RS485 Basics:

Remote Lab instructions

Serial RS485

V1

Created By:	Danie Stoop, Deon Reynders	Date:	Pre-2/08/14
Reviewed By:	Steve Steyn	Date:	10/11/14

EIT ENGINEERING INSTITUTE OF TECHNOLOGY

Instructions
RS485 Basics

Getting Started

- Logon to Electromeet (Follow the How to Connect to Remote Labs_Electromeet_HTML5_Remote_Lab instructions document)
- The software is installed on **Remote Lab 6**

Hardware:

*As you will be doing this exercise by way of a remote lab please note that all the hardware is already set up and connected to **Remote Lab 6**.* ***Any other remote lab will not have the correct hardware setup.***

Software & Hardware used:

The lab was set up with the following hardware:

- Laptop with at two USB ports
- 1x MOXA UPort 1130 USB-to-serial (RS-485) adapter
- 1x PicoScope 2204 USB oscilloscope, with probes

The following software has been installed:

- Listen32 (for sending characters via the COM port)
- Picoscope6 software

RS485

This exercise will build upon the previous (RS-232) exercise, and will introduce the use of RS-485.

1.0 Implementation

1.1 Basic port setup

As you will be doing this exercise by way of a remote lab, all of the equipment will already be set up, but please read the instructions below to familiarize yourself with the procedure.

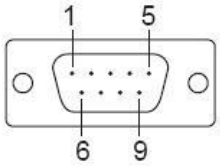
The Port setup is exactly the same as the last lab, except this time we will be using **Port 6**.

So everywhere in this exercise please select **Port 6**. You may want to verify this via Device manager, as in the case of RS-232.

And instead of using a Moxa1110 as in the last exercise we are using a **Moxa1130** this time.

The UPort 1130 is identical in appearance to the 1110, though the outputs are different.

The outputs are accessed through a screw terminal with the following configuration.

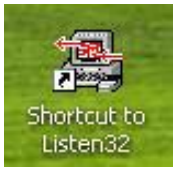
		UPort 1110	UPort 1130	
DB9 (male)	Pin	RS-232	RS-422 4-wire RS-485	2-wire RS-485
	1	DCD (in)	TxD-(A)	-
	2	RxD (in)	TxD+(B)	-
	3	TxD (out)	RxD+(B)	Data+(B)
	4	DTR (out)	RxD-(A)	Data-(A)
	5	GND	GND	GND
	6	DSR (in)	-	-
	7	RTS (out)	-	-
	8	CTS (in)	-	-

The UPort 1130 has been set up (via Device manager) to operate in RS-485 'two wire' (2W) mode. The PicoScope probes have been attached to pins 3 and 4 on the screw connector: oscilloscope Channel A to RS-485 line 'A' (Data-) and oscilloscope Channel B to RS-485 line 'B' (Data+).

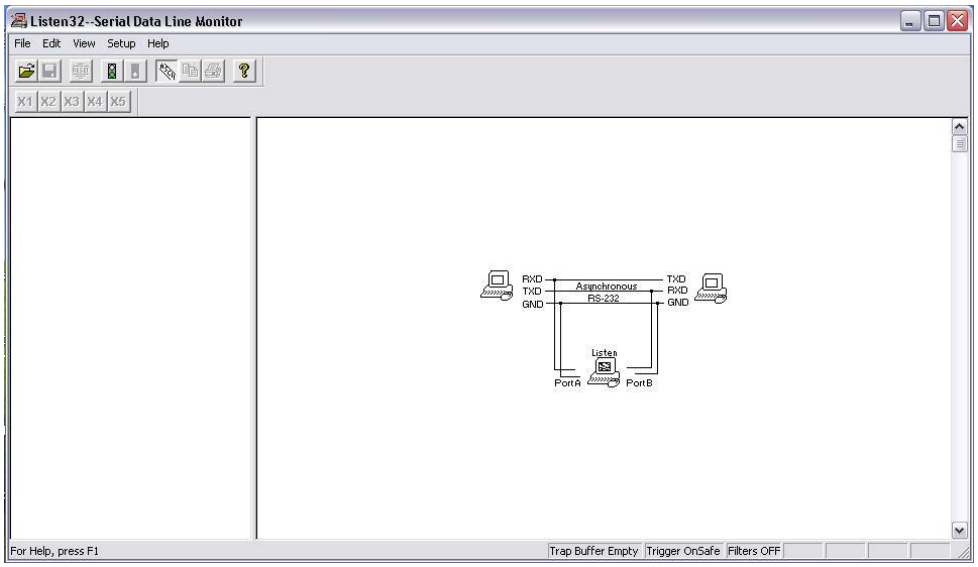
The oscilloscope ground clips have not been attached, as grounding in this case takes place via the USB cable.
NB: NEVER attach a ground clip to either line A or line B, as this will short the line to ground and render it inoperative.

1.2 Prepare to send characters

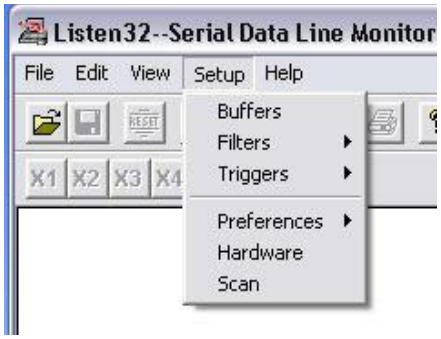
1. Run Listen32 by clicking on the desktop icon or on the taskbar at the bottom of the screen.



