ADVANTECH Enabling an Intelligent Planet

Advantech AE Technical Share Document

Date	2015/08/14	**SR#**	1-2187476555
Category	□FAQ ■SOP	**Related OS**	N/A
Abstract	ADAM-40XX, ADAM-41XX, How to use oscilloscope test the RS-485 signal		
Keyword	ADAM-40XX, ADAM-41XX, RS-485, Signal, Oscilloscope		
Related Product	ADAM-40XX, ADAM-41XX		

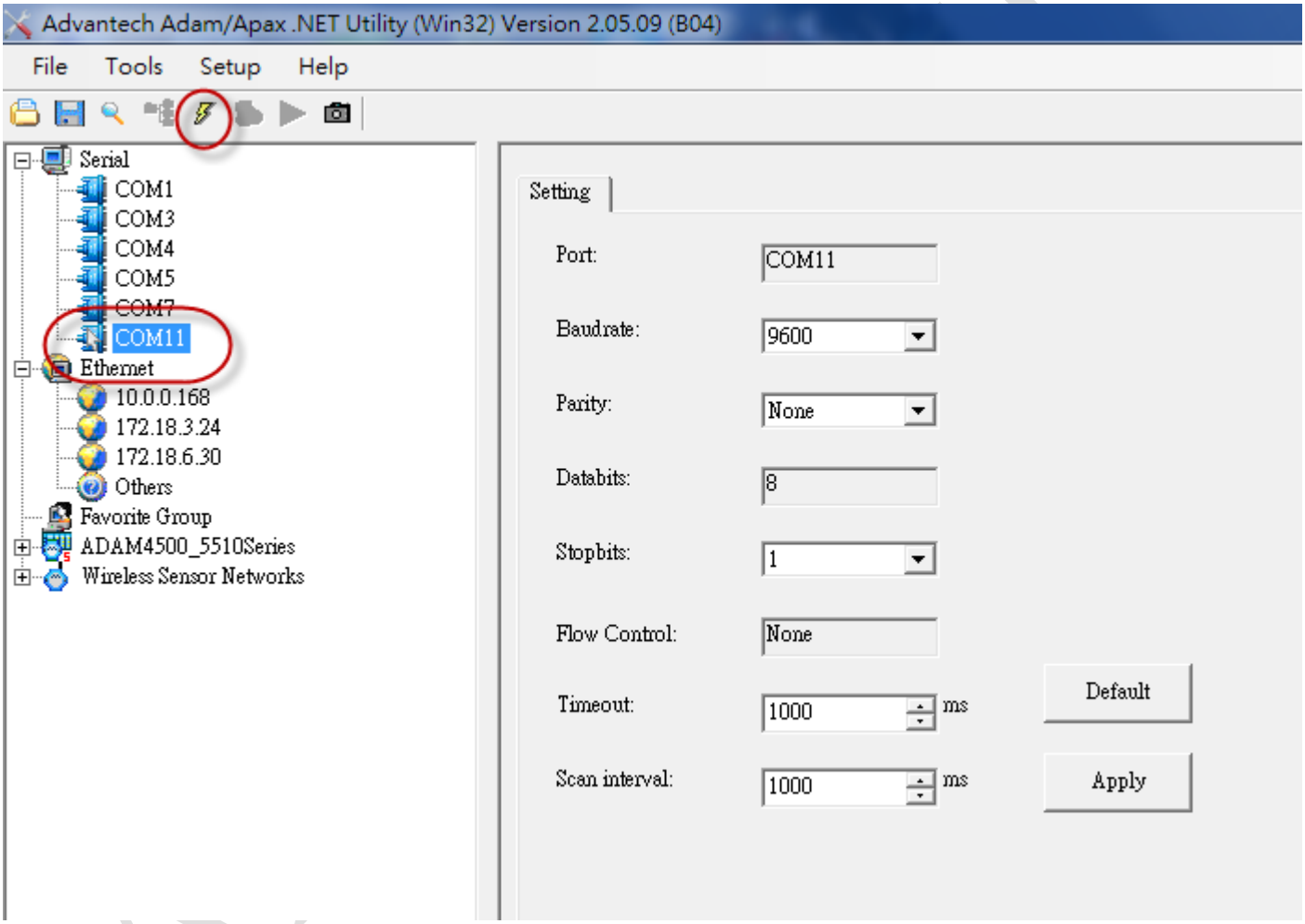
■ Problem Description:

This documentation shows the steps to test the RS-485 signal with ADAM module and oscilloscope.
Users can analysis the waveform of the RS-485 signal to understand whether the module works properly.

