BIOS

Современный персональный компьютер (или ПК) состоит из центрального процессора (ЦП или CPU) для выполнения вычислений, а также некоторого объема памяти для хранения данных, которые используются процессором. Для того, чтобы такое устройство было полезным мы подключаем к нему периферийные устройства, такие как клавиатура, "мышь", монитор, жесткий диск, CD или DVD привод, принтер, сканер и сетевая карта, позволяющие нам вводить, хранить, печатать, отображать и передавать данные.

В описанном компьютере память, используемая процессором, называется памятью с произвольным доступом (Random Access Memory -- RAM) [Прим.пер.: В русскоязычной литературе также широко используется термин ОЗУ -- Оперативное Запоминающее Устройство]. В стандартном ПК эта память является временной, то есть для хранения данных ей необходимо электричество. Выключите компьютер и эта память очистится. Посмотрим с другой стороны: когда мы выключаем ПК, мы превращаем его в набор устройств, которые ничего не делают, до тех пор, пока не будут перепрограммированы. Перепрограммирование происходит в момент включения машины; этот процесс назван начальной загрузкой или загрузкой компьютера.

Процесс начальной загрузки и BIOS

Процесс загрузки включает в себя загрузку операционной системы с внешнего устройства хранения информации, такого как floppy-диск (дискета), CD, DVD, жесткий диск, ключа защиты. Программа, выполняющая эту начальную загрузку постоянно хранится в компьютере и называется Базовой системой ввода/вывода (Basic Input Output System -- BIOS). BIOS хранится в постоянной памяти, иногда называемой Память только для чтения (Read Only Memory -- ROM) [Прим.пер.: В русскоязычной литературе также широко используется термин ПЗУ -- Постоянное Запоминающее Устройство]. В ранних ПК ROM-чип (микросхема ROM) был впаян или вставлен в специальное гнездо материнской платы). Обновление BIOS означало замену микросхемы ROM. Позднее стали использовать электрически перепрограммируемую память только для чтения (Electrically Erasable Programmable Read Only Memories -- EEPROMs). EEPROM позволяет обходиться без

специального оборудования при обновлении BIOS. Сегодня наиболее часто встречающейся формой постоянной памяти является Flash-память, которая используется также и в цифровых камерах и ключах защиты. Flash-память также позволяет обновлять BIOS.

[Прим.пер.: Изначально под ПЗУ понималась память именно постоянная, то есть микросхемы памяти, содержимое которых нельзя было изменить. Именно поэтому в старых материнских платах при обновлении BIOS необходимо было менять саму микросхему, а не микрокод, как в настоящее время. Микросхемы же, допускавшие возможность перепрограммирования, назывались ППЗУ -- Перепрограммируемое Постоянное Запоминающее Устройство]

Кроме управления начальной загрузкой ПК, в настоящее время, программа BIOS обычно позволяет пользователю установить или проверить некоторые параметры конфигурации компьютера. Это включает в себя проверку установленных устройств, таких как RAM, жесткий диск, оптический привод, клавиатура, "мышь", и, возможно, встроенный монитор, звуковую карту и сетевые соединения. Пользователь может разрешить или запретить некоторые функции. Например, можно отключить встроенную в материнскую плату звуковую подсистему, чтобы использовать установленную звуковую карту. Пользователь также может выбрать устройство, с которого следует загружать систему, и установить пароль на доступ к компьютеру.

Чтобы получить доступ к экрану настройки BIOS вам потребуется подключенная к компьютеру клавиатура. При включении компьютера выполняется Power On Self Test [Самотестирование при включении] или POST. На некоторых компьютерах выводится сообщение с просьбой нажать определенную клавишу, чтобы запустить настройку, в противном случае будет продолжена нормальная загрузка, на других вы должны сами знать какую клавишу следует нажать до того, как запустится процесс стандартной загрузки, в то время как сообщение не появляется вообще или исчезло вследствие предыдущего изменения настроек. На некоторых компьютерах вам могут быть предложены и другие варианты кроме запуска программы настройки BIOS, например, как показано на Рисунке 1. В итоге вы должны увидеть окно вроде показанного на Рисунке 2.

Рисунок 1. Доступ к настройкам BIOS

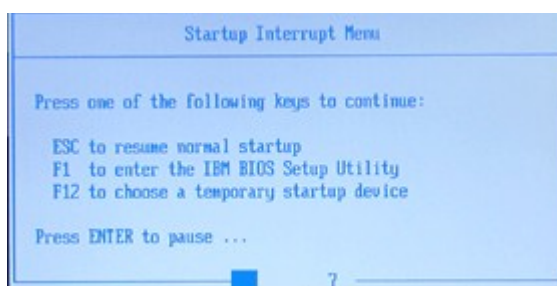
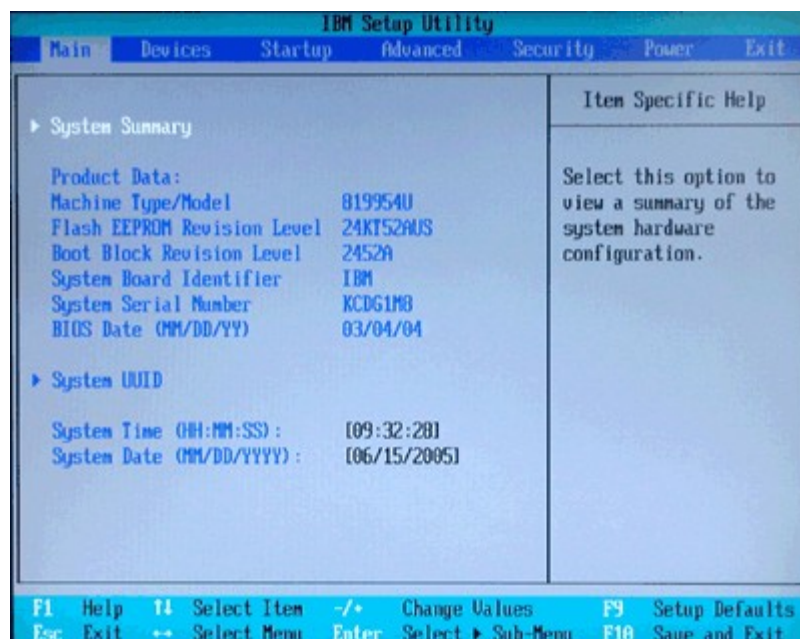


Рисунок 2. Содержание настроек BIOS



Приведенные иллюстрации - это примеры того, что вы можете увидеть, вообще экраны настроек BIOS весьма разнообразны, поэтому не удивляйтесь, если на вашем компьютере они будут выглядеть иначе.

Рисунок 2 показывает нам, что Flash EEPROM (или система BIOS) имеет версию 24KT52AUS и датируется 4 Марта, 2004 тогда как текущая дата -- 9 Июня, 2005. Проверка сайта поддержки производителя (IBM) показывает, что имеется несколько версий BIOS, вышедших позднее, так что, возможно, хорошей мыслью будет обновить системную BIOS.

На Рисунке 2 можно увидеть несколько других пунктов меню. Мы рассмотрим их в следующих разделах этого учебника. Перед тем как сделать это, давайте еще поговорим о внутренней работе ПК.

Шины, порты, IRQ, и DMA.

Шины PCI и ISA

Периферийные устройства, включая те, что могут быть встроены в материнскую плату, взаимодействуют с процессором посредством шины. Наиболее широко используемый в настоящее время тип шины это шина Peripheral Component Interconnect (связь периферийных компонентов) или PCI которая практически заменила более раннюю шину Industry Standard Architecture (Стандартная промышленная архитектура) или ISA. Шина ISA иногда обозначалась как шина AT после IBM PC-AT в котором она была использована впервые в 1984. Во время перехода с ISA на PCI, многие компьютеры содержали обе шины со слотами (гнездами), позволявшими использовать или ISA, или PCI периферию. Шина ISA поддерживала 8-битные и 16-битные карты, тогда как шина PCI поддерживает 32-битные устройства.

Существует еще пара стандартных шин, о которых вы также должны знать. Многие компьютеры имеют слот Accelerated Graphics Port (Ускоренный графический порт) или AGP,

являющийся специальным слотом, основанным на спецификациях шины PCI 2.1, но оптимизированным по пропускной способности и скорости отклика, что необходимо для графических карт. Он медленно вытесняется новой шиной PCI Express или PCI-E, которая лишена многих ограничений базовой конструкции PCI.

Многое о файловых системах Linux мы узнаем в последующих учебниках этой серии, но сейчас мы рассмотрим файловую систему `/proc`. Это не реальная файловая система на диске, а "псевдо-файловая система", предоставляющая информацию о работающем компьютере. В этой файловой системе файл `/proc/pci` содержит информацию об устройствах, подключенных к шине PCI. Ниже приводятся некоторые соображения о том как избежать работы с этим специфичным файлом, поскольку команда `lspci` предоставляет ту же информацию. Выполните команду `cat /proc/pci` и вы увидите нечто похожее на Листинг 1.

Листинг 1. /proc/pci

PCI devices found:

Bus 0, device 0, function 0:

Host bridge: Intel Corp. 82845G/GL [Brookdale-G] Chipset Host Bridge
(rev 1).

Prefetchable 32 bit memory at 0xd0000000 [0xdfffffff].

Bus 0, device 2, function 0:

VGA compatible controller: Intel Corp. 82845G/GL [Brookdale-G] Chipset
Integrated Graphics Device (rev 1).

IRQ 11.

Prefetchable 32 bit memory at 0x88000000 [0x8fffffff].

Non-prefetchable 32 bit memory at 0x80000000 [0x8007ffff].

Bus 0, device 29, function 0:

USB Controller: Intel Corp. 82801DB USB (Hub #1) (rev 1).
IRQ 11.

I/O at 0x1800 [0x181f].

Bus 0, device 29, function 1:

USB Controller: Intel Corp. 82801DB USB (Hub #2) (rev 1).
IRQ 10.

I/O at 0x1820 [0x183f].

Bus 0, device 29, function 2:

USB Controller: Intel Corp. 82801DB USB (Hub #3) (rev 1).
IRQ 5.

I/O at 0x1840 [0x185f].

Bus 0, device 29, function 7:

USB Controller: Intel Corp. 82801DB USB2 (rev 1).
IRQ 9.

Non-prefetchable 32 bit memory at 0xc0080000 [0xc00803ff].

Bus 0, device 30, function 0:

PCI bridge: Intel Corp. 82801BA/CA/DB/EB PCI Bridge (rev 129).
Master Capable. No bursts. Min Gnt=4.

Bus 0, device 31, function 0:

ISA bridge: Intel Corp. 82801DB LPC Interface Controller (rev 1).

Bus 0, device 31, function 1:

IDE interface: Intel Corp. 82801DB Ultra ATA Storage Controller
(rev 1).

IRQ 5.

I/O at 0x1860 [0x186f].

Non-prefetchable 32 bit memory at 0x60000000 [0x600003ff].

Bus 0, device 31, function 3:

SMBus: Intel Corp. 82801DB/DBM SMBus Controller (rev 1).

IRQ 9.

I/O at 0x1880 [0x189f].

Bus 0, device 31, function 5:

Multimedia audio controller: Intel Corp. 82801DB AC'97 Audio Controller (rev 1).

IRQ 9.

I/O at 0x1c00 [0x1cff].

I/O at 0x18c0 [0x18ff].

Non-prefetchable 32 bit memory at 0xc0080c00 [0xc0080dff].

Non-prefetchable 32 bit memory at 0xc0080800 [0xc00808ff].

Bus 2, device 8, function 0:

Ethernet controller: Intel Corp. 82801BD PRO/100 VE (LOM) Ethernet Controller (rev 129).

IRQ 9.

Master Capable. Latency=66. Min Gnt=8. Max Lat=56.

Non-prefetchable 32 bit memory at 0xc0100000 [0xc0100fff].

I/O at 0x2000 [0x203f].

Вы можете сравнить это с результатом получаемым по команде `lspci`. Обычно суперпользователь (root) не указывает путь для запуска этой команды, но обычным пользователям скорее всего потребуется его указать: `/sbin/lspci`. Попробуйте выполнить это на своей машине.

