

Re: Why is the serial port so slow in Linux?

Source: <http://linux.derkeiler.com/Newsgroups/comp.os.linux.misc/2007-02/msg00157.html>

- *From:* Richard <rkm@xxxxxxxxxxx>
 - *Date:* Sun, 04 Feb 2007 09:21:28 -0500
-

Floyd L. Davidson wrote:

Richard <rkm@xxxxxxxxxxx> wrote:

I am doing a loop-back test with the serial port, I have connected pins 2 and 3 (the

Testing software design with a loopback is not a valid concept on any multitasking system.

transmit and receive pins) on the DB9 connector together. I am sending a series of single bytes and reading them back as soon as they are available. The program works, the only problem is that the time it takes to transmit and receive the bytes is about 30 times longer than for a DOS program I wrote that does the same thing.

Rather **clearly** your software is at fault. The Linux serial ports run at full speed, and work extremely well.

A long while ago I posted in a Linux newsgroup asking why the serial port is so slow in Linux, and I got a reply that the problem is the FIFO in the UART. The

That is not true, and would almost certainly result in 1) higher overhead for the system and 2) slower throughput on the serial port.

Post the code you are using to open, to configure, to write and to read the serial port. The smallest possible program that

Re: Why is the serial port so slow in Linux?

will demonstrate how each functionality is coded, while still providing a complete program that can be compiled and run would be the best.

Here is the source code:

```
/*  
This program sends single bytes over the serial port and reports  
any echoes. The file "serial_comm.in" must exist in the same  
directory and it specifies icom (currently 1 or 2) and the  
integer baud. Here is a sample serial_comm.in file:
```

```
icom =  
1  
baud =  
115200
```

Valid values for baud are as follows:

```
0  
50  
75  
110  
134  
150  
200  
300  
600  
1200  
1800  
2400  
4800  
9600  
19200  
38400  
57600  
115200  
230400
```

(Note that baud=0 disconnects the serial port).

To compile serialcomm.c, the ncurses library must be linked to it.
Here is one possible compile command:

```
gcc -lncurses -g -o serial_comm serial_comm.c
```

The program serial_comm is run without any command line options.

```
*/
```

Re: Why is the serial port so slow in Linux?

```
#include <stdio.h> /* Standard input/output definitions */
#include <stdlib.h>
#include <string.h> /* String function definitions */
#include <unistd.h> /* UNIX standard function definitions */
#include <fcntl.h> /* File control definitions */
#include <errno.h> /* Error number definitions */
#include <termios.h> /* POSIX terminal control definitions */
#include <curses.h>

int ascii_loopback;
long int num_sent_loopback,num_received_loopback,num_loopback,i_read_loopback;

void exit_curses(void);
int setup_serial_port(int icom, long int baud);
void reset_canonical_input_mode (void); /* resets terminal back to canonical mode */

void set_noncanonical_input_mode (void); /* sets terminal to non-canonical mode */

struct termios saved_attributes; /* to save terminal attributes before switching modes */
void to_next_line(FILE *input_file);
char inbuf[200];

/* ----- */
/*
The following procedure reads to the end of the current line. It stops
when either \n or eof is found.
*/

void to_next_line(FILE *input_file)

{

char ascii_read_nl;

if(!feof(input_file))
{
do
{
fscanf(input_file,"%c",&ascii_read_nl);
}
while(!( ascii_read_nl=='\n') || (feof(input_file)) );

}

}; /* end procedure to_next_line */
/* ----- */
```

Re: Why is the serial port so slow in Linux?

```
/* ----- */

int setup_serial_port(int icom, long int baud)
{

int _fd;
struct termios options;

/*
* Open serial port
*/

if(icom==1)
{
_fd = open("/dev/ttyS0", O_RDWR | O_NOCTTY | O_NDELAY);
}
else
{
_fd = open("/dev/ttyS1", O_RDWR | O_NOCTTY | O_NDELAY);
};
if (_fd == -1)
{
/*
* Could not open the port.
*/
perror("open_port: Unable to open serial port ");
return _fd;
}
else
fcntl(_fd, F_SETFL, 0);

/* to make the read function return immediately (non-blocking mode): */
fcntl(_fd, F_SETFL, FNDELAY);

/*
* Get the current options for the port...
*/

tcgetattr(_fd, &options);

/*
* Set the baud rate...
*/
```

