ECE/CSE 474:
Intro to Embedded Systems

Using PuTTY for UART Communication

The instructions in this document are slightly biased toward the Windows OS. Moreover, this document assumes that you are using the TM4C1294XL Tiva Board.

A Computer's Serial Interface: PuTTY

If you are transmitting data from your microcontroller to your computer, you are going to need a way of seeing it on your computer: this is where a serial interface comes in handy. A common software application that provides a serial interface, available on Windows, macOS, and Linux, is PuTTY. You can install PuTTY from this website. Note that for MAC and Linux users, the most convenient way to use PuTTY is with the UNIX command line provided with your OS.

To use PuTTY as a serial interface with your microcontroller, you'll first need to find the COM port associated with your microcontroller. In Windows, this can be found in the Device Manager menu, usually labeled as "Stellaris Virtual Serial Port" (make sure that you have connected your TM4C to your computer). Figure 1 shows how you can find the Device Manager menu in Windows OS. Figure 2 shows what you should expect to find, except you may find a different COM port (COM5 instead of COM4, for example).

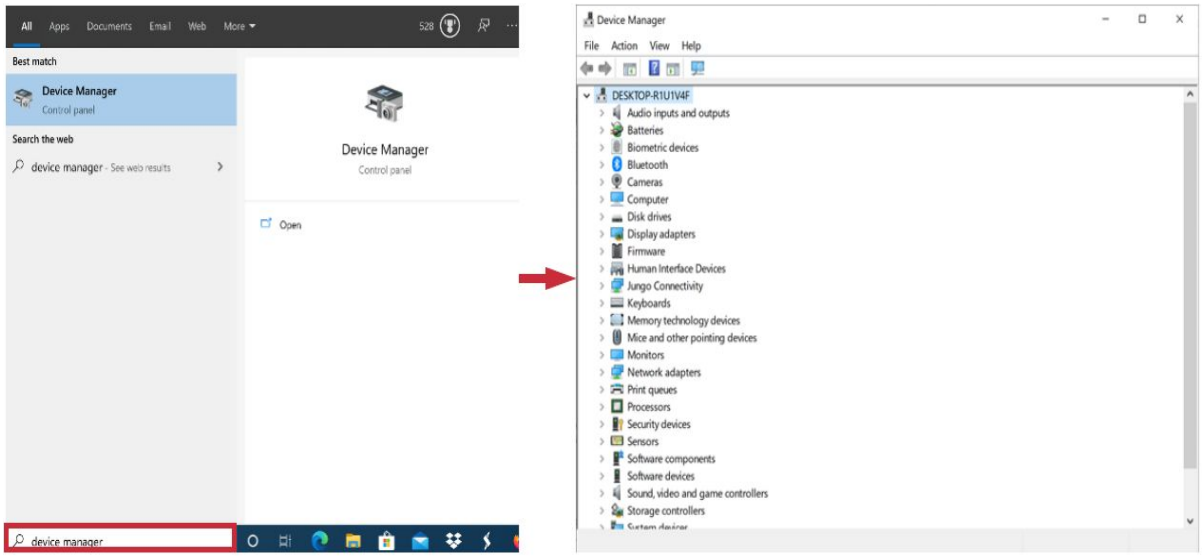


Figure 1. Steps to find Device Manager in Windows

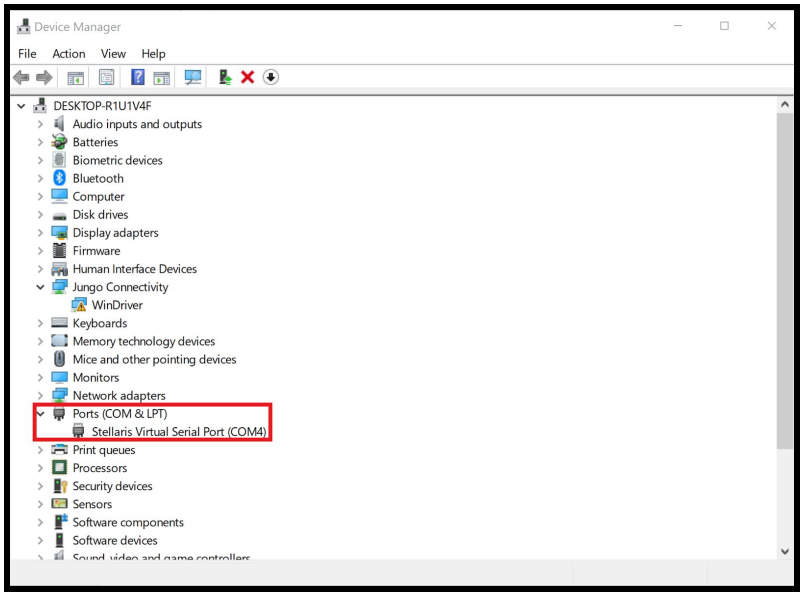


Figure 2. Device Manager Window showing the virtual serial port provided by the Tiva Board.

