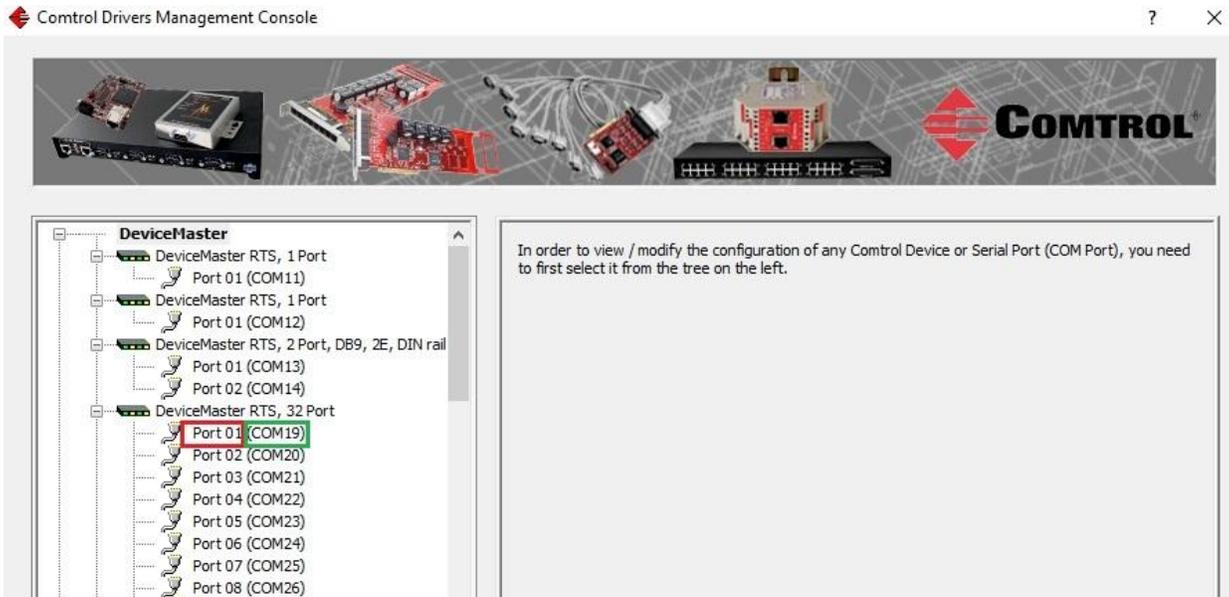


Control Device Master Serial Data T.S.

How can I tell I'm getting valid data from my application to the DeviceMaster serial port? It looks like there are random characters in the data stream.

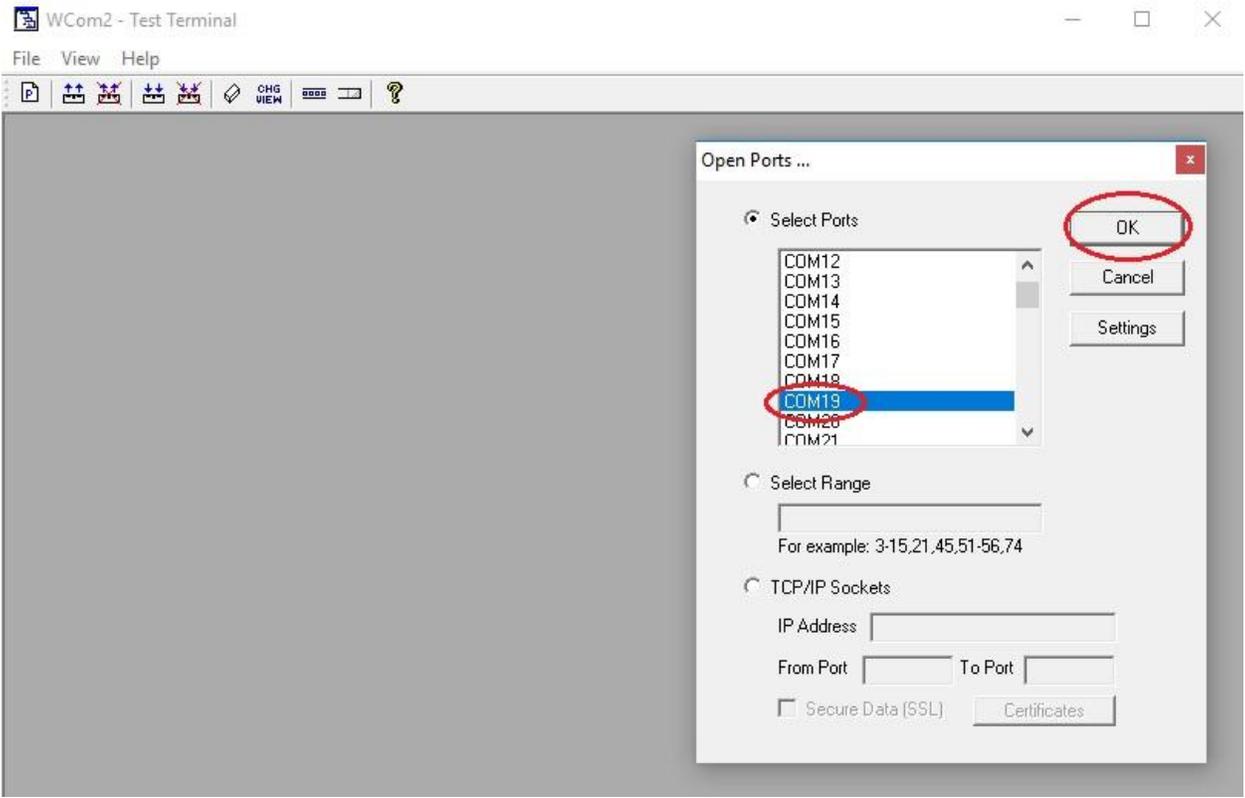
1. Determine whether this issue is from the computer to the DeviceMaster.
 - a. Open the DeviceMaster Driver Management Console
 - b. Determine which DeviceMaster physical port is mapped to your computer's comm port.
In the screenshot below, you see the 32port Device Master's physical port 01 (Red Box) is mapped to the computer's comm port 19 (Green Box).