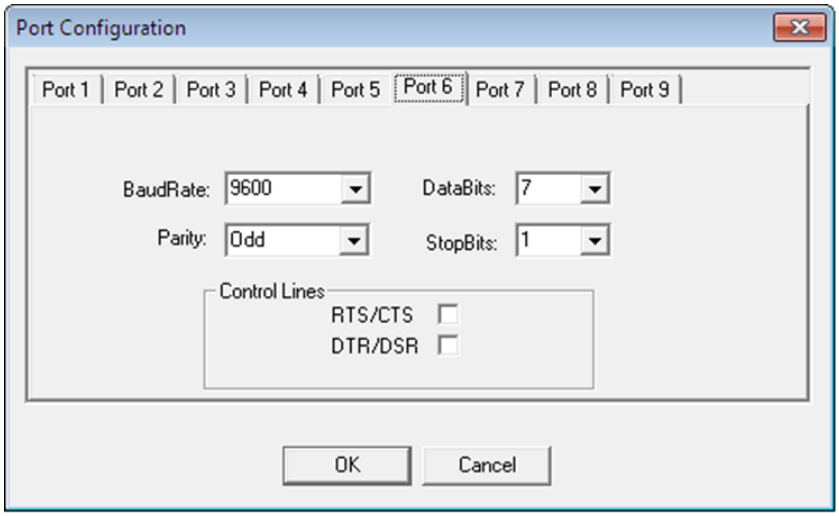
Listen32 will open up:



2. Click Setup->Hardware:

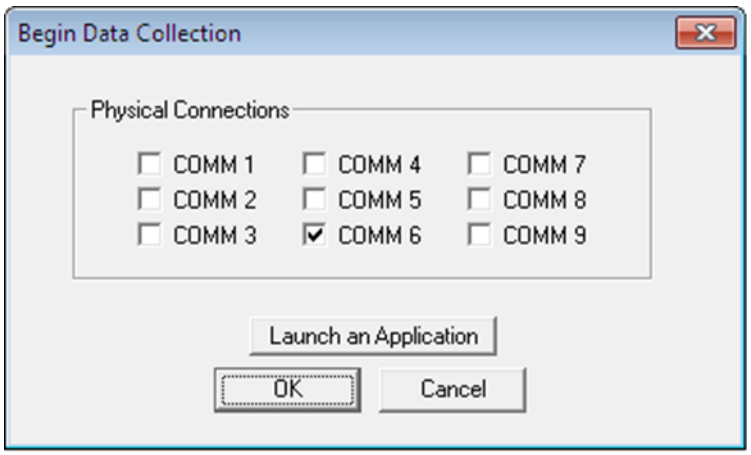


3. Then set the UART parameters for the appropriate COM port (**COM6** in this case). Set the parameters exactly as per the following figure: 9600, 7, O, 1.



4. Next, start the port monitoring process by clicking on the green traffic light (click red to stop – remember to click green light again to start it again – the red light is shown while the connection is active).

5. The 'Begin Data Collection' window will appear. Select the appropriate port (**COM6** in this case).



Listen32 is now running and the UART on the Moxa device is configured correctly
Now you must confirm the idle voltage

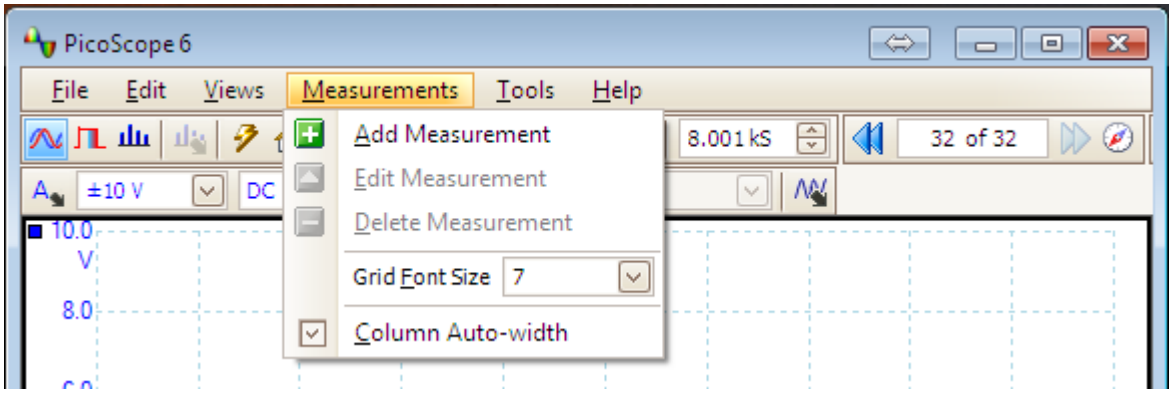
1.3 Confirmation of idle voltages

We are doing this remotely thus we can measure the TxD line idle voltage with the Picoscope.

