

## 8.2 USB to RS485 Converter Configuration

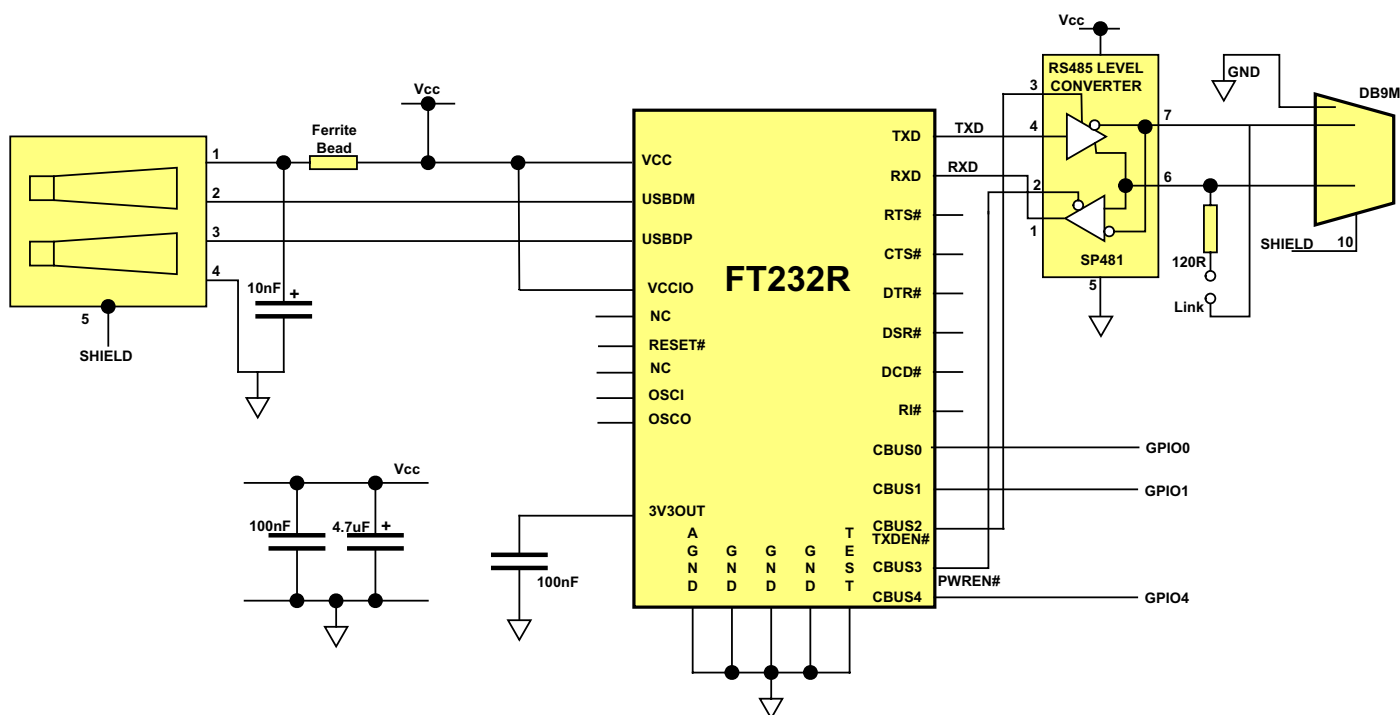


Figure 16 - Example USB to RS485 Converter Configuration

Figure 16 illustrates how to connect the FT232R's UART interface to a TTL – RS485 Level Converter I.C. to make a USB to RS485 converter. This example uses the Sipex SP481 device but there are similar parts available from Maxim and Analog Devices amongst others. The SP481 is a RS485 device in a compact 8 pin SOP package. It has separate enables on both the transmitter and receiver. With RS485, the transmitter is only enabled when a character is being transmitted from the UART. The TXDEN signal CBUS pin option on the FT232R is provided for exactly this purpose and so the transmitter enable is wired to CBUS2 which has been configured as TXDEN. Similarly, CBUS3 has been configured as PWREN#. This signal is used to control the SP481's receiver enable. The receiver enable is active low, so it is wired to the PWREN# pin to disable the receiver when in USB suspend mode. CBUS2 = TXDEN and CBUS3 = PWREN# are the default device configurations of these pins. See [Section 10](#).

RS485 is a multi-drop network – i.e. many devices can communicate with each other over a single two wire cable connection. The RS485 cable requires to be terminated at each end of the cable. A link is provided to allow the cable to be terminated if the device is physically positioned at either end of the cable.

In this example the data transmitted by the FT232R is also received by the device that is transmitting. This is a common feature of RS485 and requires the application software to remove the transmitted data from the received data stream. With the FT232R it is possible to do this entirely in hardware – simply modify the schematic so that RXD of the FT232R is the logical OR of the SP481 receiver output with TXDEN using an HC32 or similar logic gate.