USB-485 Isolated USB-RS422/RS485 Interface Converter

Installation and Operation

General

The USB485 is a general purpose USB to RS422/RS485 isolated interface converter. It is intended to provide an RS422/485 port on a PC which has only a USB port available. Drivers are included to create a virtual COM port within the operating system, compatible with standard Windows application programs. The isolation barrier is between the USB interface and the RS422/485 (DB9) interface.

Installation

1. With Windows running, insert driver CD into computer's CD drive
2. Connect USB connector to computer USB port
3. Windows should report that a new device has been found, offering a choice of locations to look for the driver. Choose the CD as the location.
4. Windows should now install the driver. Now you need to find out which COM port number has been allocated: open up Control Panel / System / Hardware / Device Manager (the exact location varies with Windows version) and look under Ports (COM and LPT). The device should appear as "FT232" or "FTDI" or similar.
5. Configure your application software to use this COM port number.
6. Unless running in a 4-wire half-duplex application, you will need to configure the RS422/485 port for the correct driver and receiver enable modes. This involves bridging some pins on the DB9 connector as shown under Connections.

Note: The USB-485 contains a unique USB device ID which ensures that whichever USB port it is plugged into, it always appears under the same COM port number. Usually, this is the desired behaviour for application software configuration reasons. However, it has two side effects:

(1) If you are going to be using multiple USB-485 converters on one PC, Windows will ask for the CD to be inserted for the installation of each one - because it thinks each one is a different device. In this case, copying the CD contents to a suitable folder on the PC and pointing the "new hardware found" function to that folder will save time.
(2) If you want the converter to always appear under the same COM port number, you need to always use the one specific USB-485 converter. If you plug in another USB-485 converter (even into the same USB port) it will appear under a new COM port number.

