

The Linux Serial HOWTO

revised by David S.Lawyer bf347@lafn.org original by Greg Hankins
Русский перевод Ilgiz Kalmetev, ilgiz@mail.rb.ru

v1.12, July 1998

Этот документ описывает возможности последовательного порта, нераскрытые в Text-Terminal-HOWTO или Serial-Programming-HOWTO. Он описывает в основном getty, мультипортовые последовательные платы и модемы.

Содержание

1 Введение	1
2 Поддерживаемые последовательные аппаратные устройства	2
3 Какими именами обозначаются последовательные порты?	6
4 Интересные программы, о которых вы должны знать	8
5 Как сделать исходящий звонок через мой модем?	9
6 Как заставить мой модем принимать входные и совершать исходящие звонки	11
7 uucgetty	13
8 Как настроить терминальный вход на моем PC?	15
9 Я могу использовать более, чем два последовательных устройства?	17
10 Как мне настроить последовательные порты на более высокие скорости? Какую скорость я должен задать для моего модема?	19
11 Коммуникационные программы и утилиты	20
12 Советы и разное по последовательным устройствам	20
13 На шаг дальше ...	22
14 Поиск неисправностей	24
15 Другие источники информации	27
16 Содействия	27

1 Введение

Это - Linux Serial HOWTO. В нем приведена информация о последовательных портах, которая не описана в других HOWTO'S. Это - информация об установке модема под Linux, некоторые советы по последовательным устройствам и советы по поиску неисправностей.

Существует другой HOWTO для терминалов (см. "Относящиеся HOWTO'S "). Этот HOWTO относится к Linux для платформы Intel x86, хотя может подойти и для других архитектур.

1.1 Авторские права

Авторские права (с) 1993 - 1997 принадлежат Greg Hankins, а в 1998 - David Lawyer. Этот документ может распространяться под условиями, изложенными в лицензии LDP на <http://sunsite.unc.edu/LDP/COPYRIGHT.html>. Этот документ не может распространяться в измененной форме без согласия автора.

1.2 Относящиеся HOWTO'S (было: Последовательный порт)

- Serial-Programming-HOWTO поможет вам писать программы на C (или куски их), чтобы работать с последовательным портом. Вы можете сделать эквивалент "stty...", открывать порты в различных режимах и пр.
- Text-Terminal-HOWTO - все о текстовых терминалах (dumb терминалы).

1.3 Примечание к выпуску

Я, David Lawyer, являюсь новым ведущим (Greg Hankins был первоначальным автором и ведущим). Внесено мало изменений, и я не проверил много чего, чтобы увидеть, является ли оно современным. Для этого надо проделать много работы. Сообщите мне, что неправильно или отсутствует. В будущем, я может быть разделю этот Howto: 1. Информация о терминалах в Text-Terminal-HOWTO. 2. Информация о модемах в новом Howto: ModemHOWTO. 3. Информация о последовательных портах, добавленная к этому Howto. Таким образом ищите новые версии в августе или сентябре 1998.

1.4 Новые версии этого документа

Новые версии Serial HOWTO будут доступны на <ftp://sunsite.unc.edu:/pub/Linux/docs/HOWTO/Serial-HOWTO> и зеркалах. Имеются другие форматы, типа Postscript и DVI в каталоге other-formats. Serial HOWTO также доступен на <http://sunsite.unc.edu/LDP/HOWTO/Serial-HOWTO.html> и будет помещаться в comp.os.linux.answers ежемесячно.

1.5 Обратная связь

Пожалуйста присылайте мне любые вопросы, комментарии, предложения или дополнительный материал. Я всегда рад услышать, что вы думаете об этом HOWTO. Также я всегда готов к разъяснениям! Сообщите мне точно, что вы не понимаете, или что могло бы быть более ясным. Вы можете найти меня по адресу bf347@lafn.org (David Lawyer).

1.6 Оговорка

Ваш mileage может измениться. Данные ответы не могут работать для всех систем и всех комбинаций установок.

2 Поддерживаемые последовательные аппаратные устройства

Linux, насколько известно, работает со следующими последовательными аппаратными устройствами.

2.1 Стандартные последовательные платы PC

- стандартные последовательные платы PC (COM1 - COM4), к которым могут быть подключены внешние последовательные устройства (модемы, последовательные мыши и т.д ...). Так как PC больше не поставляются с ними (но имеют чипы для этой цели, установленные на материнской плате), то они должны быть в розничной продаже.
- стандартные внутренние модемы PC (COM1 - COM4)
- Quickpath Systems Port-Folio 550e (позволяет IRQ - 3, 4, 5, 9, 10, 11, 12 и 15). Примечание: из-за конфликтов адресов, вы не можете использовать COM4 и видеокарту IBM8514 одновременно. Это из-за ошибки в плате IBM8514.

2.2 Plug And Play (PnP) модемы

Большинство модемов сегодня - PnP, и большинство из них будет (после некоторых усилий) работать с Linux. Самые лучшие из них - те, которые имеют переключатели для отключения PnP. Во многих случаях, если вы просто устанавливаете переключатели для некоторых значений COM и IRQ, PnP отключится. Вы можете также захотеть отключить PnP в BIOS вашего компьютера. Если вы не можете отключить PnP, тогда попробуйте вот что:

- загрузите Win95/98, сконфигурируйте модем и посмотрите, сохраняются ли установки, когда вы загружаете Linux. Если нет, то:
- загрузите Win95/98, сконфигурируйте модем и затем сделайте горячую загрузку Linux так, чтобы установки сохранились. Вы можете сделать это загрузчиком loadlin.
- возьмите инструментальные средства PnP с <http://www.redhat.com/linux-info/pnp>

2.3 Dumb многопортовые последовательные платы (с микросхемами 8250/16450/16550A)

Они также называются "последовательные адаптеры".

* = > "setserial" показывает подробности конфигурирования

- AST FourPort и аналоги (4 порта) *
- Accent Async-4 (4 порта) *
- Arnet Multiport-8 (8 портов)
- Bell Technologies HUB6 (6 портов)
- Boca BB-1004 (4 портов), BB-1008 (8 портов), BB-2016 (16 портов) *
- Boca IOAT66 (6 портов)
- Boca 2by4 (4 последовательных порта, 2 параллельных порта)
- Byterunner (claims low prices)
- Computone ValuePort V4-ISA (совместим с AST FourPort) *
- Digi PC/8 (8 портов)
- GTEK BBS-550 (8 портов) См. mini-howto по нему.
- HUB-6 См. Bell Technologies.
- Longshine LCS-8880, Longshine LCS-8880+ (совместим с AST FourPort)

- Моха C104, Моха C104+ (совместим с AST FourPort)
- PC-COMM (4 порта)
- *Sealevel Systems* COMM-2 (2 порта), COMM-4 (4 порта) и COMM-8 (8 портов)
- SIIG I/O Expander 2S IO1812 (4 порта)
- STB-4COM (4 порта)
- Twincom ACI/550
- Usenet Serial Board II (4 порта) *

В общем-то, Linux будет поддерживать любую последовательную плату, которая использует 8250, 16450, 16550, 16550A, 16650 (или совместимые) UART, или внутренний модем, который эмулирует один из вышеупомянутых UART.

Обратите внимание: ВВ-1004 и ВВ-1008 не поддерживают сигналы DCD и RI, и таким образом не пригодны для использования в качестве модемов, обслуживающих входящие звонки. В остальном они прекрасно будут работать.

2.4 Интеллектуальные многопортовые последовательные платы

Удостоверьтесь, что совместимый с Linux драйвер доступен. Этот список слегка устарел.

- Comtrol RocketPort (36MHz ASIC; 4, 8, 16 или 32 порта)
контактный адрес: info@comtrol.com или <http://www.comtrol.com>
состояние драйвера: поддерживается Comtrol
расположение драйвера: <ftp://tsx-11.mit.edu/pub/linux/packages/comtrol>
- Computone IntelliPort II (16MHz 80186; 4, 8 или 16 портов), IntelliPort II EXpandable (20MHz 80186; 16 - 64 порта)
контактный адрес: Michael H. Warfield, mhw@wittsend.atl.ga.us
состояние драйвера: "pre-ALPHA"
- Cyclades Cyclom-Y (Cirrus Logic CD1400 UARTS; 8 - 32 порта), Cyclom-Z (25MHz MIPS R3000; 8 - 128 портов) контактный адрес: si@wimpol.demon.co.uk или <http://www.cyclades.com>
состояние драйвера: поддерживается Cyclades
расположение драйвера: <ftp://ftp.cyclades.com/pub/cyclades> и включен в ядро Linux, начиная с версии 1.1.75
- Decision PCCOM8 (8 портов)
контактный адрес: pccom8@signum.se
расположение драйвера: <ftp://ftp.signum.se/pub/pccom8>
- Digi PC/XI (12.5MHz 80186; 4, 8, или 16 портов), PC/XE (12.5/16MHz 80186; 2, 4, или 8 портов), PC/XR (16MHz IDT3041; 4 или 8 портов), PC/XEM (20MHz IDT3051; 8 - 64 порта)
контактный адрес: si@wimpol.demon.co.uk или uksales@dgii.com или <http://www.dgii.com>
состояние драйвера: поддерживается Digi
расположение драйвера: <ftp://ftp.dgii.com/drivers/linux> и включен в ядро Linux, начиная с версии 2.0
- Digi COM/XI (10MHz 80188; 4 или 8 портов)
контактный адрес: Simon Park, si@wimpol.demon.co.uk
состояние драйвера: ALPHA
примечание: Саймон - часто задерживается с ответом по email в течение месяцев из-за его работы. Mark Hatle, fray@krypton.mankato.msus.edu - доброволец, любезно согласившийся сделать драйвер доступным, если вам он нужен. Mark не поддерживает и не занимается драйвером.