Порты ввода/вывода (IO Ports)

Когда процессору необходимо связаться с периферийными устройствами, он делает это через порт ввода/вывода (иногда его называют просто порт). Когда процессору необходимо передать данные или управляющую информацию для периферии он записывает ее в порт. Когда на устройстве есть данные или оно имеет статус готовности для процессора, то процессор читает данные или статус из порта. Большинство устройств имеют больше одного порта, ассоциированных с ними, обычно их число измеряется первыми степенями двойки, такими как 8, 16 или 32. Передача данных обычно производится одним или двумя байтами. Устройства не могут использовать порты одновременно, поэтому если у вас есть ISA-карты, вы должны убедиться, что каждое устройство имеет связанный с ним порт или порты. Изначально это делалось при помощи переключателей и перемычек на карте, некоторые поздние ISA-карты использовали систему под названием Plug and Play (Подключи и Работай) или PnP, которая будет обсуждаться позже в этом же разделе. PCI-карты все имеют PnP-настройки.

В файловой системе `/proc`, файл `/proc/ioports` говорит нам о портах ввода/вывода доступных в системе. Выполните команду `cat /proc/ioports`, чтобы увидеть результат (он будет похож на Листинг 2).

Листинг 2. /proc/ioproports

```
0000-001f : dma1
0020-003f : pic1
0040-005f : timer
0060-006f : keyboard
0070-007f : rtc
0080-008f : dma page reg
00a0-00bf : pic2
00c0-00df : dma2
00f0-00ff : fpu
0170-0177 : ide1
01f0-01f7 : ide0
02f8-02ff : serial(auto)
0376-0376 : ide1
0378-037a : parport0
03c0-03df : vga+
03f6-03f6 : ide0
03f8-03ff : serial(auto)
0cf8-0cff : PCI conf1
1800-181f : Intel Corp. 82801DB USB (Hub #1)
    1800-181f : usb-uhci
1820-183f : Intel Corp. 82801DB USB (Hub #2)
    1820-183f : usb-uhci
1840-185f : Intel Corp. 82801DB USB (Hub #3)
    1840-185f : usb-uhci
1860-186f : Intel Corp. 82801DB Ultra ATA Storage Controller
    1860-1867 : ide0
    1868-186f : ide1
1880-189f : Intel Corp. 82801DB/DBM SMBus Controller
18c0-18ff : Intel Corp. 82801DB AC'97 Audio Controller
    18c0-18ff : Intel ICH4
1c00-1cff : Intel Corp. 82801DB AC'97 Audio Controller
    1c00-1cff : Intel ICH4
2000-203f : Intel Corp. 82801BD PRO/100 VE (LOM) Ethernet Controller
    2000-203f : e100
```

Порты нумеруются при помощи шестнадцатиричный чисел. Без сомнения из того, что вы увидите, кое-что вам покажется знакомым, например клавиатура (keyboard), таймер (timer), параллельный порт (parallel -- принтер), последовательный порт (serial -- модем) и видеокарта (vga+). Сравните это со некоторыми стандартными ассоциациями портов ввода/вывода, показанными на Листинге 3. Следует отметить, что, например, первый параллельный порт (parport0) владеет адресами в диапазоне от 0378 до 037A, что отображено в листинге /proc/ioproports, но стандарт (LPT1) допускает использование для него диапазона от 378 до 37F.

Листинг 3. Стандартные установки для портов ввода/вывода

1F0-1F8 - Hard Drive Controller, 16-bit ISA
200-20F - Game Control
210 - Game I/O
220 - Soundcard
278-27F - LPT2
2F8-2FF - COM2
320-32F - Hard Drive Controller, 8-bit ISA
378-37F - LPT1
3B0-3BF - Monochrome Graphics Adapter (MGA)
3D0-3DF - Colour Graphics Adapter (CGA)
3F0-3F7 - Floppy Controller
3F8-3FF - COM1

Прерывания

Итак, как процессор узнает когда заканчивается последний вывод или когда появляются данные для чтения? Обычно эта информация берется из регистра статуса, который может быть доступен при чтении из одного (или нескольких) портов, связанных с устройством. Но тут возникают две очевидные проблемы. Во-первых, процессор тратит время на проверку статуса. Во-вторых, если на устройстве есть данные, которые откуда-то поступают, вроде модема, то данные должны быть своевременно считаны, в противном случае они будут перезаписаны следующей порцией данных..

Эти две проблемы: бесполезных пустых циклов процессора и уверенности в своевременном считывании или записи данных решается посредством концепции прерываний. Прерывания также называются Запросы на прерывание или IRQ. Если с устройством происходит что-то о чём должен знать процессор, устройство вызывает прерывание и процессор временно останавливается, чтобы он не делал в настоящий момент.

Вспоминая материал предыдущего раздела, вряд ли будет удивительно, что информация о прерываниях также хранится в файловой системе /proc, в /proc/interrupts. Выполните команду `command cat /proc/interrupts`, чтобы увидеть результат, похожий на Листинг 4.

Листинг 4. /proc/interrupts

```
CPU0
0: 226300426      XT-PIC timer
1:  92913        XT-PIC keyboard
2:    0          XT-PIC cascade
5:    0          XT-PIC usb-uhci
8:    1          XT-PIC rtc
9: 2641134       XT-PIC ehci-hcd, eth0, Intel ICH4
10:    0         XT-PIC usb-uhci
11: 213632       XT-PIC usb-uhci
14: 1944208      XT-PIC ide0
15: 3562845      XT-PIC ide1
```

NMI: 0
ERR: 0

Теперь прерывания нумеруются при помощи десятичных чисел в диапазоне от 0 до 15. И вновь сравните свой результат со стандартным распределением прерываний для ПК, показанным в Листинге 5.

Листинг 5. Стандартные настройки IRQ

IRQ 0 - System Timer
IRQ 1 - Keyboard
IRQ 2(9) - Video Card
IRQ 3 - COM2, COM4
IRQ 4 - COM1, COM3
IRQ 5 - Available (LPT2 or Sound Card)
IRQ 6 - Floppy Disk Controller
IRQ 7 - LPT1
IRQ 8 - Real-Time Clock
IRQ 9 - Redirected IRQ 2
IRQ 10 - Available
IRQ 11 - Available
IRQ 12 - PS/2 Mouse
IRQ 13 - Math Co-Processor
IRQ 14 - Hard Disk Controller
IRQ 15 - Available

Изначально каждое устройство имеет свое собственное IRQ. Заметим, например, что в Листинге 5 IRQ5 часто используется или звуковой картой, или вторым параллельным портом (принтер). Если вам необходимы оба устройства, вы можете найти карты, которые могут быть настроены (обычно при помощи перемычек) на использование другого прерывания, например, IRQ15.

В настоящее время, PCI-устройства используют IRQ совместно, таким образом, когда что-то останавливает процессор, обработчик прерывания проверяет ему ли предназначено это прерывание и если нет, то передает следующему в цепочке. Листинги 4 и 5 не говорят нам об этом совместном использовании. Мы изучим командуггер в следующем учебнике, но сейчас мы сможем использовать ее для фильтрации вывода результата команды `dmesg` чтобы увидеть сообщения начальной загрузки о прерываниях (IRQ), как показано в Листинге 6, в котором совместно используемые прерывания мы выделили.

Листинг 6. Прерывания, обнаруженные при начальной загрузке.

```
[ian@lyrebird ian]$ dmesg | grep -i irq
PCI: Discovered primary peer bus 01 [IRQ]
PCI: Using IRQ router PIIX [8086/24c0] at 00:1f.0
PCI: Found IRQ 5 for device 00:1f.1
PCI: Sharing IRQ 5 with 00:1d.2
Serial driver version 5.05c (2001-07-08) with MANY_PORTS MULTIPORT
  SHARE_IRQ SERIAL_PCI ISAPNP enabled
ttyS0 at 0x03f8 (irq = 4) is a 16550A
ttyS1 at 0x02f8 (irq = 3) is a 16550A
PCI: Found IRQ 5 for device 00:1f.1
PCI: Sharing IRQ 5 with 00:1d.2
ICH4: not 100% native mode: will probe irqs later
ide0 at 0x1f0-0x1f7,0x3f6 on irq 14
ide1 at 0x170-0x177,0x376 on irq 15
PCI: Found IRQ 11 for device 00:1d.0
PCI: Sharing IRQ 11 with 00:02.0
usb-uhci.c: USB UHCI at I/O 0x1800, IRQ 11
PCI: Found IRQ 10 for device 00:1d.1
usb-uhci.c: USB UHCI at I/O 0x1820, IRQ 10
PCI: Found IRQ 5 for device 00:1d.2
PCI: Sharing IRQ 5 with 00:1f.1
usb-uhci.c: USB UHCI at I/O 0x1840, IRQ 5
PCI: Found IRQ 9 for device 00:1d.7
ehci-hcd 00:1d.7: irq 9, pci mem f885d000
parport0: irq 7 detected
PCI: Found IRQ 9 for device 02:08.0
PCI: Found IRQ 9 for device 02:08.0
parport0: irq 7 detected
PCI: Found IRQ 11 for device 00:02.0
PCI: Sharing IRQ 11 with 00:1d.0
PCI: Found IRQ 9 for device 00:1f.5
PCI: Sharing IRQ 9 with 00:1f.3
i810: Intel ICH4 found at IO 0x18c0 and 0x1c00, MEM 0xc0080c00 and
  0xc0080800, IRQ 9
```

DMA

Ранее мы упоминали, что связь с периферийными устройствами через порты ввода/вывода производится одним или двумя байтами одновременно. Для быстрых устройств обслуживание прерываний может поглотить большую часть возможностей процессора. Более быстрым методом является использование Direct Memory Access (Прямого доступа к памяти) или DMA, при котором несколько инструкций ввода/вывода сообщают устройству куда в оперативной памяти (ОЗУ) можно писать или откуда читать данные и затем контроллер DMA осуществляет низкоуровневое управление реальными потоками данных между ОЗУ и периферийными устройствами.

Поднимите руку те, кто догадался где можно найти информацию об используемых каналах DMA. Если вы сказали, что в `/proc/dma`, то вы правы. Выполните команду `cat /proc/dma` Чтобы увидеть нечто подобное Листингу 7.

Листинг 7. /proc/dma

```
4: cascade
```

Это все? Важно запомнить, что большинство устройств будут использовать только один из ограниченного числа DMA-каналов при реальных приеме/передаче, поэтому `/proc/dma` часто будет выглядеть практически пустым, как в нашем примере. Мы также можем просканировать сообщения загрузки для выявления DMA-совместимых устройств также, как мы делали это в предыдущем случае с IRQ. Листинг 8 показывает типичный результат.

Листинг 8. /proc/dma

```
[ian@lyrebird ian]$ dmesg | grep -i dma
ide0: BM-DMA at 0x1860-0x1867, BIOS settings: hda:DMA, hdb:pio
ide1: BM-DMA at 0x1868-0x186f, BIOS settings: hdc:DMA, hdd:DMA
hda: 312581808 sectors (160042 MB) w/8192KiB Cache,
    CHS=19457/255/63, UDMA(100)
hdc: 398297088 sectors (203928 MB) w/7936KiB Cache,
    CHS=24792/255/63, UDMA(33)
ehci-hcd 00:1d.7: enabled 64bit PCI DMA
```

Plug and play

Ранние ПК обладали фиксированным числом портов и IRQ для отдельных устройств, таких как клавиатура или параллельный порт принтера. Это осложняло добавление новых устройств или даже использование двух однотипных, вроде двух модемов или двух принтеров, первый последовательный порт обычно назывался COM1, а второй -- COM2. Linux обычно ссылается на них как `ttyS0` и `ttyS1`. Некоторые карты можно было настраивать при помощи перемычек, которые позволяли модему, например, работать по порту COM1 или COM2. По мере увеличения устройств, исходного количества выделяемых адресов под порты и прерывания стало не хватать, и была разработана технология Plug and Play или PnP. Идея заключалась в разрешении устройству сообщать системе сколько и каких ресурсов ему необходимо, а BIOS затем сообщала устройству какие из имеющихся ресурсов ему следует использовать. Это полуавтоматическое конфигурирование было представлено в IBM PS/2 который использовал шинную архитектуру, названную `microchannel`. Позднее, эта идея и имя `plug and play` было использовано и в ISA-картах, в основном в модемах и звуковых картах, которые были наиболее популярными картами-расширениями в то время. Шина PCI продвинула идею дальше и все PCI-устройства от рождения являются `plug and play`.

