USING PCMCIA SERIAL WITH LINUX

This document contains instructions to help you install and configure the National Instruments serial hardware for Linux. This document includes information about the PCMCIA-232, PCMCIA-232/2, PCMCIA-485, and PCMCIA-485/2 interfaces.

This document assumes that you are already familiar with Linux.

Contents

Related Documentation 2
Contributions 2
Gather What You Need to Get Started 2
Setup 3
Configure /etc/pcmcia/config to Recognize Your PCMCIA Card .. 3
Signal the Card Manager to Reload /etc/pcmcia/config 4
Find Out What Devices Were Assigned to Your Card 4
Configuration 5
View Your Hardware Resources 5
Enable FIFO Buffers 5
FIFO Example 6
Configure struct termios 6
Select RS-485 Transceiver Mode 6
rs485 Example 7
Test the Configuration 7
Sample /etc/pcmcia/config after Edit 8

`natinst.com`™, *National Instruments™, and NI-Serial™ are trademarks of National Instruments Corporation. Product and company names mentioned herein are trademarks or trade names of their respective companies.*

 © Copyright 1999 National Instruments Corp. All rights reserved.

Related Documentation

The following documents contain information that you might find helpful as you read this document:

- *Linux Serial-Programming-HOWTO* by Peter Baumann. You can find the latest version of this document at the following locations:

  `ftp://metalab.unc.edu/pub/Linux/docs/HOWTO/Serial-Programming-HOWTO`

  `http://metalab.unc.edu/LDP/HOWTO/Serial-Programming-HOWTO.html`

- *Linux Serial-HOWTO* by David Lawyer. You can find the latest version of this document at the following locations:

  `ftp://metalab.unc.edu/pub/Linux/docs/HOWTO/Serial-HOWTO`

  `http://metalab.unc.edu/LDP/HOWTO/Serial-HOWTO.html`

- *Linux PCMCIA-HOWTO* by David Hinds. You can find the latest version of this document at the following locations:

  `ftp://metalab.unc.edu/pub/Linux/docs/HOWTO/PCMCIA-HOWTO`

  `http://metalab.unc.edu/LDP/HOWTO/PCMCIA-HOWTO.html`

Contributions

Thanks to Vern Howie for providing suggestions and examples from his serial suite. Also, thanks to David Hinds, David Lawyer, Greg Hankins, and Peter Baumann for providing so much information in their HOWTOs.

Gather What You Need to Get Started

Before you install your PCMCIA serial card for Linux, make sure you have the following:

- Linux kernel version 2.2.5 or later. The product has been thoroughly tested with kernel version 2.2.5; however, the product might work with earlier kernel versions.

  If you do not have kernel version 2.2.5 or later, or if you do not have the following options already compiled into your kernel, you need to recompile your kernel. Include the following Character Devices options when you configure and recompile the kernel using `make menuconfig`:

  – Standard/generic dumb serial support

  – Extended dumb serial driver options

– Support more than four serial ports
– Support for sharing serial interrupts

- `Card Services (pcmcia-cs)` 3.0.9 or later. To find the version of `Card Services`, enter the following:

  ```
  linux# cardctl -V
  ```

  The most recent version of `Card Services` can be found at `ftp://csb.stanford.edu/pub/pcmcia`.

- `PCMCIA-SERIAL.tar.gz`. You can download this file from the National Instruments FTP site at `ftp://ftp.natinst.com/support/ind_comm/serial/linux/`

  After you have the file, extract and unzip it by entering the following:

  ```
  linux# tar –zxvf PCMCIA-SERIAL.tar.gz
  ```

  The `tar` command extracts and unzips `PCMCIA-SERIAL.tar.gz` and creates the subdirectory `PCMCIA-SERIAL`. Enter the following to make sure all necessary files are included:

  ```
  linux#                cd PCMCIA-SERIAL
  linux PCMCIA-SERIAL#  ls
     FIFOtrigger    rs485        serialtest
     FIFOtrigger.c  rs485.c      serialtest.c
     termios_program.c
  ```

- You need superuser privileges to access to most of the steps and program segments in this document. To get superuser privileges, you can either log in as root or enter the following:

  ```
  linux$ su
  Password: <enter root password>
  ```

  To return to regular user privileges, enter the following:

  ```
  linux# exit
  ```

Setup

You need to complete the first two steps in this section before you insert your PCMCIA serial card. You can use any text editor you are comfortable with.