To do this you must

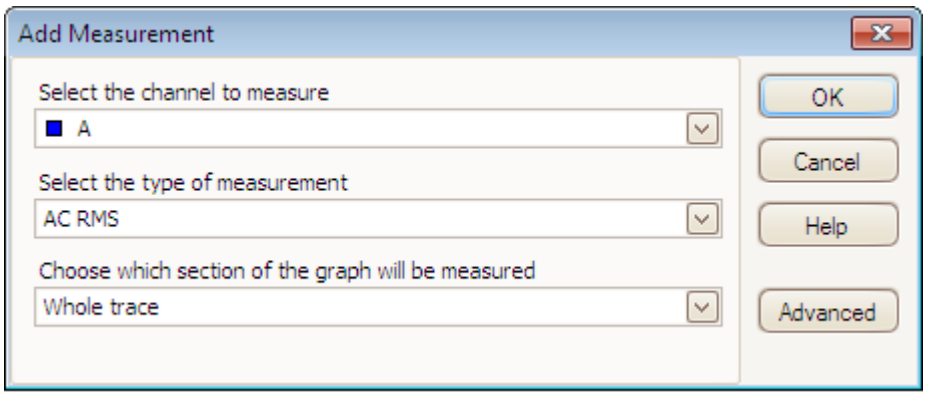
6. Run the Picosope6 software. Double-click the PicoScope6 icon on the desktop of the remote computer (Remote Lab 5)

7. We can add a measurement function to the PicoScope, which will allow us to obtain a DC voltage reading. Firstly click “Measurements” at the top of the PicoScope window

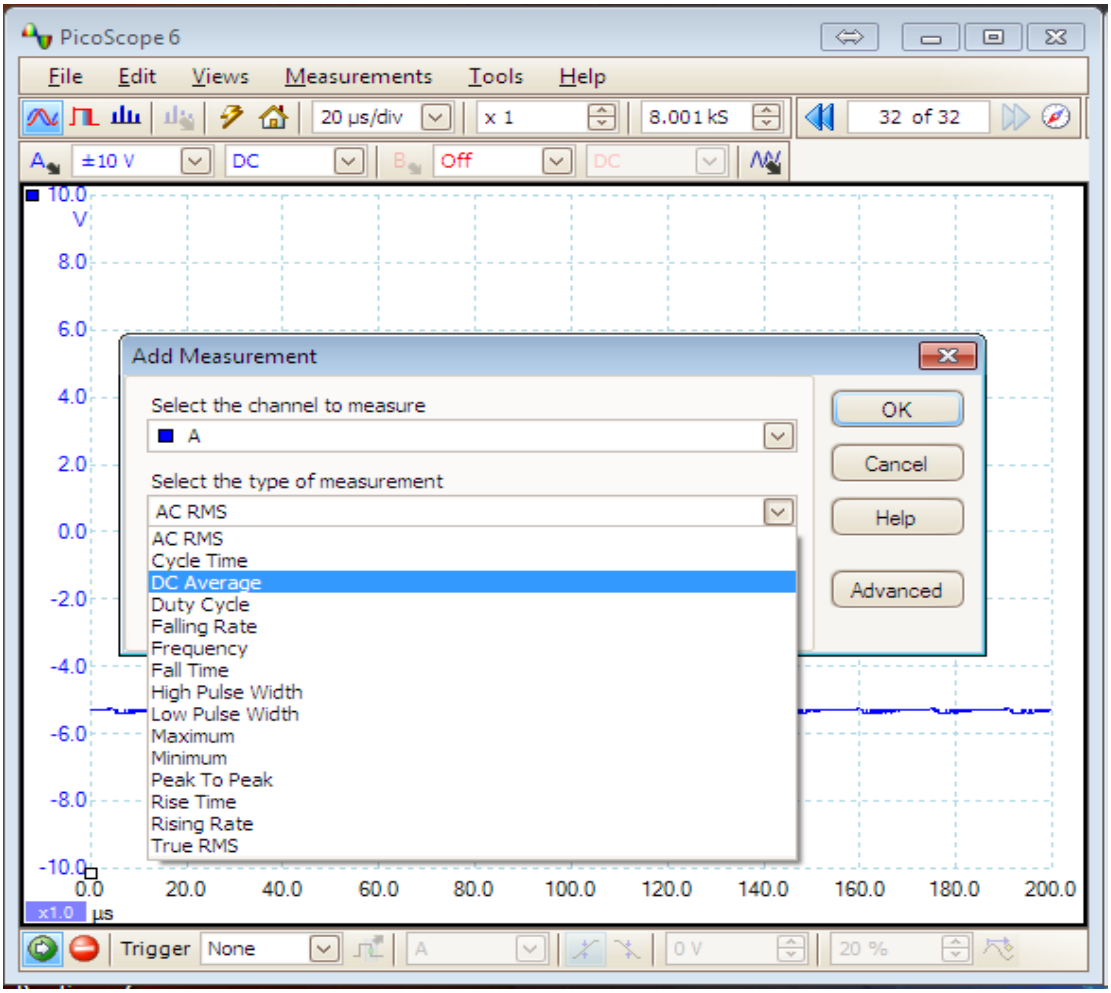


8. Click “Add Measurement”

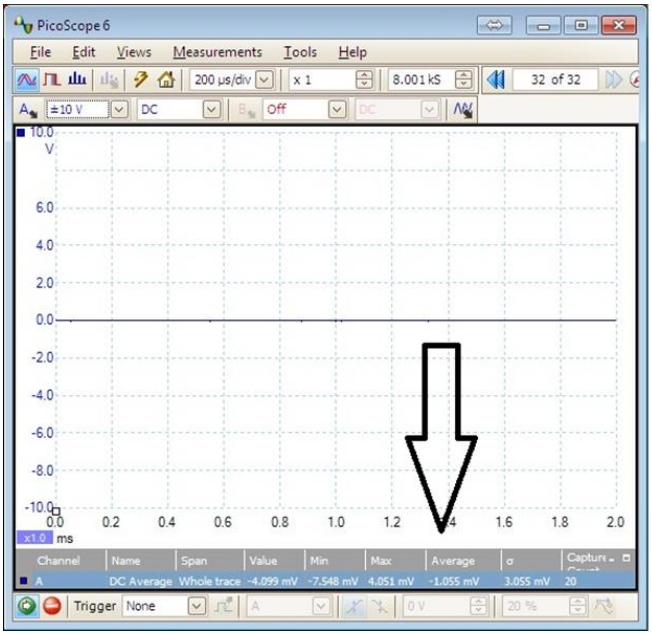
9. This will bring up the “Add Measure Window”