Re: Why is the serial port so slow in Linux?

```
switch (baud)
{
case 0: /* note: case 0 disconnects the line */
cfsetispeed(&options, B0);
cfsetospeed(&options, B0);
break;
case 50:
cfsetispeed(&options, B50);
cfsetospeed(&options, B50);
break;
case 75:
cfsetispeed(&options, B75);
cfsetospeed(&options, B75);
break;
case 110:
cfsetispeed(&options, B110);
cfsetospeed(&options, B110);
break;
case 134:
cfsetispeed(&options, B134);
cfsetospeed(&options, B134);
break;
case 150:
cfsetispeed(&options, B150);
cfsetospeed(&options, B150);
break;
case 200:
cfsetispeed(&options, B200);
cfsetospeed(&options, B200);
break;
case 300:
cfsetispeed(&options, B300);
cfsetospeed(&options, B300);
break;
case 600:
cfsetispeed(&options, B600);
cfsetospeed(&options, B600);
break;
case 1200:
cfsetispeed(&options, B1200);
cfsetospeed(&options, B1200);
break;
case 1800:
cfsetispeed(&options, B1800);
cfsetospeed(&options, B1800);
break;
case 2400:
cfsetispeed(&options, B2400);
cfsetospeed(&options, B2400);
break;
case 4800:
```

Re: Why is the serial port so slow in Linux?

Re: Why is the serial port so slow in Linux?

```
cfsetispeed(&options, B4800);
cfsetospeed(&options, B4800);
break;
case 9600:
cfsetispeed(&options, B9600);
cfsetospeed(&options, B9600);
break;
case 19200:
cfsetispeed(&options, B19200);
cfsetospeed(&options, B19200);
break;
case 38400:
cfsetispeed(&options, B38400);
cfsetospeed(&options, B38400);
break;
case 57600:
cfsetispeed(&options, B57600);
cfsetospeed(&options, B57600);
break;
case 115200:
cfsetispeed(&options, B115200);
cfsetospeed(&options, B115200);
break;
case 230400:
cfsetispeed(&options, B230400);
cfsetospeed(&options, B230400);
break;
default:
printf("Error in baud rate specification\n");
return -1;
}
```

```
/*
 * Enable the receiver and set local mode...
 */
```

```
options.c_cflag |= (CLOCAL | CREAD);
```

```
/* no parity (8N1): */
options.c_cflag &= ~PARENB;
options.c_cflag &= ~CSTOPB;
options.c_cflag &= ~CSIZE;
options.c_cflag |= CS8;
```

```
/* disable hardware flow control: */
options.c_cflag &= ~CRTSCTS;
```

```
/* setup raw input: */
```

Re: Why is the serial port so slow in Linux?

Re: Why is the serial port so slow in Linux?

```
options.c_iflag &= ~(ICANON | ECHO | ECHOE | ISIG);

/* select raw output: */
options.c_oflag &= ~OPOST;

/* disable software flow control */
options.c_iflag &= ~(IXON | IXOFF | IXANY);

/*
 * Set the new options for the port...
 */

tcsetattr(_fd, TCSANOW, &options);

/* serial port is now configured */

return _fd;

};
/* ----- */

/* ----- */

void exit_curses(void)
{
endwin();
};
/* ----- */

/* ----- */
/*
The following procedure resets the terminal to canonical (normal) mode,
assuming that the terminal attributes were saved in saved_attributes
before switching to noncanonical mode.
*/

void reset_canonical_input_mode (void)
{
tcsetattr (STDIN_FILENO, TCSANOW, &saved_attributes);
}; /* end reset_canonical_input_mode */
/* ----- */

/* ----- */
```

Re: Why is the serial port so slow in Linux?

```
/*
The following procedure sets the terminal to noncanonical input
mode; this is necessary when immediate single character input is
required by the program. Before changing the mode, the original
attributes of the terminal are saved in saved_attributes, so that
they may be restored before exiting the program.
*/

void set_noncanonical_input_mode (void)
{
struct termios tattr;
char *name;

/* Make sure stdin is a terminal. */
if (!isatty (STDIN_FILENO))
{
printf ("Error – Not a terminal.\n");
exit (1);
}

/* Save the terminal attributes so we can restore them later. */
tcgetattr (STDIN_FILENO, &saved_attributes);
atexit (reset_canonical_input_mode); /* in case program is not exited normally */

/* Set noncanonical, nonecho mode: */
tcgetattr (STDIN_FILENO, &tattr);
tattr.c_lflag &= ~(ICANON|ECHO); /* Clear ICANON and ECHO. */
tattr.c_cc[VMIN] = 0; /* so that read exits immediately */
tattr.c_cc[VTIME] = 0; /* so that read exits immediately */
tcsetattr (STDIN_FILENO, TCSAFLUSH, &tattr);
}; /* end set_noncanonical_input_mode */
/* ----- */

int main()
{

char cd[2],cs[2];
int fd; /* File descriptor for the serial port */
int quit_program,byte_sent,byte_received;
unsigned char char_sent,char_received;
int ns,nd,icom;
long int baud;
const char file_name[] = "serial_comm.in";
FILE *input_file;
int io_result;

if( (input_file=fopen(&file_name[0],"r")) == NULL)
```