Если вам случится работать за компьютерами с ISA PnP устройствами, то помните, что вам следует избегать конфликтов портов и прерываний между устройствами. Порты не могут быть использованы двумя устройствами одновременно; каждое устройство должно иметь свой собственный порт. Это же имеет место и для DMA каналов. За некоторыми исключениями, ISA-устройства не могут совместно использовать и IRQ. Если у вас есть не-PnP устройства, то вы должны вручную сконфигурировать каждое устройство, чтобы оно не пересекалось с другими. PnP должно выполнять конфигурирование автоматически. Однако,

с некоторыми ISA-устройствами это не всегда хорошо работает. Вы можете разрешить конфликты, используя `isapnptools`, обсуждаемую ниже, или вы можете переназначить некоторые порты и IRQ на PnP устройствам, чтобы система заработала.

Для систем на основе ядер, предшествующих 2.4, пакет `isapnptools` позволял пользователям настраивать PnP-устройства. Команда `isapnp` считывает файл конфигурации (обычно `/etc/isapnp.conf`) для настройки PnP-устройств. Обычно это выполняется во время загрузки Linux. Команда `rnpdump` сканирует PnP-устройства и формирует список ресурсов, которые или необходимы, или предпочтительны для использования вашими PnP-картами. Формат полученного списка подходит для использования командой `isapnp`, как только вы раскомментируете команды, которые хотите использовать. Вы должны убедиться, что конфликты ресурсов устранены. Обратитесь к man-страницам команд `isapnp` и `rnpdump` для дополнительной информации по их использованию.

Начиная с ядра 2.4, поддержка PnP внедрена в ядро и пакет `isapnptools` стал не нужен. К примеру, он был удален из Red Hat 7.3, выпущенной в мае 2002. Поддержка PnP схожа с поддержкой PCI, описанной ранее. Вы можете использовать команду `lsnpnp` (часть пакета `kernel-rcmstia-cs`) для отображения информации о PnP-устройствах. Эта информация содержится в файле `/proc/bus/pnp`. Этот файл отсутствует в системах, содержащих только PCI.

Жесткие диски IDE

В современных ПК наиболее распространены жесткие диски Integrated Drive Electronics или IDE [Прим.пер.: На самом деле сейчас (середина 2006 г.) наиболее распространены диски SATA, обсуждаемые ниже, а IDE медленно исчезают]. Они также известны как AT Attachment или ATA диски, что появилось после IBM PC-AT. Другим используемым типом дисков является также достаточно популярный интерфейс Small Computer System Interface или SCSI, особенно на серверах. IDE диски имеют преимущество низкой цены, а SCSI интерфейс позволяет подключить большее количество дисков, с большим потенциалом перекрытия операций для различных дисков на одной шине и, как следствие, большей производительностью.

Недавно на рынке появился новый тип дисков, под названием Serial ATA или SATA. Спецификации SATA призваны устранить некоторые ограничения спецификаций ATA, обеспечив хорошую совместимость с ATA.

BIOS и размер IDE дисков

IDE диски разбиты на сектора (sector), единица данных в 512 байт. Жесткий диск может состоять из нескольких вращающихся дисков-пластин, так что сектора располагаются в концентрических окружностях, каждая из которых называется цилиндром (cylinder). Данные с каждого отдельного диска считываются или записываются при помощи головки (head). Чтобы найти некоторый сектор, диск перемещает набор головок, связанных с цилиндром, выбирает соответствующую головку и ждет когда требуемый сектор окажется под головкой. Отсюда возникает термин CHS (Cylinder [Цилиндр], Head [Головка] и Sector [Сектор]). Можно также услышать другое название геометрия диска.

К сожалению, исторически, ранние BIOS содержали ограничение величины каждого из параметров C, H и S, а DOS, популярная операционная система для ПК, содержала дополнительные ограничения. На протяжении 1990-ых, размеры дисков быстро превзошли искусственные ограничения CHS, накладываемые BIOS и DOS. Привлекались различные стратегии для перевода реальных значений CHS в "виртуальные", из-за чего возникали проблемы связанные либо с самой BIOS, либо с низкоуровневыми приложениями типа Ontrack Disk Manager.

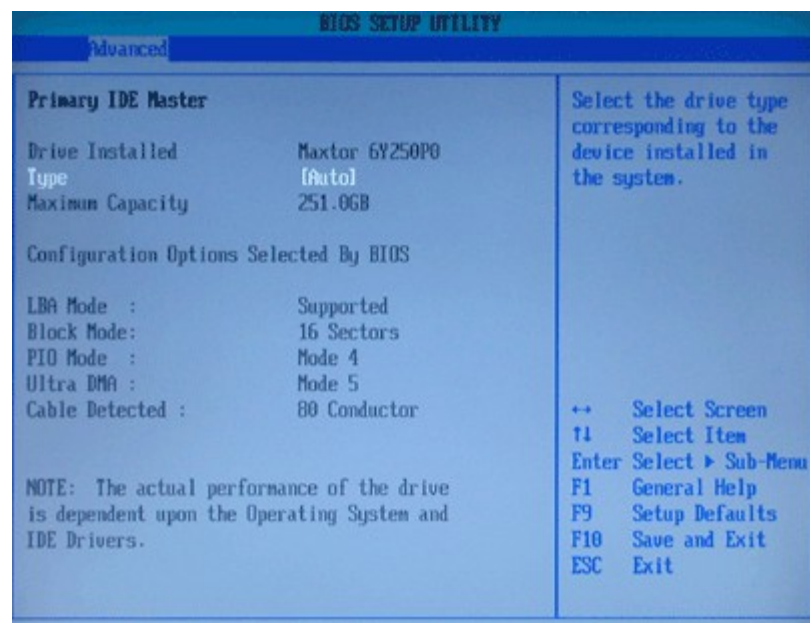
Даже без искусственных ограничений BIOS или DOS, идея CHS допускала максимум 65536 цилиндров, 16 головок, и 255 секторов/дорожек. Это ограничивает вместимость 267386880 секторами, или приблизительно 137 ГБ. Заметьте, что размер дисков, в отличие от других размеров в ПК измеряется в степенях 10, так что 1ГБ=1,000,000,000 байт.

Идея была в том, чтобы система игнорировала геометрию и оставляла ее обработку самому диску. Вместо обращения к значению CHS, система просто запрашивает Logical Block Address (Адрес Логического блока) или LBA, а электроника диска вычисляет к какому реальному сектору идет обращение на чтение/запись. Этот процесс был стандартизирован в 1996 с принятием стандарта ATA-2 (ANSI стандарт X3.279-1996, AT Attachment Interface with Extensions [Интерфейс АТ с расширениями]).

Как говорилось ранее, BIOS необходима для загрузки системы, поэтому ей необходимо знать об организации диска, чтобы загрузить стартовую программу. Старые BIOS, не понимающие LBA-диски могут иметь ограничение по загрузке с первых 1024 цилиндров диска, или, по крайней мере, с первых 1024 цилиндров, в понимании геометрии диска самой BIOS! Такие BIOS вероятно в настоящее время весьма редки, но если вам необходимо работать с одной из них, она может иметь установки для поддержки LBA, а также вам, вероятно, понадобится разместить каталог /boot в разделе в первых 1024 цилиндрах. Даже когда ваша система будет прекрасно загружаться с самого конца самого большого диска, многие инструменты разбивки диска в Linux будут предупреждать вас о том, что раздел находится за пределами первых 1024 цилиндров.

На Рисунке 3 показана информация, доступная в BIOS моей материнской платы Intel для 250ГБ IDE диска на одной из моих Linux-систем.

Рисунок 3. Вид BIOS для большого LBA-диска



В листинге 9 показана часть результата применения команды `hdparm -l /dev/hda` для диска подобного приведенному на Рисунке 3 в системе Linux (в данном случае Fedora Core 3). Заметьте, что значения CHS ограничены 4,128,705 секторами, значение LBA установлено в 268,435,455 секторов или 137ГБ. А реальный объем измеряется в единицах LBA48. Это 490,234,752 сектора или 251ГБ.

Листинг 9. Результат команды `hdparm -l /dev/hda`

`/dev/hda:`

ATA device, with non-removable media

Model Number: Maxtor 6Y250P0

Serial Number: Y638VBWE

Firmware Revision: YAR41BW0

Standards:

Supported: 7 6 5 4

Likely used: 7

Configuration:

| Logical | max | current |
|---------|-----|---------|
|---------|-----|---------|

| | | |
|-----------|-------|-------|
| cylinders | 16383 | 65535 |
|-----------|-------|-------|

| | | |
|-------|----|---|
| heads | 16 | 1 |
|-------|----|---|

| | | |
|---------------|----|----|
| sectors/track | 63 | 63 |
|---------------|----|----|

--

CHS current addressable sectors: 4128705

LBA user addressable sectors: 268435455

LBA48 user addressable sectors: 490234752

device size with M = 1024*1024: 239372 MBytes

device size with M = 1000*1000: 251000 MBytes (251 GB)

Capabilities:

LBA, IORDY(can be disabled)

Queue depth: 1

...

По умолчанию ПК загружается с первого IDE диска компьютера. Некоторые компьютеры имеют пункты BIOS, позволяющие изменить это, однако большинство загружаются именно так. Сначала компьютер загружает маленький кусочек кода из master boot record (главной загрузочной записи) которая, в свою очередь, предоставляет информацию с какого раздела грузиться. Мы рассмотрим подробнее начальные загрузчики для Linux в следующих учебниках.

Если вы хотите узнать больше об истории больших дисков, смотрите ссылку [Large Disk HOWTO](#) (Большие диски HOWTO) в Ресурсах, доступный посредством проекта Linux Documentation Project.

Имена дисков Linux

Большую часть о том, как Linux использует диски мы рассмотрим в последующих учебниках данной серии. Однако, сейчас хорошо бы представить вам другую важную файловую систему Linux -- `/dev`. Это, подобно `/proc`, псевдо-файловая система описывающая устройства, которые могут быть в Linux системах. Внутри файловой системы `/dev` вы обнаружите такие записи, как `/dev/hda`, `/dev/hda5`, `/dev/sda`, `/dev/sdb1` и так далее. Вы также найдете множество других записей для других типов устройств, но сейчас давайте взглянем на те из них, которые начинаются или с `/dev/hd`, или с `/dev/sd`.

Устройства, начинающиеся с `/dev/hd`, такие как `/dev/hda` или `/dev/hda5` ссылаются на IDE-диски. Первый диск первого контроллера IDE это `/dev/hda`, а второй, если имеется, это

/dev/hdb. Точно также первый диск второго контроллера IDE это /dev/hdc, а второй -- /dev/hdd. Как видно из Листинга 10, в /dev прописано намного больше, нежели имеется в вашей системе.

Листинг 10. Записи /dev/hd? и /dev/sd?

```
[ian@lyrebird ian]$ ls /dev/hd?
/dev/hda /dev/hdd /dev/hdg /dev/hdj /dev/hdm /dev/hdp /dev/hds
/dev/hdb /dev/hde /dev/hdh /dev/hdk /dev/hdn /dev/hdq /dev/hdt
/dev/hdc /dev/hdf /dev/hdi /dev/hdl /dev/hdo /dev/hdr
[ian@lyrebird ian]$ ls /dev/sd?
/dev/sda /dev/sde /dev/sdi /dev/sdm /dev/sdq /dev/sdu /dev/sdy
/dev/sdb /dev/sdf /dev/sdj /dev/sdn /dev/sdr /dev/sdv /dev/sdz
/dev/sdc /dev/sdg /dev/sdk /dev/sdo /dev/sds /dev/sdw
/dev/sdd /dev/sdh /dev/sdl /dev/sdp /dev/sdt /dev/sdx
```

Также как и для IRQ ранее, мы можем использовать команду dmesg, чтобы узнать какие дисковые устройства обнаружены во время загрузки, Вывод для одной из моих машин приведен в Листинге 11.