10. From the Type of measurement selection field select “DC Average”



11. Then Click OK. This will add the measuring function to the bottom of the PicoScope

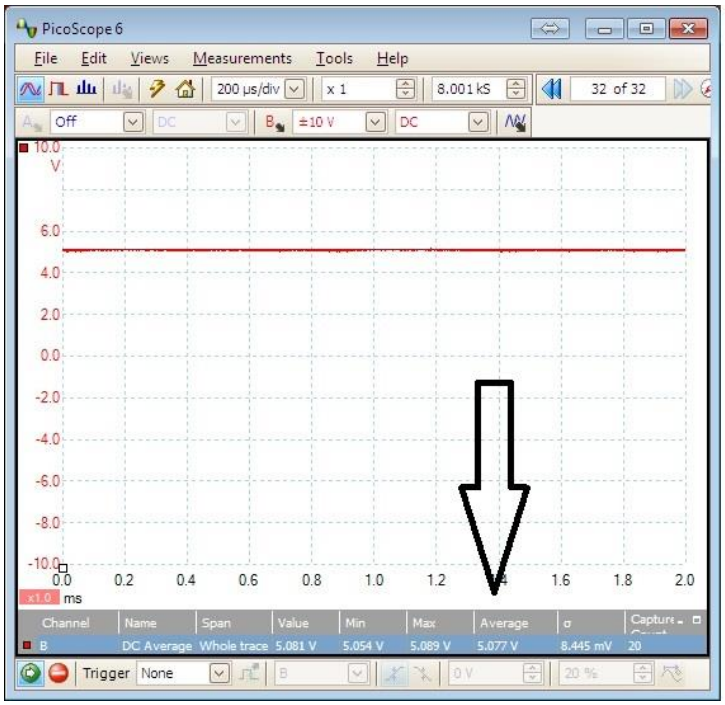


12. You can then read the average DC voltage level.

13. **Question:** What is the average DC voltage on the A wire when it is idling? State the answer in your assignment, rounded to 2 decimal places.

14. **Instruction**: Take a screenshot of this and submit it as evidence

15. Temporarily enable Channel B on the PicoCcope and repeat the DC measurement, but this time for the B wire



16. **Question :** What is the average DC voltage on the B wire when it is idling? State the answer in your assignment, rounded to 2 decimal places

17. **Instruction:** Take a screenshot of this and submit it as evidence

1.4 Transmitting a single character and then capturing it on the Picosocpe

The PicoScope must be configured as before, but this time we are using both Channel A and Channel B. We will first check the waveform on the A wire, then on the B wire, and finally combine the two traces.

NOTE: THIS IS A TRAP FOR THE UNWARY. The Start/Stop buttons on the PicoScope complement every time you click on them. So, if the PicoScope has stopped capturing, and you click twice on the Start button, you are still in 'stop' mode. However, if you observe the backgrounds of the Stop and Start buttons you will notice that they change from yellow to blue when 'true'. 

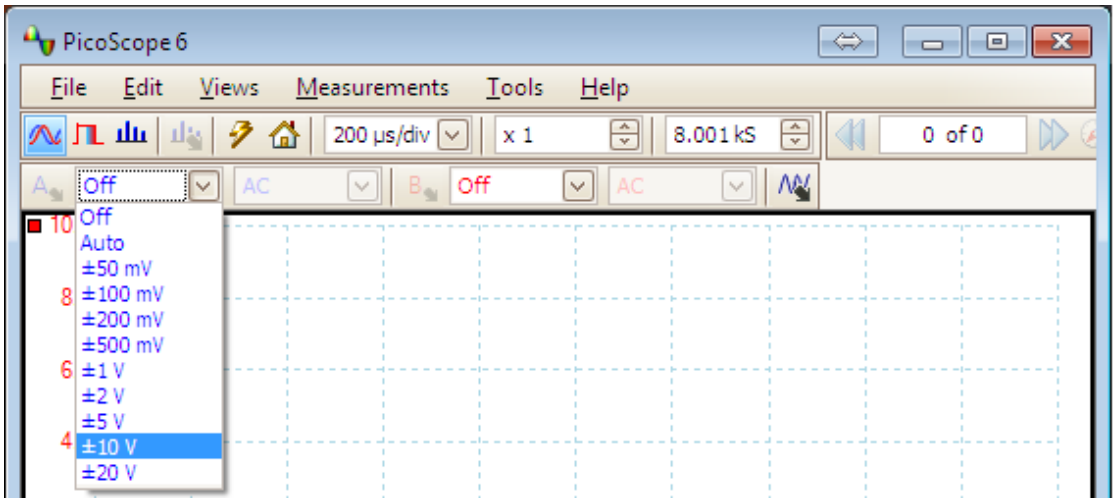
1.4.1 Capturing the signal on the 'A' wire

We will set up **Channel A** first. The PicoScope settings should be as follows:

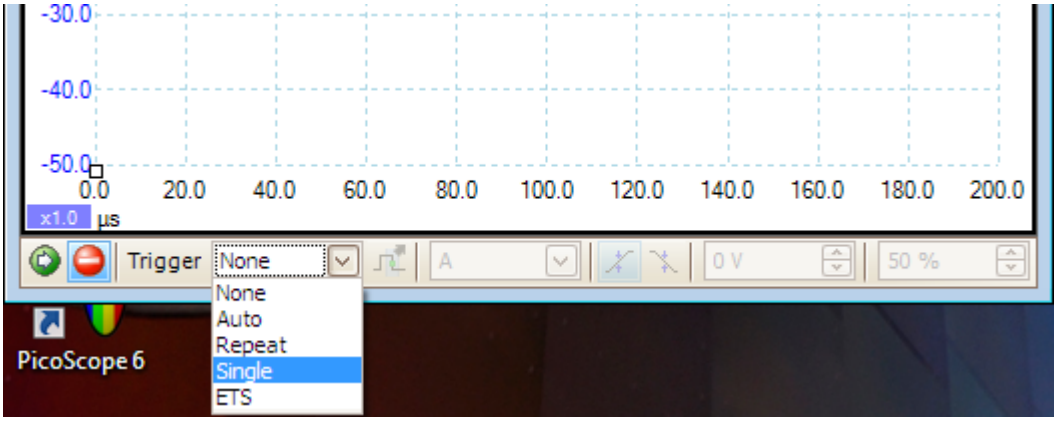
- Trigger: Single trigger, Channel A , rising edge
- Trigger level 2V. This is necessary because, unlike with RS-232, we are operating with positive voltages only. In theory the 'A' wire will be at '0' when no data is being sent. However, you have already ascertained that it is not. So, should the trigger level be set to 0V, the oscilloscope will not detect the low-to-high transition. If you cannot push the trigger level up high enough, check that the X-axis scale is on +/- 10V, and not on 'auto'.
- X-axis: 200 microseconds per division
- Magnification: X1
- Y-axis full scale: +/-10V
- Coupling: DC

Here is how we change to settings:

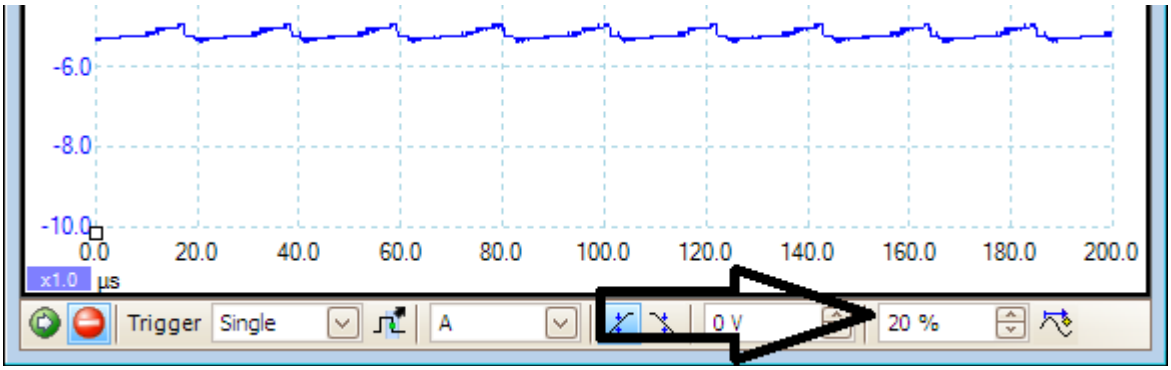
18. First we will set the Y axis. At the top of the PicoScope under the "Input Range" field for ChA select +/-10V



