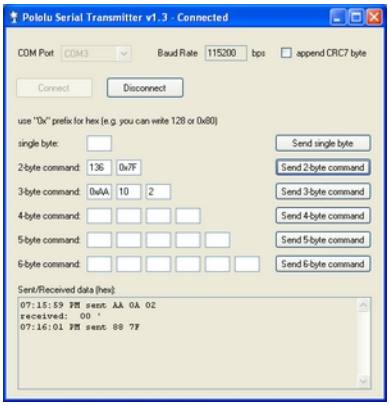
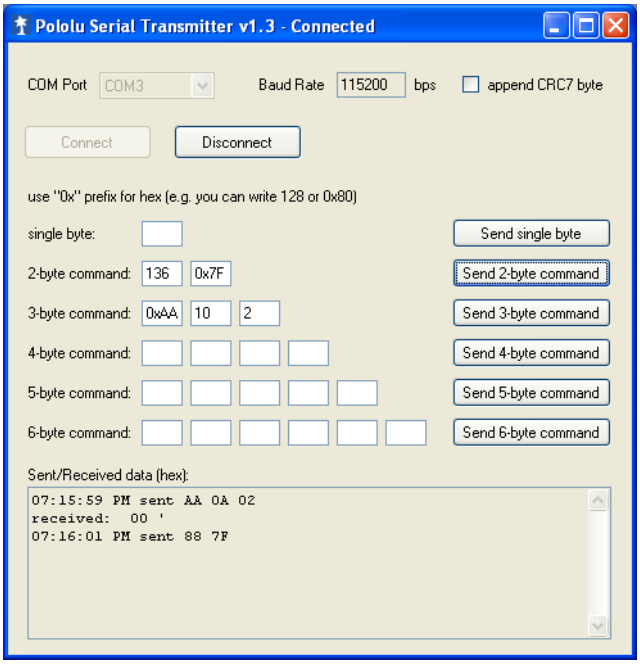
Pololu Serial Transmitter utility for Windows



This is a simple serial transmitter utility for Windows that lets you transmit sequences of bytes to a selectable COM port. The most recent version allows you to specify the baud rate, and it reports back any serial data it receives while you are connected. This program can be used to send commands to our serially controllable devices, which can help you troubleshoot problems by letting you determine if the fault lies with your serial code or with the device itself. To install the program, download and unzip the following archive:

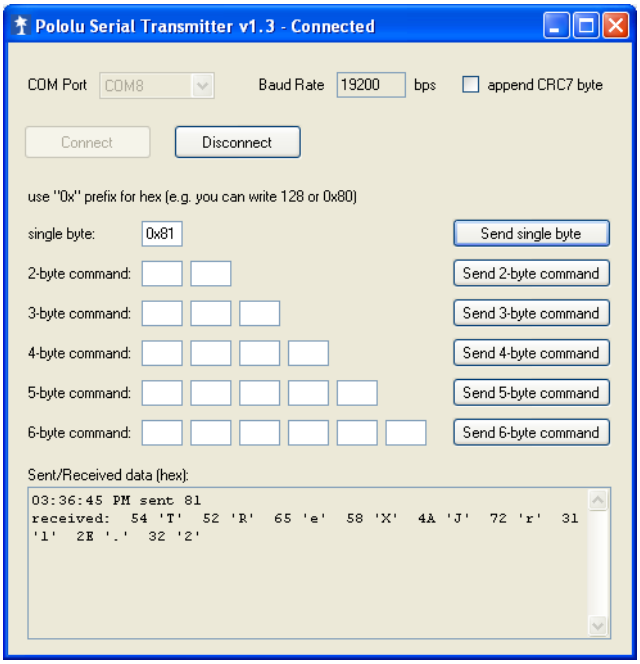
Pololu Serial Transmitter [http://www.pololu.com/file/download/PololuSerialTransmitterV1.3.zip?file_id=0J132] (256k zip)

and open the file setup.exe. The installed program will appear in your Start menu as **Pololu Serial Transmitter** under the **Pololu** category.



You can send a single byte at a time using the **Send single byte** button, or you can send entire packets up to six bytes in length. Simply enter the values of the bytes you wish to transmit in the text boxes. These values are interpreted as *decimal* (base-10) integers unless they are preceded by the **0x** prefix, in which case they are interpreted as *hexidecimal* (base-16) integers. For example, to send a byte that has a value of 128, you can either type **128** or **0x80**. The log at the bottom of the window keeps a record of the packets that have been sent and received. This log displays each sent byte as a two-digit hex number and each received byte as a two-digit hex number followed by its ASCII character representation surrounded by single quotes. Note that some bytes (e.g. 0) represent special ASCII characters that don't display well.

The following screen capture shows what happens if you use the serial transmitter utility to send the command byte 0x81 ("Get Signature") to the **TReX Jr motor controller [http://www.pololu.com/catalog/product/767]**:


Pololu Serial Transmitter v1.3 - Connected
COM Port
COM8
Baud Rate
19200
bps
append CRC7 byte
Connect
Disconnect
use "0x" prefix for hex (e.g. you can write 128 or 0x80)
single byte:
0x81
Send single byte
2-byte command:
Send 2-byte command
3-byte command:
Send 3-byte command
4-byte command:
Send 4-byte command
5-byte command:
Send 5-byte command
6-byte command:
Send 6-byte command
Sent/Received data (hex):
03:36:45 PM sent 81
received: 54 'T' 52 'R' 65 'e' 58 'X' 4A 'J' 72 'r' 31
'1' 2E '.' 32 '2'