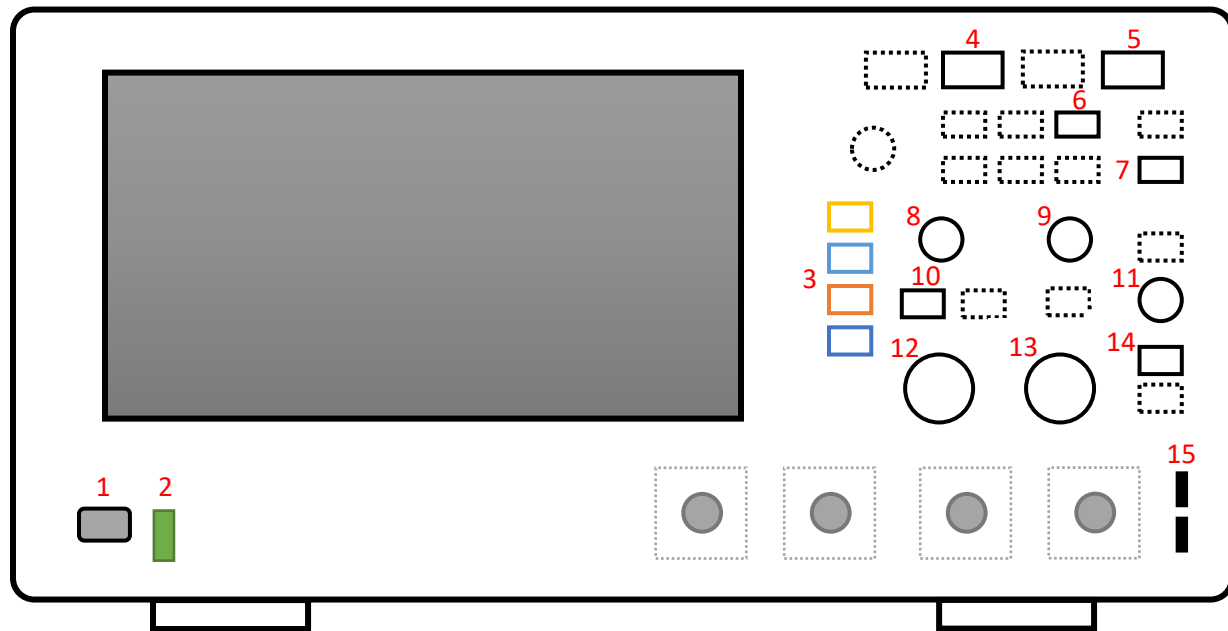


# Protocol Decoding

How to capture and decode communication protocols with the Rigol DS1054Z.

## The Instrument

An overview of the front panel of the Rigol DS1054Z digital oscilloscope can be seen in the following figure:



The control elements which are used to capture and decode protocols are the following ones:

|                   |                  |                   |                  |
|-------------------|------------------|-------------------|------------------|
| 1: Power Key      | 5: SINGLE        | 9: Trigger Pos.   | 13: Time Div.    |
| 2: USB Interface  | 6: Storage       | 10: MATH          | 14: MENU         |
| 3: Channel Select | 7: Print         | 11: Trigger Level | 15: Compensation |
| 4: AUTO           | 8: Vertical Pos. | 12: Volt Div.     |                  |

There is a Quick Guide<sup>1</sup> and a User's Guide<sup>2</sup> for the instrument which describe the functions of the oscilloscope in detail. Here a short introduction on how to capture communication signals and how to decode them shall be given.