19. On the PicoScope under the trigger Field (at the bottom of the screen) select “Single”

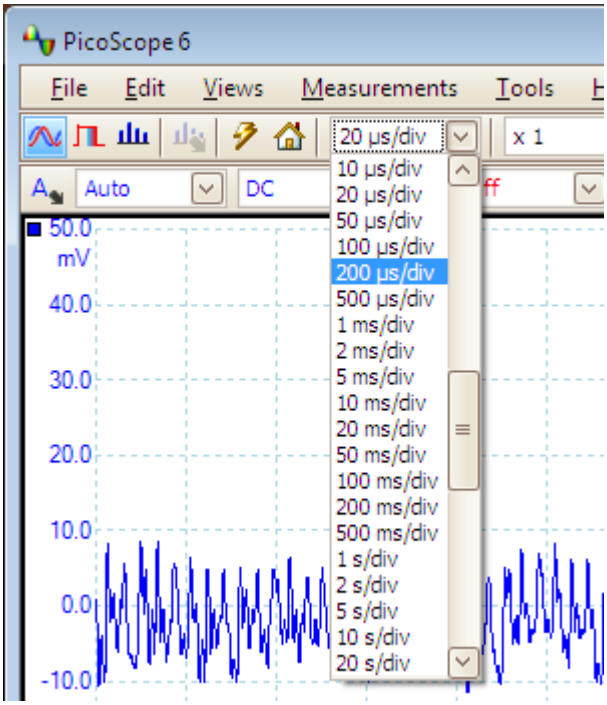


20. Set the pre-trigger to 20%

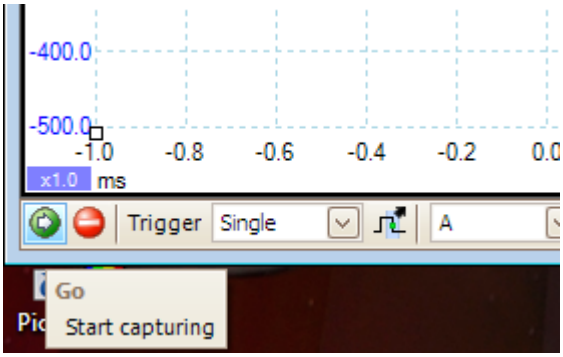


21. The other trigger settings (slope and level) are shown below

22. At the top of the PicoScope interface under the Time/div field select 200µS/div



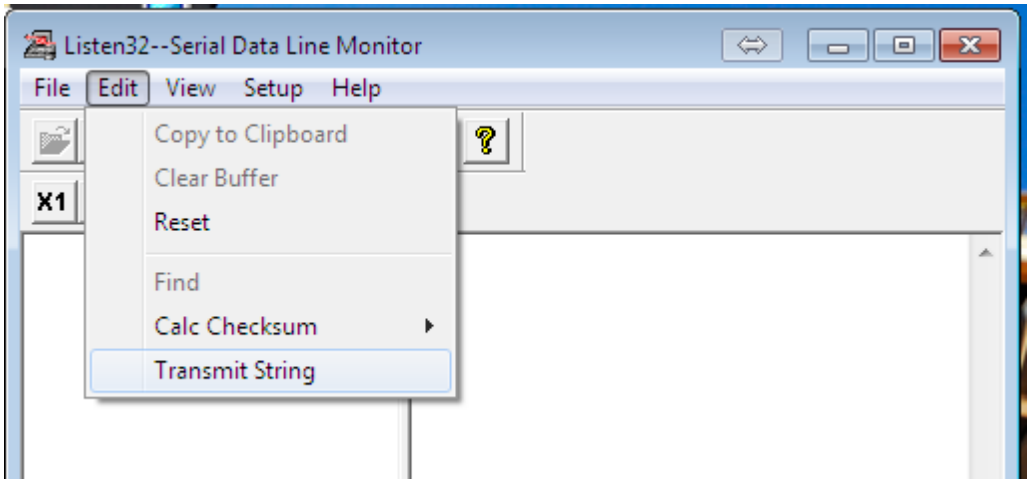
23. Now click the green “Start Capturing” icon (green Icon at the bottom left of the PicoScope window



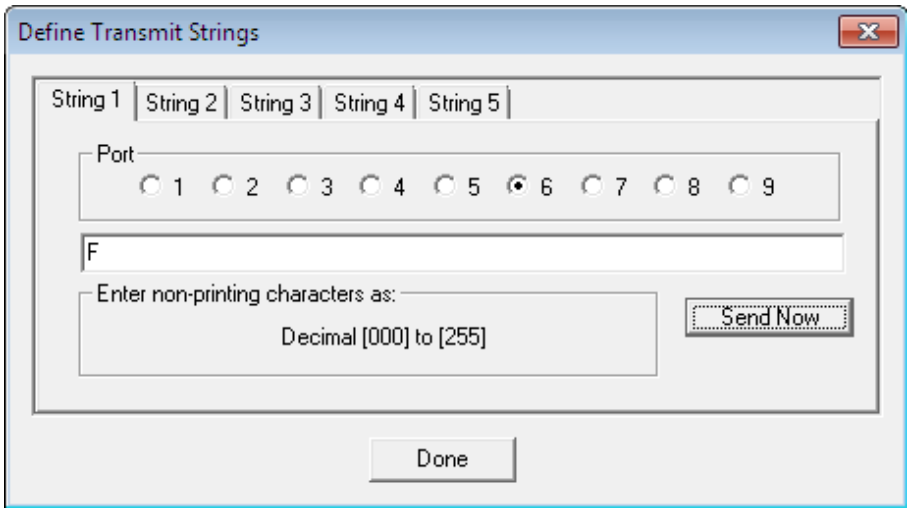
24. The PicoScope is now ready to capture the character we are about to transmit on Channel A, which is the A line or Data- of the RS-485

25. We already have Listen32 running (it will be in the toolbar at the bottom of the screen if the window is not already open)

26. On the Listen32 window click Edit->Transmit string. The ‘Define Transmit Strings’ window will appear.



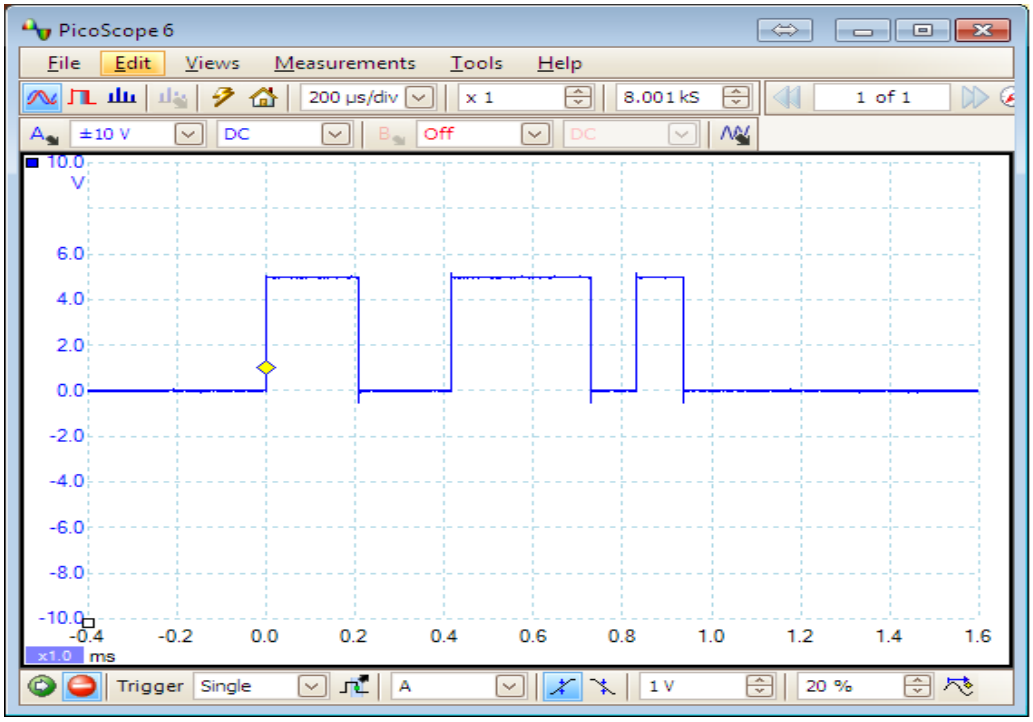
- The “Define Transmit Strings” window will now appear