- Equinox SuperSerial Technology (30MHz ASIC; 2 - 128 портов)
контактный адрес: sales@equinox.com или <http://www.equinox.com>
состояние драйвера: поддерживается Equinox
расположение драйвера: <ftp://ftp.equinox.com/library/sst>
- GTEK Cyclone (16C654 UARTS; 6, 16 и 32 порта), SmartCard (24MHz Dallas DS80C320; 8 портов), BlackBoard-8A (16C654 UARTS; 8 портов), PCSS (15/24MHz 8032; 8 портов)
контактный адрес: spot@gtek.com или <http://www.gtek.com>
состояние драйвера: поддерживается GTEK
расположение драйвера: <ftp://ftp.gtek.com/pub>
- Hayes ESP (COM-BIC; 1 - 8 портов)
контактный адрес: Andrew J. Robinson, arobinso@nyx.net или <http://www.nyx.net/~arobinso>
состояние драйвера: поддерживается автором
расположение драйвера: <http://www.nyx.net/~arobinso> и включен в ядро Linux, начиная с версии 2.1.15
- Maxpeed SS (Toshiba; 4, 8 и 16 портов)
контактный адрес: info@maxpeed.com или <http://www.maxpeed.com>
состояние драйвера: поддерживается Maxpeed
расположение драйвера: <ftp://maxpeed.com/pub/ss>
- Моха С218 (12MHz 80286; 8 портов), Моха С320 (40MHz TMS320; 8 - 32 порта)
контактный адрес: info@moxa.com.tw или <http://www.moxa.com.tw>
состояние драйвера: поддерживается Моха
расположение драйвера: <ftp://ftp.moxa.com.tw/drivers/c218-320/linux>
- SDL RISC0M/8 (Cirrus Logic CD180; 8 портов)
контактный адрес: sales@sdlcomm.com или <http://www.sdlcomm.com>
состояние драйвера: поддерживается SDL
расположение драйвера: <ftp://ftp.sdlcomm.com/pub/drivers>
- Specialix SIO (20MHz Zilog Z280; 4 - 32 порта), XIO (25MHz Zilog Z280; 4 - 32 порта)
контактный адрес: Саймон Аллен, simonallen@cix.compulink.co.uk
состояние драйвера: БЭТА
расположение драйвера: <ftp://sunsite.unc.edu/pub/Linux/kernel/patches/serial>
- Stallion EasyIO-4 (4 порта), EasyIO-8 (8 портов), EasyConnection (8 - 32 порта) - каждый с Cirrus Logic CD1400 UARTs, Stallion (8MHz 80186 CPU; 8 или 16 портов), Brumby (10/12 MHz 80186 CPU; 4, 8 или 16 портов), ONBOARD (16MHz 80186 CPU; 4, 8, 12, 16 или 32 порта), EasyConnection 8/64 (25MHz 80186 CPU; 8 - 64 порта)
контактный адрес: sales@stallion.com или <http://www.stallion.com>
состояние драйвера: поддерживается Stallion
расположение драйвера: <ftp://ftp.stallion.com/drivers/ata5/linux> и включен в ядро linux, начиная с 1.3.27

Обзор изделий Comtrol, Cyclades, Digi, и Stallion был напечатан в июне 1995 в Linux Journal. Статья доступна на <http://www.ssc.com/lj/issue14>.

2.5 Неподдерживаемые последовательные аппаратные средства

2.5.1 Winмодемы не работают

Модемы, помеченные "Winmodems", например USR Sportster Winmodem и IBM Aptiva MWAVE, не поддерживаются в Linux. Эти модемы используют частные разработки, которые требуют специальных драйверов Windows.

Избегайте модемов этих типов.

2.5.2 Драйверы Rockwell (RPI) могут работать

Модемы, которые требуют драйверы Rockwell RPI, пригодны для использования не полностью, так как программное обеспечение драйвера не работает под Linux. RPI делает сжатие и исправление ошибки, используя программное обеспечение MS на CPU вашего компьютера. Если вы желаете эксплуатировать модем без использования RPI (и таким образом обходиться без сжатия и исправления ошибок), то вы можете легко отключить RPI, пошлав в модем (в виде строки инициализации) команду отключения RPI: +H0

3 Какими именами обозначаются последовательные порты?

Порт ввода-вывода - это способ передать данные в/из компьютера. Имеются множество типов портов ввода-вывода типа последовательных портов, параллельных портов, контроллеров дискового, плат локальной сети ethernet и т.д. Мы будем иметь дело с последовательными портами, так как модемы и терминалы - последовательные устройства. Каждый последовательный порт должен иметь адрес ввода-вывода и прерывание (IRQ). Имеются четыре последовательных порта, соответствующие COM1 - COM4:

```
ttyS0 (COM1) address 0x3f8 IRQ 4
ttyS1 (COM2) address 0x2f8 IRQ 3
ttyS2 (COM3) address 0x3e8 IRQ 4
ttyS3 (COM4) address 0x2e8 IRQ 3
```

Если Linux не обнаруживает последовательные порты при загрузке, то удостоверьтесь, что поддержка последовательных устройств разрешена и вкомпилирована в ядро. В этом документе, я обращаюсь к COM1 как ttyS0, COM2 как ttyS1, COM3 как ttyS2, и COM4 как ttyS3. Обратите внимание, что по умолчанию эти устройства имеют перекрывающиеся IRQ. Вы не можете использовать все порты с этими значениями по умолчанию, и вы должны переназначить другие IRQ. См. раздел "Я могу использовать более, чем два последовательных устройства?" при установке IRQ.

3.1 Устройства: модем, мышь

На некоторых инсталляциях, будут созданы два дополнительных устройства, /dev/modem для вашего модема и /dev/mouse для вашей мыши. Оба из них - символические связи с соответствующими устройствами в /dev, которые вы определили при установке (если у вас bus mouse, то /dev/mouse укажет на устройство bus mouse).

В отношении /dev/mouse и /dev/modem ведутся споры. Я строго против использования этих связей. В частности, если вы планируете использовать ваш модем для приема входящих звонков, то вы можете столкнуться с проблемами, потому что файлы блокировки не могут работать правильно, если вы используете /dev/modem. Используйте эту связь, если это вам нравится, но удостоверьтесь, что она указывает на правильное устройство. Однако, если вы измените или удалите эту связь, то некоторые приложения (minicom, например) могут потребовать перенастройки.

3.2 Устройство cua

Каждое устройство ttyS имеет соответствующее устройство cua. Велись некоторые разговоры об отмене cua, так что возможно самое лучшее - это использовать ttyS. Основное различие между cua и ttyS - то, что для обычной команды "open"(без каких-либо флажков) в прикладной программе, cua откроет порт, даже если сигналы управления модема (типа DCD) просят этого не делать.

(Чтобы проверить сигналы управления модема должен быть установлен stty.) Порт ttyS отказался бы открываться в этом случае, но его можно принудительно открыть, задавая некоторые флажки в команде "open".

Таким образом порт ttyS может делать все, что может порт cua. За исключением того, что результаты "принудительного"открытия in read statements in the program работают по-другому, но программисты могут обойти это, изменяя свои программы. Удаление cua привело бы Linux в большее соответствие с Posix стандартом и решило бы некоторые проблемы с файлами блокировки.

3.3 Устройства и числа последовательных портов в /dev

```
/dev/ttyS0 major 4, minor 64    /dev/cua0 major 5, minor 64
/dev/ttyS1 major 4, minor 65    /dev/cua1 major 5, minor 65
/dev/ttyS2 major 4, minor 66    /dev/cua2 major 5, minor 66
/dev/ttyS3 major 4, minor 67    /dev/cua3 major 5, minor 67
```

Обратите внимание, что все дистрибутивы должны приходить с этими устройствами, уже правильно установленными (если cua не отменен). Вы можете проверить это, напечатав:

```
linux% ls -l /dev/cua* linux% ls -l /dev/ttyS*
```

3.3.1 Создание ttyS устройств в /dev

Если у вас нет устройства, то вы должны будете создать его командой mknod. Например, предположим, что вам нужно создать устройство ttyS0:

```
linux# mknod -m 666 /dev/cua0 c 5 64
linux# mknod -m 666 /dev/ttyS0 c 4 64
```

Вы можете использовать скрипт MAKEDEV, который лежит в /dev. Он упрощает создание устройств. Например, если вам нужно создать устройств ttyS0, то вы напечатаете:

```
linux# cd /dev
linux# ./MAKEDEV ttyS0
```

Эти строки создают устройства для входящих и исходящих устройств и должны установить правильные права доступа.

3.4 Примечания для dumb многопортовых плат

Использование ваших многопортовых плат зависит от того какая плата у вас есть. Некоторые из них перечислены подробно в rc.serial или в 0setserial, который приходит с пакетом setserial. Я строго рекомендую взять последнюю версию setserial, если вы пытаетесь использовать многопортовые платы. Вероятно, вы должны будете создать эти устройства.

Используйте или команду mknod, или скрипт MAKEDEV. Устройства для многопортовых плат создаются сложением "64 + номер_порта". Так, если вы хотели создать устройство ttyS17, то напечатайте:

```
linux# mknod -m 666 /dev/cua17 c 5 81
linux# mknod -m 666 /dev/ttyS17 c 4 81
```

Обратите внимание, что "64 + 17 = 81". При использовании скрипта MAKEDEV, вы должны ввести:

```
linux# cd /dev
linux# ./MAKEDEV ttyS17
```

Обратите внимание: руководство SIIG для IO1812 распечатывает для COM5-COM8 неправильные значения.

Они должны быть COM5 = 0x250, COM6 = 0x258, COM7 = 0x260 и COM8 = 0x268.

Обратите внимание: Digi PC/8 Interrupt Status Register - в 0x140.

