

RS232 Connections, and wiring up serial devices

RS232 Pin Assignments (DB25 PC signal set)

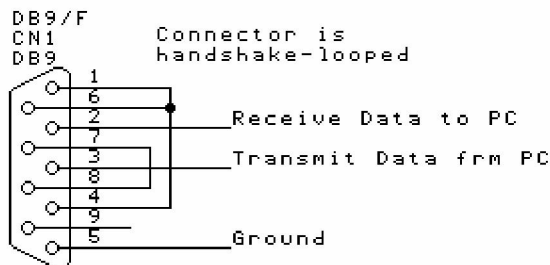
- Pin 1 Protective Ground
- Pin 2 Transmit Data
- Pin 3 Received Data
- Pin 4 Request To Send
- Pin 5 Clear To Send
- Pin 6 Data Set Ready
- Pin 7 Signal Ground
- Pin 8 Received Line Signal Detector
(Data Carrier Detect)
- Pin 20 Data Terminal Ready
- Pin 22 Ring Indicator

RS232 Pin Assignments (DB9 PC signal set)

- Pin 1 Received Line Signal Detector
(Data Carrier Detect)
- Pin 2 Received Data
- Pin 3 Transmit Data
- Pin 4 Data Terminal Ready
- Pin 5 Signal Ground
- Pin 6 Data Set Ready
- Pin 7 Request To Send
- Pin 8 Clear To Send
- Pin 9 Ring Indicator

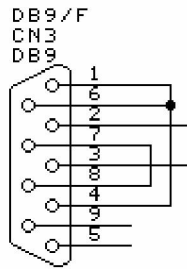
Wiring up something nice and simple, for instance a plain old "dumb terminal", is just a matter of connecting Tx, Rx and Ground, right?

Usually Not. While the normal PC hardware might well run with just Tx, Rx and Ground connected, most driver software will wait forever for one of the handshaking lines to go to the correct level. Depending on the signal state it might sometimes work, other times it might not. The reliable solution is to loop back the handshake lines if they are not used.



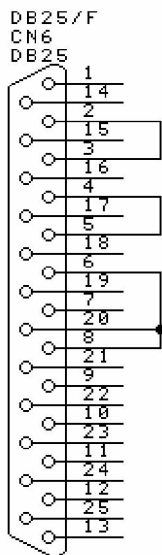
Handshake looping a PC serial connector

When the lines are handshake looped, the RTS output from the PC immediately activates the CTS input - so the PC effectively controls its own handshaking



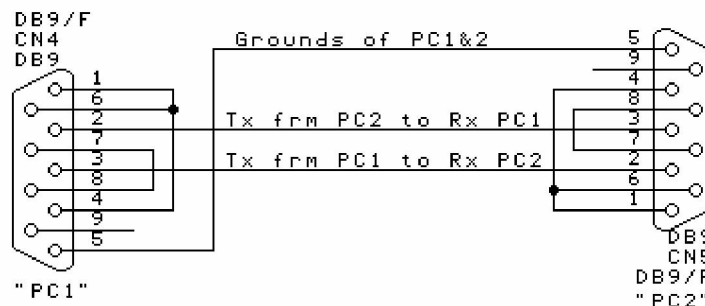
RS232 DB9 PC Loopback test plug

The PC loopback plug is a useful diagnostic tool. The loopback plug connects serial inputs to serial outputs so that the port may be tested. There is more than one way to wire up a loopback plug - but this is the most common.



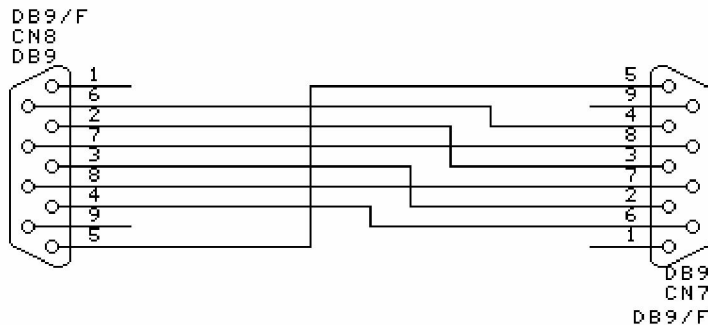
RS232 DB25 PC Loopback test plug

Connecting together two serial devices involves connecting the Rx of one device to the Tx of the other, and vice versa. The diagram below indicates how you would go about connecting two PC's together, without handshaking.



Connecting two PCs together using RS232, without handshaking

When Handshaking is required, generally RTS of one device connects to CTS of the other, and vice versa, and also DSR of one device connects to DTR of the other device, and vice versa. The particular requirements for different equipment *may* vary



Connecting two PCs together using RS232, with handshaking

Using a Breakout box or LED box to work out cabling

If you have problem with RS232 cabling, your best "emergency" tool may be a breakout box (sometimes called an LED box). Generally these units only come in the DB25 size, but with a couple of DB9 to DB25 adaptors, they can be used with DB9 cables as well. The units have an LED for each signal line in the cable, and the LED lights green or red dependent on the signal state. The Breakout box also allows you to disconnect certain lines in the cable, and patch in new ones - good for trying new cable wiring possibilities.