■ **Brief Solution - Step by Step:**

1. Use Adam/Apax .Net utility to send the ASCII command to the module.

In order to make the application scenario simpler, we would use ADAM-4018 as the end-device for the SOP, it will be set at the initial mode (slave ID=0)
First, click the COM port that you want to use in the utility, and click the flash icon to lunch the Adam Commander

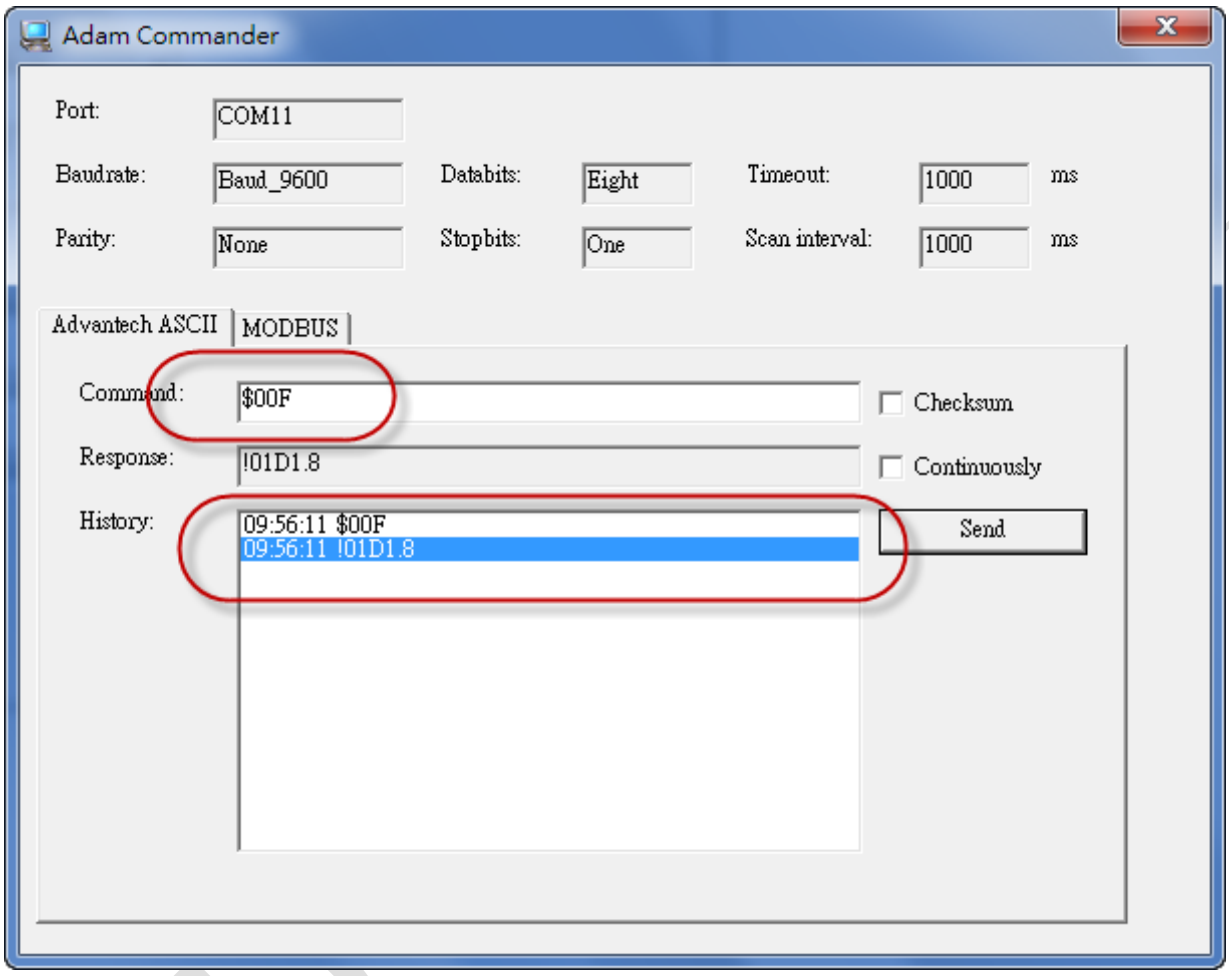


The user can enter the command here, then click “send” to send the ASCII command to the module.

Take the following picture as example, with the ASCII command “$00F”, ADAM will response its firmware version.

