

# Not reading, or sending, fast enough on my serial application

---

*Source:* <http://unix.derkeiler.com/Newsgroups/comp.unix.programmer/2006-06/msg00375.html>

---

- *From:* [af300wsm@xxxxxxxxxx](mailto:af300wsm@xxxxxxxxxx)
  - *Date:* 23 Jun 2006 08:25:40 -0700
- 

Hello,

Thanks to those who responded earlier this week to my posting about needing help with my serial programming. I'm past that hurdle (the hurdle I was trying to clear ended up being one of my own "genius"), but now I'm seeing that my application drops packets.

Now, I'm trying isolate where I'm having difficulty. The code places the serial driver in raw, non-canonical, mode for all the read and write calls. So I have two questions.

Pertinent information: packet size is 255 bytes.

1) on the send, or write, side of the app. I simply open the serial line and write the 2500 packets to it. This part of the application takes, approx., 5 seconds to complete. I'm wondering, since the serial line is set to 115200 BAUD, am I'm writing the data to the device faster than the kernel can send it and therefore bits are being dropped in the buffer before being sent?

2) on the receive, read, side; I again place the serial line in raw mode and then set the `c_cc[VMIN]` member to the packet size and then set the `c_cc[VTIME]` member to 0. I also call `fcntl` to set the `FNDELAY` flag on the descriptor and then use `select()` with a timeout of 30 seconds on the file descriptor (the extremely long timeout is only to allow the user to start the sender on another computer before the timeout). What I'm wondering is, am I incorrectly using these facilities? Since I'm setting `VMIN` to 255, is `select` extraneous? Also, since I'm setting `VMIN` to 255 AND I'm setting the flags to `FNDELAY`, is the read still being blocked until the 255 characters are present?

Andy

.