27. Select String 1 and the appropriate port, in this case **port 6**, and enter upper case F

28. Click “Send Now” to transmit. The transmission should be captured on the PicoScope now.

29. Select the PicoScope again (remember it will be on the tool bar at the bottom of the screen if not on screen) And you should have a wave form similar to the one below:



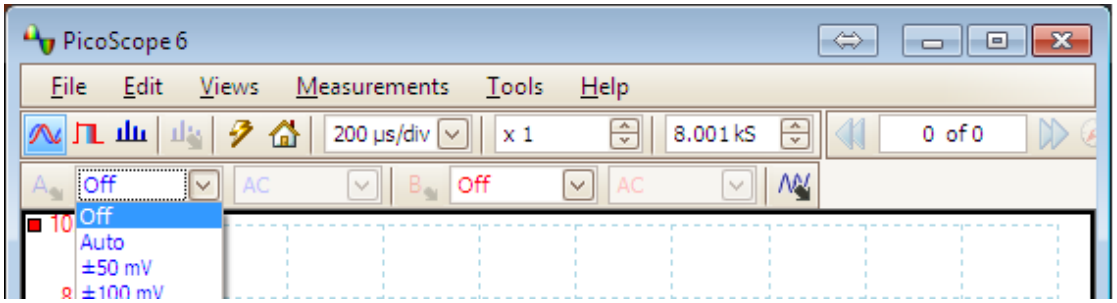
30. Note that the shape of the waveform is identical to that of RS-232, but instead of alternating between -5V and +5V it now only alternates between 0V and +5V

31. **Instruction**: Take a screenshot of this waveform and paste it into your assignment

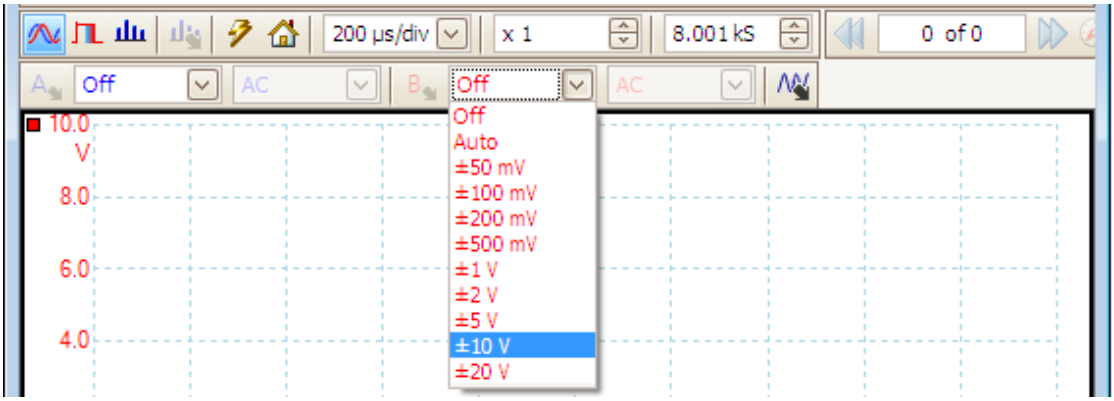
1.4.2 Capturing the signal on the 'B' wire

The steps are similar to those for the A wire, but there are a few minor differences.

32. First turn off Channel A in the Input range field



33. Now turn on Channel B in the input range field, and select +/-10V



34. Set the trigger for channel B. Keep in mind that the captured signal is expected to be a mirror image of what we saw on Channel A, so it will initially sit at approximately +5V, and then drop down to 0V at the beginning of the 'start' pulse. So setting the trigger will involves the same steps as for Channel A,

    BUT

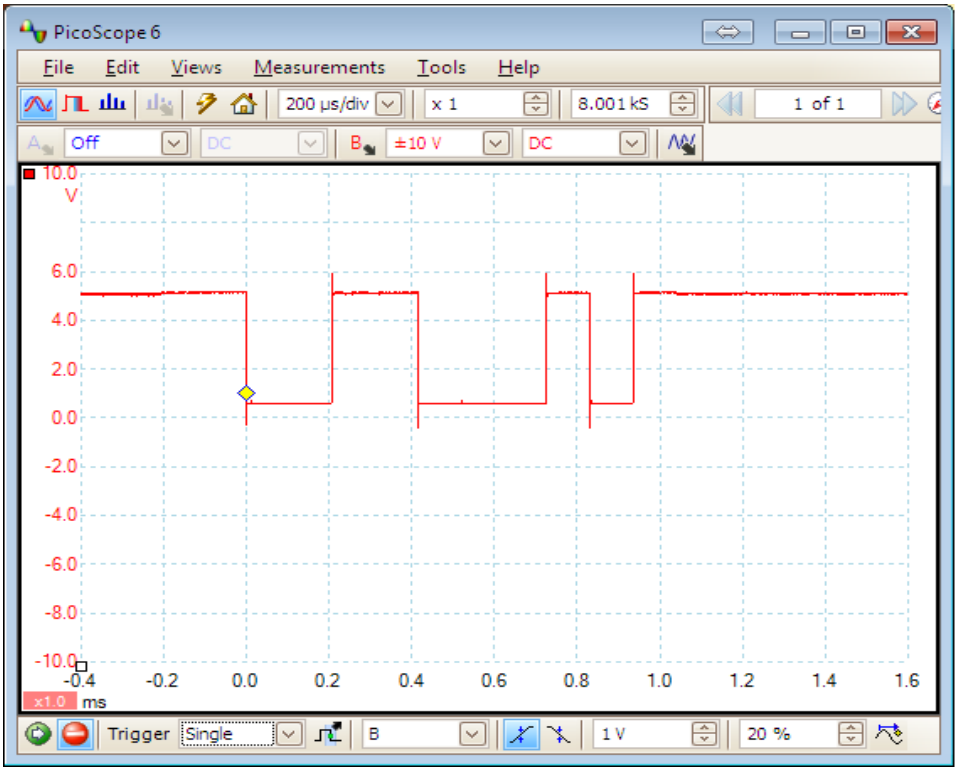
    We have to trigger on channel **B**, and on the **downward** edge. See below