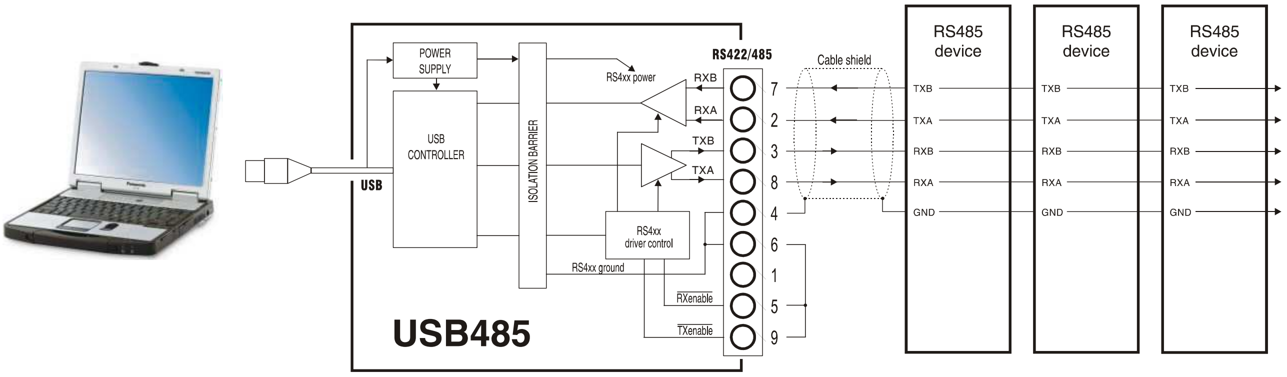
Connections

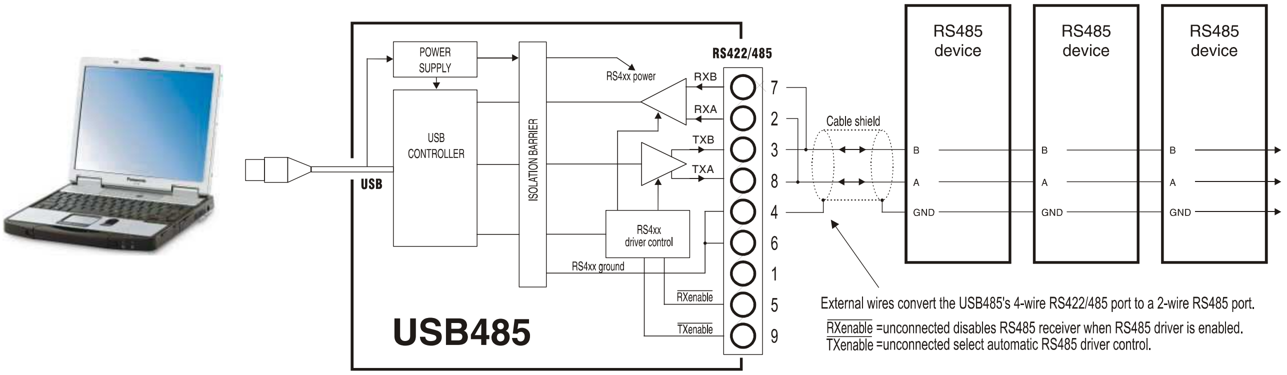
USB Connector	Pin Function		RS4xx DB9 Connector	Pin Function
1	USB +5V		1	No Connection
2	-- Data		2	RX-A
3	+ Data/Clock		3	TX-B
4	USB Ground		4	GROUND
Shield	Shield		5	/RXenable (note 1)
			6	GROUND
			7	RX-B
			8	TX-A
			9	/TXenable (note 2)

Note 1: To permanently enable the RS4xx receiver, connect pin 5 to GROUND
Note 2: To permanently enable the RS4xx driver, connect pin 9 to GROUND

2-wire RS485: Interconnect TX-A and RX-A and this gives the "A" connection. Interconnect TX-B and RX-B and this gives the "B" connection.
Ground: always connect the RX4xx ground terminal to the comms ground of the equipment.
A or B: Following RS422/485 standards, the RS422/485 connections above are marked A or B. They are defined as follows: when the start bit of the data is a "1" , the A output is HIGH (+5V nominally) and the B output is LOW (0V nominally). The quiescent state (no data present) of these signals is thus TX-A=0V; TX-B=+5V.

RS422, or a Master on 4-wire RS485





External wires convert the USB485's 4-wire RS422/485 port to a 2-wire RS485 port.
$\overline{\text{RXenable}}$ =unconnected disables RS485 receiver when RS485 driver is enabled.
$\overline{\text{TXenable}}$ =unconnected select automatic RS485 driver control.

Configuration

The USB-485 has no switch or software configuration. The only customisable features are whether the RS4xx driver is to be be permanently enabled (for RS422 point to point systems, or for use as a Master on a 4-wire RS485 multidrop system) or whether the RS4xx receiver is to be forced to be permanently enabled for receipt of all transmitted data (usually this is for RS422 point to point systems). These selections are done by grounding pins 9 and 5 respectively. The selection of baud rate and character format is done by the Windows application program.

Power Requirements

The USB-485 is powered entirely from the USB interface. It contains an isolated inverter which powers the isolated RS422/485 interface. The USB power consumption depends somewhat on the RS4xx cable/terminator loading; the 90mA is the worst case with a 100 ohm terminator. However, if connecting several USB-485 converters to a hub, the hub may need to be a powered hub because many laptops do not have the full-spec (500mA) USB ports.

Software Drivers

A driver is required within the operating system, to create the virtual COM port. There are two main vendors of USB-serial chips on the market: Prolific and FTDI. Most USB-RS232 converters on the market, including the KK Systems USB-232, use the Prolific chip and drivers. Nearly The USB-485 uses the FTDI chip and drivers, so if you are using one of the other converters you will still need to install the new FTDI drivers which are enclosed with this product. Furthermore, most of the Prolific-based products do not have a unique USB device identifier and plugging them into different USB ports generates new COM port numbers.

RS485 Driver/Receiver Control

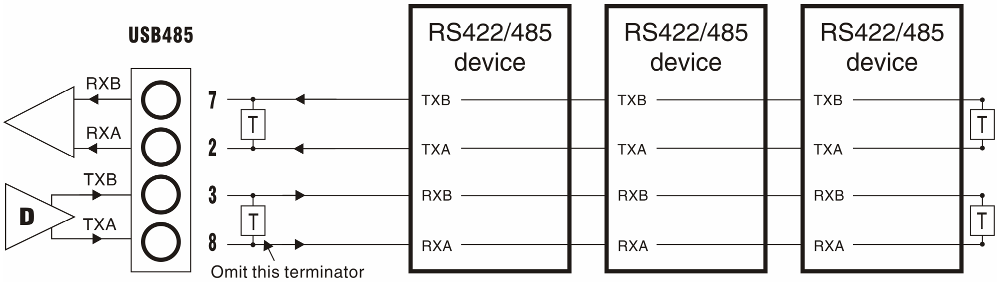
The USB-485 performs automatic **driver** enable (ADE) automatically, enabling the RS485 driver for each transmitted character. There is no need for your application software to provide RTS Control of any kind. This is overriden by grounding pin 9 of the DB9 connector which permanently enables the driver, for RS422 (or 4-wire RS485 Master) applications.