Configure /etc/pcmcia/config to Recognize Your PCMCIA Card

Add your card to the list of cards and modify the `serial_cs` device so that the PCMCIA card manager knows what driver to link to the card.

1. To modify the device in the `/etc/pcmcia/config` file, enter the following:

   ```
   linux# pico /etc/pcmcia/config
   ```

2. In the file, edit the `device "serial_cs"` section to the following:

```
device "serial_cs"
   class "serial" module "misc/serial","serial_cs"
```

Also, add the following to the file after the declaration of the devices, but before the declaration of the other cards. (To view an edited file, refer to *Sample /etc/pcmcia/config after Edit*.) If you are using card services version 3.0.13 or later, you do not need to add the National Instruments cards to `/etc/pcmcia/config`.

```
card "National Instruments PCMCIA-232"
   manfid 0x10b, 0x0d50
   bind "serial_cs"
card "National Instruments PCMCIA-232/2"
   manfid 0x10b, 0x0d51
   bind "serial_cs"
card "National Instruments PCMCIA-485"
   manfid 0x10b, 0x0d52
   bind "serial_cs"
card "National Instruments PCMCIA-485/2"
   manfid 0x10b, 0x0d53
   bind "serial_cs"
```

Signal the Card Manager to Reload /etc/pcmcia/config

Enter the following. Notice that the "`" is a forward single quote.

```
linux# kill -HUP `cat /var/run/cardmgr.pid`
```

Find Out What Devices Were Assigned to Your Card

Insert your PCMCIA serial card. You should hear two consecutive high beeps. To see what serial device the card manager assigned to your card, enter the following:

```
linux# more /var/run/stab
Socket 0: National Instruments PCMCIA-485
   0    serial    serial_cs 0   ttyS2  4    66
   0    serial    serial_cs 1   ttyS3  4    67
Socket 1: empty
```

The devices listed as `ttyS<port number>` under National Instruments are your serial ports.

Configuration

View Your Hardware Resources

To see what system resources your serial card is using, use the `setserial` command, as follows:

```
linux# setserial –gv /dev/ttyS<port number>
```

For example, to view the resources of `/dev/ttyS2`, enter the following:

```
linux# setserial –gv /dev/ttyS2
```

Something similar to the following should appear:

```
/dev/ttyS2, UART: 16550A, Port: 0x0af8, IRQ: 3
```

Enable FIFO Buffers

Use `FIFOtrigger` (from your `PCMCIA-SERIAL` directory) to enable and set the trigger level of the receive FIFO. `FIFOtrigger` enables the receive FIFO of only one serial port. To enable the receive FIFO for your other serial ports, rerun `FIFOtrigger` with a different serial port specified in the command line. Enter the following to use `FIFOtrigger`:

```
linux PCMCIA-SERIAL#./FIFOtrigger <port number>
                    <rx_trigger>
```

Table 1. rx_trigger Values

Receive FIFO Trigger Level	rx_trigger
1	0x00
4	0x40
8	0x80
14	0xC0

The hardware issues a *receive full* interrupt when the number of characters in the receive FIFO rises above the trigger level. For more information about FIFO buffers, refer to your PCMCIA serial getting started manual.

If `FIFOtrigger` does not work immediately or if it causes a segmentation fault, enter the following to recompile `FIFOtrigger.c` and rerun `FIFOtrigger`. Also, the source code for `FIFOtrigger` is available for

viewing and editing at `FIFOtrigger.c`, provided in the `PCMCIA-SERIAL` directory.

```
linux PCMCIA-SERIAL#gcc –O FIFOtrigger.c –o FIFOtrigger
linux PCMCIA-SERIAL#./FIFOtrigger <port number>
                    <rx_trigger>
```

FIFO Example

Enter the following to set the receive FIFO trigger level to 8 for `/dev/ttyS5`:

```
linux PCMCIA-SERIAL# ./FIFOtrigger 5 0x80
```

Configure struct termios

Every serial port has an associated `struct termios`. By using this `struct termios` in a program, you can set the baud rate, character size (number of data bits), parity, control characters, flow control, and input and output mode. For more information about the termios structure, refer to the `termios man page`. To view the `termios man page`, enter the following:

```
linux#  man termios
```

To configure your serial port, use a program segment similar to `termios_program.c` in your `PCMCIA-SERIAL` directory.