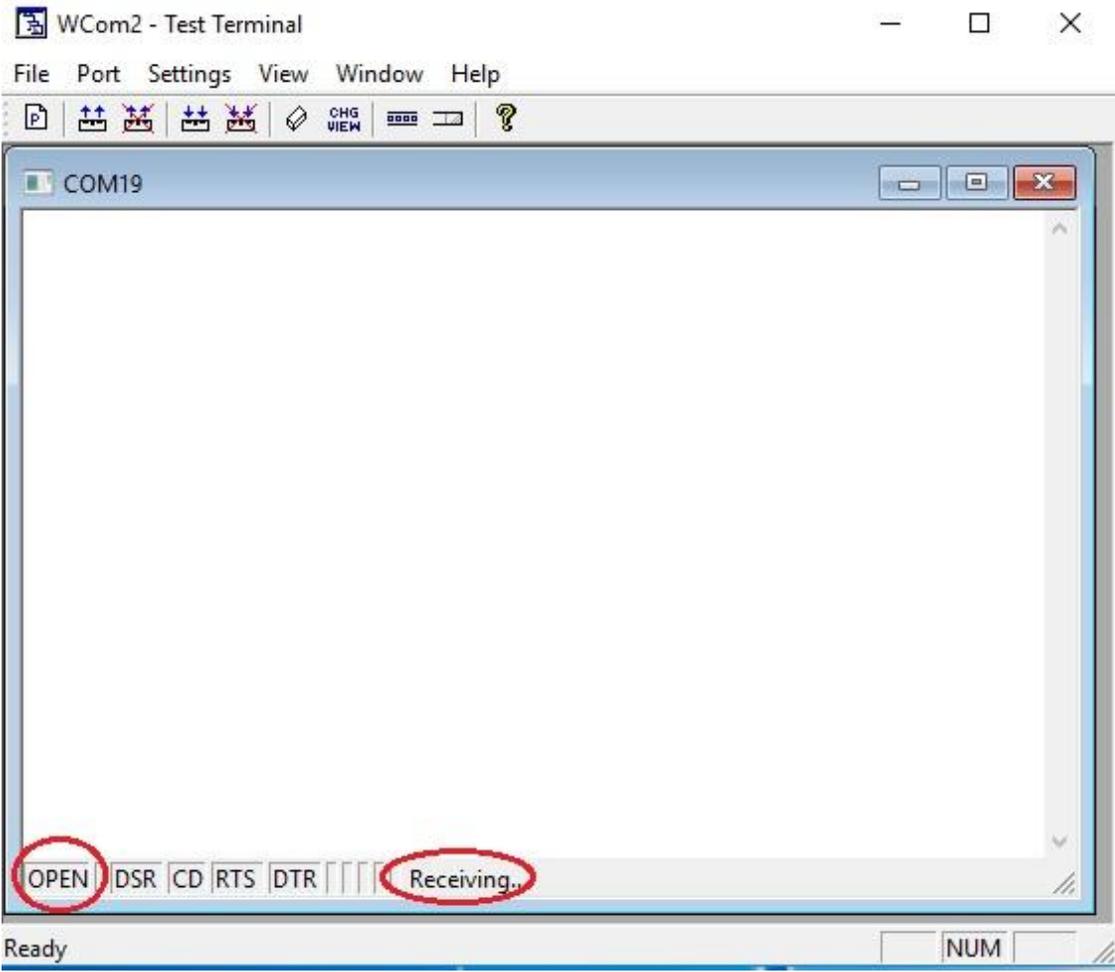
- c. Go to Start Menu -> Control -> PortVision DX and open the Application.
- d. After the GUI comes up, go to the top menu 'Tools' -> 'Applications' -> 'Test Terminal (WCom2)'.
- e. In the WCom2 – Test Terminal GUI, go to 'File' -> 'Open Port' -> Com19 (see step b) and click ok.