Open up PuTTY, and the PuTTY window in Figure3 should appear. Figure 3 also points out the only two configuration menus you'll have to deal with. Select Serial Mode by pressing on the "Serial" button, emphasized by the red arrow in Figure 3.

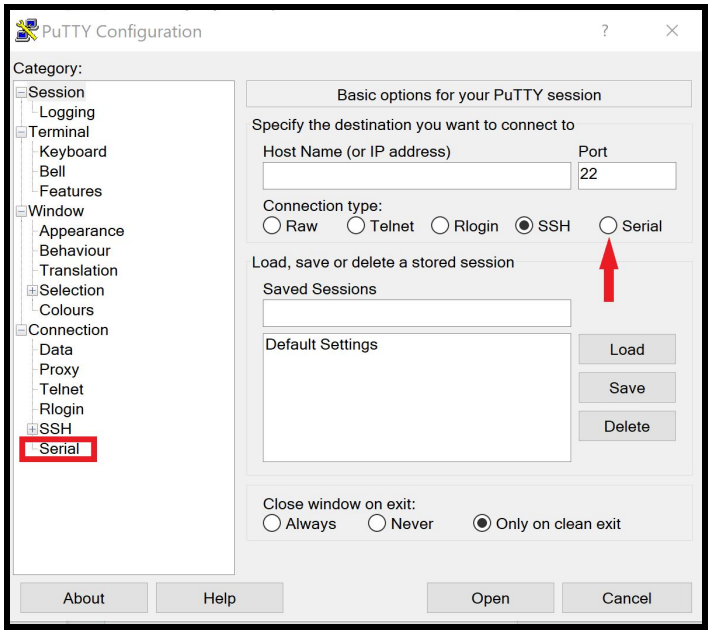
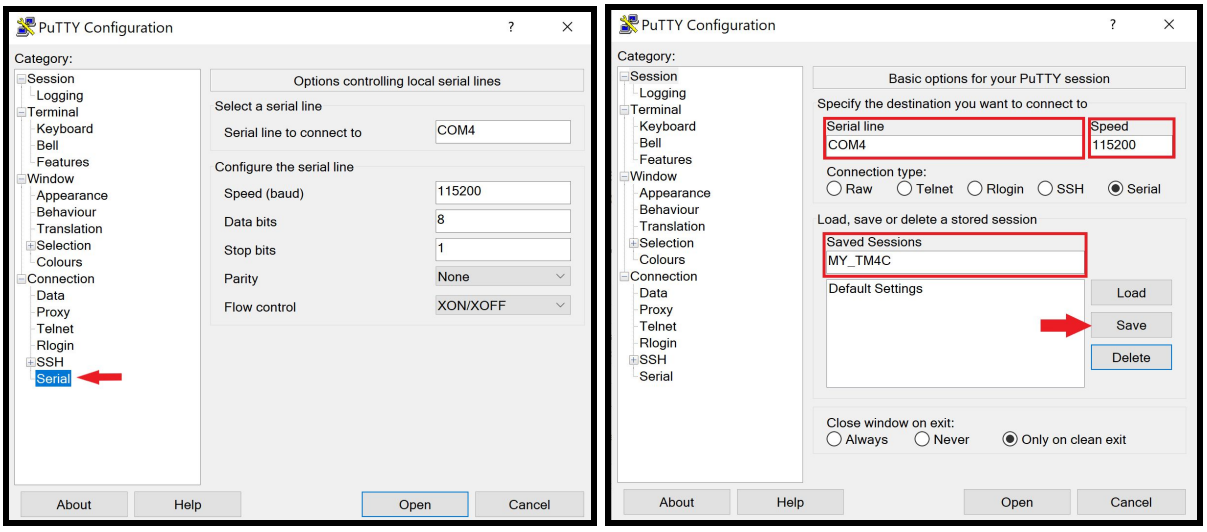


Figure 3. PuTTY window when opened for the first time.

Figures 4-5 (Figure 4 left, Figure 5 right) show the configuration windows for the serial interface. **The default settings of UART are displayed in these windows**. It is recommended that you use these settings when configuring your UART. Make sure you match the serial line with the COM port that the Virtual Serial Port is connected to. You can save a serial setting by writing a descriptive name in the "Saved Sessions" box and clicking on the "save" button.