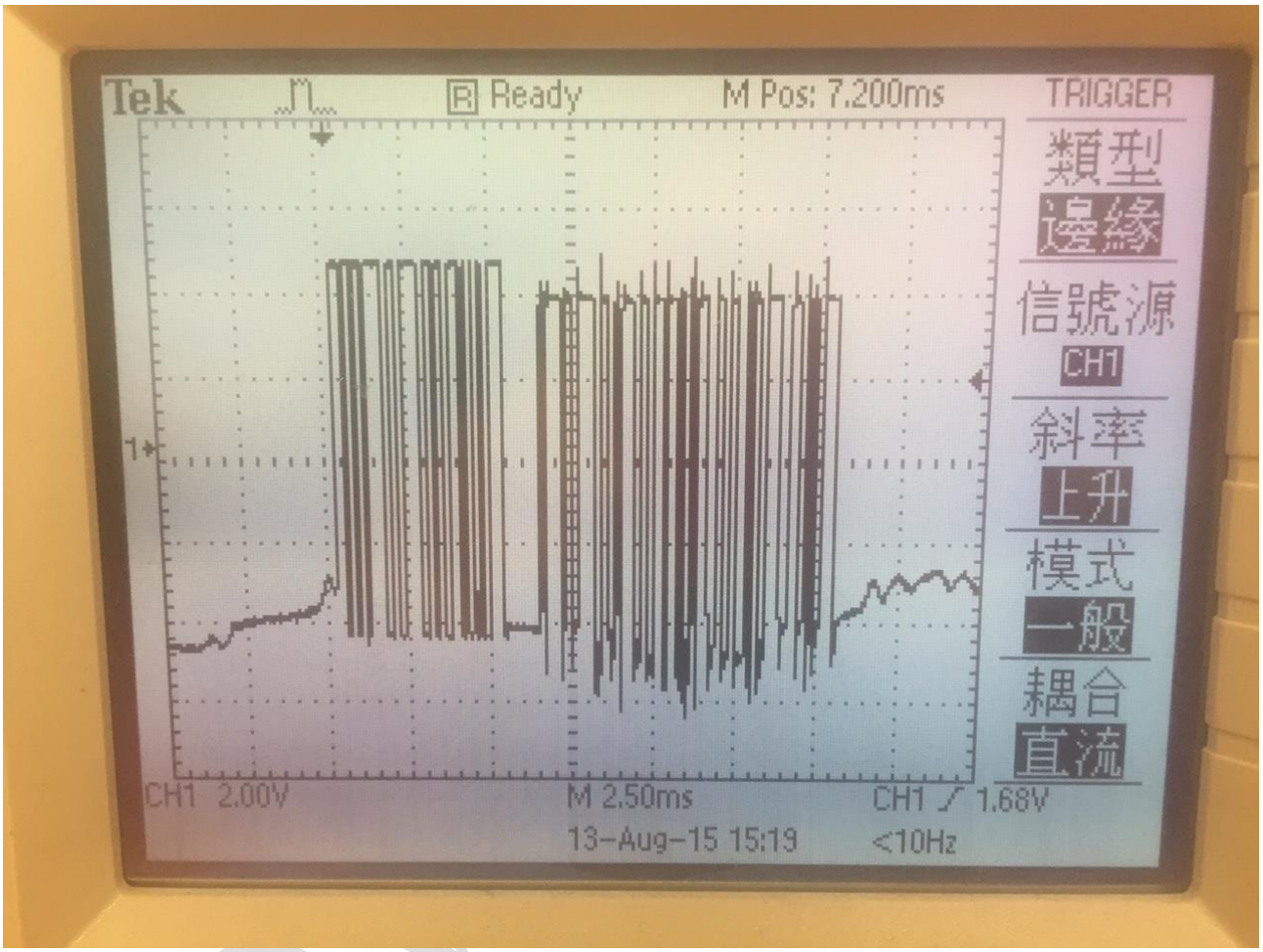
Query: $00F

Response:!01D1.8



2. Check the waveform detected by the oscilloscope

Here's the waveform screen-shot of the detected signal.
As illustrated in the picture, the first part of the signal is the query part and the second part is the signal that ADAM-4018 response to the host.



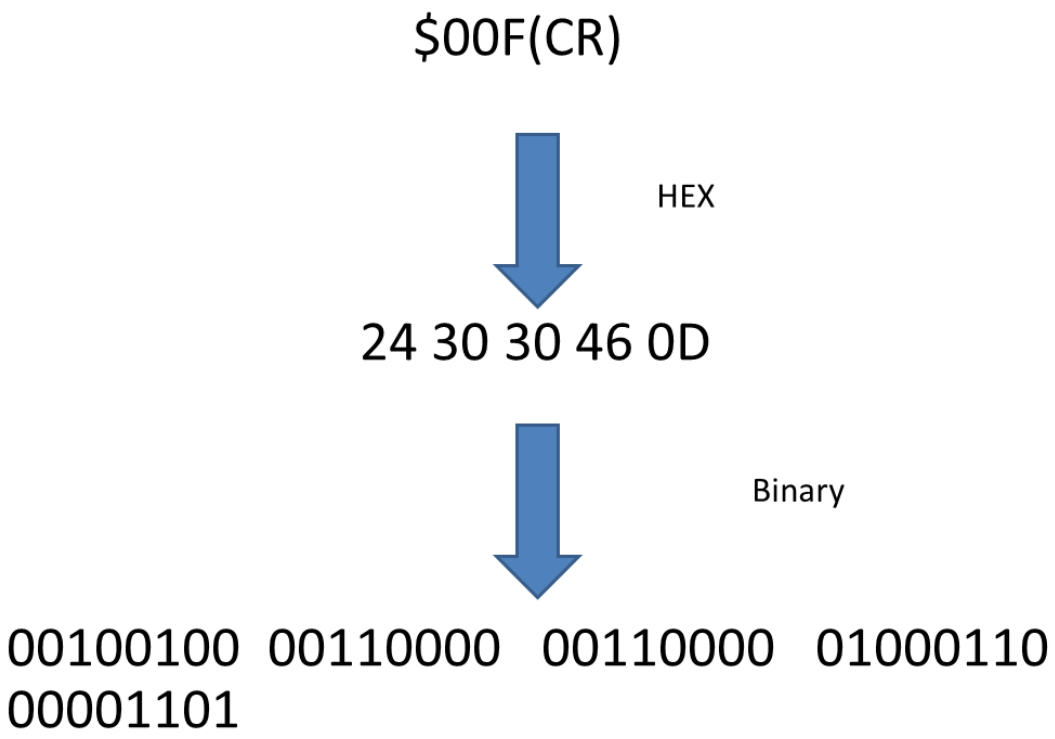
3. Analysis the waveform of the signal.

In normal situation, the screen-shot of the waveform should match the ASCII command that we send and get through the utility.
However, when there's the noise or compatibility issue between host and ADAM, which cause the ADAM not response correctly to the signal, we can check the signal by the oscilloscope.

According to the definition of RS-485, the voltage level for logic0 and logic 1 should be like the following picture.

