Re: [SOLVED] Serial buffer corruption [was Re: FTDI usb−serial possible bug]

Source: http://linux.derkeiler.com/Mailing−Lists/Kernel/2007−05/msg02598.html

- *From*: "Antonino Ingargiola" <tritemio@xxxxxxxxx>
- *Date*: Sat, 5 May 2007 11:53:39 +0200

On 5/4/07, Paul Fulghum <paulkf@xxxxxxxxxxxxx> wrote:

> Antonino:
>
> Can you try two tests (with my patch applied):
>
> 1. comment out the tty_flush_buffer() call in tty_ldisc_flush() and test
>
> 2. uncomment (reenable) the above call and comment out the
> tty_flush_buffer() call in tty_ioctl() and test

I assume you meant tty_buffer_flush(). I've built kernel 1). In kernel 2), do you mean:

/*if (ld−>ioctl)
tty_buffer_flush(tty);*/
tty_ldisc_deref(ld);

right? This is what I'm building... I'll report these new tests soon.

While waiting for kernel building I'll document the testing procedure. In this way someone else can easily try to reproduce the problem.

1. Hardware.

Two serial ports required. Connect the two port witth a null−modem cable, or patch, for each port the Tx pin with the Rx of the other port[0].

2. Software

I assume python is installed. Install also pyserial[1] (in debian python−serial), if you manually download the package you can put the "serial" dir in the dir you use to perform the test (no need to install system−wide). If you can, install also the ipython shell that has colored output and auto−completition.

3. Test

From the python shell:

```
import serial
s0 = serial.Serial(0, timeout=1)

open the /dev/ttyS0 port with default values, hit 's0' [enter] to see
the serial parameters.

s1 = serial.Serial(1, timeout=1)

with the previous comma we open /dev/ttyS1 with the same serial port
settings. To write to one serial port:

s0.wr
```