Figures 4-5. PuTTY Serial Setup Guide

Now that you have your PuTTY setup, you can create a PuTTY terminal to see UART data. Figure 6 visually describes this process. If you saved your settings, you can click on the saved session and press "Load" to access your saved settings. With your correct settings selected, press the "open" at the bottom right of the PuTTY terminal. If you followed all the steps correctly in setting up your microcontroller, you should now be able to see UART data transferred serially! The data shown in the terminal is an example outcome of Task2.1 of Lab3.

One more note to complete this discussion on serial communication. Humans, in general, have a hard time reading binary when bits are not squished together; you can imagine that data displayed as 1's and 0's without any clear delineation between them would be an absolute nightmare to read. PuTTY (or any useful serial interface, for that matter) will not display data in binary. Instead, it will treat your transmitted data as ASCII characters.

When transmitting data, you must convert your values into strings (or, character arrays in C) so that you can actually recognize the data in your serial interface. Otherwise, you'll get some weird mumbo-jumbo, and not the good kind. **It is anticipated that you will use C libraries to convert data to readable character arrays, especially floating-point values**.

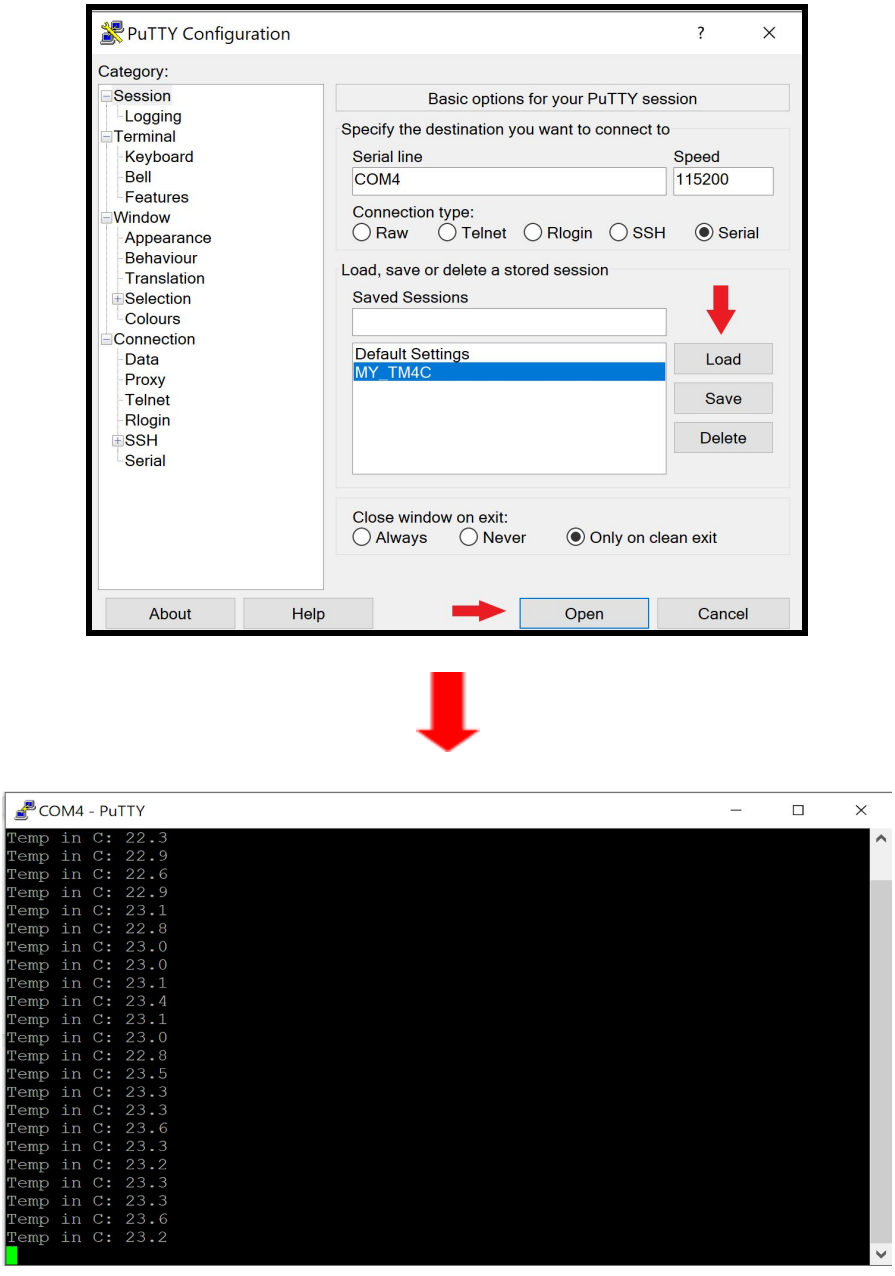


Figure 6: Transition from PuTTY serial configuration to PuTTY Terminal