Обратите внимание: для AST Fourport, вам возможно потребуется указать skip_test в rc.serial.

3.5 Примечания для интеллектуальных многопортовых плат

Читайте информацию, которая приходит с драйвером. Эти платы используют специальные устройства и нестандартные. Эта информация изменяется в зависимости от ваших аппаратных средств.

4 Интересные программы, о которых вы должны знать

4.1 Что такое getty?

getty - программа, которая обслуживает процесс входа в систему, когда вы регистрируетесь на Unix машине. Вы должны будете использовать getty, если хотите звонить модемом на вашу Linux машину. Вы не должны использовать getty, если вы хотите только звонить наружу вашим модемом. Имеются три версии, которые обычно используются в Linux: getty_ps, mgetty и agetty. Синтаксис для этих программ отличается, так что убедитесь, что вы проверили и удостоверились, что вы используете правильный синтаксис для того getty, которым пользуетесь.

4.1.1 О getty_ps

Большинство дистрибутивов приходит с установленным пакетом getty_ps. (Debian использовал agetty, но теперь использует mgetty). Он содержит две программы: getty - используется для консоли и устройств терминала, и ugetty для модемов. Я использую эту версию getty, так что на него я и буду ориентироваться.

4.1.2 О mgetty

mgetty - версия getty, главным образом для использования с модемами. Она может использоваться для аппаратных терминалов, но документация приблизительно на 99%, связана с модемами. В дополнение ко входу в систему по телефонной линии, mgetty также обеспечивает поддержку факсов и автоматическое обнаружение PPP. Документация по mgetty (поставляется в формате texinfo) достаточно хороша, и не нуждается в дополнении.

Пожалуйста, обратитесь к ней для инструкций по установке. Вы можете найти последнюю информацию о mgetty по адресу <http://www.leo.org/doing/mgetty/>.

4.1.3 О agetty

Agetty - третья разновидность getty. Это простая полнофункциональная реализация getty, которая больше подходит для виртуальных консолей или терминалов, чем для модемов.

4.2 Что такое setserial?

setserial - программа, которая позволяет вам смотреть и изменять различные атрибуты последовательного устройства, включая адрес порта, прерывание и другие опции последовательного порта. Вы можете посмотреть версию вашей копии этой программы, запустив setserial без параметров. Если параметр - /dev/ttyS1, и т.д., то вы увидите некоторую информацию об этом порте.

Когда ваша Linux система загружается, настраиваются только ttyS{0-3}, используя значения по умолчанию IRQS 4 и 3. Так что, если у вас есть любые другие последовательные порты,

обеспеченные другими платами или если `ttyS{0-3}` имеют необычный IRQ, то вы должны использовать `setserial`, чтобы сконфигурировать эти последовательные порты. Полный список опций есть в `man`.

5 Как сделать исходящий звонок через мой модем?

5.1 Аппаратные требования

Вы можете использовать или внешний или внутренний модем. Внутренние стоят меньше и не занимают место на столе или корпусе компьютера. Но внешний проще в установке и имеет лампочки, которые дают вам сведения о том, что происходит. Однако внешние модемы нужно выключать, когда они не используются, и вероятно, они немного потребляют электричество даже когда выключены.

5.1.1 Внешние модемы

Сначала, удостоверьтесь, что у вас правильный кабель. Ваш модем требует прямой кабель, без пересекающихся штырьков. Любой компьютерный магазин должен иметь их. Удостоверитесь, что вам дали то, что нужно. Если вы используете DB25 последовательный порт, то он всегда будет "папа" DB25 (входящий разъем). Не путайте его с параллельным портом, который является "мамой" DB25 (охватывающим). Подключите ваш модем к одному из ваших последовательных портов. Проконсультируйтесь с руководством по вашему модему об том, как сделать это, если вам нужна справка.

5.1.2 Внутренние модемы

Для внутреннего модема вам не нужен кабель. Внутренний модем не нуждается в последовательном порте, он в него встроен. Все, что вы должны сделать - настроить его на незанятые прерывание и адрес порта ввода-вывода. Проконсультируйтесь с вашим руководством по модему, если вы затрудняетесь. Также, см. раздел "Я могу использовать более чем два последовательных устройства?", если вам нужна справка при выборе прерываний или адресов.

На некоторых материнских платах вы должны будете отключить последовательный порт, который заменяет модем, чтобы избежать конфликтов. Это может быть выполнено переключками или в установках BIOS, в зависимости от вашей материнской платы.

Проконсультируйтесь с руководством по вашей материнской плате.

Если у вас видеоплата IBM8514, то вы должны знать о том, что в ней есть ошибка. Вы можете столкнуться с проблемами, если захотите, чтобы ваш внутренний модем работал на `ttyS3`. Если Linux не обнаруживает ваш внутренний модем на `ttyS3`, вы можете использовать `setserial`, и модем будет прекрасно работать. Внутренние модемы на `ttyS{0-2}` не должны иметь каких-либо проблем. Linux не производит автоматической конфигурации на `ttyS3` из-за ошибки этой видеоплаты.

5.2 Общение с вашим модемом

Используйте `kermit`, `minicom` или другую программу связи, чтобы проверить установку, прежде, чем вы перейдете к более сложным вещам типа SLIP или PPP. Вы можете найти последнюю версию `kermit` на <http://www.columbia.edu/kermit/>.

Например, надо сказать, что ваш модем был на `ttyS3`, и его быстроедействие было 115200 бит\сек. Вы должны были бы сделать следующее:

```
linux# kermit
C-Kermit 6.0.192, 6 Sep 96, for Linux
Copyright (C) 1985, 1996,
  Trustees of Columbia University in the City of New York.
```

```

Default file-transfer mode is BINARY Type ? or HELP for help.
C-Kermit>set line /dev/ttyS3
C-Kermit>set carrier-watch off
C-Kermit>set speed 115200
/dev/ttyS3, 115200 bps
C-Kermit>c
Connecting to /dev/ttyS3, speed 115200.
The escape character is Ctrl-\ (ASCII 28, FS)
Type the escape character followed by C to get back,
or followed by ? to see other options.
ATE1Q0V1                ; you type this and then the Enter key
OK                       ; modem should respond with this

```

Если ваш модем отвечает на команды AT, то вы можете считать, что ваш модем на Linux стороне работает правильно. Теперь попробуйте позвонить на другой модем, напечатав:

```
ATDT7654321
```

где 7654321 - номер телефона. Используйте ATDP вместо ATDT, если у вас номер набирается пульсом. Если обращение проходит, то ваш модем работает.

Чтобы вернуться к подсказке kermit, нажав клавишу Ctrl, нажмите клавишу наклонной черты влево, затем отпустите клавишу Ctrl, и нажмите клавишу C:

```

Ctrl-\-C
(Back at linux)
C-Kermit>quit
linux#

```

Это было только тест с использованием примитивного дозвона "вручную".

Нормальный способ состоит в том, чтобы позволить kermit звонить с помощью встроенной базы данных модемов и возможностями автоматическими набора, например, для модема USRobotics (USR):

```

linux# kermit
C-Kermit 6.0.192, 6 Sep 1997, for Linux
  Copyright (C) 1985, 1996,
  Trustees of Columbia University in the City of New York.
Default file-transfer mode is BINARY
Type ? or HELP for help
C-Kermit>set modem type usr          ; Select modem type
C-Kermit>set line /dev/ttyS3         ; Select communication device
C-Kermit>set speed 115200            ; Set the dialing speed
C-Kermit>dial 7654321                ; Dial
Number: 7654321
Device=/dev/ttyS3, modem=usr, speed=115200
Call completed.<BEEP>
Connecting to /dev/ttyS3, speed 115200
The escape character is Ctrl-\ (ASCII 28, FS).
Type the escape character followed by C to get back,
or followed by ? to see other options.

Welcome to ...

login:

```

См. раздел "Программы связи" о программах связи, если вам нужны какие-либо указания.

Когда вы делаете исходящий звонок вашим модемом, установите быстродействие последовательного порта самым высоким, которое поддерживает ваш модем (обычно это не более 115200 бит\сек). This is not the same as the "advertised" speed of the modem such as 56K. Версии Linux с libc версии выше 5.x имеют поддержку для скоростей до 115200 бит\сек. В Linux работают даже более высокие скорости, чем эти.

5.3 Настройка модема на исходящие звонки

Для использования модема только для исходящих звонков, вы можете сконфигурировать ваш модем.

Если вы предполагаете использовать ваш модем для входящих звонков, вы должны настроить ваш модем на то же самое быстродействие, которое вы предполагаете указать в `getty`. Однако, большинство модемов сегодня устанавливает свое быстродействие автоматически на одинаковое быстродействие, используя команды модема. Таким образом, `getty`, запущенный на скорости 115200, автоматически установит модем на это быстродействие, когда `getty` пошлет `init` строку на последовательный порт, на котором установлен модем. Вообще, фабричные значения по умолчанию, которые дают возможность исправления ошибок и аппаратного управления потоком данных - это самая лучшая установка для исходящих звонков модемом, проконсультируйтесь с руководством по вашему модему для этих установок.

5.4 Аппаратное управление потоком данных

Если ваш модем поддерживает аппаратное управление потоком данных (RTS/CTS), я строго рекомендую вам его использовать. Это особенно важно для модемов, которые поддерживают сжатие данных. Сначала, вы должны разрешить RTS/CTS управление потоком данных на последовательном порту непосредственно. Это лучше всего выполнять при запуске, например, в `/etc/rc.d/rc.local` или `/etc/rc.d/rc.serial`. Удостоверьтесь, что эти файлы выполняются из главного файла `rc.sysinit`! Вы должны сделать следующее для каждого последовательного порта, на котором вы хотите разрешить аппаратное управление потоком данных:

```
stty crtscts < /dev/ttyS3
```

Если не разрешено по умолчанию, то также разрешите RTS/CTS на вашем модеме. В этом вам может помочь ваша программа связи. Проконсультируйтесь с руководством по вашему модему, если необходимо, и затем сохраните настройку вашего модем, если ваш модем поддерживает сохранение профилей.

