Users often ask how to use our ELM327 IC with the Bluetooth wireless system. Bluetooth offers several advantages over conventional (ie. RS232 or USB) interfaces, such as eliminating the cable between the ELM327 and the computer, isolating the vehicle wiring from the computer wiring, and allowing connections with devices such as PDAs or internet tablets that do not have the capability to connect by RS232 or USB.

The following provides a few ideas on how you might add Bluetooth to your ELM327 circuit.

Using Serial to Bluetooth Adapters

There are a few RS232 to Bluetooth converters on the market that simply connect to the 9 pin RS232 connector, and provide Bluetooth capability. Two of these devices are the Firefly from Roving Networks (www.rovingnetworks.com) and the GBS301 from IOGEAR (www.iogear.com).

Both the FireFly and the IOGEAR provide a connector for an external power source, but also allow DC to be provided through the RS232 connector (on pin 9). Your ELM327 circuit can be modified as shown in Figure 1 to provide this power, so that you do not need to connect the separate DC supply each time:

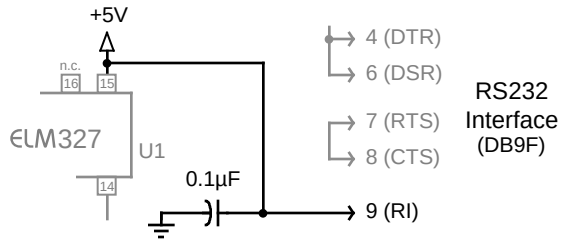


Figure 1. 5V for the Bluetooth Adapter

Note that these modules draw a significant current when operating (about 50mA), so you may need to modify your ELM327's 5V supply in order to provide the required power. Check that your ELM327 circuit uses a full size 7805 type regulator, and not the small 78L05 listed for the basic (Figure 9) ELM327 circuit. If you do have the smaller regulator, we advise removing it, and installing the larger one in its place.

If you are using the RS232 interface as shown in the ELM327 data sheet, you will find that it does not always work reliably at 115200 baud - it should only be used at speeds of 57600 baud or less. Since the FireFly only offers 9600 or 115200 from the dip switch, this leaves a choice of only 9600 baud for it. The IOGEAR model offers 9600, 19200, 38400, and 57600 baud rates from the dip switch, however, making it easier to use for 38400 baud, or even 57600 baud. Decide on which rate you prefer, and then make sure that you set the ELM327's pin 6 logic level to allow it.

Regardless of which Bluetooth adapter that you choose, the next step should be to 'pair' this device with the computer. Follow the manufacturer's instructions on how to do this. Do not be surprised if there is no software needed, as many modern operating systems will have default Bluetooth drivers built in. Simply follow the procedure to add this device (as a 'Serial Port' device) and then use it as you would any physically connected one.

Using Serial to Bluetooth Modules

RS232 to Bluetooth converters are convenient in that they can easily be connected to almost any RS232 connector, quickly creating a virtual serial port. If you use an external power adapter, there is no need to modify any circuitry, which keeps the connection simple. This type of approach does work well, but it limits the baud rates available, and it adds unnecessary complexity, as the RS232 interface is still required.

Several modules that contain a complete serial Bluetooth system (including an antenna) are now available in the marketplace. These provide a logic level RS232 interface for your circuitry and a standard Bluetooth protocol for 'connecting' to your controlling device, whether it be a PC, a netbook, or a PDA, etc. One such module is the LMX9838 from National Semiconductor (www.national.com).

The LMX9838 is a terrific little (1.0cm x 1.7cm) solution, but it operates with a 3.3V supply. The ELM327 will only operate at 5V, so it would seem that the two might not be compatible. We often get questions concerning this type of interface, so will address it here.

Interfacing to 3.3V

Connecting to a 3.3V system usually requires some form of voltage conversion. Some 3.3V circuits have inputs that are '5V tolerant' (but not all offer this feature), and most 5V logic (the ELM327 included) will not tolerate 3.3V on an input. We have found that an interface circuit such as the one shown in Figure 2 works well for most baud rates that you would encounter with the ELM327.