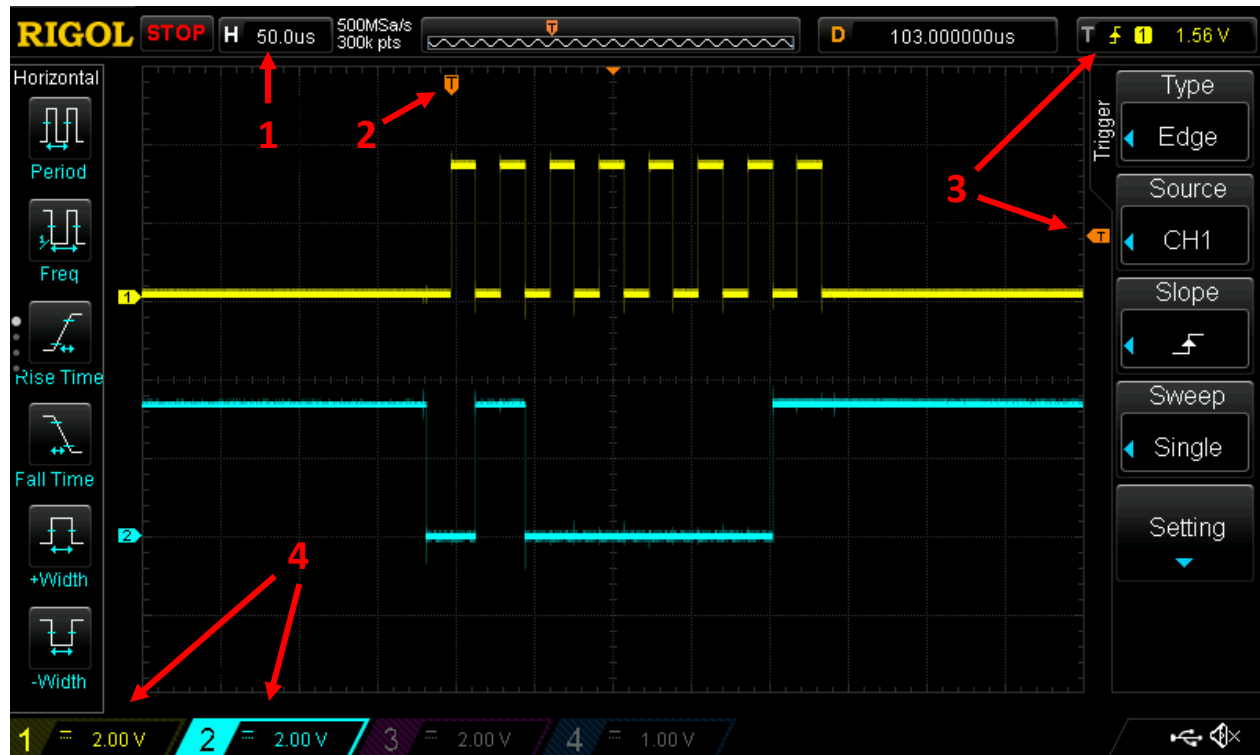
After power-on, the oscilloscope should be reset to its default configuration and the probes should be compensated. Section *Function Inspection* on page 6 of the Quick Guide explains how this is done. Please follow the steps described there.

The signals of a communication protocol can either be captured or decoded. If the signals are captured the progression of the curve is shown. No information on the meaning of the signals is acquired. When a communication protocol is decoded the signals will be analyzed and the data transmitted will be shown. This requires, however, some knowledge on the protocol and an extensive configuration.

<sup>1</sup> [http://beyondmeasure.rigoltech.com/acton/attachment/1579/f-0506/1/-/-/-/MSO1000Z%26DS1000Z\\_QuickGuide.pdf](http://beyondmeasure.rigoltech.com/acton/attachment/1579/f-0506/1/-/-/-/MSO1000Z%26DS1000Z_QuickGuide.pdf)

<sup>2</sup> [http://beyondmeasure.rigoltech.com/acton/attachment/1579/f-050a/1/-/-/-/MSO1000Z%26DS1000Z\\_UserGuide.pdf](http://beyondmeasure.rigoltech.com/acton/attachment/1579/f-050a/1/-/-/-/MSO1000Z%26DS1000Z_UserGuide.pdf)

In order to capture or to decode a communication protocol the signal lines have to be connected with the analog input. It depends on the type of protocol how many lines have to be probed. An appropriate voltage and time base have to be selected to get good signals. This is done by knob 12 and 13. It is also important to select the correct trigger type, level and position. The mode is selected by button 14. The trigger level and its position are adjusted by knob 9 and 11. The actual settings of these parameters are shown on the screen. There are four important regions of interest:



1. Time Base: Makes the signal narrower or wider. Here 50.0  $\mu$ s per division.
2. Trigger Start: Places the captured signal vertically on the screen.
3. Trigger Level: Determines at which voltage level the signal is captured. Here 1.56 V.
4. Voltage Base: Makes the signal higher or lower. Here 2.00 V per division.