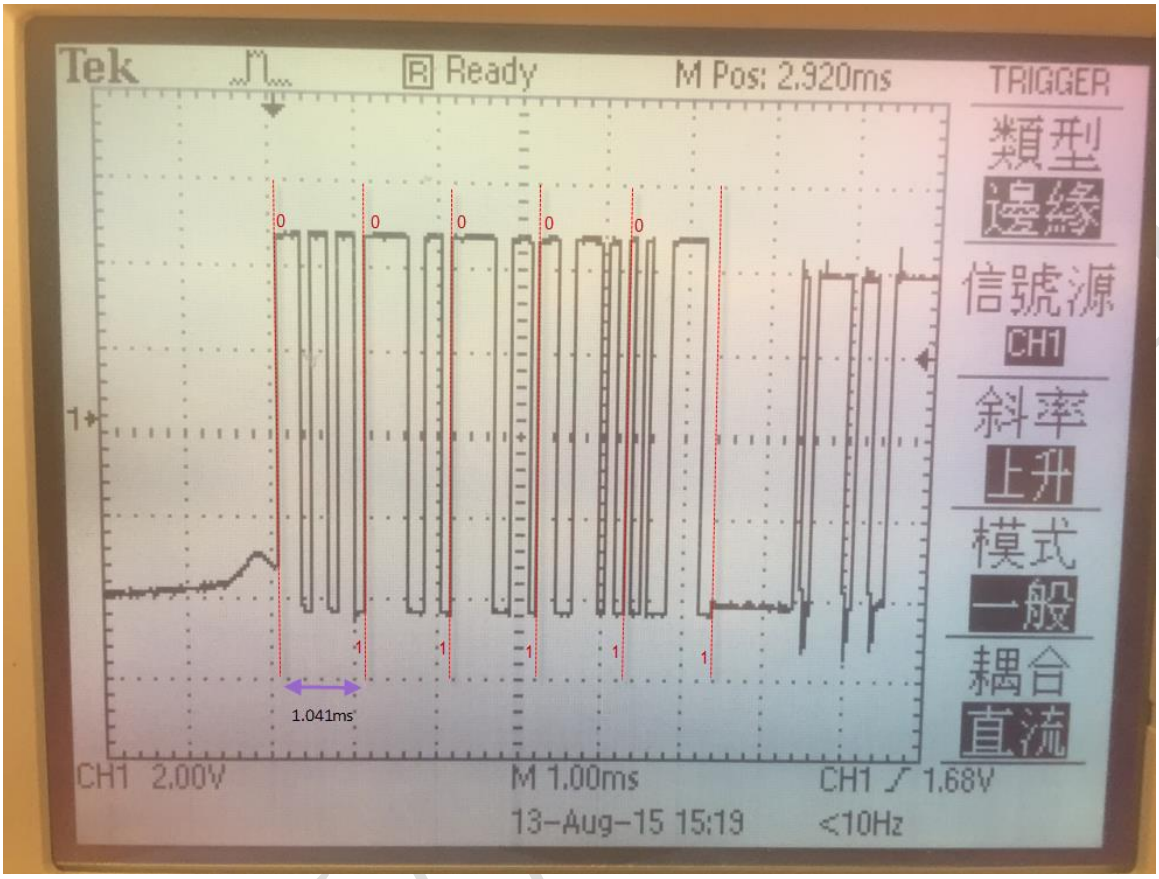
- Logic State:
  - Logic 0(Space): (Data+) – (Data–) > 200mV
  - Logic 1(Mark): (Data+) – (Data–) < –200mV

And we can change the data format of the ASCII command into the binary format.



For our test, the baudrate is 9600 bps, so every bits will take 1/9600= 0.1041ms

With the start bit logic 0 and stop bit logic1 and the information of baudrate, we could know that each byte will takes about 1.041ms, so that we could separate the signal into different parts, and each part means a byte, or more clearly, a ASCII code.



We can take a step further and convert the signal back to the original ASCII command that we send by the utility