6 Как заставить мой модем принимать входные и совершать исходящие звонки

Настройте ваш модем правильно на исходящие звонки. Если вы не читали раздел "Как сделать исходящий звонок через мой модем?", то прочитайте его сейчас! Он содержит очень важную информацию об установке. Вы не должны читать этот раздел, если вы хотите только звонить наружу вашим модемом.

6.1 Настройка модема на входящие и исходящие звонки

Для использования входящих и исходящих звонков, вы должны установить ваш модем некоторым образом (снова, используя AT команды на вашем модеме):

```
E1      включение эхоотображения
Q0      сообщать коды результата
```

```
V1      verbose ON
S0=0    не поднимать трубку (uugetty обрабатывает эту функцию опцией WAITFOR)
```

Если вы не установили эти параметры правильно, ваша INIT строка в вашем файле настроек может не сработать, нарушая всю работу. Но более подробная информация о файлах конфигурации расположена ниже ...

```
&C1     включать DCD только после соединения
&S0     DSR всегда включен
DTR on/off сброс модема (зависит от производителя - RTFM)
```

Они воздействуют на то, что ваш модем делает, когда он начинает и заканчивает звонить. Если ваш модем не поддерживает сохранение профилей, то вы можете установить их через INIT строку в вашем файле конфигурации. См. ниже. Некоторые модемы приходят с DIP переключателями, которые воздействуют на установки регистров. Убедитесь, что они тоже установлены правильно.

Я начал коллекционировать установки модемов для различных типов модемов. Пока, я только имею несколько из них, если вы хотите послать мне вашу рабочую конфигурацию, пожалуйста, сделайте это! Вы можете получить их на <ftp://ftp.cc.gatech.edu/pub/people/gregh/modem-configs>

Обратите внимание: чтобы заставить мой модем USR Courier V.34 сброситься правильно, когда падает DTR, я должен был установить &D2 и S13=1 (установка бита 0 регистра S13).

На модеме USR Sportster V.34 это также работает.

Обратите внимание: некоторые модемы Supra обрабатывают DCD иначе, чем другие модемы. Если вы используете Supra, попробуйте установить &C0, а не &C1. Вы должны также установить &D2, чтобы корректно обрабатывать DTR.

6.2 Установка getty_ps

Получить последнюю версию можно на sunsite.unc.edu:/pub/Linux/system/serial.

В частности, если вы хотите использовать высокие скорости (57600 и 115200 бит\сек), вы должны взять версию 2.0.7j или выше. Вы должны также иметь libc 5.x или выше.

По умолчанию, getty_ps будет сконфигурирован в соответствии с Linux FSSTND (Стандарт Файловой системы), что означает, что бинарники будут в /sbin, а файлы конфигурации будут называться /etc/conf.{uu}getty.ttySN.

Это не указано в документации! Также ожидается, что файлы блокировки будут в /var/lock. Удостоверьтесь, что у вас есть каталог /var/lock.

Если вы не хотите совместимости с FSSTND, бинарники расположатся в / и т.д, файлы конфигурации расположатся в /etc/default/{uu}getty.ttySN, и файлы блокировки в /usr/spool/uucp. Я рекомендую этот способ, если вы используете UUCP, потому что с UUCP возникнут проблемы, если вы переместите файлы блокировки туда, где он не ищет их.

getty_ps может также использовать syslogd, чтобы регистрировать сообщения. См. man для syslogd(1) и syslog.conf(5) для настройки syslogd, если он еще у вас не запущен. Сообщения регистрируются с приоритетом LOG_AUTH, ошибки используют LOG_ERR и отладка использует LOG_DEBUG. Если вы не хотите использовать syslogd, вы можете отредактировать tune.h в исходных файлах getty_ps, чтобы использовать регистрационный файл для сообщений вместо файла регистрации по умолчанию, именуемого /var/adm/getty.

Решите, что вы хотите - FSSTND совместимость или возможности syslog. Вы можете также выбирать комбинацию из двух. Отредактируйте Makefile, tune.h и config.h, чтобы отразить ваше решение. Затем скомпилируйте и установите согласно инструкциям, включенных в пакет.

> От этой отметки, все относящееся к getty будет относиться к getty_ps.

Ссылки на uugetty относятся к uugetty, который приходит с пакетом getty_ps. Эти инструкции не будут работать с mgetty или agetty.

7 uugetty

7.1 Установка uugetty

Удостоверьтесь, что ваше выходное и входное устройство для последовательного порта - ваш модем включен. Если у вас модем стоит на ttyS3, вам понадобятся устройства /dev/cua3 и /dev/ttyS3. Если у вас неправильные устройства, см. раздел "Создание ttyS устройств в /dev" о том, как создать устройства, и создайте устройства. Если вы хотите звонить наружу вашим модемом в то время, как uugetty ожидает на порту регистрации в системе, то используйте устройство /dev/cuaN вместо устройства /dev/ttySN

[Можно было бы обойтись и без cua, если программное обеспечение не требует их.]
uugetty делает проверку важного файла блокировки. Подредактируйте /etc/gettydefs, чтобы включить в него запись для вашего модема. Когда вы завершите редактирование /etc/gettydefs, вы можете проверить правильность синтаксиса, сделав:

```
linux# getty -c /etc/gettydefs
```

7.1.1 Современные модемы

Если у вас модем со сжатием данных и скоростью 9600 бит\сек или более быстрый, вы можете заблокировать ваш последовательный порт на одну скорость. Например:

```
# 115200 fixed speed
F115200# B115200 CS8 # B115200 SANE -ISTRIP HUPCL #S @L @B login: #F115200
```

Если у вас ваш модем настроен на аппаратное управление потоком данных RTS/CTS, вы можете добавить в запись CRTSCTS:

```
# 115200 fixed speed with hardware flow control
F115200# B115200 CS8 CRTSCTS # B115200 SANE -ISTRIP HUPCL CRTSCTS #S @L @B login: #F115200
```

7.1.2 Старые медленные модемы

Если у вас медленный модем (меньше 9600 бит\сек), то вместо одной строки для одной скорости, вам потребуется несколько строк, чтобы попробовать ряд скоростей.

Обратите внимание, что эти строки связаны друг другу последним "словом" в строке, типа #38400. Между каждой записью необходимо оставлять пустые строки.

```
# Modem entries
115200# B115200 CS8 # B115200 SANE -ISTRIP HUPCL #S @L @B login: #57600

57600# B57600 CS8 # B57600 SANE -ISTRIP HUPCL #S @L @B login: #38400

38400# B38400 CS8 # B38400 SANE -ISTRIP HUPCL #S @L @B login: #19200

19200# B19200 CS8 # B19200 SANE -ISTRIP HUPCL #S @L @B login: #9600

9600# B9600 CS8 # B9600 SANE -ISTRIP HUPCL #S @L @B login: #2400

2400# B2400 CS8 # B2400 SANE -ISTRIP HUPCL #S @L @B login: #115200
```

vv

<sect2>Сообщение при входе в систему

<p>

Если хотите, вы можете заставить uugetty печатать интересные вещи при входе в систему. В моих примерах, я ставлю имя системы, линию и текущую скорость бит\сек. Вы можете добавить и другие вещи:

<tscreen><verb>

```

@B Текущая скорость (оценивается при определении значения @B )
    в бит/сек.
@D Текущая дата в формате ММ/ДД/YY.
@L Последовательная линия, с которой работает getty.
@S Имя системы.
@T Текшее время в формате НН:ММ:SS (24-часовой).
@U Число работающих в данный момент пользователей. В качестве
    этого значения берется количество записей в файле /etc/utmp,
    которые имеют непустое поле ut_name.
@V Значение VERSION, как задано в файле defaults.
Чтобы отобразить символ '@', используйте или '\@', или '@@'.

```

7.2 Настройка uugetty

Имеется большое количество параметров, которые вы можете изменить для каждого имеющегося у вас порта.

Они выполнены в виде отдельного настроечного файла для каждого порта. Файл /etc/conf.uugetty будет использоваться всеми процессами uugetty, а /etc/conf.uugetty.ttySN будет использоваться только этим одним портом. Пример файлов конфигурации по умолчанию, может быть найден в исходниках getty_ps, которые есть в большинстве Linux-дистрибутивах. Из-за экономии места они здесь не приводятся. Обратите внимание, что, если вы используете старые версии getty (старее, чем 2.0.7e), или не используете FSSTND, то заданный по умолчанию файл будет /etc/default/uugetty.ttySN. Файл /etc/conf.uugetty.ttyS3 Грэга примерно такой:

```

# пример настроечного файла uugetty для Hayes-совместимых модемов
# с разрешением входящих звонков
#
# проверка наличия lock файла... Если этот lock файл есть, то uugetty
# перезапускается так, чтобы модем был заново проинициализирован
ALTLOCK=cua3
ALTLINE=cua3
# какой порт инициализировать
INITLINE=cua3
# время таймаута, после которого связь обрывается ...
TIMEOUT=60
# строка инициализации модема...
# формат: <ожидаемое> <отправляемое> ... (чат-последовательность)
INIT="" AT\r OK\r\n
WAITFOR=RING
CONNECT="" ATA\r CONNECT\sA
# эта строка устанавливает задержку перед выдачей сообщения при входе в
# систему
DELAY=1
#DEBUG=010

```

Добавьте следующую строку в ваш /etc/inittab, так, чтобы uugetty запускался на вашем последовательном порту (заменяя значения в соответствии с вашей системой - расположение файла конфигурации, порт, быстродействие и заданный по умолчанию тип терминала):

```
S3:456:respawn:/sbin/uugetty -d /etc/default/uugetty.ttyS3 ttyS3 F115200 vt100
```

Рестарт init:

```
linux# init q
```

Скорость в вашем /etc/inittab указывается самой высокой, которую поддерживает ваш модем. Теперь Linux будет следить за появлением запросов на соединение на вашем последовательном порту. Можно звонить с других машин и входить в вашу систему.

uugetty имеет еще множество опций, см. полное описание в man по getty(1m). Среди прочего имеется возможность планирования и возможность обратного звонка.

