

# ***EZ-232*** RS-232 Serial Interface

The EZ-232 Interface, designed by Jim Brain, provides a low-speed serial port for Commodore 8-bit computers. It can operate at speeds of up to 2400bps, when configured as a standard interface, and at speeds of up to 9600bps, when configured as a UP9600 interface.

This is my second batch of interfaces, and I've decided to have solder mask applied to the boards. This makes soldering them easier, and they look a bit more professional. Because I doubled the size of the run this time, I was able to keep the prices the same as the last batch.

I'll first explain the jumpers and switches on this board for those of you who purchased one already assembled.

The jumpers on the board are used to change between 2400bps and 9600bps modes. **As the board layout shows, if you short JP1, JP2 and Pins 2-3 of JP3, it'll be configured as a UP9600 interface.** Novaterm 9.6 has a serial driver for this interface, and it will allow you to communicate up to 9600BPS on the user port. I usually leave mine jumpered this way, as it still seems to work fine at lower speeds, and with Q-Link at 1200bps. **Shorting pins 1-2 of JP3, and unjumpering JP1 and JP2 return it to a standard 2400bps interface.**

**Note:** Leave the EZ-232 configured as a standard RS-232 interface when using with a Commodore 128. The C128 is already capable of 9600BPS on the User Port with a standard RS-232 interface. **Configuring it as a UP9600 interface will cause the C-128 to lock up.**

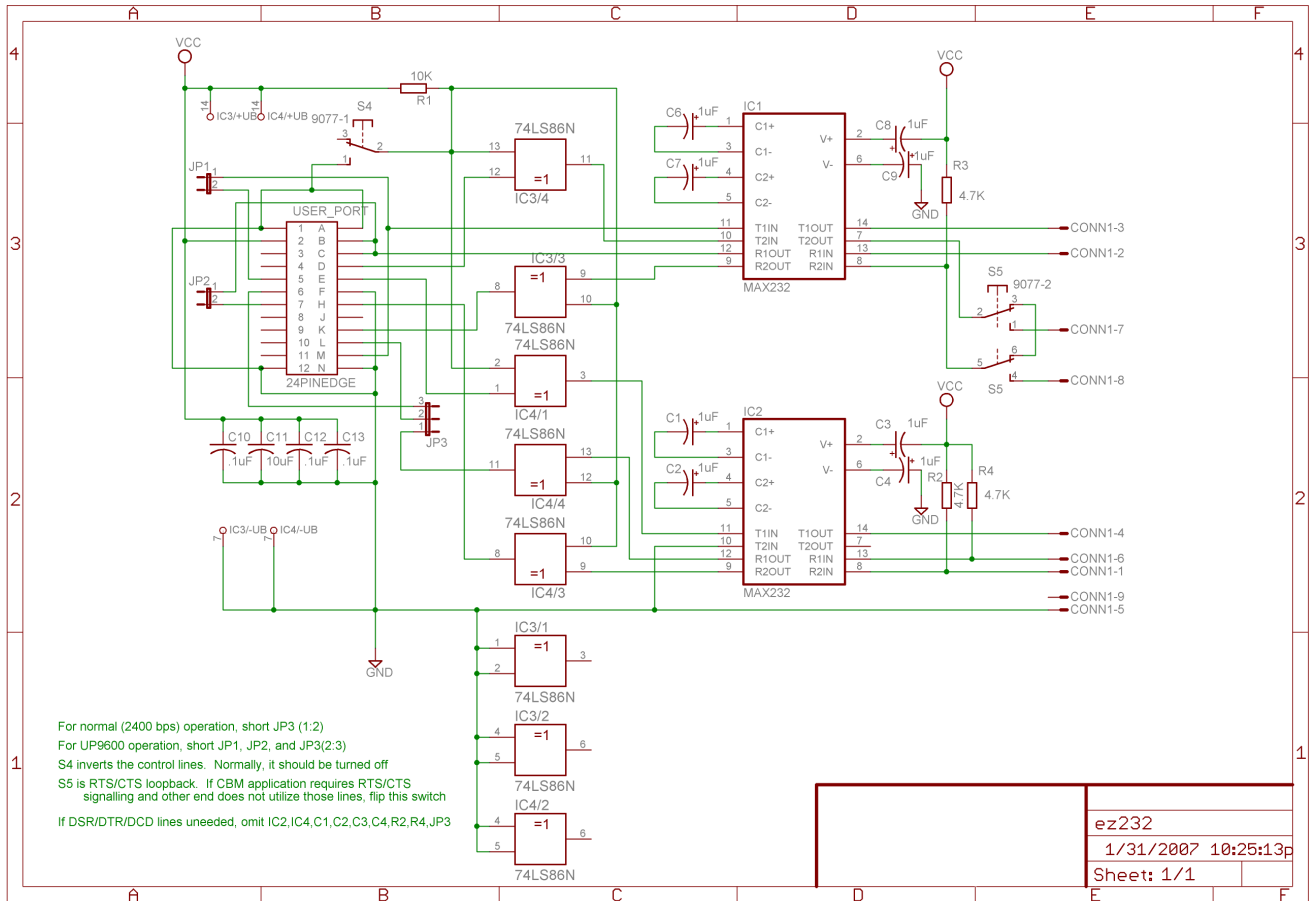
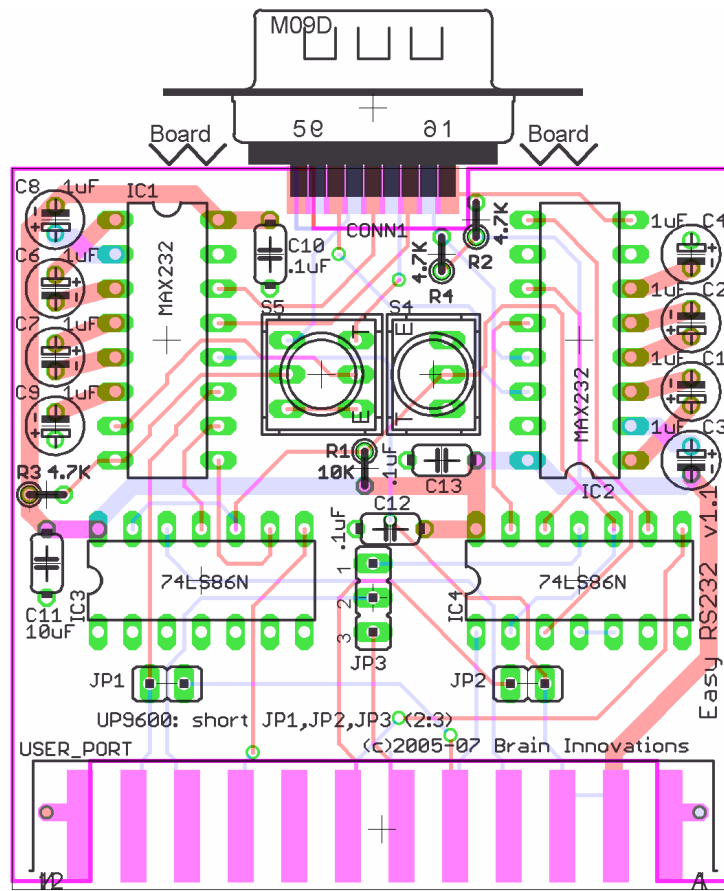
**Switch S4** will invert the control lines (DCD and DTR). This is used if you wish to have an inverted carrier detect signal used. **Normally this switch is in the off position, towards the DB9.**