The first thing to remember, is that there is a good chance the two devices you are trying together will actually work if you can get the cable correct. If you have some other way to actually prove this - for instance by trying each of the devices on another system - do it.

Given a hypothetical example - for instance connecting a standard PC with a DB25M to a 200 disk CD changer with a DB25M, the first thing I would try and do is get a cable that I think would work. In this instance, I would either purchase or build a null-modem cable (DB25F to DB25F) - similar to the last example, basically the cable used to connect two PCs together with handshaking, only DB25F to DB25F instead of DB9F to DB9F.

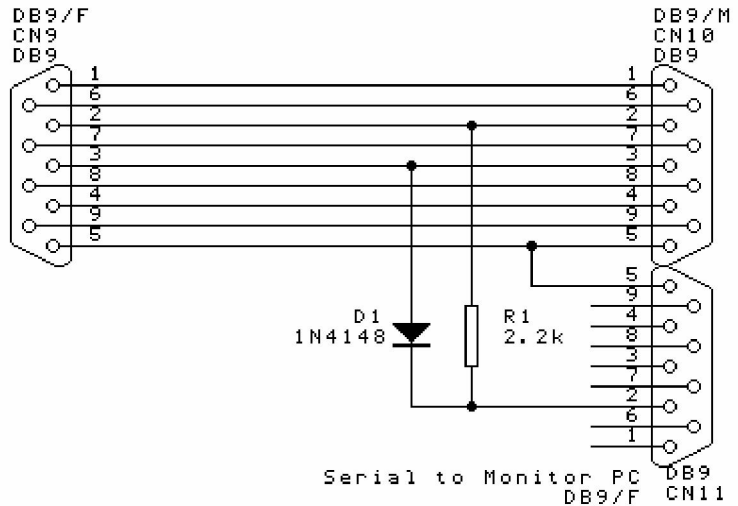
Given the cable that I believe will work, connect the cable, LED box and two devices all together. Before powering on both devices, unplug just one of them. Power the devices on and make a note of which LEDs are lit. Then unplug the connected device and plug in the disconnected one, without rearranging the cabling otherwise. Again make a note of which LEDs are lit. If any single LED is lit by both of the devices, then there is an output conflict, and the cable wiring is incorrect. By this, I mean that one line in the cable has an output driving it from both ends - and this is not correct for RS232 - so that means that the cable wiring is not correct for the devices. Pay particular attention to Tx and Rx.

To continue with the example above, if I saw that two ends were driving the same lines, I would assume the null modem cable was not correct, and I would try a one-for-one gender changer instead.

If each end drives its own set of LEDs, connect the two ends together. In normal situations, you should see all the LEDs light up - but there are some devices which will not light up all the

LEDs. Having said that, if one of the devices is a PC and any LED except RI (Ring indicator) is not lit up, the cable will probably not work.

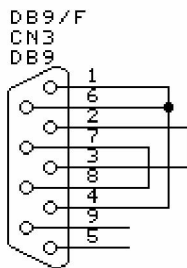
Normally, other cabling problems will involve handshake lines. An LED box will be an invaluable guide, but there is no trivial test to determine the solution. An LED Box will also show the lines as they change state, although it is usually quite hard to see the serial communications themselves unless the comms are continuous, or at a low baud rate (9600 baud or lower is usually visible).



Using a 'T' plug and a PC to monitor comms

The gadget below is a quick 10 minute project that is really great for monitoring RS232 Comms using a PC.

A gadget for monitoring RS232 Comms between two devices



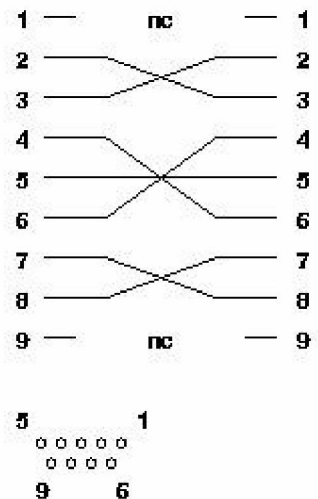
There are three sockets on our monitoring gadget. Two of them are connected straight through - you plug them in series with the devices you wish to monitor - and the third goes off to another monitoring PC.

The monitoring PC "Sees" on its serial port both sides of the serial conversation - that is it sees what is sent by PC1 and also what is sent by PC2. This can be a positive advantage, because you can see the serial conversation as it progresses

between the two devices. Some serial protocols, however, talk "full duplex" meaning that one end can start transmitting while it is still receiving from the other end. This unit cannot monitor full duplex Comms - you will see gobble-dee-gook where the two transmissions overlap.

If you try this unit, you will be surprised how useful it is, and how often it works - mostly because many supposedly full duplex installations still talk half duplex in any case, because that is the sensible way to write the software. *Not bad for one diode and a resistor, huh?*

Null-modem cable RS232



For information on other types of connectors, check out:

<http://www.hardwarebook.net/cable/index.html>

or for PC HW: http://www.epanorama.net/links/project_pc.html