NOTE- If the Comm Port can't be opened, this is usually because:

1. Comm port is busy (is being accessed by computer's application)
2. Driver is not able to communicate to the Device Master
3. Another computer is using that Device Master's Physical Port



f. Observe that the Comm Port is 'Open' and 'Receiving'. See below.



- g. Attach the loopback connector to the back of the DeviceMaster's Physical Port that was mapped to Comm 19. In this case, it would be Physical Port 1. If there is no loopback plug available, then you can fabricate one from a surplus serial cable. Please see the below wiring for creating a loopback plug.

RJ45 Loopback Plugs (RS-232/422)

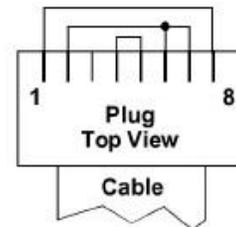
Loopback connectors are RJ45 serial port plugs, with pins wired together as shown, that are used in conjunction with application software ([Test Terminal](#)) to test serial ports.

The DeviceMaster is shipped with a single loopback plug for testing RS-232/422 mode.

Pins 4 to 5

Pins 1 to 8

Pins 2 to 6 to 7



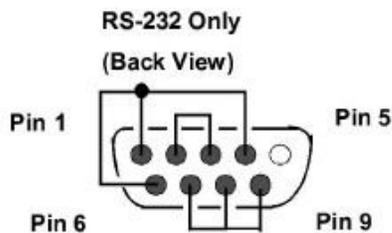
The RS-232 loopback plug also works for RS-422.

DB9 Loopback Plugs (RS-232/422)

Loopback connectors are DB9 female serial port plugs, with pins wired together as shown, that are used in conjunction with application software ([Test Terminal](#)) to test serial ports. Each DeviceMaster is shipped with a single loopback plug for testing RS-232/422 mode.

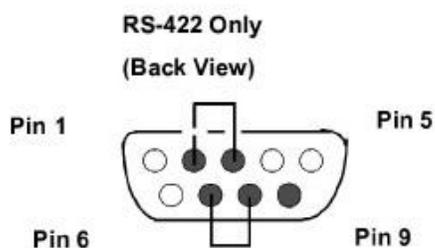
Wire the pins together to build additional plugs or replace a missing RS-232 loopback plug.

