Title:
Cisco CCNA Exam Tutorial: Directly Connected Serial Interfaces

Word Count:
594

Summary:
Configuring directly connected serial interfaces seems simple enough, but there are important details you must know to make it work and pass the CCNA exam! Learn these details from Chris Bryant, CCIE #12933.

Keywords:
Ccna, exam, free, pass, dte, dce, serial, interface, directly, connected, clockrate, certification, line, protocol, show, interface, controllers

Article Body:
To pass the CCNA exam, you've got to master quite a few services and routing protocols that may be new to you. Between RIP, IGRP, EIGRP, OSPF, and switching, there are hundreds of details you've got to absorb! It's easy to spend all your time on those topics and not pay proper attention to "easier" technologies, and then all of a sudden on exam day you can't quite remember the details of those particular services.

One setup you've got to be more than familiar with is directly connecting serial interfaces on Cisco routers. This is also a valuable skill to have in your home lab, since it allows you to add segments to your network setup.

A Cisco serial interface is operating as a DTE by default. The problem is that when you take a cable and connect two routers directly by their serial interfaces (with a DTE/DCE cable, that is!), they're both waiting for the other to send them a clock rate. One of the interfaces must act as the DCE and that interface must send the clock rate.

If you can see the DTE/DCE cable, you can tell by looking which router has the DCE interface connected to it - the letters "DTE" or "DCE" will either be molded into the connector itself, or if it's an older cable there should be a little piece of tape on the cable that tells you what the interface type is. But what if you have no access to the cable, or there are other cables all around it and you can't see what type it is?

Run the command "show controller serial x", with x representing the interface number the cable's connected to. There will be quite a bit of output from this command, but the information you need is right at the top:

```
R1#show controller serial 1

HD unit 1, idb = 0x1DBFEC, driver structure at 0x1E35D0

buffer size 1524 HD unit 1, V.35 DTE cable
```

I left off the 16 or so rows of information that comes after this, but this is the information we need right now. If R1's got the DTE cable end, the other router should have the DCE end:

```
R3#show controller serial 1

HD unit 1, idb = 0x1C44E8, driver structure at 0x1CBAC8

buffer size 1524 HD unit 1, V.35 DCE cable
```

We know now that R3 needs to supply a clock rate to R1. There's a hint of a problem in just that little bit of command output - do you see what it is? Let's run show interface serial1 to get more information.

```
R3#show int s1

Serial1 is up, line protocol is down
```

The line protocol is down because there is no clockrate being supplied by R3. If there has been, we would have seen that in the output of show controllers serial 1.

This is simple enough to fix, though! We'll use the command clockrate 56000 on R3's serial1 interface, and the line protocol will soon come up.

```
R3(config)#int s1

R3(config-if)#clockrate 56000

1w2d: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial1, changed state to up
```

This is a simple concept, but there are a few details you must keep in mind! For a home lab configuration, you'll need a DTE/DCE cable to make this work. If you cannot see the cable connectors, run show controllers serial x to see if the router has the DTE or DCE end of the cable attached. On the interface with the DCE attached, use the clockrate command to bring the line protocol up. It's just that simple!