The USB-485 RS4xx **receiver** is disabled when the transmitter is enabled. If this is not desired (for full-duplex usage e.g. RS422), connect pin 5 of the DB9 connector to ground.

Terminators

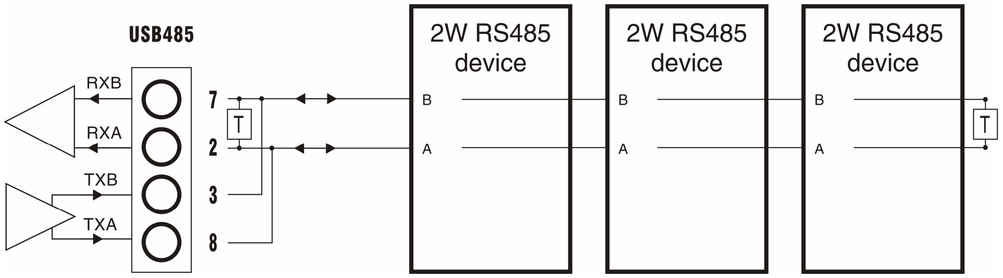
These are not normally needed for cable runs under 300 metres. The following diagrams show where they should be connected. The resistor value should equal the characteristic impedance of the cable - typically 100-120 ohms for proper grade RS422/485 cable.

RS422 / 4-wire RS485



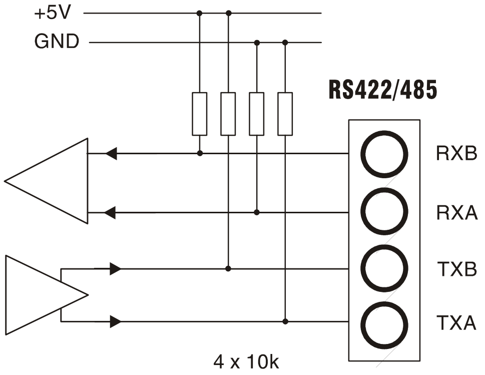
Omit this terminator
if driver D is always enabled
(if /TXenable is grounded)

2-wire RS485



Pullup Resistors

To ensure proper bus state in RS485 systems, the USB-485 contains internal pullup resistors as shown below:



Troubleshooting

The application program does not find the RS422/485 device
(1) Look in Control Panel / System / Hardware / Device Manager / Ports and check the COM port number appearing there for the USB-485 matches your application program configuration.
(2) Check the COM port settings (baud rate, etc) in your application program.
(3) Check the RS422/485 connections - there is no standard in the wiring and most RS422/485 devices need a custom cable to be constructed.

The USB-485 installs under a too-high COM port number and the application program doesn't support it
This often happens if you have been "playing" with Bluetooth. Windows does not offer a clean means of removing unused port numbers; please email KK Systems for instructions. Alternatively it is usually possible to force the USB-485 driver (under Ports / Properties / Advanced) to use a different COM port number; be sure it isn't used by another device.

Specification

USB interface:	USB1.1 (USB2 compatible). Current consumption 90mA max.
RS4xx interface:	Standard specification drivers and receivers. Receiver Zin 96k ohm min. Max common-mode voltage -7V to +12V.
Serial parameters:	All baud rates supported by the Win32 API up to and including 115200 baud
Isolation:	Test voltage 1500V AC RMS for 1 second
Environmental:	Operating 0 to +50C, storage -40C to +70C RH 0 to 90% non-condensing
CE compliance:	Emissions EN50081-2. Immunity EN50082-2
Dimensions:	130mm x 53mm x 21mm (case only)

Contact

KK Systems Ltd
Tates, London Road
Pyecombe, Brighton
BN45 7ED
Great Britain

tel +44 1273 857185
fax +44 1273 857186
email: support@kksystems.com
http://www.kksystems.com

Edition 5   12th June 2007 E & O.E.