
MSB Protocol Scanner

Michael Hungershausen, www.iftools.com

November, 1st 2019

In the following document we take a closer look to a great feature of our Analyzers. It is called the protocol scanner.

We explain when it is useful, how it works, and what the reason can be if it does not work.

What does the scanner do?

The scanner examines the recorded bits on the bus in real time and tries to find out the following parameters of the low level protocol:

Bit rate, number of data bits and if the last bit of a bit frame could be an odd or even parity bit.

This is an important tool for inspecting data communication when the parameters are not known.

But: the scanner can only give a proposal for the protocol parameters, the user is responsible to use these parameters directly or modified. Therefore the results are not automatically used as the working parameters for the analyzer, before they must be acknowledged by the user.

How does it work?

At first the bit rate is determined. This is done by measuring the time between all occurring bit edges and generating mean values in groups. The lowest mean time is assumed to be the time for one single bit which results in the bit rate. This bit rate is the basis for the following examination.

The next step is the determination of the length of the data frame, the total number of bits. For that the internal UART varies the number of data bits from 5 to 9 waiting for framing errors. Some bytes have to be recorded with different data content, a real 'living' bit stream is necessary. When for a while no framing error occurs the number of data bits are assumed as correct.

The task is to find out if a parity bit is used instead of the last data bit, e.g. if a protocol is 8N1 or 7E1, meaning 8 real data bits or 7 data bits plus even parity.

Another number of data frames are examined and checked if the last data bit is **always** the parity of the first bits, even or odd. The more frames are recorded the more safe is the result.

The scanner always assumes one stop bit, more stop bits are simply additional gaps between the data frames and of no importance.

The measuring intervals are 250ms. If within three seconds not a sufficient number of data was available for evaluation a display appears, indicating that not enough data was provided.

Why does it not work?

Based on the above explanations the following issues can lead to wrong or no data:

The data stream must contain enough data, at least 50 bytes per second.

The bus must work correctly without errors and external influences. The line quality must be sufficient. Check that the wires are not exchanged in a RS485 bus, check the signal window.

Check if the measured bit rate can be roughly true. Use the signal window of the analyzer to verify the measurement. Without correct measured rate (within some percent) the data and parity results will be wrong. An unreasonable measured rate can be a strong sign for a bad bus signal quality, you should follow this hint first, perhaps this is the key to solve your searched bus faults.

The higher the bit rate the higher the precision of the measurement. Depending on the bus and transmitter quality the measured rate will be higher than the real one.

In general take the measured bit rate as a hint. If you do not await an unusual bit rate try to set the bit rate from the standard list near to the measured one for the final settings. E.g.: Measured 19490, the chance is high that the real bit rate is 19200 (+1.5%).

Standard rates are: 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 115200, 230400, 460800 and 921600 Bd.