- Pins 1 to 4 to 6
- Pins 2 to 3
- Pins 7 to 8 to 9

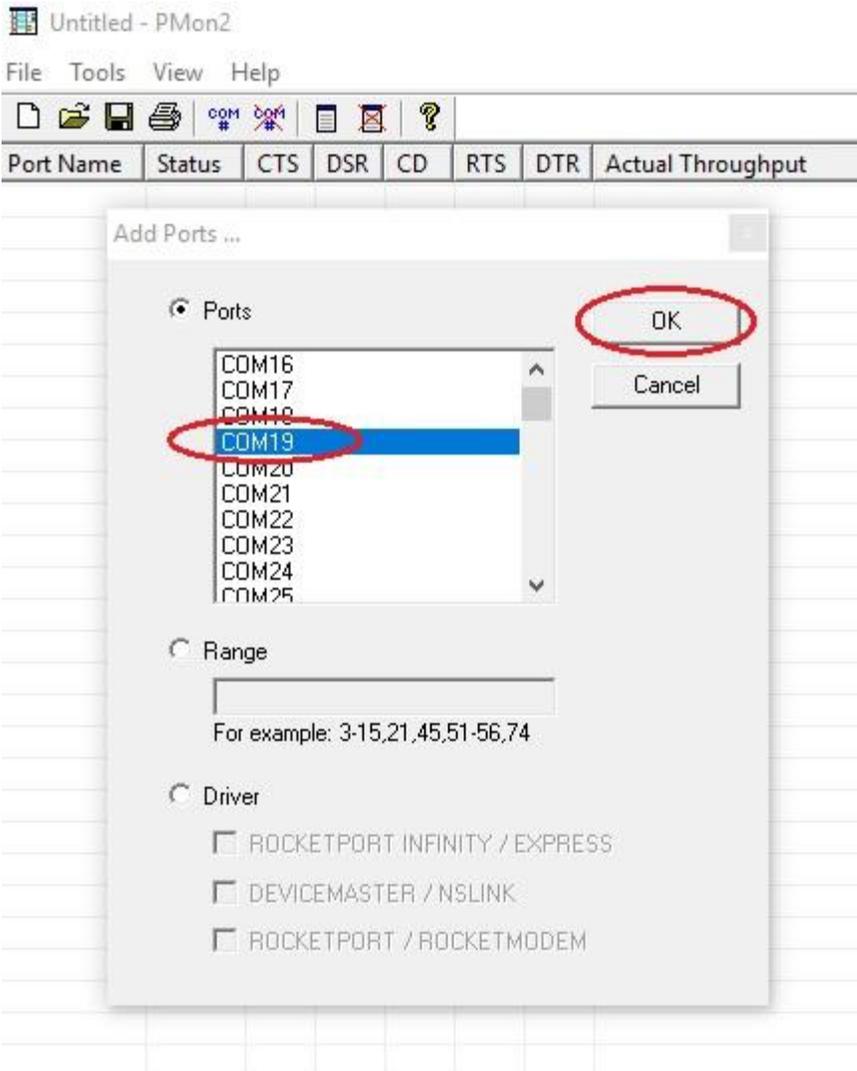


Wire the pins together for an RS-422 loopback plug.

- Pins 2 to 3
- Pins 7 to 8



- h. After the loopback plug is attached to the port, go back to PortVision and open menu item 'Tools' -> 'Applications' -> Port Monitor (PMon2).
- i. In the PMon2 GUI, go to 'Tools' -> 'Add Ports' -> Select 'Com19' and click 'OK'.