8 Как настроить терминальный вход на моем PC?

Детально это описывается в Text-Terminal-HOWTO (но он все еще незавершен). Инструкции в этом разделе будут работать как для подсоединенных терминалов, так и для других компьютеров, подключенных к последовательным портам вашей Linux машины.

8.1 Аппаратные требования

Удостоверьтесь, что у вас соответствующий кабель. Нуль-модемный кабель, купленный в компьютерном магазине, может подойти, но вероятно, что он не сможет работать с аппаратным управлением потоком данных. Но он должен быть нуль- модемным кабелем! Многие компьютерные магазины выставляют этот кабель как последовательный кабель принтера. Удостоверьтесь, что вы используете ваш последовательный порт (разъем-"папу"(входящий в ответную часть) DB25 или DB9), а не ваш параллельный порт (разъем-"маму"(охватывающий) DB25 или centronics).

Для разъема DB25, вам нужно минимум:

PC "папа" DB25			Terminal DB25	
TxD	Передача данных	2 -->	3	RxD Прием данных
RxD	Прием данных	3 <--	2	TxD Передача данных
SG	Сигнальная земля	7 ---	7	SG Сигнальная земля

Этот нуль-модемный кабель хорошо работает между двумя компьютерами, которые используют сигналы хендшейкинга RTS/CTS, но многие терминалы используют сигнал хендшейкинга DTR, которое не может быть установлен в linux с помощью программы "stty". Включите в stty crtscts и соедините штырек терминала 20 (DTR) со штырьком PC 5 (CTS).

PC "папа" DB25			Terminal DB25	
TxD	Передача данных	2 -->	3	RxD Прием данных
RxD	Прием данных	3 <--	2	TxD Передача данных
RTS	Запрос передачи	4 -->	5	CTS Готов к передаче
CTS	Готов к передаче	5 <--	4	RTS Запрос передачи
DSR	Data Set Ready		6	
DCD	Обнаружение несущей	8 <--	20	DTR Готовность данных терминала
SG	Сигнальная земля	7 ---	7	SG Сигнальная земля
			6	DSR Data Set Ready
DTR	Готовность данных терминала	20 -->	8	DCD Обнаружение несущей

Если на вашем последовательном порту стоит разъем DB9, то попробуйте следующую распайку:

PC DB9			Terminal DB25	
TxD	Передача данных	2 -->	2	RxD Прием данных

RxD	Прием данных	3 <-- 3	TxD	Передача данных
SG	Сигнальная земля	5 --- 7	SG	Сигнальная земля

В качестве альтернативы, вот полный DB9-DB25 нуль-модемный кабель (Вряд ли будет работать с аппаратным хендшейкингом терминала; см. выше):

PC DB9			Terminal DB25	
TxD	Передача данных	2 --> 2	RxD	Прием данных
RxD	Прием данных	3 <-- 3	TxD	Передача данных
		6	DSR	Data Set Ready
DTR	Готовность данных терминала	4 --> 8	DCD	Обнаружение несущей
GND	Сигнальная земля	5 --- 7	GND	Сигнальная земля
DCD	Обнаружение несущей	1		
DSR	Data Set Ready	6 <-- 20	DTR	Готовность данных терминала
RTS	Запрос передачи	7 --> 5	CTS	Готов к передаче
CTS	Готов к передаче	8 <-- 4	RTS	Запрос передачи
(RI	Ring Indicator	9 не обязательно)		

(Да, штырьки 2 и 3 в разъеме DB9 действительно имеют значения противоположные значениям в разъеме DB25!)

Если вы не используете полный нуль-модемный кабель, то вам возможно придется применить следующий прием: на разъеме с компьютерной стороны, соедините RTS и CTS вместе, а также вместе соедините DSR, DCD и DTR. Таким образом, когда компьютер захочет получить сигнал хендшейкинга, он получит его (от самого себя).

Теперь, когда у вас правильный кабель?, соедините ваш терминал с вашим компьютером. Если можете, скажите терминалу игнорировать сигналы управления модемом. Пробуйте использовать в установках терминала скорость 9600 бит\сек, 8 информационных битов, 1 стоповый бит, без битов контроля по четности (или разрешите контроль по четности, если вы хотите дополнительных сложностей).

8.2 Установка getty (часть getty_ps)

Проконсультируйтесь с вашим руководством по getty для других программ getty, например, agetty. Добавьте записи для getty в файл /etc/gettydefs, чтобы использовать ваш терминал, если их там до сих пор нет:

```
# 38400 bps Dumb Terminal entry
DT38400# B38400 CS8 CLOCAL # B38400 SANE -ISTRIP CLOCAL #@S @L login: #DT38400

# 19200 bps Dumb Terminal entry
DT19200# B19200 CS8 CLOCAL # B19200 SANE -ISTRIP CLOCAL #@S @L login: #DT19200

# 9600 bps Dumb Terminal entry
DT9600# B9600 CS8 CLOCAL # B9600 SANE -ISTRIP CLOCAL #@S @L login: #DT9600
```

Если хотите, вы можете заставить ugetty печатать интересные вещи при входе в систему. В моих примерах, я ставлю имя системы и текущую линию. Вы можете добавить и другие вещи:

```
@B Текущая скорость (оценивается при определении значения @B )
в бит/сек.
@D Текущая дата в формате ММ/ДД/YY.
@L Последовательная линия, с которой работает getty.
@S Имя системы.
```



```

@T    Текщее время в формате HH:MM:SS (24-часовой).
@U    Число работающих в данный момент пользователей. В качестве
      этого значения берется количество записей в файле /etc/utmp,
      которые имеют непустое поле ut_name.
@V    Значение VERSION, как задано в файле defaults.
      Чтобы отобразить символ '@', используйте или '\@', или '@@'.

```

Когда вы завершили редактирование файла /etc/gettydefs, вы можете проверить правильность синтаксиса, сделав:

```
linux# getty -c /etc/gettydefs
```

Удостоверьтесь, что у вас нет файла конфигурации getty или uugetty для последовательного порта, к которому подключен ваш терминал (/etc/default/{uu}getty.ttySN или /etc/conf.{uu}getty.ttySN), поскольку он вероятно будет пересекаться с getty, запущенном на этом терминале. Удалите файл, если он существует.

Отредактируйте ваш файл /etc/inittab, чтобы запустить getty на последовательном порту (установив для вашего окружения правильные порт, скорость и заданный по умолчанию тип терминала):

```
S1:456:respawn:/sbin/getty ttyS1 DT9600 vt100
```

Перезапустите init:

```
linux# init q
```

В этом месте, вы должны на терминале увидеть подсказку для входа в систему. Вам, вероятно, придется нажать Enter, чтобы привлечь внимание терминала.

8.3 Примечания о настройке PC в качестве терминала

Многие люди настраивают другие PC как терминалы, соединенные с Linux машинами. Например, старые 8088 или 80286 PC отлично подходят для этой цели. Все, в чем вы нуждаетесь для запуска - это загрузочный диск DOS, содержащий версию DOS, подходящую вашему PC-терминалу, и программы связи для вашего PC-терминала. Kermit очень хорошо подходит для этой цели. Вы можете найти скомпилированные версии kermit для каждой OS на <http://www.columbia.edu/kermit/>. Также хорошо будут работать другие популярные программы связи DOS типа telix и procomm. Убедитесь, что ваши настройки PC-терминала содержат правильную информацию о последовательном порте.

9 Я могу использовать более, чем два последовательных устройства?

Вы не должны читать этот раздел, если не хотите использовать три или более последовательных устройства ... (принимая, что у вас нет многопортовой платы).

Если у вас есть другой запасной последовательный порт, то да, вы можете. Число последовательных портов, которые вы можете использовать, ограничено числом прерываний (IRQs) и адресов портов ввода-вывода. Это - ограничение не системы Linux, а ограничение шины PC. Каждому последовательному устройству должны быть назначены адрес и собственное прерывание.

Если два устройства совместно используют одно и тоже прерывание, то они могут нормально работать, при условии, что эти два устройства не будут действовать одновременно. (В большинстве

случаев, две программы не должны даже выполняться одновременно.) Последовательное устройство может быть последовательным портом, внутренним модемом или многопортовой последовательной платой.

Многопортовые последовательные платы специально разработаны для получения множества последовательных портов, которые совместно используют одно и то же IRQ для всех последовательных портов на плате.

Lіnux получает данные из них, используя различные адреса ввода-вывода для каждого порта на плате.

9.1 Выбор прерываний последовательных устройств

Обычный PC имеет ttyS0 и ttyS2 на IRQ 4, и ttyS1 и ttyS3 на IRQ 3. Вы можете увидеть, какие IRQs используются, напечатав: `setserial /dev/ttyS2`, и т.д. Просмотр `/proc/interrupts` покажет некоторых из них. Чтобы использовать более, чем два последовательных устройства, вы должны будете переназначить прерывания. Хороший подход состоит в переназначении прерывания для параллельного порта. В PC обычно для ваших параллельных портов установлены прерывания IRQ 5 и IRQ 7, но мало кто использует два параллельных порта. Вы можете переназначать одно из прерываний на последовательное устройство, и прекрасно использовать оставшийся параллельный порт. чтобы сделать это, вам нужна будет программа `setserial`. Кроме того, вы должны поиграть с перемычками на материнской плате, свериться с документацией на вашу плату. Установите перемычки на IRQ, который вы хотите назначить каждому порту.