Листинг 11. Жесткие диски, обнаруженные при загрузке

```
[ian@lyrebird ian]$ dmesg | grep "[hs]d[a-z]"
Kernel command line: ro root=LABEL=RHEL3 hdd=ide-scsi
ide_setup: hdd=ide-scsi
   ide0: BM-DMA at 0x1860-0x1867, BIOS settings: hda:DMA, hdb:pio
   ide1: BM-DMA at 0x1868-0x186f, BIOS settings: hdc:DMA, hdd:DMA
hda: WDC WD1600JB-00EVA0, ATA DISK drive
hdc: Maxtor 6Y200P0, ATA DISK drive
hdd: SONY DVD RW DRU-700A, ATAPI CD/DVD-ROM drive
hda: attached ide-disk driver.
hda: host protected area => 1
hda: 312581808 sectors (160042 MB) w/8192KiB Cache,
    CHS=19457/255/63, UDMA(100)
hdc: attached ide-disk driver.
hdc: host protected area => 1
hdc: 398297088 sectors (203928 MB) w/7936KiB Cache,
    CHS=24792/255/63, UDMA(33)
hda: hda1 hda2 hda3 hda4 < hda5 hda6 hda7 hda8 hda9 hda10 hda11 >
hdc: hdc1 < hdc5 hdc6 hdc7 hdc8 >
hdd: attached ide-scsi driver.
```

Из Выделенных строк Листинга 11, мы можем видеть, что в системе имеется два IDE-диска (hda и hdc), а также привод DVD-RW (hdd). Заметьте, hdb нет, что говорит о том, что на первом IDE-контроллере второго диска нет. Диск может иметь четыре основных раздела (primary) и неограниченное количество логических (logical). Рассмотрев диск hdc Листинга 11, мы можем увидеть, что он имеет один primary-раздел (hdc1) и четыре логических (hdc5, hdc6, hdc7, и hdc8). В теме 104 в последующем учебнике этой серии мы увидим, что на самом деле hdc1 это контейнер (или расширенный (extended) раздел) для логических разделов.

Исторически такие устройства, как sda и sdb являлись SCSI-дисками, которые мы рассмотрим далее, при изучении настройки SCSI-устройств для ядра 2.4, IDE CD и DVD устройства обычно управляются через эмуляцию SCSI. Такие устройства часто появляются в /dev похожие на /dev/cdrom, что является символьной ссылкой на эмулируемое SCSI-устройство. Для описанной системы, Листинг 12 показывает, что /dev/cdrom это ссылка на /dev/scd0, а не на /dev/hdd, как можно было бы ожидать. Заметьте, что параметр ядра hdd=ide-scsi в Листинге 11 является указанием, что ide-scsi привод был присоединен к hdd.

Листинг 12. IDE SCSI-эмуляция

```
[ian@lyrebird ian]$ ls -l /dev/cdrom
lrwxrwxrwx 1 root root 9 Jan 11 17:15 /dev/cdrom -> /dev/scd0
```

Сейчас вы убедитесь, что и USB, и SATA устройства хранения обозначаются как sd, а не hd.

Стандартная периферия

Выше мы упоминали такую периферию, как последовательный и параллельный порты, которые обычно интегрированы в материнскую плату, и рассмотрели некоторые стандартные порты ввода/вывода, а также IRQ ассоциируемые с этими устройствами. Последовательные порты, в действительности, использовались для соединения различных устройств и исторически трудно настраиваются. С появлением устройств стандарта IEEE 1394, известного также, как Firewire и Universal Serial Bus (Универсальной шины данных) или USB, автоматическая настройка и "горячее" подключение устройств повсеместно заменило рутину корректной настройки последовательных и параллельных портов. Фактически, legacy-free (свободные от наследства) системы не поддерживают стандартные последовательный и параллельный порты. В них не поддерживается ни флоппи-драйв, ни PS/2 клавиатура, ни PS/2 "мышь".

Теперь мы обсудим некоторые основные настройки BIOS, которые вам может понадобится изменить.

Serial ports (COMn) [Последовательные порты]

Стандартные последовательные порты нумеруются от COM1 до COM4. Если в вашей системе есть единственное гнездо последовательного порта (изначально 25-контактный DB25 разъем, но в настоящее время обычно 9-контактный разъем DB9), то вероятно используются стандартные для COM1 адреса и прерывания (IRQ), а именно порт ввода/вывода (IO port) 3F8 и IRQ 4. Стандартные адреса и прерывания для последовательных портов приведены в Таблице 3.

Таблица 3. Параметры последовательного порта

| Имя | Адрес | IRQ |
|------|---------|-----|
| COM1 | 3F8-3FF | 4 |
| COM2 | 2F8-2FF | 3 |
| COM3 | 3E8-3EF | 4 |
| COM4 | 2E8-2EF | 3 |

Вы можете заметить, что COM1 и COM3 совместно используют IRQ 4, а COM2 и COM4, в свою очередь, IRQ 3. Поскольку драйвер и устройство в действительности не могут совместно использовать прерывания, и устройство не может вообще не использовать прерывание, то это означает, что реальная система использует только COM1 и COM2.

Иногда, вам может понадобиться или отключить встроенный последовательный порт, или настроить его на использование другого адреса и IRQ. Наиболее вероятной причиной сделать это является конфликт между PnP модемом и ISA-слотом или желание использовать PnP-модем на COM1. Мы рекомендуем изменять это только в том случае, если Linux не может определить вашу конфигурацию.

Parallel ports (LPTn) [Параллельные порты]

Стандартные параллельные порты нумеруются от LPT1 до LPT4, хотя обычно присутствует только два. Если в вашем компьютере есть единственное гнездо параллельного порта, то он, вероятно, по умолчанию использует адрес и IRQ для LPT1, а именно порт ввода/вывода 378 и IRQ 7. Стандартные адреса портов ввода/вывода и IRQ для параллельных портов приведены в Таблице 4.

Таблица 4. Параметры параллельного порта

| Имя | Адрес | IRQ |
|------|---------|-----|
| LPT1 | 378-37F | 7 |
| LPT2 | 278-27F | 5 |
| LPT* | 3BC-3BE | |

Заметьте, что порты ввода/вывода 3BC-3BE изначально использовались графическим адаптером Hercules, который также имеет параллельный порт. Многие системы BIOS присваивают этот диапазон LPT1 и затем два других диапазона становятся LPT2 и LPT3 соответственно, вместо LPT1 и LPT2.

Многие системы не используют прерываний для принтеров, поэтому IRQ реально может использоваться, а может и нет. Не редкость также совместное использование IRQ 7 для печати и звуковой карты (совместимой с Sound Blaster).

Параллельные порты изначально обычно использовались для печати с данными, поступающими на принтер и несколькими линиями, зарезервированными для отчета о статусе. Позднее, параллельные порты использовались для подключения различных устройств (включая ранние CD-ROM и ленточные приводы), в связи с чем направленность только на вывод данных сменилась двунаправленным потоком данных.

Текущий стандарт параллельных портов это IEEE Std. 1284-1994 Standard Signaling Method for a Bi-Directional Parallel Peripheral Interface for Personal Computers (Стандарт Метода Передачи сигналов для Двунаправленного Параллельного Интерфейса Периферии Персонального Компьютера), который определяет пять сигнальных режимов. Ваш BIOS во время настройки может предоставить вам на выбор один из них bi-directional (двунаправленный), EPP, ECP и EPP and ECP. ECP расшифровывается как Enhanced Capabilities Port (Порт с Расширенными Возможностями) и разработан для использования с принтерами. EPP расшифровывается как Enhanced Parallel Port (Улучшенный Параллельный Порт) и разработан для таких устройств, как CD-ROMы и ленточный приводы, которым необходим значительный поток данных в обоих направлениях. Выбор BIOS по умолчанию скорее всего это ECP. Как и для последовательных портов, изменять это следует только если у вас есть устройство, не функционирующее должным образом.

Порт Floppy дисковод

Если в вашей системе присутствует стандартный контроллер floppy диска, то он использует порты 3F0-3F7. Если вы установите стандартный floppy-дисковод в компьютер, который продавался без него, то вам необходимо будет включить соответствующие опции в BIOS. Просмотрите документацию производителя для выяснения деталей.

Клавиатура и "мышь"

Контроллер клавиатура/"мышь" использует порты 0060 и 0064 для стандартной клавиатуры и "мыши", подсоединенных к гнезду PS/2. Многие системы выдают ошибку Power-On-Self-Test (POST) [Самотестирование При Включении], если клавиатура не подключена. Большинство машин разработанных для использования в качестве серверов и многие настольные в настоящее время имеют опции BIOS для нормального запуска без клавиатуры.

Работать на компьютере без клавиатуры (или "мыши") проблематично. Но сервера часто работают именно так. Управление осуществляется посредством сети используя инструменты web-администрирования или интерфейс командной строки, такой как telnet или (предпочтительнее) ssh.

Установка безклавиатурной системы обычно подразумевает использование терминала (или эмулятора терминала), подсоединенного к последовательному порту. Обычно вам необходима клавиатура и монитор, чтобы убедиться, что BIOS настроена правильно и последовательный порт включён. Также вам может потребоваться сформированный загрузочный диск или CD для осуществления установки Linux.

Другой подход используется в системах JS20 blade server это эмуляция последовательного соединения в сети.

Модемы и звуковые карты

Модемы

Модем (modem) (образовано от modulator/demodulator) это устройство для преобразования цифровых сигналов, используемых компьютером в последовательные потоки аналоговых данных, передаваемых по телефонной линии. В первые дни ПК, модемы были внешними устройствами, которые подсоединялись к последовательному порту. Позднее, модемы стали выпускаться в виде плат, которые могли быть вставлены внутрь системного блока, что снижало стоимость размещения и подключения к сети, ликвидируя необходимость в кабеле между последовательным портом и модемом. Следующим снижением стоимости стало переложение некоторых функций, обычно выполнявшихся модемом, на программное обеспечение ПК. Модемы такого типа можно назвать, в зависимости от терминологии, soft-модемам, HCF модемами, HSP модемами, HSF модемами или модемами без контроллера (controllerless modem). Такие модемы были разработаны для снижения стоимости систем, которые в основном работают под Microsoft Windows. Термин win-модем часто используется в отношении этих устройств, хотя Win-модем® является зарегистрированной торговой маркой U.S. Robotics, которая производит различные модемы под этим именем.

Большинство внешних и полнофункциональных внутренних модемов будут работать без проблем и в Linux. Некоторые из модемов, требующих помощи программного обеспечения Операционной системы ПК, также будут работать в Linux и список модемов этой категории постоянно пополняется. Soft-модемы, работающие в Linux часто называют lin-модемами и есть сайт, посвященный им (linmodems.org). Если у вас такой модем, то в начале следует посетить указанный сайт lin-модемов (смотри Ресурсы) и загрузить последнюю версию инструмента scanmodem. Она сообщит все что известно о доступном драйвере (если он есть) для вашего модема.

Если у вас ISA-модем, то вам нужно убедиться, что порты, IRQ и DMA каналы не

конфликтуют с другими устройствами. Для дополнительной информации смотри предыдущий раздел Настройки BIOS.

Модемы, рассматриваемые в данном разделе являются асинхронными модемами. Есть также другой класс модемов, называемых синхронными модемами, использующихся для HDLC, SDLC, BSC или ISDN. Упрощенно, можно сказать, что асинхронная передача применяется для передачи отдельных байтов информации, а синхронная связь используется для передачи блоков информации.

Большинство соединений Linux используют Internet Protocol или IP. Поэтому Linux-системам необходимо нечто вроде IP для асинхронных линий, которые изначально разрабатывались не для блочных протоколов вроде IP. Первый вариант реализации этого назывался Serial Line Interface Protocol (Протокол интерфейса последовательной линии) или SLIP. Его вариант, использующий сжатые заголовки, назвали CSLIP. В наши дни, Большинство Internet-провайдеров (ISP -- Internet Service Provider) поддерживают dialup-соединения с использованием Point-to-Point Protocol или PPP.