Select RS-485 Transceiver Mode

If you are using an RS-485 interface, you can select the transceiver mode for each device. For more information on the transceiver control modes, refer to your PCMCIA serial getting started manual. Use the `rs485` program (from your `PCMCIA-SERIAL` directory) to select the transceiver mode. `rs485` sets the transceiver mode for only one serial port. To set the transceiver mode for other serial ports, rerun `rs485` with a different serial port specified in the command line. If you do not know which transceiver mode to use, choose Four-Wire mode.

Table 2. Transceiver Modes

Transceiver Mode	Mode
Four-Wire Mode	0
Two-Wire Mode: DTR with echo	1
Two-Wire Mode: DTR controlled	2
Two-Wire Mode: TXRDY auto control	3

Enter the following to use `rs485`:

```
linux PCMCIA-SERIAL# ./rs485 <port number> <mode>
```

If `rs485` does not work immediately or if it causes a segmentation fault, recompile `rs485.c` and enter the following to rerun `rs485`. Also, the source code for `rs485` is available for viewing and editing at `rs485.c`, provided in the `PCMCIA-SERIAL` directory.

```
linux PCMCIA-SERIAL# gcc –O rs485.c –o rs485
linux PCMCIA-SERIAL# ./rs485 <port number> <mode>
```

rs485 Example

Enter the following to select the Four-Wire Mode for `/dev/ttyS5`:

```
linux PCMCIA-SERIAL# ./rs485 5 0
```

Test the Configuration

After you connect the cable to the ports (as shown in your PCMCIA serial getting started manual), run the `serialtest` program (from your `PCMCIA-SERIAL` directory) to verify your setup. Make sure you specify two different ports for the `serialtest` program, as shown in the following:

```
linux PCMCIA-SERIAL#./serialtest <receive port number>
                    <transmit port number>
```

If the test is successful, it will display a `SUCCESS` message. If the test hangs, type <ctrl-c> to exit the program.

To test `/dev/ttyS2` and `/dev/ttyS3`, connect a cable between the two ports and enter the following:

```
linux PCMCIA-SERIAL# ./serialtest 2 3
```

If `serialtest` does not work immediately or if it causes a segmentation fault, enter the following to recompile `serialtest.c` rerun `serialtest`. Also, the source code for `serialtest` is available for viewing and editing at `serialtest.c`, provided in the `PCMCIA-SERIAL` directory.

```
linux PCMCIA-SERIAL#gcc serialtest.c –o serialtest
linux PCMCIA-SERIAL#./serialtest <receive port number>
                    <transmit port number>
```

Sample /etc/pcmcia/config after Edit

```
#
# config.sample 1.87 1999/02/16 01:33:35 (David Hinds)
#
# config.opts is now included at the very end
#
# Device driver definitions
#
device "3c589_cs"
       class "network" module "3c589_cs"
device "ibmtr_cs"
       class "network" module "ibmtr_cs"
device "nmclan_cs"
       class "network" module "nmclan_cs"
device "oti12_cs"
       class "cdrom" module "oti12_cs"
device "pcnet_cs"
       class "network" module "net/8390", "pcnet_cs"
device "smc91c92_cs"
       class "network" module "smc91c92_cs"
device "wavelan_cs"
       class "network" module "wavelan_cs"
device "memory_cs" needs_mtd
       class "memory" module "memory_cs"
device "ftl_cs" needs_mtd
       class "ftl" module "ftl_cs"
device "serial_cs"
       class "serial" module "misc/serial","serial_cs"
#
# Other device driver definitions
#
#
# National Instruments serial card definitions
#
card "National Instruments PCMCIA-232"
       manfid 0x10b, 0x0d50
       bind "serial_cs"
card "National Instruments PCMCIA-232/2"
       manfid 0x10b, 0x0d51
       bind "serial_cs"
card "National Instruments PCMCIA-485"
       manfid 0x10b, 0x0d52
       bind "serial_cs"
```

```
card "National Instruments PCMCIA-485/2"
       manfid 0x10b, 0x0d53
       bind "serial_cs"
#
# Other card definitions
#
#
# Include local configuration settings
source ./config.opts
```

322542A-01 Aug99