Вы должны установить все так, чтобы имелось одно и только одно прерывание для каждого последовательного устройства. Вот как Грэг устанавливает его в `/etc/rc.d/rc.local` - вы должны сделать это в файле, который выполняется после запуска:

```
/sbin/setserial /dev/ttyS0 irq 3      # моя последовательная мышь
/sbin/setserial /dev/ttyS1 irq 4      # мой Wyse dumb терминал
/sbin/setserial /dev/ttyS2 irq 5      # мой модем Zoom
/sbin/setserial /dev/ttyS3 irq 9      # мой модем USR
```

Стандартные назначения IRQ:

IRQ 0	Таймер, канал 0
IRQ 1	Клавиатура
IRQ 2	Cascade for controller 2
IRQ 3	Последовательный порт 2
IRQ 4	Последовательный порт 1
IRQ 5	Параллельный порт 2
IRQ 6	Накопитель на гибких дисках
IRQ 7	Параллельный порт 1
IRQ 8	Часы реального времени
IRQ 9	Перенаправлен на IRQ2
IRQ 10	не назначен
IRQ 11	не назначен
IRQ 12	не назначен
IRQ 13	Математический сопроцессор
IRQ 14	Накопитель на жестких дисках 1
IRQ 15	Накопитель на жестких дисках 2

В действительности нет никакого "правильного способа" назначения прерывания. Только удостоверьтесь, что оно не используется материнской платой или любыми другими платами.

Номера прерываний 2, 3, 4, 5 или 7 - это хороший выбор. "не назначен" означает, что в настоящее время ничего стандартно не использует эти IRQS. Также обратите внимание, что IRQ 2 - это тоже, что и IRQ 9. Вы можете вызывать его или как 2 или как 9, последовательный драйвер очень

10. Как мне настроить последовательные порты на более высокие скорости? Какую скорость я должен задать

понятливый. Если у вас последовательная плата с 16-разрядным разъемом шины, то вы можете также использовать IRQ 10, 11, 12 или 15.

Только удостоверьтесь, что вы не используете IRQ 0, 1, 6, 8, 13 или 14! Они используются вашей материнской платой. Вы можете заработать множество неприятностей, используя эти IRQ не по назначению. Когда вы закончите, посмотрите `/proc/interrupts` и удостоверьтесь, что конфликтов нет.

9.2 Установка адресов последовательных устройств

Затем, вы должны установить адрес порта. Проверьте в руководстве по вашей плате установки переключателей. Подобно прерываниям, одно последовательное устройство может иметь только один адрес. Ваши порты обычно будут приходиться с такими установками:

```
ttyS0 адрес 0x3f8
ttyS1 адрес 0x2f8
ttyS2 адрес 0x3e8
ttyS3 адрес 0x2e8
```

Выберите адреса, которые вы хотите использовать для каждого последовательного устройства и соответственно установите переключатели. Мой модем установлен на `ttyS3`, мышь на `ttyS0`, а мой терминал на `ttyS2`.

Когда вы перезагрузитесь, Linux должен увидеть ваши последовательные порты по тем адресам, на которые вы их установили. IRQ, который видит Linux, может не соответствовать IRQ, который вы устанавливаете переключателями. Не волнуйтесь об этом. Linux не делает попыток определить IRQ при загрузке, потому что определение IRQ рискованно и может быть неправильным. Используйте `setserial`, чтобы сообщить Linux, какое IRQ используется портом. Вы можете проверить `/proc/ioprots`, чтобы увидеть, какие адреса порта ввода-вывода используются после загрузки Linux.

10 Как мне настроить последовательные порты на более высокие скорости? Какую скорость я должен задать для моего модема?

Этот раздел должен помочь вам выяснить, какую скорость нужно использовать в программах связи или `getty`, которые работают в вашем модеме.

Обычно вы только устанавливаете скорость параметром, заданным в `getty` или меню вашей коммуникационной программы (или в конфигурационном файле). Если по некоторым причинам, это не работает для высоких скоростей, то имеется возможность задать специальные опции в `setserial`: с помощью опций `spd_hi` или `spd_vhi`, задаваемыми в `setserial`, установка 38400 фактически даст скорость 57600 (`spd_hi`) или 115200 (`spd_vhi`).

Удостоверьтесь, что для более высоких скоростей у вас имеется по крайней мере 16550A UARTS. Существуют некоторые предлагаемые скорости:

- Если у вас 28.8K (V.34), 33.6K (V.34) или 56K (V.90) модем, используйте 115200 бит\сек.
- Если у вас 14400 бит\сек (V.32bis) модем со сжатием данных V.42bis, используйте 57600 бит\сек.
- Если у вас 9600 бит\сек (V.32) модем со сжатием данных V.42bis, используйте 38400 бит\сек.
- Если ваш модем работает медленнее, чем на 9600 бит\сек (V.32), то установите скорость на самую высокую, какую только поддерживает ваш модем. Например, 300 бит\сек (V.21/Bell 103), 1200 бит\сек (V.22/Bell 212A) или 2400 бит\сек (V.22bis).

Если вы используете `setserial`, то потестируйте его команды сперва в командной строке, а затем, когда убедитесь, что все работает, поместите их в `/etc/rc.d/rc.serial` или `/etc/rc.d/rc.local` или `/etc/rc.boot/0setserial` так, чтобы они выполнялись при загрузке системы. Удостоверьтесь, что вы используете правильный путь для `setserial` и правильное имя устройства. Вы можете проверить установки последовательного порта командой:

```
setserial -a /dev/ttyS3
```

11 Коммуникационные программы и утилиты

Вот список нескольких коммуникационных программ, которые доступны в Интернет через FTP, если их нет в вашем дистрибутиве.

- `esu` - программа связи
- *C-Kermit* - переносная scriptable, последовательная и TCP/IP связь, включая передачу файлов и трансляцию символьных наборов
- `minicom` - telix-подобная программа связи
- `procomm` - procomm-подобная коммуникационная программа с протоколом `zmodem`
- `seyon` - коммуникационная программа под X
- `xs` - коммуникационный пакет `xcomm`
- также имеются полезные программы `term` и `SLiRP`. Они предоставляют функциональные возможности TCP/IP, используя учетную запись `shell`.
- `screen` - другая многосессионная программа. Она ведет себя как виртуальные консоли.
- `callback` - программа, которая заставит ваш модем звонить вам обратно.
- `mgetty+fax` работает с факсами и является альтернативой `ps_getty`.
- `ZyXEL` - управляющая программа для модемов `ZyXEL U-1496`. Она обеспечивает защиту входящих, исходящих и обратных звонков, обработку факсов и голосовые почтовые ящики.
- Программное обеспечение `SLIP` и `PPP` можно найти на <ftp://sunsite.unc.edu/pub/Linux/system/network/serial>.
- Другие вещи можно найти на <ftp://sunsite.unc.edu/pub/Linux/system/serial> и <ftp://sunsite.unc.edu/pub/Linux/apps/serialcomm> или на одном из зеркал. Это каталоги, где хранятся программы для работы с последовательными устройствами.

12 Советы и разное по последовательным устройствам

Вы можете счесть полезными несколько советов ...

12.1 kermit и zmodem

Чтобы использовать `zmodem` с `kermit`, добавьте следующее к вашему `.kermitrc`:

```
define rz !rz < /dev/ttyS3 > /dev/ttyS3 define sz !sz \%0 > /dev/ttyS3 < /dev/ttyS3
```

Убедитесь, что указали правильный порт, к которому подключен ваш модем. Затем, чтобы использовать его, нужно только напечатать `rz` или `sz <имя файла>` в командной строке `kermit`.

12.2 Автоматическая настройка типов терминала

Чтобы тип вашего терминала автоматически устанавливался, когда вы регистрируетесь в системе, добавьте тип терминала в `/etc/inittab`. Если бы у меня к порту `ttyS1` был подключен терминал `vt100`, то я добавил бы "vt100" в команду `getty`:

```
S1:456:respawn:/sbin/getty ttyS1 DT9600 vt100
```

Другие `getty` имеют подобные опции. Вы можете также использовать `tset`, который может устанавливать характеристики терминала, когда вы регистрируетесь в системе, и не зависит от каких-либо значений по умолчанию.

12.3 Цвета ls при последовательной связи

Если `ls` выдает результат своей работы в цвете на терминалах, поддерживающих цвет, отключите это. Включение цвета осуществляется опциями `ls -color` и `ls -colour`.

В некоторых дистрибутивах `ls` настроен с цветом по умолчанию. Проверьте в файлах `/etc/profile` и `/etc/csh.cshrc` псевдонимы `ls`. Также вы можете установить псевдоним `ls` в значение `ls -no-color`, если не хотите изменять системные значения по умолчанию.

12.4 Печать на принтер, соединенный с терминалом

Имеется программа, называемая `vtprint`, доступная на <ftp://ftp.sdsu.edu/pub/vtprint> и на <http://www.sdsu.edu/garrett>.

Другая программа, которая будет делать это, называется `xprt`. Ее можно найти на <ftp://sunsite.unc.edu/pub/Linux/system/printing>.

12.5 Может ли Linux настроить последовательные устройства автоматически?

Да. Если это еще не настроено (или почти настроено), то вы можете заставить Linux обнаруживать и настраивать последовательные устройства автоматически при запуске. Если необходимо добавьте строку:

```
/sbin/setserial /dev/ttyS3 auto_irq skip_test autoconfig
```

в ваш файл `/etc/rc.d/rc.local` или `/etc/rc.d/rc.serial` или в файл `/etc/rc.boot/0setserial` файл. Прделайте это для каждого последовательного порта, который вы хотите настроить автоматически. Убедитесь, что используете имена реально существующих в вашей системе устройств.