Linux Networking-HOWTO и Руководство сетевого администратора доступные в проекте Linux Documentation Project (смотри Ресурсы) предоставляют информацию о настройке SLIP, CSLIP и PPP.

При связи посредством модема, существует множество параметров настройки, которые вам возможно придется изменить на вашем компьютере с Linux. Наиболее важно установить скорость соединения системы с модемом. Обычно она выбирается больше чем номинальная скорость соединения в линии, и часто устанавливается максимально возможное значение для последовательного порта и модема. Одним из способов настроить или посмотреть параметры модема, используемые драйвером последовательного порта, является программа `setserial`. Команда `setserial` проиллюстрирована Листингом 13. Заметьте, что параметр `-G` приводит к форматированному выводу, удобному для применения в качестве установочных параметров с `setserial`. В данном случае, UART (Universal Asynchronous Receiver Transmitter) имеет буфер 16550, что является стандартом для UART в современных ПК. Установлена скорость 115,200 bps, которая также обычно используется с этим UART для большинства современных внешних модемов 56kbps. Следует отметить, что скорость по умолчанию на некоторых более новых системах может быть установлена вплоть до 460,800bps. Если ваш модем не откликается, то это первое, что вы должны проверить.

Листинг 13. Команда setserial

```
[root@attic4 ~]# setserial /dev/ttyS0
/dev/ttyS0, UART: 16550A, Port: 0x03f8, IRQ: 4
[root@attic4 ~]# setserial -G /dev/ttyS0
/dev/ttyS0 uart 16550A port 0x03f8 irq 4 baud_base 115200 spd_normal skip_test
```

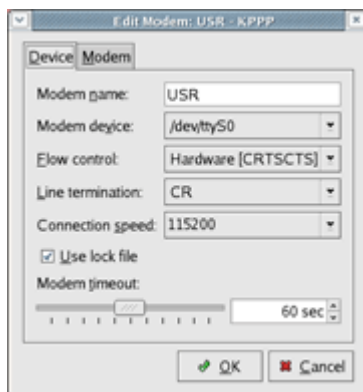
Единственное, что следует сказать особо о `setserial` так это то, что она не обращается к самому устройству. Все, что она выводит -- это параметры, используемые драйвером последовательного порта, если только вы не используете параметры `autoconfig` и `auto_irq` parameters. В этом случае `setserial` попросит ядро опросить устройство. Для более полной информации об этих и других параметрах команды смотрите man-страницы `setserial`.

Мы рассмотрим сеть более подробно в учебнике для экзамена LPI 102 (Смотри Ресурсы). А между тем если вы хотите настроить PPP-соединение, то имеется несколько превосходных инструментов, которые помогут вам сделать это. Программа `kppp` имеет приятный графический интерфейс и проста в использовании. Команда `wvdial` является хорошим инструментом командной строки для настройки dialup соединений. В добавок к этому, дистрибутивы могут иметь другие инструменты или специально для PPP или dialup

соединений, или как часть более общих инструментов настройки сети, такие как `system-config-network` в Fedora Core 4.

Другим аспектом модемных соединений, который также обычно контролируется коммуникационной программой, но может быть изменён или иметь значение по умолчанию записанное в самом модеме, является `flow control` (управление потоком данных). Он определяет способ, при помощи которого один модем передаёт другому команду подождать, пока не будет очищен буфер получения данных. Такая команда может выдаваться программным обеспечением путем отправки символов `XON` и `XOFF`, но более хороший способ, используемый также для PPP-соединений, называется `hardware flow control` (аппаратное управление потоком данных) при котором для индикации готовности к приему данных используются значения сигнальной линии модема. Используемые сигналы называются `Clear to Send` (Нечего передавать) или `CTS` и `Ready to Send` (Готов к передаче) или `RTS`, так что вы часто будете видеть их или нечто подобное при описании управления потоком данных с использованием `RTS/CTS`. На Рисунке 4 показано как настроить скорость и аппаратное управление потоком данных с использованием программы `kppp`.

Рисунок 4. Настройка параметров модема в `kppp`



Звуковые карты

Большинство персональных компьютеров, продаваемых в настоящее время содержат звуковую карту.

Звуковой порт (Sound Blaster)

Серия звуковых карт The Creative Labs Sound Blaster series стала *de facto* промышленным стандартом для звуковых карт. Хотя существует множество превосходных звуковых карт других брендов, большинство из них предоставляют режим совместимости для одной или нескольких серий Sound Blaster. Оригинальная карта Sound Blaster была 8-битной и работала в оригинальном IBM PC. Позднее 16-битные модели для PC-AT и совместимые с ними использовали 16-битную шину PC-AT или ISA. Сейчас, большинство таких карт используют шину PCI. Многие материнские платы содержат даже встроенный чип, совместимый с Sound Blaster. Звуковые устройства могут быть также подсоединены через USB, хотя мы не будем рассматривать их.

Порты, используемые ISA-картой Sound Blaster это 0220-022F, хотя часто можно было выбрать базовый адрес 240, 260 или 280. Подобно этому, IRQ обычно выбирались из стандартного набора 2, 5, 7, или 10. По умолчанию использовалось IRQ 5. Также обычно можно было настраивать DMA каналы.

Как и для всех ISA-устройств, вы должны убедиться, что порты, IRQ и DMA каналы не

конфликтуют с другими устройствами. Для дополнительной информации смотри предыдущий раздел Настройки BIOS.

MIDI порт (MPU-401)

Многие звуковые карты также имеют интерфейс для подключения устройства MIDI (от Musical Instrument Digital Interface -- Цифровой Интерфейс Музыкальных Инструментов). Вообще этот интерфейс эмулирует Roland MPU-401. Стандартные порты, используемые ISA-интерфейсом MPU-401 это 0200-020F.

Как и для всех ISA-устройств, вы должны убедиться, что порты, IRQ и DMA каналы не конфликтуют с другими устройствами. Для дополнительной информации смотри предыдущий раздел Настройки BIOS.

Настройка поддержки звука в Linux

Современные ядра 2.4 и 2.6 имеют поддержку звука для огромного числа разнообразных звуковых устройств, встроенную в ядро, обычно в качестве модуля. Как и для других устройств, мы можем использовать команду `rpndump` для ISA-устройств, или команду `lspci` для PCI устройств, чтобы вывести информацию об устройстве. Листинг 14 содержит вывод команды `lspci` для звуковой системы от Intel (Intel sound system), встроенной в материнскую плату.

Листинг 14. Использование `lspci` для отображения звуковых ресурсов

```
[root@lyrebird root]# lspci | grep aud
00:1f.5 Multimedia audio controller: Intel Corporation 82801DB/DBL/DBM
      (ICH4/ICH4-L/ICH4-M) AC'97 Audio Controller (rev 01)
```

Модули ядра -- это предпочтительный способ для обеспечения поддержки различных устройств. Необходимо только загрузить модули, соответствующие устройствам реально присутствующим в системе, причем они могут выгружаться и подгружаться без перезагрузки системы. Для ядра версии 2.4 и более ранних, информация о конфигурации ядра хранится в `/etc/modules.conf`. Для ядер 2.6, система модулей ядра была изменена и теперь информация хранится в `/etc/modprobe.conf`. В любом случае, команда `lsmod` отформатирует содержимое `/proc/modules` и отобразит состояние загруженных модулей.

В Листинге 15 приведено содержимое `/etc/modprobe.conf` для ядра 2.6, а Листинг 16 содержит вывод команды `lsmod`, связанный со звуковыми устройствами этой системы.

Листинг 15. Пример `/etc/modprobe.conf` (Ядро 2.6)

```
[root@attic4 ~]# cat /etc/modprobe.conf
alias eth0 e100
alias snd-card-0 snd-intel8x0
install snd-intel8x0 /sbin/modprobe --ignore-install snd-intel8x0 &&\
  /usr/sbin/alsactl restore >/dev/null 2>&1 || :
remove snd-intel8x0 { /usr/sbin/alsactl store >/dev/null 2>&1 || : ; } \
  /sbin/modprobe -r --ignore-remove snd-intel8x0
alias usb-controller ehci-hcd
alias usb-controller1 uhci-hcd
```

Листинг 16. Вывод команды `lsmod`, связанный со звуком (Ядро 2.6)

```
[root@attic4 ~]# lsmod | egrep '(snd)|(Module)'
Module                Size Used by
snd_intel8x0          34689  1
snd_ac97_codec        75961  1 snd_intel8x0
snd_seq_dummy         3653  0
snd_seq_oss           37057  0
snd_seq_midi_event    9153  1 snd_seq_oss
snd_seq               62289  5 snd_seq_dummy,snd_seq_oss,snd_seq_midi_event
snd_seq_device        8781  3 snd_seq_dummy,snd_seq_oss,snd_seq
snd_pcm_oss           51185  0
snd_mixer_oss         17857  1 snd_pcm_oss
snd_pcm              100169  3 snd_intel8x0,snd_ac97_codec,snd_pcm_oss
snd_timer             33605  2 snd_seq,snd_pcm
snd                   57157  11 snd_intel8x0,snd_ac97_codec,snd_seq_oss,
snd_seq,snd_seq_device,snd_pcm_oss,snd_mixer_oss,snd_pcm,snd_timer
soundcore             10913  1 snd
snd_page_alloc        9669  2 snd_intel8x0,snd_pcm
```

В Листинге 17 приведено содержимое `/etc/modules.conf` для ядра 2.4, а Листинг 18 содержит вывод команды `lsmod`, связанный со звуковыми устройствами этой системы. Заметьте, что файлы `modules.conf` и `modprobe.conf` схожи.

Листинг 17. Пример `/etc/modules.conf` (Ядро 2.4)

```
[root@lyrebird root]# cat /etc/modules.conf
alias eth0 e100
alias usb-controller usb-uhci
alias usb-controller1 ehci-hcd
alias sound-slot-0 i810_audio
post-install sound-slot-0 /bin/aumix-minimal -f /etc/.aumixrc -L >/dev/null 2>&1 || :
pre-remove sound-slot-0 /bin/aumix-minimal -f /etc/.aumixrc -S >/dev/null 2>&1 || :
```

Листинг 18. Вывод команды `lsmod`, связанный со звуком (Ядро 2.4)

```
Module                Size Used by Not tainted
smbfs                 43568  1 (autoclean)
i810_audio            28824  0 (autoclean)
ac97_codec            16840  0 (autoclean) [i810_audio]
soundcore             6436  2 (autoclean) [i810_audio]
st                   30788  0 (autoclean) (unused)
```

Поддержка звука на многих системах 2.4 и более ранних обеспечивается благодаря драйверам Open Sound System (OSS) Free. Сегодня многие системы используют драйвера Advanced Linux sound architecture или ALSA. Утилита `sndconfig` была создана Red Hat для конфигурирования ISA PnP звуковых карт. Она также работает и с PCI звуковыми картами. Эта утилита может присутствовать и в системах, не использующих драйвера ALSA, хотя

поддержка современных модулей сделала ее практически не нужной. Эта утилита опрашивает звуковую карту, воспроизводя речь Линуса Торвальдса (Linus Torvalds) в качестве теста, а затем обновляет файл /etc/modules.conf. Типичная операция показана на Рисунках 5 и 6.