35. Select Listen32 again (that will still be running in the background) and click "Send Now" on the "Define Transmit Strings" again as we did for Channel A

36. Bring up the PicoScope window again and you should have a waveform that looks similar to this:

37. Note that this signal is a mirror image of that on the A wire

38. Also note that despite the fact that, in theory, the B signal should go down to zero, it does not. However, what is important is the *difference* between A and B



39. **Instruction:** Take a screenshot of this waveform and paste it into your assignment

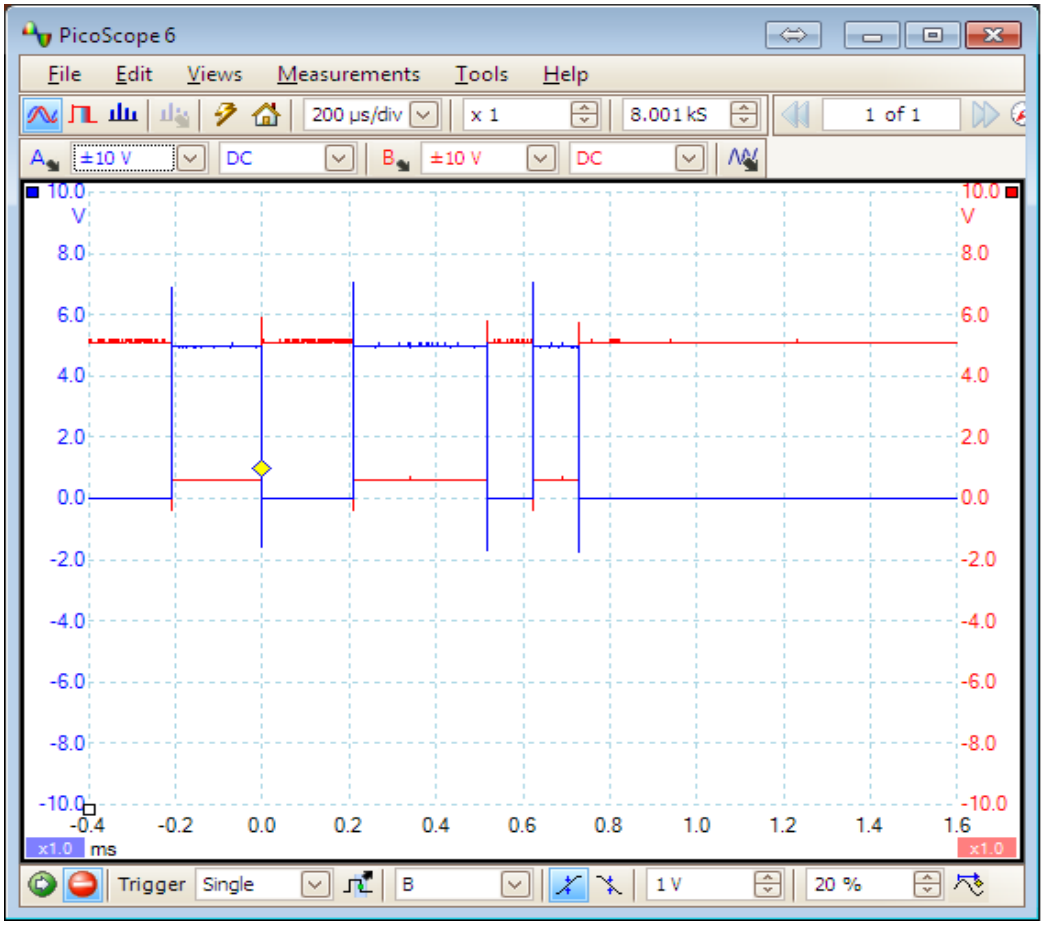
1.4.3 Capturing A and B simultaneously

40. We now must enable Channel A again in order to capture both A and B. So leave Channel B enabled and re-enable Channel A by again selecting +/-10V under the “Input Range” field

41. Reset the trigger again as above (Single) then click the green “Go” button

42. Again select Listen32 that will still be running in the background and click “Send Now” on the “Define Transmit Strings” again as we did earlier

43. Bring up the PicoScope window and you should have a waveform that looks similar to this:



- At first glance it might seem like a string of rectangles, but this is because the low-to-high transition of the one signal coincides with the high-to-low transition of the other.

- **Instruction:** Take a screenshot of this waveform and paste it into your assignment

End