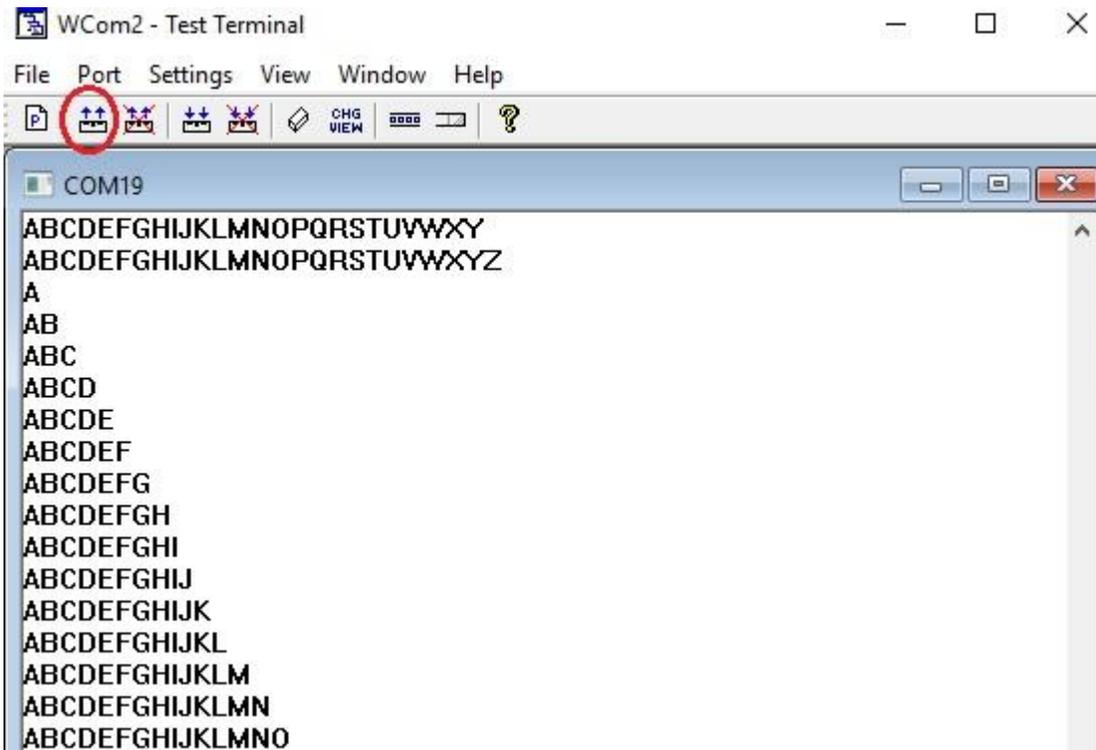


You should see the following:

The image shows the main window of the PMon2 application. The 'COM19' port has been added to the monitoring table. The table has the following columns: Port Name, Status, CTS, DSR, CD, RTS, DTR, Actual Throughput, TxTotal, RxTotal, Total Errors, OverrunErrors, FramingErrors, and ParityErrors.

Port Name	Status	CTS	DSR	CD	RTS	DTR	Actual Throughput	TxTotal	RxTotal	Total Errors	OverrunErrors	FramingErrors	ParityErrors
COM19	OPEN	OFF	ON	ON	ON	ON	0	0	0	0	0	0	0

- j. Now you should have Test Terminal and Port Monitor open, with a loopback connector on the Physical Port of the Device Master. In Test Terminal, initiate the data loopback by selecting the menu item illustrated below. You should see scrolling alpha characters.



- k. Examine Port Monitor for errors. In the screenshot below, notice there are no errors, but the CTS is not active. The data loopback will pass (TxD and RxD), but a control signal loopback test will fail.

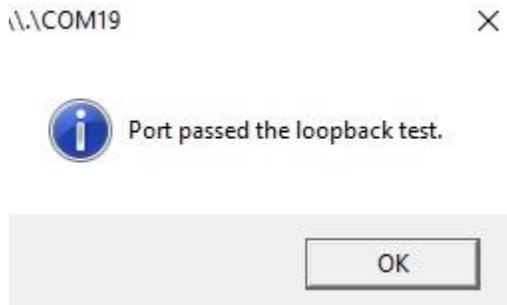
The screenshot shows a window titled "Untitled - PMon2". The menu bar includes "File", "Tools", "View", and "Help". The toolbar contains several icons, with the "Port Monitor" icon circled in red. The main area displays a table with the following data:

Port Name	Status	CTS	DSR	CD	RTS	DTR	Actual Throughput	TxTotal	RxTotal	Total Errors	OverrunErrors	FramingErrors	ParityErrors
COM19	OPEN	OFF	ON	ON	ON	ON	115085	331692	328173	0	0	0	0

- l. To illustrate the failure, go to Test Terminal GUI, select 'Port' -> 'Loopback Test'. A failure appears as:



Note- This should be investigated. In the above example the loopback plug was not inserted all the way. Once inserted correctly, the CTS appeared as 'On' and the Loopback test was successful.



The above example illustrates possible issues customers have if a cable is not inserted securely to the port. Data integrity may be affected as well as proper serial operation. Attention to detail is important when running tests.

- m. If the loopback test is successful, then this proves communication between the computer's driver -> computer's comm port -> Ethernet connection from computer to Device Master -> Device Master's serial port output -> back to computer's driver (send and receive) are good.

Factors that can affect data integrity after leaving the DeviceMaster's serial port are:

- i. Cable length too long
 - ii. Cable wiring issues (worn, incorrect wiring)
 - iii. Noisy environment where the cable is picking up environment electrical noise.
 - iv. Serial Port configuration mis-match (baud rate, parity, stop bits, flow control, etc.
- n. Try attaching the loopback connector at the end of the serial cable prior where it is connected to the end-point equipment. Re-run the test and check for errors. If there are errors when running the loopback test at the end of the cable but not at the end of the Devicemaster, then the issue has to be associated with the cable.
 - o. If there are no errors at the end of the cable, then additional troubleshooting is required.
 - i. Has the issue always been this way or did it just start?
 - ii. If it just started, when did it start (day), what changed?
 - iii. Does this happen at a particular time or day or during a particular step in the communication process / application?
 - iv. Is this only with one DeviceMaster port or does this happen with more than one or a different port? Have the application use a different port, if possible, and change the physical port cable to match. See if error persists.
 - p. If there were errors when doing the data loopback test monitoring the Test Terminal and Port Monitor output, then a program called 'Comsniff' along with 'Wireshark' should be used along with Test Terminal to further isolate the problem.
 - i. Comsniff utility download and instructions are located here:
<http://downloads.comtrol.com/contribs/utilities/comtrol/ComSniff/> Note- Comsniff acts as an intermediary between the application and the driver. Comsniff should be started first, then the application. Comsniff will intercept the application's data prior to the driver and pass this to the

driver. If the data is correct within the Comsniff application, then we need to look at what happens after it leaves the driver (Wireshark).