12.5.1 Примечания для многопортовых плат

Адреса и IRQS платы смотрите в файлах `rc.serial` или `/etc/rc.boot/0setserial`, которые поставляются с программой `setserial`. В них можно найти множество подробностей о многопортовых платах, включая адреса ввода-вывода и имена устройств.

12.6 Использование последовательной консоли

Имеется статья в выпуске *36 Linux Journal*, которая объясняет, как использовать последовательную консоль. См. `Text-Terminal-HOWTO`.

12.7 Повышение производительности последовательных устройств

Если вы наблюдаете медленную производительность и последовательный порт слишком быстр для системы с (E)IDE дисками, то вы можете взять пакет `hdparm`. Это утилита, которая может изменять параметры (E)IDE устройств, включая `unmasking` других IRQs при выполнении дискового IRQ. Это ускорит отклик и поможет решить проблему обслуживания последовательных устройств. Убедитесь, что вы очень тщательно прочитали `man` страницу, так как некоторые комбинации привода/контроллера могут быть некорректными, и файловая система может разрушиться. Также обратите внимание на утилиту `irqtune`, который изменяет приоритет IRQ устройства, например последовательного порта, к которому подключен ваш модем. Это может улучшить производительность последовательного устройства на вашей системе. `Irqtune FAQ` можно найти на <http://www.best.com/cae/irqtune>.

13 На шаг дальше ...

Этот раздел не обязателен для прочтения, но может дать вам более глубокие знания Unix и мира телекоммуникаций.

13.1 Что такое lock файлы (файлы блокировки)?

Файл блокировки - это просто файл, говорящий, что указанное устройство в настоящее время занято (используется).

Они хранятся в `/usr/spool/uucp` или `/var/lock`. В Linux файлы блокировки именовются `LCK.name`, где `name` - это или имя устройства, или имя UUCP сайта. Некоторые процессы создают эти файлы блокировки для того, чтобы иметь монопольный доступ к устройству. Например, если вы делаете исходящий звонок вашим модемом, появится файл блокировки, сообщая другим процессам, что кто-то уже использует модем. Файл блокировки в основном содержат PID процесса, который занял устройство. Большинство программ проверяют наличие файла блокировки, и пытаются определить, является ли эта блокировка все еще допустимой, производя в таблице процессов поиск процесса, который заблокировал устройство. Если определяется, что файл блокировки действителен, то программа (возможно) заканчивает работу. Если нет, то некоторые программы удаляют устаревший файл блокировки, и используют устройство, создавая собственный файл блокировки. Другие программы только завершают работу и сообщают вам, что устройство занято процессом.

13.2 "бод" или "бит\сек"

"бод" и "бит\сек" являются возможно одним из наиболее часто неправильно применяемых терминов в области компьютеров и телекоммуникаций. Многие люди считают эти термины взаимозаменяемыми, хотя фактически это не так!

бод (baud)

Скорость в бодах - это количество посланных секунду сигналов, например одно посланное модемом (модем - это сокращение от "модулятор-демодулятор") изменение. Например, скорость в 1200 бод подразумевает, что один сигнал изменяется каждые 833 микросекунды. Обычные скорости модемов в бодах - 50, 75, 110, 300, 600, 1200 и 2400. Большинство высокоскоростных модемов могут работать со скоростями 2400 бод. Из-за ограничений ширины полосы частот на голосовых телефонных линиях, достичь скорости более 2400 бод тяжелее, и работает только на линиях с `pristine` качеством. В одном боде можно закодировать несколько бит, и таким образом получится, что скорость в битах превышает скорость в бодах. "бод" назван по имени Эмиля Баудота (Emile Baudot), изобретателя асинхронного телеграфного принтера.

бит\сек (bps)

Скорость бит\сек - количество переданных за одну секунду бит. Раньше для модемов обычными скоростями бит\сек были 50, 75, 110, 300, 1200, 2400, 9600. Сегодня это - 14.4Кб, 28.8Кб, 33.6Кб, и 56Кб (они не соответствуют скоростям в кабеле "последовательный порт-модем", которые дополняют старые модемные скорости бит\сек (50-9600): 19.2Кб, 38.4Кб, 57.6Кб и 115.2Кб). При использовании модемов со сжатием V.42bis (максимальное сжатие 4:1), возможно достижение скорости до 115.2Кб/сек. Вот что имеет в виду большинство людей, когда они неправильно применяют слово "бод".

Итак, если высокоскоростные модемы работают на 2400 бод, как они умудряются посылать 14400 бит\сек (или больше)? Модемы достигают скоростей бит\сек больших, чем скорости в бодах, кодируя несколько битов при каждом изменении сигнала или смене фазы. Таким образом, когда в одном бод закодировано 2 или больше битов, скорость бит\сек превышает скорость в бодах. Если ваш модем соединяется со скоростью 14400 бит\сек, он собирается посылать 6 битов при каждом изменении фазы, со скоростью 2400 бод.

Как начался этот бардак? Ну, когда модемы работали медленно, скорость бит\сек фактически равнялась скорости в бодах. Одним изменением фазы кодировался один бит. Термины бит\сек и бод были взаимозаменяемыми, потому что имели одинаковые значения. Например, модем 300 бит\сек также имел скорость 300 бод. Все изменилось, когда появились более быстрые модемы, и скорость передачи информации в битах превысила скорость в бодах.

13.3 Что такое UART? Как они воздействуют на производительность?

UART (Universal Asynchronous Receiver Transmitter - Универсальный Асинхронный Приемопередатчик) - это микросхема на вашей мультикарте или материнской плате. Функции UART может выполнять и другая микросхема, которая занимается кроме этого другими вещами.

Назначение UART состоит в том, чтобы преобразовать данные в биты, посылать биты по последовательной линии и затем восстанавливать данные на другом конце линии. UART имеет дело с данными, разбитыми побайтно, что удобно для представления символов ASCII.

Скажем, у вас есть терминал, подключенный к PC. Когда вы печатаете символ, терминал передает этот символ своему передатчику (тоже UART). Передатчик посылает этот байт наружу по последовательной линии побитно с определенной скоростью. UART на PC конце линии получает все эти биты и восстанавливает из них байт и помещает его в буфер.

Есть два различных типа UART. Вы вероятно слышали о dumb UART - 8250 и 16450, и FIFO UART - 16550A. Чтобы понять их различия, сначала давайте исследуем, что случается когда UART послал или получил байт.

UART непосредственно не может делать что-либо с данными, он только посылает и получает их. Для первоначальных UART, ЦП получает запрос на прерывание от последовательного устройства каждый раз, когда байт был послан или получен.

Затем ЦП перемещает полученный байт из буфера UART в область оперативной памяти или передает UART другой байт для передачи. В UART 8250 и 16450 буфера однобайтные. Это означает, что за одно прерывание ЦП посылается или принимается только 1 байт. Для низких скоростей это нормально. Но при высоких скоростях передачи, ЦП настолько загружается работой с UART, что хуже обслуживает другие задачи. В некоторых случаях, ЦП не успевает обработать прерывание вовремя, и байт затирается вновь пришедшим, потому что байты поступают в буфер UART слишком быстро.

В этой ситуации полезен UART 16550A. Эти чипы приходят с 16-байтным FIFO. Это означает, что такая микросхема может получать или передавать до 14 байтов прежде, чем она должна будет сформировать запрос на прерывание ЦП. В свою очередь ЦП может передать сразу 14 байтов за один раз. Хотя порог прерывания не всегда устанавливается в 14, все равно имеется значительное преимущество перед другими UART, которые имеют только однобайтный буфер.

ЦП получает меньшее количество прерываний, и имеет больше времени для выполнения других вещей.

Данные не теряются, и все счастливы. (Существует также 16550 UART, но он обрабатывается как 16450, так как имеет дефект.)

В общем, UART 8250 и 16450 должны прекрасно подходить для скоростей до 38400 бит\сек. На скоростях больше 38400 бит\сек, вы могли бы заметить потерю данных. Другие операционные системы для PC (definition used loosely here), вроде DOS, не многозадачные, так что они могли бы лучше работать с 8250 или 16450s. Именно поэтому некоторые люди не замечают потерь данных, пока не начинают работать в Linux.

В не-UART и интеллектуальных многопортовых платах стоят DSP чипы, которые имеют дополнительные буферы и более развитое управление, и таким образом ЦП менее загружается. Например, платы Cyclades Cyclom и Stallion EasyIO используют Cirrus Logic CD1400 RISC UART, а многие платы используют ЦП 80186 или даже специальные RISC ЦП, для обработки последовательного ввода-вывода.

Имейте в виду, что dumb UART не плохи, просто они не очень хороши для высоких скоростей. У вас не должно возникать проблем с подключением терминалов или мышей к этим UART. Но, для высокоскоростных модемов 16550A - определенно обязателен.

На большинстве новых PC (486's, Pentiums или выше) установлены 16550A. Если ваша техника действительно старая, то вы можете нарастить ее вычислительные возможности купив микросхем 16550A и заменив ваши 16450 UART. Если используется другой тип микросхемы, то вам не повезло. Если UART вставляется в "кроватьку", то обновление упрощается (если вы можете найти замену). Новые и старые микросхемы совместимы по значениям выводов. Возможно, более правильно купить последовательную плату через Интернет (некоторые розничные торговцы их продают).

14 Поиск неисправностей

14.1 Я получаю сообщение "line NNN of inittab invalid"

Удостоверьтесь, что вы используете правильный для вашей версии init синтаксис.

Различные init используют различный синтаксис в файле /etc/inittab. Удостоверьтесь, что вы используете правильный синтаксис для вашей версии getty.