Re: Why is the serial port so slow in Linux?

```
{
printf("Error – file serial_comm.in not found\n");
exit(1);
};

to_next_line(input_file); /* skip first line of comments */
io_result = fscanf(input_file, "%d",&icom);

if(io_result < 1)
{
printf("Incorrect format in file serial_comm.in\n");
printf("Program will be terminated\n");
exit(1);
};

to_next_line(input_file); /* advance to next line */
to_next_line(input_file); /* skip next line of comments */
io_result = fscanf(input_file, "%d",&baud);

if(io_result < 1)
{
printf("Incorrect format in file serial_comm.in\n");
printf("Program will be terminated\n");
exit(1);
};

fclose(input_file);

fd = setup_serial_port(icom,baud);

if(fd<0)
{
printf("Error opening serial port\n");
printf("Program will be terminated\n");
exit(1);
};

initscr(); /* start curses mode */
atexit(exit_curses); /* so that we exit curses mode at exit */
clear();
nocbreak();

quit_program = 0;

do
{ /* main do loop */

clear();
```

Re: Why is the serial port so slow in Linux?

```
printw("S to send byte over serial port and observe echoes\n");
printw("L for serial port loopback test\n");
printw("Q to quit program\n");
refresh();
cbreak(); /* so don't wait for eol */
noecho();
nodelay(stdscr,1); /* so getch doesn't wait for characters */
do
{
inbuf[0] = getch();
}
while(inbuf[0]<=0);

if(toupper(inbuf[0])=='L')
{ /* loopback */
clear();
nocbreak();
echo();
nodelay(stdscr,0);
printw("Enter byte to send in loopback test (prefix with 0x if in hex):\n");
refresh();
scanw("%i",&byte_sent);
char_sent = byte_sent;
printw("Enter number of bytes to send in loopback test:\n");
refresh();
scanw("%d",&num_loopback);

cbreak(); /* so don't wait for eol */
nodelay(stdscr,1); /* so getch doesn't wait for characters */
noecho();

num_sent_loopback = 0;
num_received_loopback = 0;
do
{ /* loopback loop */

ns = write(fd,&char_sent,1);

if(ns <= 0)
{
endwin(); /* exit curses mode */
printf("Error – byte not sent successfully\n");
printf("Program will be terminated\n");
close(fd);
exit(1);
};

num_sent_loopback++;

i_read_loopback=0;
```

Re: Why is the serial port so slow in Linux?

```
do
{
i_read_loopback++;
ns = read(fd,&char_received,1);
nd = getch();
}
while(!( ns>0) || (nd>0) || (i_read_loopback >= 100000) ) );

if(ns>0)
{
byte_received = char_received;
if(byte_received != byte_sent)
{
printw("Warning – byte received not the same as byte sent\n");
refresh();
};
num_received_loopback++;
};

} /* loopback loop */
while( !( (nd > 0) || (num_sent_loopback >= num_loopback) ) );

printw("%s%d%s%d%s", "Sent ", num_sent_loopback, " bytes, received ", num_received_loopback,
" bytes\n");
refresh();
printw("Press any A/N key to continue\n");

cbreak(); /* so don't wait for eol */
noecho(); /* typed characters not echoed on screen */
nodelay(stdscr,0); /* so getch waits for characters */
ascii_loopback = getch();

nocbreak(); /* so wait for eol again */
echo();
nodelay(stdscr,0); /* so getch waits for characters again */

if(toupper(nd) == 'Q')
{
quit_program = 1;
};

}; /* loopback */

if(toupper(inbuf[0])=='S')
{ /* send byte */
clear();
nocbreak();
echo();
nodelay(stdscr,0);
printw("Enter byte (prefix with 0x if in hex):\n");
```

Re: Why is the serial port so slow in Linux?

```
refresh();
scanw("%i",&byte_sent);
char_sent = byte_sent;
ns = write(fd,&char_sent,1);

if(ns <= 0)
{
endwin(); /* exit curses mode */
printf("Error – byte not sent successfully\n");
printf("Program will be terminated\n");
close(fd);
exit(1);
};

cbreak(); /* so don't wait for eol */
nodelay(stdscr,1); /* so getch doesn't wait for characters */
noecho();

do
{
ns = read(fd,&char_received,1);
nd = getch();

if(ns>0)
{
byte_received = char_received;
printw("%s%x%s%d%s","0x",byte_received,"(",
byte_received," decimal) \n");
refresh();
};

}
while( !(nd > 0) );

nocbreak(); /* so wait for eol again */
echo();
nodelay(stdscr,0); /* so getch waits for characters again */

if(toupper(nd) == 'Q')
{
quit_program = 1;
};

}; /* send byte */

if(toupper(inbuf[0])=='Q')
{
quit_program = 1;
};

} /* main do loop */
```

Re: Why is the serial port so slow in Linux?

```
while(quit_program == 0);  
  
endwin(); /* quit curses mode */  
close(fd);  
  
}; /* end of main */  
.
```