

Bitscope Logic tips

Example: capturing a signal on an UART asynchronous serial connection

The signal is connected to channel A.

Trigger
Rise / Fall / High / Low / None

The screenshot shows the BitScope Logic 1.2 software interface. At the top, there is a 'Trigger' dropdown menu set to 'Rise / Fall / High / Low / None'. Below this is a list of channels: Ch A (yellow), Ch B (green), Logic 7 (magenta), Logic 6 (cyan), Logic 5 (teal), Logic 4 (yellow), Logic 3 (orange), Logic 2 (red), Logic 1 (brown), and Logic 0 (white). Each channel has a trigger icon and a 'T' in a box. The waveform display shows a signal on Ch A (yellow) and Logic 7 (magenta). The capture time is set to 20.00 mS, the sample rate is 303.0 ks/S, and the pre-trigger is set to PRE 25 %. The RUN button is highlighted with a red arrow.

INSPECT ? BS000501

0.00 S 2.00 mS 4.00 mS 6.00 mS 8.00 m

Ch A T
Ch B T
Logic 7 T
Logic 6 T
Logic 5 T
Logic 4 T
Logic 3 T
Logic 2 T
Logic 1 T
Logic 0 T

0.000 S 0.000 S 0.000 S 0.000 Hz

20.00 mS 303.0 ks/S PRE 25 % BitScope Logic RUN

Capture time Sample rate Pre-Trigger Capture control: RUN / ARMED / CAPTURE / ACQUIRE / DECODE / IDLE

To capture the data:

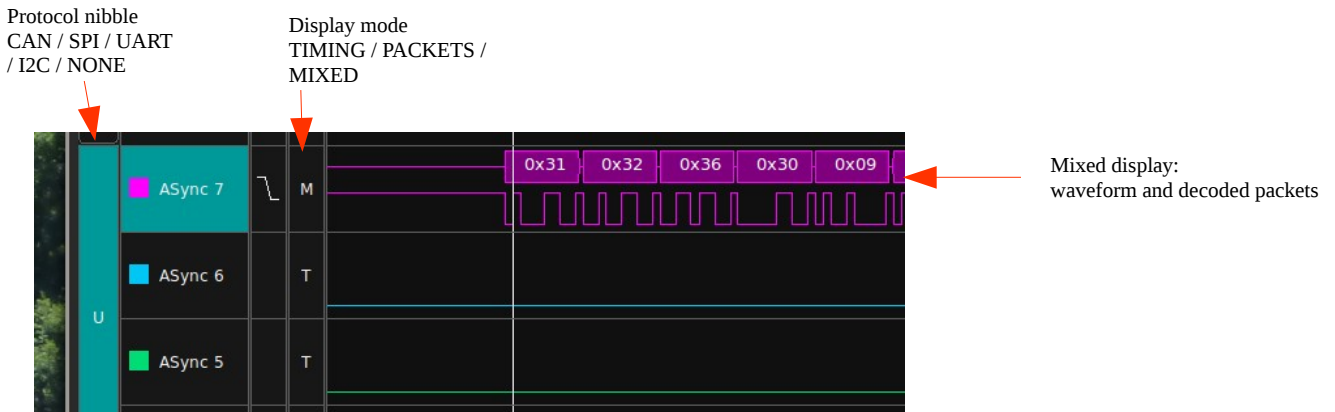
- Connect to the scope
- Set capture time. This automatically sets the sample rate. The sample rate should be at least 4 times the baudrate of the signal.
- Set the trigger condition (Right click in the trigger field)
- Klick the RUN button. The button text turns to ARMED. Now we are waiting for the trigger condition.

When the scope triggers, the waveform is displayed and the button text turns to RUN again.

The signal is displayed as analog signal on channel A, and automatically via the internal comparator, on Logic channel 7.

Decode the captured data

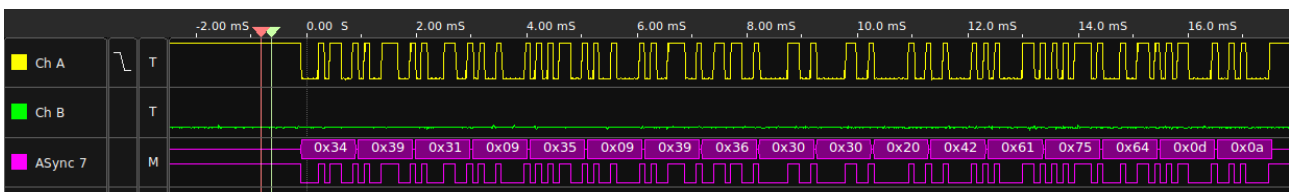
Bitscope Logic has 2 decoders, one for the high and one for the high (7..4) and one for the low (3..0) data nibble. How many signals can be decoded depends on how many signals are needed by the protocol. In our UART example we could decode 4 independant signals per nibble, but we need just one.



- Right click in the protocol nibble field and select protocol UART
The names 4 high nibble fields turn to ASync7...ASync4
- Set display mode to MIXED, to see waveform and decoded data
- Set up baudrate etc.: left click the protocol nibble field

If the baudrate is unknown, we can measure the length of one bit, e.g. the start bit, with the cursors

Now the signal and the data are displayed:



We can experiment with the number of stop bits, or the parity if these settings are unknown.

Unfortunately the polarity is not changeable, and fixed to „Inverted“. This means that the signal coming from TxD of a microcontroller can be analysed. For such a signal, Low means 0 and High means 1, so I would describe it as „non-inverted“. I suppose the idea of the bitscope authors was to say „Inverted“ as compared to an RS232 interface.

In the current version there is no possibility to log the data to a file or display the ASCII values.