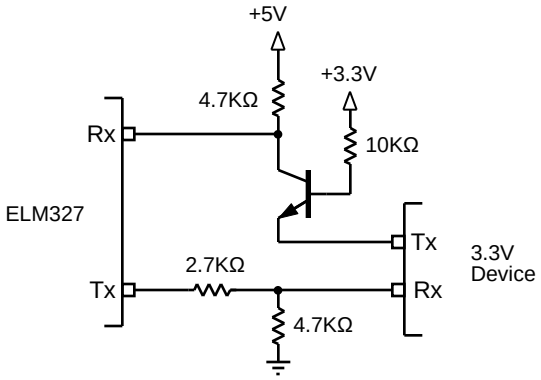


Figure 2. A 5V to 3.3V Interface

This circuit uses a simple voltage divider to reduce the ELM327's Tx output to approximately 3.3V from 5V. The 2.7KΩ and 4.7kΩ resistors shown allow for operation over a wide range of frequencies while not presenting a large load to the ELM327.

Interfacing a 3.3V output to the ELM327's 5V input requires a little more circuitry. Amplification is needed to increase the 3.3V level, so we add a transistor to that portion of the circuit, as shown. When the 3.3V device's Tx output is above about 2.8V, the transistor will be turned off (since Vbe is ~0V), and the ELM327's Rx input will have 5V applied to it, through the 4.7kΩ resistor. When the 3.3V output switches to a low level, the transitor will be driven into saturation and the Rx input will go to about 0V.

Since the transistor is operated in saturation, there will be a slight delay in turning off due to the charge storage. Because of this delay, the rising edge of the ELM327 Rx input will show a very slight pause in the voltage while the charge is depleted. The ELM327 has a schmitt trigger input so this does not affect the operation here, but if you wish to adapt this circuit for use at higher frequencies, it is something that you should be aware of.

Connecting to the Bluetooth module

Figure 3 shows the LMX9838 Bluetooth module connected to a typical ELM327 circuit (based on the one in the ELM327 data sheet).

The module needs a 3.3V power supply, so an additional voltage regulator has been added. We show an MIC5200-3.3 from Micrel (www.micrel.com), but there are several alternatives available (the Microchip MCP1700, etc). The regulator has been connected directly to the +5V supply, and filter capacitors have been connected to its output. To meet the extra current requirements of the module (65mA maximum), be sure that you have a 7805 type 5V regulator, and not a 78L05 type.

The LMX9838 module can be selected for 9600 baud or 115200 baud by changing the logic levels on pins 25 and 26. Initially, your circuit should be constructed for operation at 9600 baud, but be sure to make provisions to allow changing the logic levels at the pins so that you can switch to 115200 baud (as 9600 baud is very slow).

In order to set the ELM327 for operation at 115200 (instead of the default 38400), we will need to change the baud rate Programmable Parameter (0C). This is best done while operating at 9600 baud, as that rate is not affected by any Programmable Parameter.

To change the default baud rate, first connect the circuit of Fig 3 for 9600 baud. Then power your circuit, and set up your PC to use bluetooth for a serial port. Follow your Operating System's procedure to 'pair' to the LMX9838 (the code is 0000), and then connect to the ELM327 through the new serial port at 9600 baud. You should be able to perform simple commands such as AT I, even if no vehicle is connected:

```
>AT I
ELM327 v1.3a
```

Once you are certain that all is working well at 9600 baud, set the ELM327 for a power on default baud rate of 115200 (when pin 6 is high). To do this, assign the value 23 to PP 0C:

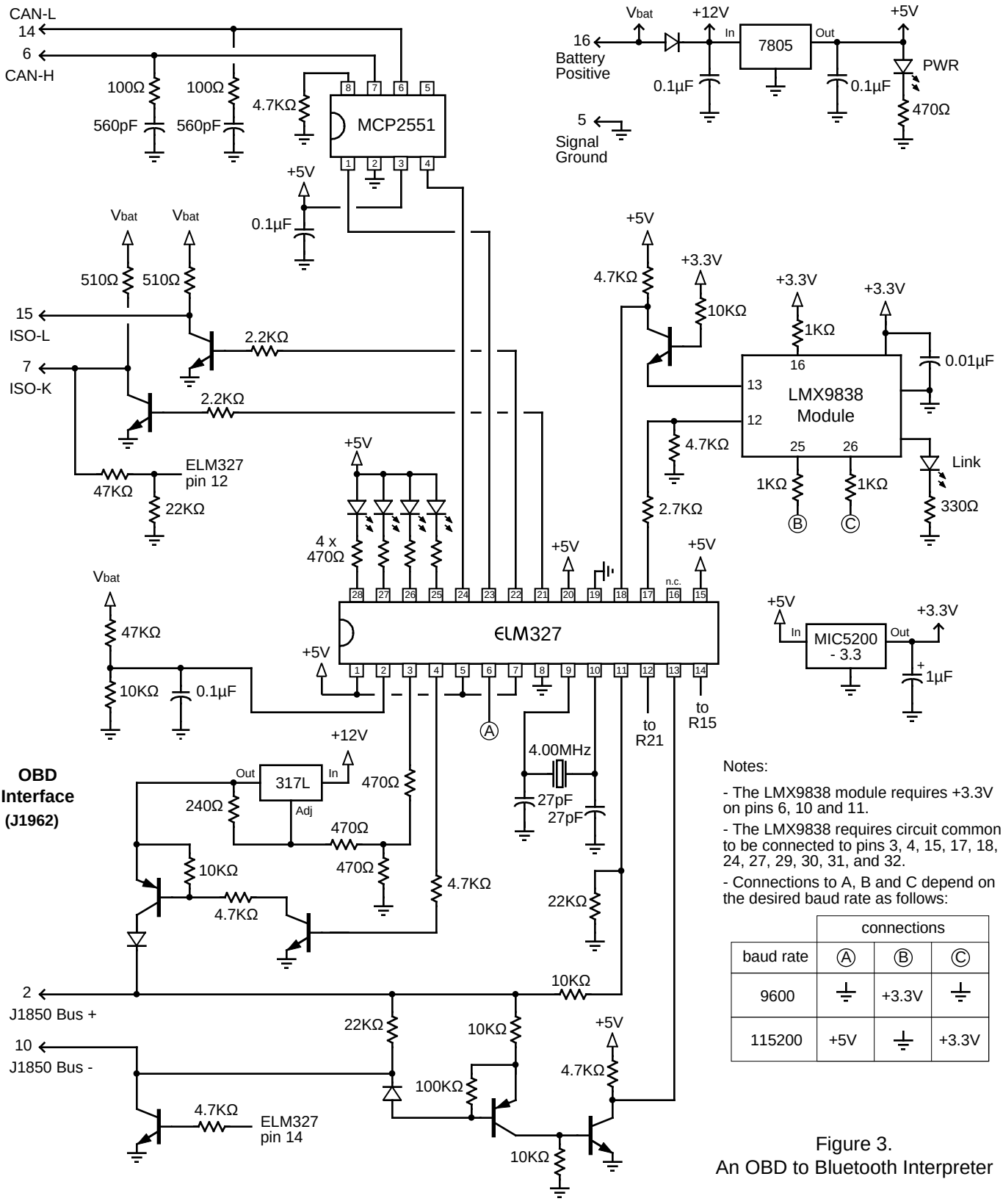
```
>AT PP 0C SV 23
```

and then enable the use of this new value:

```
>AT PP 0C ON
```

That's all that is required to change the ELM327. Now, change the wiring so that the ELM327 powers up to the high speed baud rate, as does the LMX9838. That is, connect the ELM327's pin 6 to 5V instead of 0V, and connect the resistors on the LMX9838 module pins 25 and 26 to 0V and 3.3V, respectively.

Apply power to the circuit, and test your Bluetooth connection. You may have to change your software for 115200 baud instead of 9600. If all has gone well, you should now have a high speed wireless OBD interpreter.



Notes:

- The LMX9838 module requires +3.3V on pins 6, 10 and 11.
- The LMX9838 requires circuit common to be connected to pins 3, 4, 15, 17, 18, 24, 27, 29, 30, 31, and 32.
- Connections to A, B and C depend on the desired baud rate as follows:

	connections		
baud rate	Ⓐ	Ⓑ	Ⓒ
9600	⏚	+3.3V	⏚
115200	+5V	⏚	+3.3V

Figure 3.
An OBD to Bluetooth Interpreter