Рисунок 5. Утилита *sndconfig*

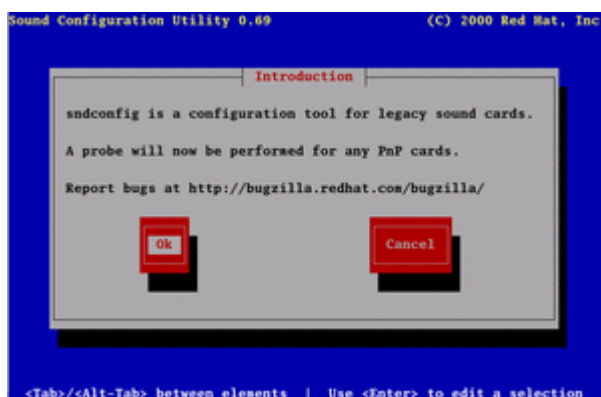
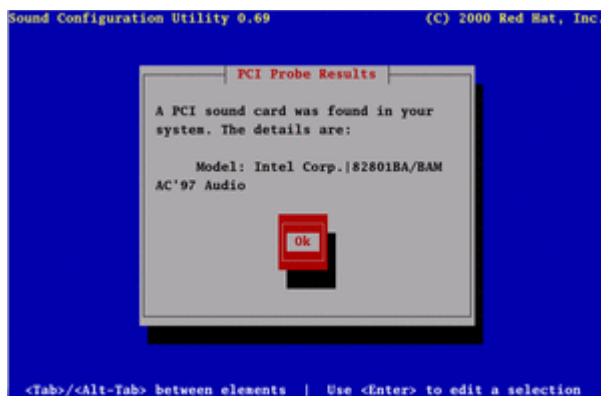


Рисунок 6. Утилита *sndconfig*



Настройка SCSI-устройств

Обзор SCSI

Small Computer System Interface (Системный Интерфейс Малых Компьютеров), больше известный как SCSI, это интерфейс, разработанный для соединения потоковых устройств, таких как ленточные и блочные устройства хранения типа дисков, CD-ROM или DVD приводов. Он также используется для других устройств, таких как сканеры и принтеры. SCSI произносится как "скази". SCSI был разработан для размещения нескольких устройств на одной шине. Одно устройство, называемое контроллер отвечает за управление шиной. SCSI-устройства могут быть как внутренними, так и внешними.

Имеется три главных версии стандартов SCSI от Американского Института Национальных Стандартов (American National Standards Institute -- ANSI).

SCSI

это оригинальный стандарт (X3.131-1986), сейчас обычно называется SCSI-1. Он появился благодаря усилиям Shugart Associates в создании стандартного интерфейса подключения

дисковых устройств. Этот стандарт поддерживал до 8 устройств на одном кабеле. SCSI-1 использует пассивную оконечную схему [passive termination] (более подробно об этом далее). Это стандарт в настоящее время исчез, хотя SCSI-1 устройства могут все еще работать на современных SCSI кабелях, предполагающих соответствующую оконечную схему. Интерфейс данных был параллельным и 8-битным с максимальной скоростью передачи 5 МБ/с (Мегабайт/сек.). Стандарт SCSI был разработан для дисков, но был очень гибок и использовался для других устройств, преимущественно сканеров и медленных устройств, таких как Zip[™]). FConnection использовал 50-ти жильный кабель, обычно Centronics, а позднее с 50-контактный D-shell, похожий на DB-25 RS-232 последовательного соединения,

SCSI-2

был принят как стандарт ANSI X3.131-1994 в 1994. Эта версия удвоила скорость шины до 10МБ/с, а также ввела так называемую широкую или 16-битную передачу данных. 16-битная шина работая на скорости 10МБ/с могла передавать 20МБ/с данных. Для 8-битных или узких SCSI-устройств использовался 50-жильный кабель, а для новых широких устройств -- 68-жильный. Была также повышена плотность кабеля, что позволило использовать более миниатюрные и дешевые соединители. SCSI-2 также стандартизировал набор команд SCSI и ввел отдельную передачу сигнала (differential signaling) для улучшения качества на высоких скоростях. Позднее это было названо Передача сигналов с High Voltage Differential (Высоковольтным разделением) или HVD. HVD требует активной оконечной схемы. 8-битные и 16-битные устройства можно подсоединять к одному и тому же кабелю, если позаботиться о соответствующей оконечной схеме. SCSI-2 поддерживает до 16 устройств на одном кабеле из которых по крайней мере 8 могут быть узкими.

SCSI-3

это набор стандартов, а не один стандарт. Это позволяет улучшать стандарты для быстроизменяющихся технологических областей, избегая необходимости пересмотра стандартов стабильной технологии. Итоговая архитектура определяется стандартом ANSI X3.270-1996, который также известен как SCSI-3 Architecture Model или SAM. Ранние стандарты SCSI теперь преобразованы в стандарты SCSI Parallel Interface (Параллельный интерфейс SCSI) или SPI. Скорость была вновь увеличена и современные 16-битные устройства способны передавать данные со скоростью до 320МБ/с при скорости шины 160МБ/с.

SCSI-3 ввел Оптоволоконные каналы SCSI (Fiber Channel SCSI) с поддержкой до 126 устройств на одну шину, поддерживающей соединение свыше 1ГБ/с или 2ГБ/с по оптоволоконным каналам на расстояния в несколько километров. Это помогает смягчить имеющиеся ограничения, связанные с использованием стандартных SCSI-кабелей. Другим важным нововведением было Single Connector Attachment или SCA, используемое только для широких (16-битных) устройств. SCA это 80-контактное соединительное звено, которое включало контакты от 68-контактного звена, а также питание и некоторые дополнительные контакты. SCA разрабатывалась для безопасного горячего подключения устройств в работающей системе и часто используется в устройствах объединенных в систему хранения Redundant Array of Independent disks (Избыточный массив независимых дисков) или RAID, а также в сетевых устройствах хранения и серверных стойках.

Выше мы упомянули оконечную схему, не объяснив что это такое. Электрические спецификации шины SCSI требуют, чтобы каждый конец шины был корректно завершен. Вы должны использовать соответствующий тип терминатора для вашей шины: пассивный, HVD или LVD. Если вы смешиваете широкие и узкие устройства, то будьте внимательны, поскольку терминатор для узких устройств может располагаться в месте отличном от терминатора для широких устройств. Если контроллер работает только с внутренней шиной или только с внешней, то он обычно обеспечивается терминатором, автоматическим или настраиваемым через BIOS. Просмотрите руководство к вашему конкретному контроллеру. Если контроллер управляет обоими и внешним и внутренним сегментами, то обычно он не должен оборудоваться терминатором.

Некоторые устройства способны играть роль терминатора, что выражается перемычками или переключателями. И вновь обратитесь к руководству вашего устройства. В противном случае завершение (termination) обычно реализуется оконечным блоком, подсоединяемым к кабелю. Какой бы тип терминатора вы не использовали, будьте очень осторожны если вы совместно используете и широкие и узкие устройства на одной шине, поскольку завершение для узких и широких устройств может оказаться в разных местах кабеля.

SCSI ID (Идентификация)

Теперь вы можете удивиться как же система управляет таким количеством устройств на одном кабеле. Каждое устройство, включая контроллер имеет свой ID, выражаемый числом. Для узких (8-битных) SCSI, ID-номера находятся в диапазоне от 0 до 7. Широкие SCSI добавляют номера от 8 до 15. Узкие устройства могут использовать только номера от 0 до 7, в то время как широкие могут использовать номера от 0 до 15. Контроллеру обычно присваивается номер 7. ID устройств могут назначаться при помощи перемычек, переключателей или при помощи циферблата на устройстве, а также программно. Устройствам, использующим Single Connector Attachment (SCA), ID обычно присваиваются автоматически, поскольку эти устройства могут подключаться во время работы.

Устройства на SCSI шине имеют приоритет. Приоритет для узких устройств изменяется от 0 (наинизший) до 7 (наивысший), так что контроллер с адресом 7 имеет наивысший приоритет. Дополнительные ID для широких устройств имеют приоритет от 8 (наинизший) до 15 (наивысший), но 15 имеет приоритет меньше 0. Поэтому реальная шкала приоритета выглядит так 8, 9, 10, 11, 12, 13, 14, 15, 0, 1, 2, 3, 4, 5, 6, 7. Более медленные устройства и устройства, которые не могут допускать задержек (такие как записывающие CD или DVD приводы) должны иметь высокоприоритетные ID, чтобы гарантированно иметь значительные ресурсы.

Устройства, подобные RAID контроллерам могут выделять единственный ID для шины, но могут объединять несколько дисков. В добавок к ID, адресация SCSI допускает а Logical Unit Number (Логическую Нумерацию Устройств) или LUN. Ленточные и отдельные дисковые приводы или не сообщают LUN, или выдают LUN равный 0.

SCSI адаптер может поддерживать более одного кабеля или канала, а также в системе может быть несколько SCSI-адаптеров. Таким образом, полный ID устройства состоит из номера адаптера, номера канала, ID устройства и LUN.

Такие устройства как пишущие CD приводы использующие ide-scsi эмуляцию и USB устройства хранения также будут появляться словно имеют свой собственный адаптер.

Имена и файлы Linux для SCSI устройств

Вернемся к разделу о BIOS когда мы обсуждали имена, присваиваемые Linux IDE приводам, такие как /dev/hda и /dev/hdc. Для IDE контроллера, который может поддерживать один или два жестких диска это просто. Второй IDE привод на втором адаптере это всегда /dev/hdd, даже если еще одним диском является ведущий (primary) на первом адаптере (/dev/hda). Для SCSI ситуация становится более сложной, поскольку мы можем подсоединить к одному кабелю жесткие диски, ленточные устройства, CD и DVD приводы, а также другие устройства.

Linux присваивает имена устройствам по мере того, как они обнаруживаются во время загрузки. Поэтому первый жесткий диск на первом канале первого адаптера станет /dev/sda, второй -- /dev/sdb, и так далее. Первое ленточное устройство будет /dev/st0, второе -- /dev/st1, и так далее. Первое CD-устройство станет /dev/sr0 или /dev/scd0, а второе -- /dev/sr1 или /dev/scd1. Устройствам, использующим эмуляцию SCSI, вроде USB устройств

хранения и (вплоть до ядра 2.6) IDE CD или DVD приводам будут также выделяться имена в пространстве имен.

Хоть мы и не хотим полностью разбираться во всех сложностях со SCSI именованием, очень важно помнить, что эта нумерация производится заново при каждой перезагрузке. Если вы добавляете или удаляете жесткий диск SCSI, то все приводы выше него при последующей перезагрузке получают другие имена. То же происходит и с устройствами других типов. В другом учебнике этой серии мы более подробно изучим разбиение дисков, метки и файловые системы, но сейчас мы хотим предостеречь вас от одной вещи. Поскольку диски могут иметь до 15 разделов, каждый из которых имеет имя, связанное с именем устройства (например, /dev/sda1, /dev/sda2 и так далее до /dev/sda15), это может стать причиной путаницы, когда ваша система попытается смонтировать файловые системы. Очень тщательно планируйте добавление нового или удаление имеющегося SCSI устройства и по возможности вместо имен устройств используйте метки SCSI дисков.

Мы познакомились с файловой системой /proc в разделе Настройки BIOS . Файловая система /proc также содержит информацию о SCSI устройствах. В Листинге 19 приведено содержимое /proc/scsi/scsi для системы с двумя SCSI устройствами: жестким диском с ID 0 и контроллером с ID 8.

Листинг 19. /proc/scsi/scsi

```
[root@waratah root]# cat /proc/scsi/scsi
Attached devices:
Host: scsi1 Channel: 00 Id: 00 Lun: 00
  Vendor: IBM-PSG  Model: DPSS-336950M  F  Rev: S94S
  Type:   Direct-Access              ANSI SCSI revision: 03
Host: scsi1 Channel: 00 Id: 08 Lun: 00
  Vendor: IBM      Model: YGLv3 S2       Rev: 0
  Type:   Processor                  ANSI SCSI revision: 02
```

Если вы хотите узнать какое реальное устройство соответствует скажем /dev/sda, то вы можете использовать команду scsi_info. Листинг 20 подтверждает, что наш первый (и единственный) SCSI жесткий диск это /dev/sda.

Листинг 20. Команда scsi_info

```
[root@waratah root]# scsi_info /dev/sda
SCSI_ID="0,0,0"
MODEL="IBM-PSG DPSS-336950M  F"
FW_REV="S94S"
```

Однако, заметьте, что некоторые системы, такие как Fedora Core 2, не содержат команды scsi_info (являющейся частью пакета kernel-pcmcia-cs).

Более поздние системы используют драйвер SCSI Generic или sg (Универсальный драйвер). При использовании sg драйвера вы сможете найти дополнительную информацию в ветке /proc/scsi/sg вашей файловой системы. Вы также будете иметь устройства вроде /dev/sg0, /dev/sg1, /dev/sg2 и так далее. Универсальные устройства обычно соответствуют другим типам устройств, типа жесткого диска как /dev/sda или ленты вроде /dev/st0.

