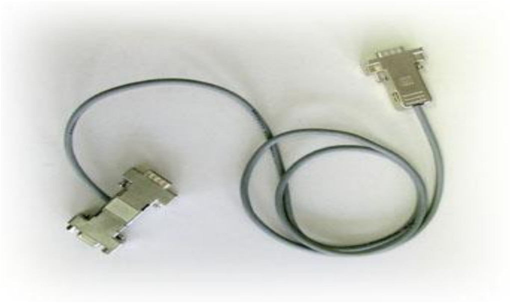
AN RS232 'SPY' CABLE

The RS232 norm defines a *one-on-one* communication link between DTE device (Data Terminal Equipment, such as a PC or a printer) to a DCE device (Data Communications Equipment, typically a modem). One DTE device talks to one DCE device, or to one other DTE device through a *null-modem* cable —a null-modem cable is a cable where the *transmit* and *receive* lines are crossed. With a special cable, though, it is possible to have a third apparatus "listen in" on the data.

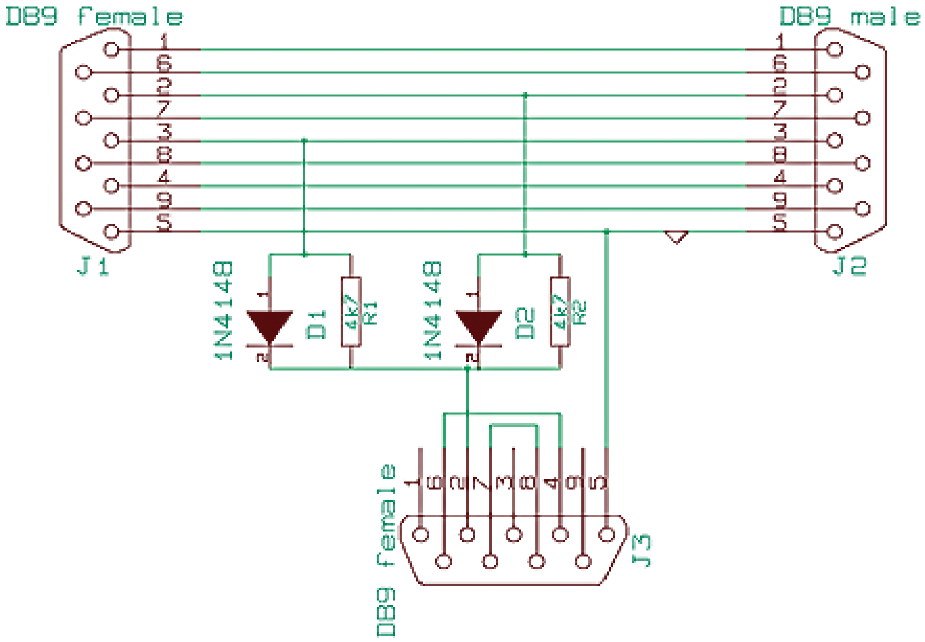


A male-female "gender changer" with a third (female) connector split off

The motivation for such a cable is that a standard debugger running on a PC is not very practical in debugging serial protocol problems (or for deciphering the protocol). Two different apparatus/programs are involved in a serial link, and if you stop one of these (on a breakpoint) for inspection, the other may time out and abort the communication.

A better approach is to pick up the stream of bytes on a separate monitoring computer and verify/decipher the protocol from this trace. In this case, both sides run at their normal communication speeds and neither is aware that what it sends gets captured by a PC "in the middle".

The RS232 connection has separate *transmit* and *receive* lines, which means that the hardware can send data and receive data at the same time ("full duplex"). However, the majority of the protocols are half duplex: also called "master-slave" or "query-reply". What the splitter cable does is to re-route the *transmit* lines of both devices to the *read*line for the monitoring PC. This will only work correctly for half duplex protocols; when both devices that are involved in the serial connection send at the same, the signals will collide on the split-off receive line.



This schematic assumes that the split-off cable has a female connector, for connection to a PC. When you need a male connector for J3, connect the diodes and resistors to pin 3 of J3 —instead of pin 2.

Here is how it works: when a device (on either side of the communication link) is not sending, it pulls its transmit line low (where *low* is typically a voltage between -12V and -5V). If neither device is sending, both transmit lines are therefore low and they pull the receive line of the split-off cable low as well. If either device is sending, the "high" pulses (between +5V and +12V) pass through a diode and pull the receive line of the split-off cable up.

The two 4k7 resistors cause some cross-talk between the transmit lines of the two communicating devices, but this should not cause any problems (alternative designs that I have seen use 2k2 or even 1k resistors, so 4k7 is probably on the save side).

On the PC that must monitor the data stream, you can run any common RS232 communication program, such as the HyperTerminal that comes with Windows (better alternatives are freely available, see the bottom of this page). Note that the monitoring PC cannot send anything: the transmit line is left unconnected.

Alternative: a PC with two RS232 ports

Instead of a special cable, a set-up that works in most cases is to use a PC or laptop that has two RS232 ports, plus special software that forwards data received from one port to the other, and vice versa. Software suitable for RS232 port monitoring and port forwarding is Termite.

Source: http://www.compuphase.com/electronics/rs232split.htm