ii. Wireshark – Instructions to use Wireshark are here:

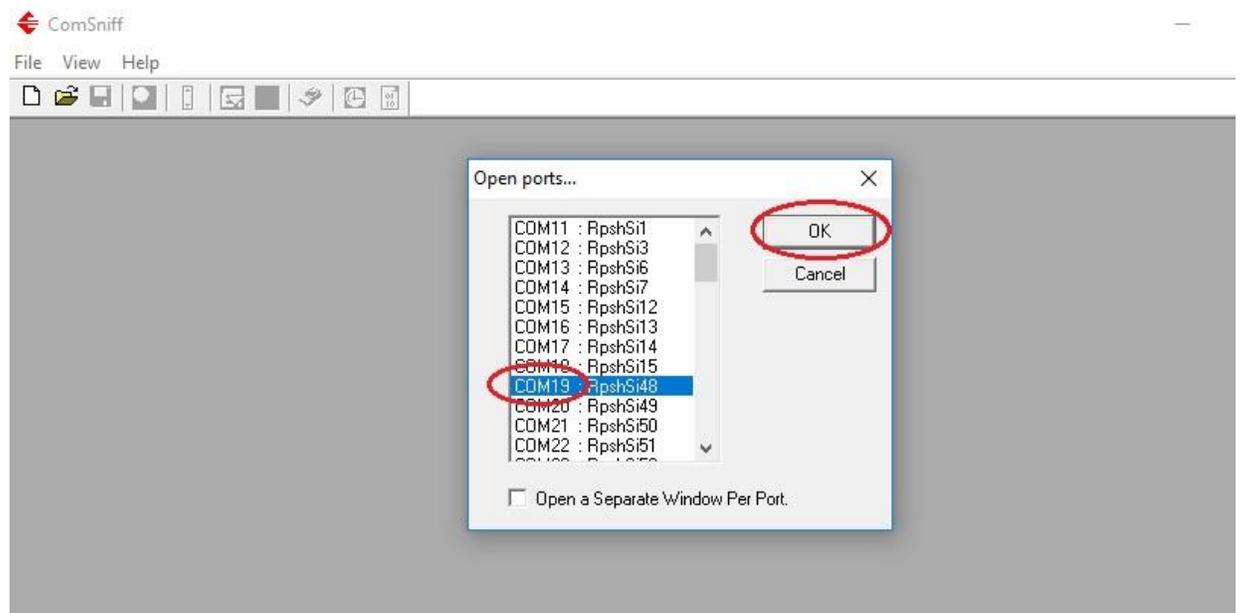
http://downloads.control.com/contribs/devicemaster/help_files/tracing/ in the wireshark version2 usage.pdf file. To get Wireshark, go to their website:

<https://www.wireshark.org/download.html>

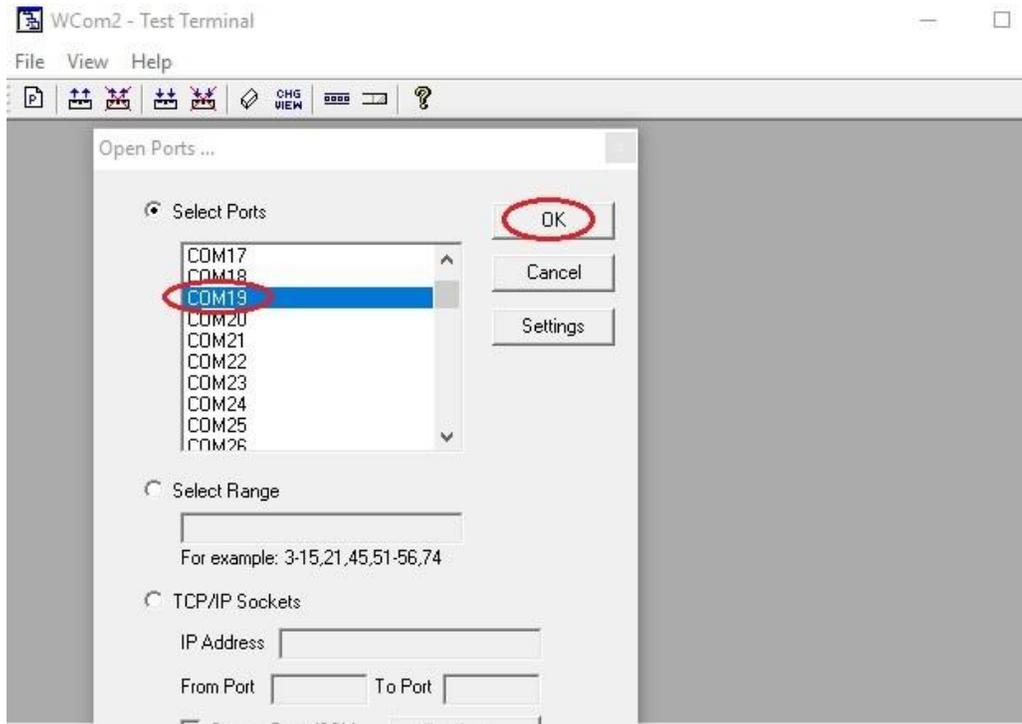
q. Run Wireshark, Comsniff and Test Terminal concurrently to determine where the spurious / corrupted data is being introduced into the data stream. Note, Test Terminal will open the Comm Port and this will prevent the Application from opening the Comm Port. A choice must be made to test using the Application OR with Test Terminal to avoid physical port conflicts.

Example: Assuming Test Terminal is acting as the application.

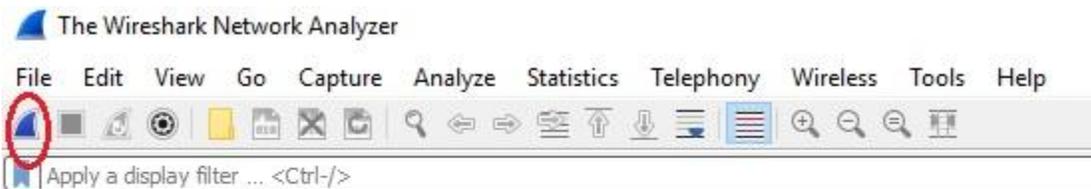
1. Open Comsniff, in the GUI select 'File' -> 'New Capture Window'
2. Select Comm Port to be tested. In this example, we are using Com19.
3. Click 'OK'. Comsniff is now waiting to intercept data coming from the Application (Test Terminal) to the Driver.



4. In Test Terminal, in the menu, go to 'File' -> 'Open Port' -> COM19. Click OK

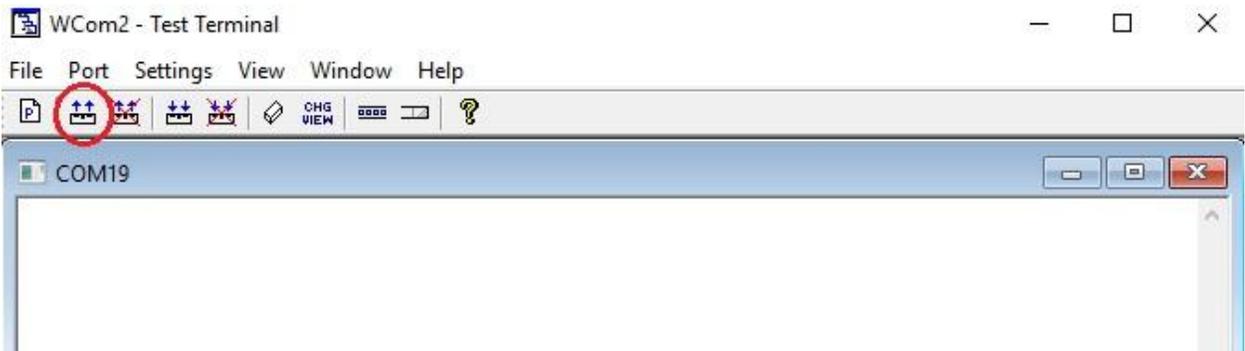


5. Data should start appearing in Comsniff as port communications occur between Test Terminal and the Driver. The Window will quickly fill with message requests, "IOCTL_SERIAL_GET_COMMSTATUS".
 - a. Filter this out by going to Comsniff's Menu -> Edit -> Capture Filter.
 - b. When the filter GUI appears, enter *GET_COMMSTATUS* in the 'Exclude' window.
 - c. Click 'Apply'. The messages should stop scrolling.
 - d. In the Comsniff Window, go to the Menu -> 'Edit' -> 'Clear Display'
6. Start Wireshark tracing by clicking on the 'Fin' at the far left of the menu.