14.2 Когда я пытаюсь сделать исходящий звонок, система говорит "/dev/cua N : Device or resource busy"

Эта проблема может возникать, когда DCD или DTR неправильно установлены. DCD должен быть установлен только когда имеется фактическое соединение (то есть кто-то позвонил), а не когда getty наблюдает за состоянием порта. Проверьте, что ваш модем сконфигурирован так, чтобы DCD устанавливался только когда имеется соединение. DTR должен быть установлен всякий раз, когда кто-то использует или наблюдает за линией, вроде getty, kermit или любой другой программы связи. Другая общая причина ошибки "device busy", - что вы настроили ваш последовательный порт на уже кем-то занятое прерывание.

Когда каждое устройство инициализируется, оно запрашивает Linux о своих правах использования аппаратных прерываний. Linux смотрит, какое прерывание назначено устройству, и если это прерывание уже занято, то устройство не сможет корректно проинициализироваться. Устройство не имеет другого способа сообщить вам о своей неправильной инициализации, кроме выдачи сообщения "device busy", когда вы попытаетесь воспользоваться этим устройством. Проверьте прерывания на всех ваших платах (последовательных, ethernet SCSI и т.д.). Поищите конфликты IRQ.

14.3 Я получаю сообщение "getty respawning too fast: disabled for 5 minutes"

Удостоверьтесь, что ваш модем сконфигурирован правильно. Проверьте регистры E и Q. Это может происходить, когда ваш модем общается с getty.

Удостоверьтесь, что в вашем /etc/inittab getty вызывается правильно.

Неправильный синтаксис или имена устройств могут вызвать серьезные проблемы.

Проверьте правильность синтаксиса вашего /etc/gettydefs, делая следующее:


```
linux# getty -c /etc/gettydefs
```

Также такое может случиться при неудачной инициализации `ugetty`. См. раздел "getty или ugetty все еще не работает".

14.4 Последовательных устройства работают медленно, или последовательные устройства могут отправлять данные только в одном направлении

Вероятно у вас конфликт IRQ. Удостоверьтесь, что это не так. Проверьте все ваши платы (последовательные, ethernet, SCSI и т.д ...). Проверьте установки перемычек и параметры `setserial` для всех ваших последовательных устройств. Также попытайтесь обнаружить конфликты в `/proc/ioports` и `/proc/interrupts`.

14.5 Мой модем виснет после того, как кто-то зависает, или ugetty не возрождается повторно

Это может случиться, когда ваш модем не сбрасывается при отсутствии DTR. Я видел как светодиоды RD и SD сходили с ума, когда это случалось. Вы должны сбросить ваш модем. В большинстве Hayes-совместимых модемов это делается командой `&D3`, но на моем USR Courier, я должен был установить `&D2` и `S13=1`. Проверьте ваше руководство по модему.

14.6 Мой терминал, соединен с моим PC, но после того, как я ввожу свое имя для входа в систему, он только блокируется

- Если вы используете `getty`: вероятно у вас не указано `CLOCAL` в файле `/etc/gettydefs` для терминала, и вы вероятно не используете полный нуль-модемный кабель. Вам нужен `CLOCAL`, который заставляет Linux игнорировать сигналы управления модемом. Вот на что это должно быть похоже:

```
# 38400 bps Dumb Terminal entry
DT38400# B38400 CS8 CLOCAL # B38400 SANE -ISTRIP CLOCAL #@S @L login: #DT38400

# 19200 bps Dumb Terminal entry
DT19200# B19200 CS8 CLOCAL # B19200 SANE -ISTRIP CLOCAL #@S @L login: #DT19200

# 9600 bps Dumb Terminal entry
DT9600# B9600 CS8 CLOCAL # B9600 SANE -ISTRIP CLOCAL #@S @L login: #DT9600
```

Затем, уничтожьте процесс `getty`, так что новый будет порожден с новой записью.

- Если вы используете `agetty`: добавьте флажок `-L` в строку `agetty` в файле `/etc/inittab`. Она заставит его проигнорировать сигналы управления модема.

Затем перезапустите `init`, напечатав `init q`. Запись должна выглядеть следующим образом:

```
s1:345:respawn:/sbin/agetty -L 9600 ttyS1 vt100
```

14.7 На высоких скоростях мой модем теряет данные

Если вы попытаетесь заставить работать ваш модем быстрее, чем на 38400 бит\сек, и у вас нет UART 16550A, то вы должны обновить вашу технику. О UART написано в разделе "Что такое UART?".

14.8 При загрузке Linux не указывает настройки последовательных устройств, которые я задал

Это не совсем так. Linux не пытается обнаружить IRQ при запуске, он делает только обнаружение последовательных устройств. Таким образом неважно, что он говорит о IRQ, потому что он принимает только стандартные IRQS.

Но когда при загрузке setserial изменяет IRQ, вы должны увидеть это на экране.

Так, например, при том, что я устанавливаю ttyS2 на IRQ 5, я вижу

```
Jan 23 22:25:28 misfits vmunix: tty02 at 0x03e8 (irq = 4) is a 16550A
```

вначале загрузки Linux. Вы должны использовать setserial, чтобы сообщить Linux IRQ, который вы используете.

14.9 gz и/или sz не работают, когда я звоню на мою машину с Linux модемом

Если Linux ищет /dev/modem, когда вы пытаетесь передавать файлы, посмотрите файлы /etc/profile или /etc/csh.cshrc. Возможно, что в этих файлах определены несколько псевдонимов, как это случается в некоторых дистрибутивах, особенно в Slackware. Эти псевдонимы запутывают программы zmodem. Уберите их или исправьте.

14.10 Мой экран отображает прикольные символы

Это случается на виртуальных консолях, когда вы посылаете двоичные данные на ваш экран, или иногда при последовательных соединениях. Способ устранить это состоит в том, чтобы напечатать echo ^[c. Делается это так:

```
linux% echo <ctrl>v<esc>c
```

14.11 getty или uugetty все еще не работают

Есть опция DEBUG, которая приходит с getty_ps. Отредактируйте ваш настроечный файл /etc/conf.{uu}getty.ttySN и добавьте DEBUG=NNN. Где NNN - одна из следующих комбинаций чисел в зависимости от того, что вы собираетесь отлаживать:

D_OPT	001	установки опций
D_DEF	002	обработка файлов по умолчанию
D_UTMP	004	обработка utmp/wtmp
D_INIT	010	инициализация линии (INIT)
D_GTAB	020	обработка файла gettytab
D_RUN	040	другая диагностика времени выполнения
D_RB	100	отладка ringback
D_LOCK	200	обработка файла блокировки uugetty
D_SCH	400	обработка schedule
D_ALL	777	все вместе

Для первой отладки отлично подходит DEBUG=010.

Если у вас запущен syslogd, отладочная информация появится в ваших log-файлах. Если syslogd не запущен, то информация появится в /tmp/getty.ttySN для отлаживаемого getty и /tmp/uugetty.ttySN для uugetty, и в log-файле /var/adm/getty.log. Посмотрите в отладочной информации, что происходит. Наиболее вероятно, что вам придется настроить некоторые из параметров в вашем файле конфигурации, и перенастроить ваш модем.

Вы могли бы также попробовать mgetty. Для некоторых он проще.

15 Другие источники информации

- man по agetty(8), getty(1m), gettydefs(5), init(1), login(1), mgetty(8), setserial(8)
- Руководство вашнго модема или терминала
- NET-3 HOWTO: все о сетях, включая SLIP, CSLIP и PPP
- PPP HOWTO: помощь в настройке PPP
- Printing HOWTO: для настройки последовательного принтера
- Serial-Programming-HOWTO: некоторые аспекты программирования последовательных портов
- Term HOWTO: все, что вы хотели узнать о программе term
- Text-Terminal-HOWTO: как они работают, как их устанавливать и настраивать
- UPS HOWTO: настройка UPS-датчиков, подключенных к вашему последовательному порту
- UUCP HOWTO: информация о настройке UUCP
- Группы новостей Usenet:

comp.os.linux.answers	FAQs, How-Tos, READMEs итд о Linux.
comp.os.linux.hardware	Аппаратная совместимость с ОС Linux
comp.os.linux.networking	Работа с сетями и связь в Linux.
comp.os.linux.setup	Установка Linux и администрирование системы.
comp.terminals	

- Почтовый список рассылки по последовательным устройствам в Linux. Чтобы подписаться на него, пошлите письмо по адресу majordomo@vger.rutgers.edu, поместив в тело сообщения строку "subscribe linux-serial". Если вы поместите в тело сообщения "help", то вы получите сообщение со справкой. Сервер также обслуживает много других списков почтовой рассылки Linux. Для получения перечня этих списков рассылки отправьте команду "lists".
- Белая Книга, обсуждающая связь по последовательным линиям и многопортовые последовательные платы, доступна на *Cyclades*.
- Часто задаваемые вопросы по модемам:
Navas 28800 Modem FAQ Curt's High Speed Modem Page
- Программирование последовательных устройств:
 - Serial-Programming-HOWTO
 - *Advanced Programming in the UNIX Environment* от W. Richard Stevens (ISBN 0-201-56317-7; Addison-Wesley)
 - *POSIX Programmer's Guide* от Donald Levine (ISBN 0-937175-73-0; O'Reilly)

16 Содействия

В основном первоначальный HOWTO был написан Greg Hankins (greg.hankins@cc.gatech.edu). Он также переработал множество вкладов от других людей, чтобы поддержать непрерывность стиля и последовательность изложения.

Он писал: "Спасибо каждому, кто способствовал работе или комментировал документ, список людей стал слишком длинен, чтобы перечислить (больше сотни) Специальная благодарность Ted T'so за ответы на вопросы о последовательных драйверах, Kris Gleason, который поддерживал getty_ps, и Gert Dцring, который поддерживает mgetty."

КОНЕЦ SERIAL HOWTO