Пакет sg3_utils содержит несколько утилит для манипулирования и определения параметров подсистем SCSI. В действительности, команда sg_map выводит таблицу соответствий (map) sg-имен и других имен устройств если они существуют. Заметьте, что сканеры не имеют другого имени, только универсальное. Листинг 21 содержит результат выполнения sg_map в

системе с оптическим диском IDE, который использует SCSI эмуляцию и двумя USB-дисками.

Листинг 21. Команда sg_map

```
[root@lyrebird root]# sg_map
/dev/sg0 /dev/scd0
/dev/sg1 /dev/sda
/dev/sg2 /dev/sdb
```

Для sg, соответствующая scsi_info утилита называется sginfo. Вы можете использовать либо универсальное имя устройства, либо более знакомое имя от sginfo. Листинг 22 содержит вывод команды sginfo для трех устройств Листинга 21. Заметьте, что sginfo не предоставила информацию о /dev/sg1, хотя как видно из листинга, команда scsi_info показывает его как USB-диск. В данном случае устройство было извлечено из системы. Информация о нем осталась (и ее можно найти в /proc/scsi/scsi). Команда sginfo для получения информации опрашивает устройства, в то время как scsi_info использует связанную информацию. Поэтому sginfo должна выполняться из под root, а scsi_info не требует этого, хотя не-root пользователям может потребоваться указать полный путь /sbin/scsi_info.

Листинг 22. Команда sginfo

```
[root@lyrebird root]# sginfo /dev/scd0
INQUIRY response (cmd: 0x12)
-----
Device Type          5
Vendor:              SONY
Product:             DVD RW DRU-700A
Revision level:      VY08
```

```
[root@lyrebird root]# sginfo /dev/sg1
INQUIRY response (cmd: 0x12)
-----
Device Type          0
Vendor:
Product:
Revision level:
```

```
[root@lyrebird root]# sginfo /dev/sg2
INQUIRY response (cmd: 0x12)
-----
Device Type          0
Vendor:              WD
Product:             2500JB External
Revision level:      0411
```

```
[root@lyrebird root]# scsi_info /dev/sg1
SCSI_ID="0,0,0"
MODEL=" USB DISK 12X"
FW_REV="2.00"
```

SCSI BIOS и последовательность загрузки.

В то время как SCSI является стандартом для большинства серверов, многие настольные компьютеры и ноутбуки обычно не поддерживают SCSI. Такие системы обычно загружаются с флоппи дисков, CD или DVD приводов или первого жесткого диска IDE в компьютере. Порядок загрузки обычно настраивается в окне настройки BIOS, так как мы видели в разделе Настройки BIOS, и иногда динамически при помощи нажатия клавиш или их комбинаций во время старта системы.

Загрузочная спецификация BIOS (смотри Ресурсы) определяет метод добавления карт, таких как SCSI-карты, выводит сообщение при включении и вызывает BIOS карты для ее конфигурирования. SCSI карты обычно используют ее для настройки подсистем SCSI, управляемых картой. Например, карта Adaptec AHA-2930U2 выводит сообщение

Press <Ctrl><A> for SCSISelect (TM) Utility!

(Нажмите Ctrl+A для запуска утилиты SCSISelect), позволяющее пользователю, нажав одновременно клавиши ctrl и A, войти в BIOS адаптера. Другие карты имеют сходную процедуру входа в BIOS карты для ее настройки.

Оказавшись в BIOS карты, вы увидите экраные страницы, при помощи которых часто можно настроить адрес SCSI контроллера (обычно 7), загрузочное SCSI устройство (обычно ID 0), скорость шины и должен ли контроллер обеспечивать оконечную схему или нет. Некоторые старые карты могут потребовать, чтобы загрузочное устройство имело ID 0, но большинство современных карт позволяют вам выбрать любое устройство. Вы можете, и вероятно захотите, настроить и другие параметры, такие как возможность форматирования жесткого диска. За подробностями обратитесь к документации производителя вашей карты. После настройки поведения шины SCSI вы обычно должны еще указать BIOS ПК на необходимость загрузки со SCSI диска, а не IDE. Проконсультируйтесь с руководством к вашей системе для определения можете ли вы загрузиться с не-IDE диска и как это можно настроить.

Платы расширения ПК

Материал, который вы должны знать для данного раздела, был изложен при обсуждении Настроек BIOS . Вам следует вновь просмотреть информацию о DMA, IRQ, портах и различных типах шин и адаптеров в разделе Шины, порты, IRQ, и DMA чтобы понять содержимое файлов /proc/dma, /proc/interrupts, и /proc/ioports, а также как их использовать для определения конфликтов. Просмотрите материал о /proc/pci и команде lspci. Также повторите раздел Plug and play, чтобы знать об ISA и Plug and Play картах. Там же вы найдете информацию о isapnp и pnpdump.

Коммуникационные устройства

В этом разделе рассматриваются разнообразные коммуникационные устройства, включая модемы ISDN адаптеры и DSL коммутаторы. Материал данного раздела разбит на две основные категории:

1. Выбор и установка коммуникационного устройства, и
2. Связь с устройством

Выбор коммуникационного устройства подобен выбору любого другого устройства для вашей системы, то есть оно должно соответствовать вашему типу шины (PCI or ISA) и необходима поддержка устройства в Linux. Вам следует просмотреть обсуждение DMA, IRQ, портов и различных типов шин и адаптеров в разделе Шины, порты, IRQ, и DMA чтобы понимать содержимое файлов `/proc/dma`, `/proc/interrupts`, и `/proc/ioports`, а также как использовать их для определения конфликтов. Просмотреть материал о `/proc/pci` и команде `lspci`. Также повторите раздел Plug and play, чтобы знать об ISA и Plug and Play картах. Там же вы найдете информацию о `isapnp` и `nnpdump`.

Ядро Linux с каждой версией поддерживает все больше и больше устройств, так что, во-первых, посмотрите есть ли поддержка нужного устройства в уже используемом вами дистрибутиве. Если поддержка уже есть, то в вашем дистрибутиве уже может быть утилита, помогающая в настройке устройства. На Рисунке 7 показан инструмент настройки сети в Fedora Core 4. Вы можете видеть, что соединение ethernet уже настроено (и активно), а также настроено модемное соединение для резервного копирования по телефонной линии с использованием PPP. Система уже поддерживает подключение ISDN, Token Ring (кольцевые ЛВС), беспроводное и xDSL соединения.

Рисунок 7. Утилита настройки сети в Fedora Core



Если вы собираетесь установить драйвер для коммуникационного устройства, прежде всего проверьте является ли требуемый драйвер частью вашего дистрибутива. Если является, но еще не установлен, то установите его. В противном случае вам следует попытаться найти пакет для вашей системы с уже скомпилированным драйвером. И только если ничего не нашли вы можете собрать драйвер из исходных текстов самостоятельно. Мы рассмотрим построение пакетов в учебнике для экзамена LPI 101, Тема 102. (Смотри Ресурсы).

Для ISDN соединения вам также понадобится синхронный драйвер PPP, поскольку обычный, используемый для работы с асинхронными модемами, разработан для режима посимвольной передачи, а не для блочного. Как мы уже указывали в разделе о модемах, мы рассмотрим настройку соединений более подробно в учебнике для экзамена LPI 102 (смотри Ресурсы).

DSL соединения могут быть одного из нескольких типов. Некоторые предоставляют ethernet порт, связанный с сетью провайдера. Аутентификация в этом случае обычно производится с использованием ethernet MAC-адреса вашего компьютера. Если вы подсоединяете маршрутизатор (или другой компьютер) к DSL модему, то вам может понадобиться изменить Mac-адрес компьютера, чтобы добиться работоспособности соединения. Проще, если провайдер использует Point-to-Point Protocol over Ethernet или PPPoE. В этом случае вам выдается имя пользователя и пароль для использования при соединении. В таком случае, если вы используете маршрутизатор, вы обычно настраиваете соединение на нём, а ваш компьютер просто использует стандартное ethernet-подключение. Реже вам потребуется использовать PPPoA или PPP через ATM соединение.

Беспроводные соединения могут потребовать от вас имя сети, к которой вы подключаетесь. Это называется Service Set Identifier (Сервис установки идентификатора) или SSID. Если в сети используется шифрование, такое как Wired Equivalent Privacy или WEP или WiFi Protected Access или WPA, то вам понадобится соответствующим образом настроить соединение.

USB устройства

Обзор USB

В этом разделе мы рассмотрим поддержку Universal Serial Bus (Универсальной последовательной шины) или USB устройств в Linux. USB была разработана консорциумом компаний с целью предоставить единственную, простую шину для подключения периферии. В разделе Настройки BIOS, мы рассмотрели аспекты управления портами, IRQ и DMA ресурсами для машин с шиной ISA. Дизайн USB позволяет устройствам подключаться на лету и использовать стандартные гнезда для подключения устройств. USB-устройства включают клавиатуры, "мыши", принтеры, сканеры, жесткие диски, flash-драйвы, камеры, модемы, сетевые адаптеры и колонки. Список продолжает расти. Имеющаяся в Linux поддержка довольно всеобъемлюща, хотя некоторые устройства требуют специальных драйверов а другие, преимущественно принтеры, могут не поддерживаться или могут поддерживаться частично.

Компьютерные системы могут обладать одним или более контроллерами или хабами, к которым могут подключаться USB устройство или другой (внешний) хаб. Хаб может поддерживать до 7 устройств, некоторые или каждое из которых могут иметь дополнительные хабы. Хаб внутри системного блока называется root hub (корневой хаб). Каждая такая звездоподобная топология может поддерживать до 127 хабов или устройств.

Замечание: Часто, мы говорим USB порт, подразумевая возможность поддержки USB в компьютере и гнездо для подключения (сравните с последовательным или параллельным портами), а не внутренний адрес порта, используемый устройством.

USB системы являются многослойными.

- Слой Bus Interface (Интерфейс шины) обеспечивает физическую, сигнальную, и пакетную связь между хостом и устройствами, обеспечивая передачу данных между хостом и устройствами.

- Слой Device (устройство) используется системным ПО для базовых USB-операций с устройствами посредством шины. Это позволяет хосту определить характеристики устройства, включая класс устройства, имя производителя, имя устройства, требования к напряжению и многие другие функции наподобие скорости устройства и уровня поддержки USB.

- Слой Function (Функция) предоставляет дополнительные возможности, специфичные для устройства. Соответствие слоев хоста и ПО устройства позволяет выполнять функции,

присущие устройству.

Ранние спецификации USB (1.0 и 1.1) поддерживали скорость до 12Мб/с (Мегабит в секунду). Устройства, соответствующие этим спецификациям соответственно являются низкоскоростными, это принтеры, "мыши", клавиатуры, сканеры и модемы. Новая спецификация USB 2.0 поддерживает скорость до 480Мб/с, что сравнимо с жестким диском и внешним CD/DVD приводом. Некоторые USB 2.0 устройства имеют обратную совместимость, что позволяет использовать их в старых системах, хотя и не все скоростные устройства обладают ею. Если ваш компьютер не имеет встроенной поддержки USB 2.0, то вы можете воспользоваться PCI картой (или PC картой для ноутбука), предоставляющей один или несколько портов USB 2.0.

USB-кабель довольно тонок и содержит 4 провода: два для передачи сигнала плюс силовой и нейтральный. Разъём, подключаемый к хабу, имеет плоский прямоугольный штеккер (называемый А штеккер), а разъём, подключаемый к устройству или к хабу более низкого уровня имеет штеккер больше похожий на квадрат (называемый В штеккер). Несколько отличается существующий mini-B штеккер, для подсоединения к компьютеру небольших устройств вроде фотоаппарата. USB устройства и хабы могут получать питание по USB шине или могут иметь собственные источники питания.