**Switch S5** is the RTS/CTS loopback. (May be 1 or 2 switches.) If you're using software that requires RTS/CTS handshaking, but the device you're connected to doesn't provide it, then flip this switch to the on position. **Normally, you'll leave it off, towards the DB9.**

That's about it for using the device, now onto some tips on assembling it yourself!

I'm going to assume you have the proper equipment, and have already honed your soldering skills on other stuff before you've decided to build this. Normally, when I put these together, I start with the sockets. They're all the same height, and you can let the board sit on them while soldering. Next, I tackle the discrete components. You'll notice the solder pads for the capacitors and resistors are quite small. The application of solder mask makes soldering these components much easier now, and you can heat the pad and the lead at the same time. The resistors should all be installed standing straight up. The larger pads for the sockets and jumpers are much easier, and you can heat the pad and the lead at the same time while applying the solder. The last things I solder are the DB9 and edge connector. I generally use more solder on the 4 end pins on the edge connector, as they're on the largest parts of the board and can bear the most load.

Before plugging the interface into the computer, you should do a quick check with a multimeter between pins M and N on the edge connector. Make sure there's not a dead short between those two pins. If there is, check for solder jumping over the void between the solder pad and the ground plane on the board. I have found this can happen quite easily, so it's best to check the board before you apply power to it with your Commodore.



For normal (2400 bps) operation, short JP3 (1:2)  
 For UP9600 operation, short JP1, JP2, and JP3(2:3)  
 S4 inverts the control lines. Normally, it should be turned off  
 S5 is RTS/CTS loopback. If CBM application requires RTS/CTS  
 signalling and other end does not utilize those lines, flip this switch  
 If DSR/DTR/DCD lines unneeded, omit IC2,IC4,C1,C2,C3,C4,R2,R4,JP3