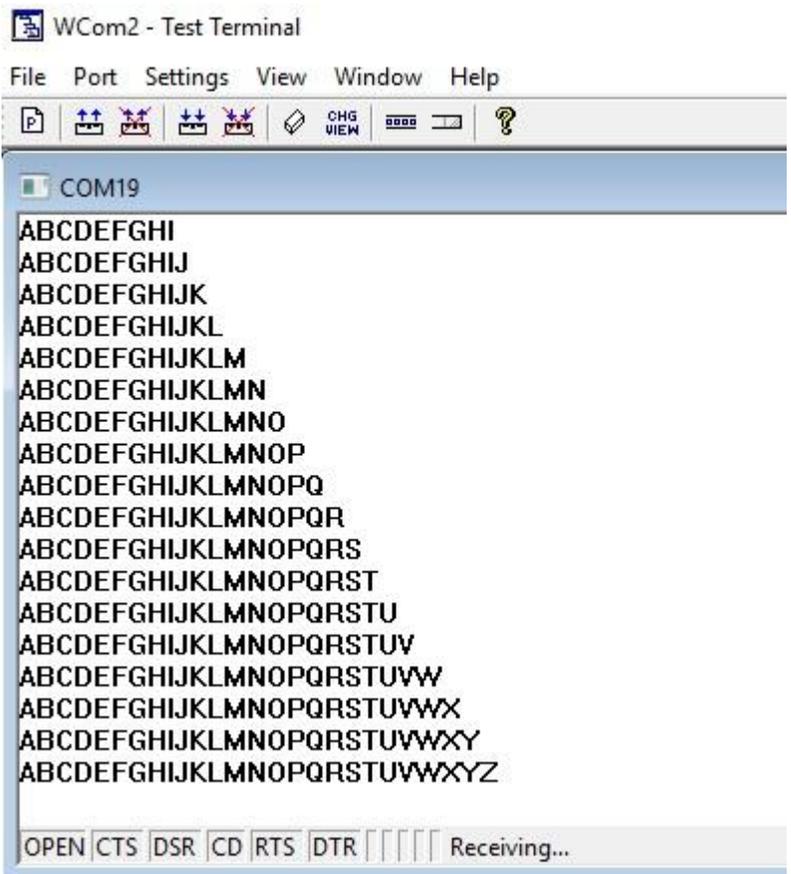


Welcome to Wireshark

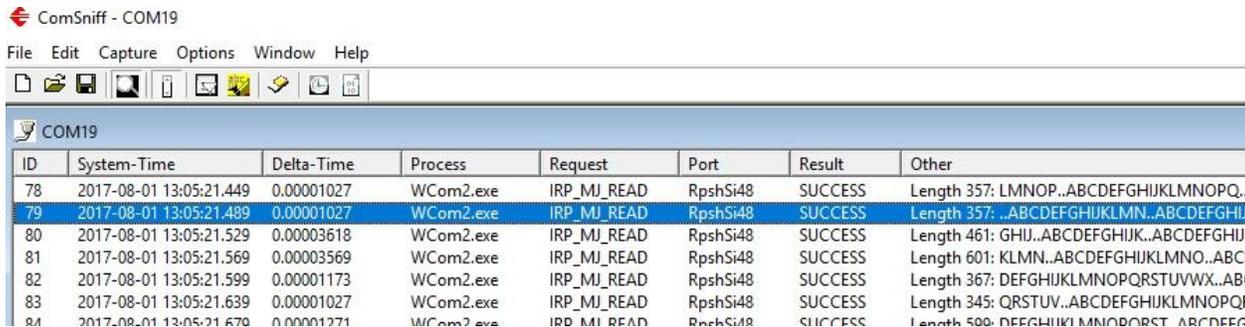
7. Proceed to do a data loopback test in Test Terminal by clicking on the box with the two blue arrows pointing up.



8. Data should be scrolling in Test Terminal.



9. ComSniff should show scrolling alpha characters:



10. In the Wireshark GUI, there will be accumulating data. Let this continue until the error occurs while being mindful of disk space. Wireshark could fill the drive easily.
11. Examine the ComSniff and Test Terminal traces for any extra / corrupted characters.