Модуль поддержки USB в Linux

USB в настоящее время полностью поддерживается в Linux. Большая часть изменений проявилась в ветке ядра 2.6. Многое было перенесено из ядер 2.4, какая-то поддержка имеется даже в ядрах 2.2. Linux поддерживает как USB 2.0, так и ранние спецификации. Ввиду подключения на лету (горячего подключения), заложенной в самой природе USB, поддержка обычно производится посредством модулей ядра, которые могут загружаться или выгружаться по необходимости. В этом учебнике мы предполагаем, что все модули, которые вам необходимы для вашего дистрибутива или доступны, или уже установлены. Если вам необходимо скомпилировать свое собственное ядро, то обратитесь к теме 201 экзамена LPI 201 (смотри Ресурсы).

После того, как вы убедитесь, что ваш компьютер имеет USB порты, вы можете посмотреть что обнаружила система Linux, используя команду `lspci`, как показано в Листинге 23. Мы отфильтровали вывод, чтобы отобразить только устройства, соответствующие USB.

Листинг 23. Вывод команды `lspci` для USB устройств

```
[root@lyrebird root]# lspci | grep -i usb
00:1d.0 USB Controller: Intel Corporation 82801DB/DBL/DBM
      (ICH4/ICH4-L/ICH4-M) USB UHCI Controller #1 (rev 01)
00:1d.1 USB Controller: Intel Corporation 82801DB/DBL/DBM
      (ICH4/ICH4-L/ICH4-M) USB UHCI Controller #2 (rev 01)
00:1d.2 USB Controller: Intel Corporation 82801DB/DBL/DBM
      (ICH4/ICH4-L/ICH4-M) USB UHCI Controller #3 (rev 01)
00:1d.7 USB Controller: Intel Corporation 82801DB/DBM
      (ICH4/ICH4-M) USB2 EHCI Controller (rev 01)
```

Вы можете видеть, что в этой системе имеется четыре USB контроллера. Поля UHCI и EHCI показывают модули-драйвера, необходимые для поддержки контроллера. Корректный драйвер USB 1.1 зависит от чипсета, используемого в вашем контроллере. USB 2.0 требует EHCI драйвер плюс USB 1.1 драйвер. Смотри Таблицу 5.

Таблица 5. Драйвера USB в Linux

| Драйвер | Чипсет |
|---------|---|
| EHCI | USB 2.0 Поддерживает - требует один UHCI, OHCI или JE |
| UHCI | Intel и VIA Чипсеты |
| JE | Это альтернатива UHCI для ядер 2.4. Если UHCI не работает, и у вас Intel или VIA чипсет, то попробуйте JE |
| OHCI | Compaq, большинство PowerMacs, iMacs, и PowerBooks, OPTi, SiS, Ali |

Мы рассматривали команду `lsmod` и файлы конфигурации модулей `/etc/modules.conf` (ядро 2.4) и `/etc/modprobe.conf` (ядро 2.6) ранее при обсуждении поддержки звука. В Листинге 24 показано несколько модулей, связанных с USB устройствами, которые загружаются в той же системе, что использовалась и в Листинге 23. В этой системе есть USB-мышь

Листинг 24. Использование `lsmod` для отображения загруженных USB модулей

```
[root@lyrebird root]# lsmod | egrep 'usb|hci|hid|mouse|Module'
Module                Size Used by    Not tainted
usbserial              23420  0 (autoclean) (unused)
mousedev               5524   1
hid                    22244  0 (unused)
input                  5888   0 [keybdev mousedev hid]
ehci-hcd               20008  0 (unused)
usb-uhci               25740  0 (unused)
usbcore                77376  1 [usbserial hid ehci-hcd usb-uhci]
```

Заметим однако, что модуль `usbcore` используется всеми остальными USB модулями наряду с модулем `hid` (human interface device -- интерфейс пользовательских устройств).

Отображение информации о USB

Итак, теперь мы кое-что знаем о модулях поддержки USB, как мы можем обнаружить, что USB Устройство было подсоединено к компьютеру? Эта информация может быть обнаружена в систем файлах `/proc/bus/usb`. Файл `/proc/bus/usb/devices` содержит сводную информацию о подключенных в настоящее время USB устройствах. Частичный список для нашей системы приведен в Листинге 25.

Листинг 25. Часть содержимого /proc/bus/usb/devices

```
[root@lyrebird root]# cat /proc/bus/usb/devices
T: Bus=04 Lev=00 Prnt=00 Port=00 Cnt=00 Dev#= 1 Spd=480 MxCh= 6
B: Alloc= 0/800 us ( 0%), #Int= 0, #Iso= 0
D: Ver= 2.00 Cls=09(hub ) Sub=00 Prot=01 MxPS= 8 #Cfgs= 1
P: Vendor=0000 ProdID=0000 Rev= 2.04
S: Manufacturer=Linux 2.4.21-32.0.1.EL ehci-hcd
S: Product=Intel Corp. 82801DB USB2
S: SerialNumber=00:1d.7
C:* #Ifs= 1 Cfg#= 1 Atr=40 MxPwr= 0mA
I: If#= 0 Alt= 0 #EPs= 1 Cls=09(hub ) Sub=00 Prot=00 Driver=hub
E: Ad=81(I) Atr=03(Int.) MxPS= 2 IvL=256ms
T: Bus=03 Lev=00 Prnt=00 Port=00 Cnt=00 Dev#= 1 Spd=12 MxCh= 2
B: Alloc= 0/900 us ( 0%), #Int= 0, #Iso= 0
D: Ver= 1.00 Cls=09(hub ) Sub=00 Prot=00 MxPS= 8 #Cfgs= 1
P: Vendor=0000 ProdID=0000 Rev= 0.00
S: Product=USB UHCI Root Hub
S: SerialNumber=1840
C:* #Ifs= 1 Cfg#= 1 Atr=40 MxPwr= 0mA
I: If#= 0 Alt= 0 #EPs= 1 Cls=09(hub ) Sub=00 Prot=00 Driver=hub
E: Ad=81(I) Atr=03(Int.) MxPS= 8 IvL=255ms
```

Выделенное Spd=480 нами выше соответствует шине USB 2.0, а Spd=12 -- устройствам USB 1.1 (или возможно USB 1.0). Далее в списке видно, что наша мышь имеет Spd=1.5. Полтора мегабита в секунду вполне достаточно для мыши.

Что касается других пунктов, которые можно увидеть в системе файлов /proc, то, возможно, вам будет приятно узнать, что имеется команда `lsusb` помогающая вам отображать эту информацию. В частности, вы можете получить древовидное изображение ваших USB устройств, используя опцию `-t`. Это отобразит иерархию подключения. Вы можете использовать опцию `-d` для вывода информации о конкретном устройстве, после того, как получите результат от системы с использованием опции `-t`. Опция `-v` генерирует подробный вывод с интерпретацией многих полей, что мы могли увидеть в Листинге 25. Для Листинга 26, мы подсоединили внешний хаб, цифровую камеру Nikon, USB-брелок и внешний жесткий диск USB 2.00, а затем выполнили команду.

Листинг 26. Использование команды lsusb

```
[root@lyrebird root]# lsusb -t
Bus# 4
`-Dev# 1 Vendor 0x0000 Product 0x0000
  |-Dev# 2 Vendor 0x0409 Product 0x0059
    |-Dev# 8 Vendor 0x04b0 Product 0x0108
    |-Dev# 4 Vendor 0x0d7d Product 0x1400
    |-Dev# 7 Vendor 0x1058 Product 0x0401
  `--Dev# 3 Vendor 0x07d0 Product 0x1202
Bus# 3
`-Dev# 1 Vendor 0x0000 Product 0x0000
Bus# 2
`-Dev# 1 Vendor 0x0000 Product 0x0000
Bus# 1
`-Dev# 1 Vendor 0x0000 Product 0x0000
  `--Dev# 2 Vendor 0x1241 Product 0x1111
[root@lyrebird root]# lsusb -d 0x0409:0x0059
Bus 004 Device 002: ID 0409:0059 NEC Corp. HighSpeed Hub
[root@lyrebird root]# lsusb -d 0x04b0:0x0108
Bus 004 Device 008: ID 04b0:0108 Nikon Corp. Coolpix 2500
[root@lyrebird root]# lsusb -d 0x0d7d:0x1400
Bus 004 Device 004: ID 0d7d:1400 Phison Electronics Corp.
[root@lyrebird root]# lsusb -d 0x1058:0x0401
Bus 004 Device 007: ID 1058:0401 Western Digital Technologies, Inc.
[root@lyrebird root]# lsusb -d 0x07d0:0x1202
Bus 004 Device 003: ID 07d0:1202 Dazzle
[root@lyrebird root]# lsusb -d 0x1241:0x1111
Bus 001 Device 002: ID 1241:1111 Belkin Mouse
[root@lyrebird root]#
```

В Листинге 27 приведена часть подробного вывода после выполнения команды `lsusb`. Это часть для USB-брелока. Заметьте, что устройство сообщает свои максимальные требования к питанию (200mA). Отметим также, что устройство может быть обнаружено как SCSI устройство. Используя команду `dmesg` или `fdisk -l` вы можете выяснить, какое SCSI устройство соответствует этому устройству. Большинство камер оборудованы USB портами, а также устройствами чтения карт (картридерами). Флэш-устройства и жесткие диски причисляемые к классу устройств хранения управляются в Linux как SCSI устройства. Многие камеры идут с программами под Windows, помогающими загружать картинки с камеры. В Linux вы можете просто смонтировать SCSI устройство, представляющее камеру и скопировать фотографии на свой жесткий диск, где вы сможете редактировать их в таких программах, как GNU Image Manipulation Program [GNU Программа Манипулирования Изображениями] (the GIMP). Затем вы можете стереть файлы с карты памяти или записать файлы на нее из Linux, используя камеру как вычурную замену флоппи-диску.

Листинг 27. Подробный вывод (его часть) команды

```
[root@lyrebird root]# lsusb -vd 0x0d7d:0x1400
```

Bus 004 Device 004: ID 0d7d:1400 Phison Electronics Corp.

Device Descriptor:

```
bLength          18
bDescriptorType   1
bcdUSB            2.00
bDeviceClass      0 (Defined at Interface level)
bDeviceSubClass   0
bDeviceProtocol   0
bMaxPacketSize0   64
idVendor          0x0d7d Phison Electronics Corp.
idProduct         0x1400
bcdDevice         0.02
iManufacturer     1
iProduct          2 USB DISK 12X
iSerial           3 0743112A0083
```

bNumConfigurations 1

Configuration Descriptor:

```
bLength          9
bDescriptorType   2
wTotalLength      32
bNumInterfaces    1
bConfigurationValue 1
iConfiguration    0
bmAttributes      0x80
MaxPower          200mA
```

Interface Descriptor:

```
bLength          9
bDescriptorType   4
bInterfaceNumber  0
bAlternateSetting  0
bNumEndpoints     2
bInterfaceClass    8 Mass Storage
bInterfaceSubClass  6 SCSI
bInterfaceProtocol 80 Bulk (Zip)
iInterface         0
```

...

Еще один кусочек информации, который мы можем извлечь о шине и ID наших USB устройств из Листинга 26, это способ определения какие модули требуются для конкретного устройства. Приведем пару примеров в Листинге 28.

Листинг 28. Подробный вывод (его часть) команды lsusb

```
[root@lyrebird root]# usbmodules --device /proc/bus/usb/004/003
usb-storage
[root@lyrebird root]# usbmodules --device /proc/bus/usb/004/007
usb-storage
hid
```

Подключение на лету (Горячее подключение)

Есть две команды, при помощи которых ваша система может использоваться для управления подключением USB устройств на лету: `usbmgr` и `hotplug`. В зависимости от того, какую из них вы будете использовать файлы настроек можно обнаружить в каталогах `/etc/usbmgr` или `/etc/hotplug` соответственно. Новые системы скорее всего уже имеют поддержку подключения на лету.

Подключение на лету USB устройств (а также PC карт) подразумевает, что пользователь подключает устройство в то время, как система уже работает. При этом система должна :

- Определить тип устройства, найти драйвер и запустить его
- Связать драйвер с устройством
- Уведомить другие подсистемы об устройстве. Что позволяет дискам смонтироваться или, например, добавиться очереди для печати.