The knobs which control these values are:

- Time Base: Knob 13
- Trigger Start: Knob 9
- Trigger Level: Knob 11
- Voltage Base: Knob 12

A screenshot of the curves can be made by pressing button 7. The screenshot will be saved on the USB flash drive connected with slot 2. Please not that only FAT32 is supported. So newer sticks will have to be reformatted.

## I<sup>2</sup>C

The I<sup>2</sup>C protocol uses two line for the communication:

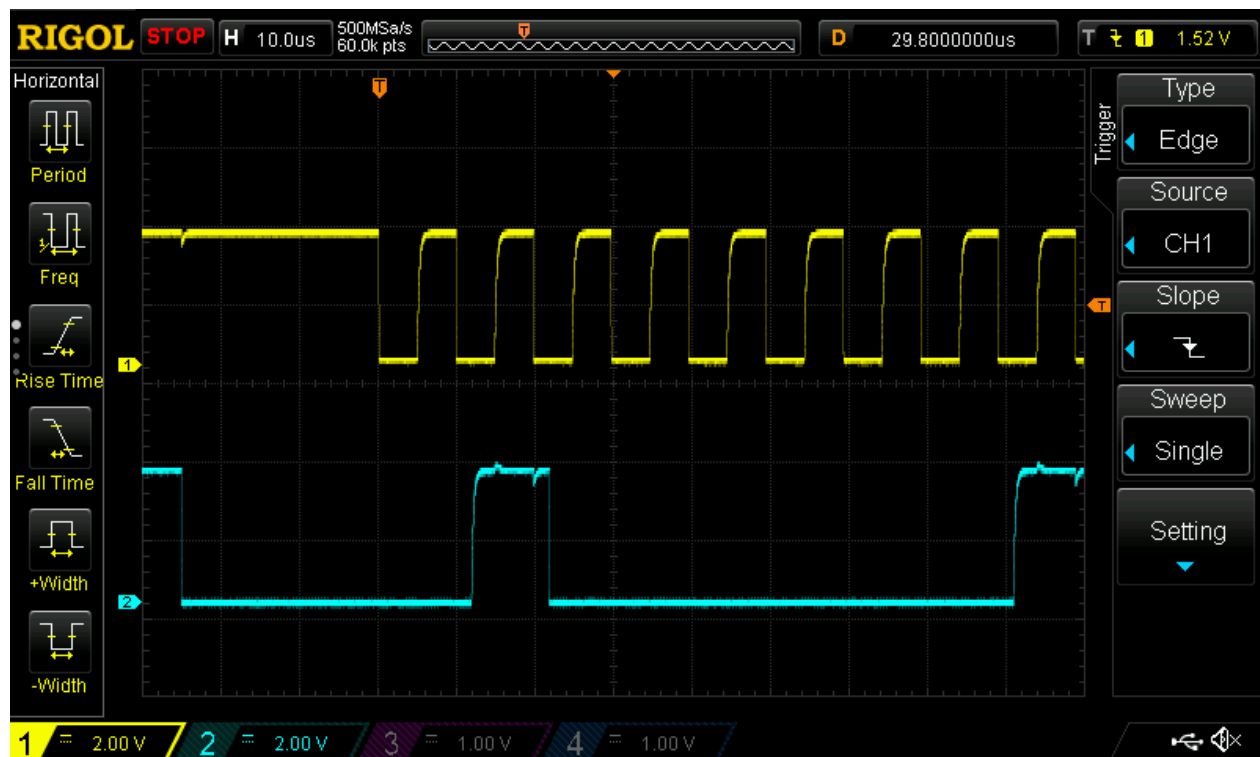
- SDA – Serial Data Line
- SCL – Serial Clock line

It is a bus protocol which uses an address token to select the addressee and communication is controlled by start and stop conditions. Please see Module 5.3.4<sup>3</sup> for more information or consult the Internet.

### Capturing

In order to capture the communication signals SCL can be used as trigger. The following steps describe the necessary settings:

- Use knob 12 to set 2 V per division.
- Use knob 13 to set 10.0  $\mu$ s per division.
- Press button 14 to place an edge trigger on the falling slope.
- Use knob 11 to set the trigger level to 1.5 V.
- Press button 5 to capture the signal.
- Use knob 8 to move the signals vertically. Use button 3 to select the signal to be moved.
- Use knob 9 to move the signals horizontally.
- Press button 5 again to capture the adjusted signal.



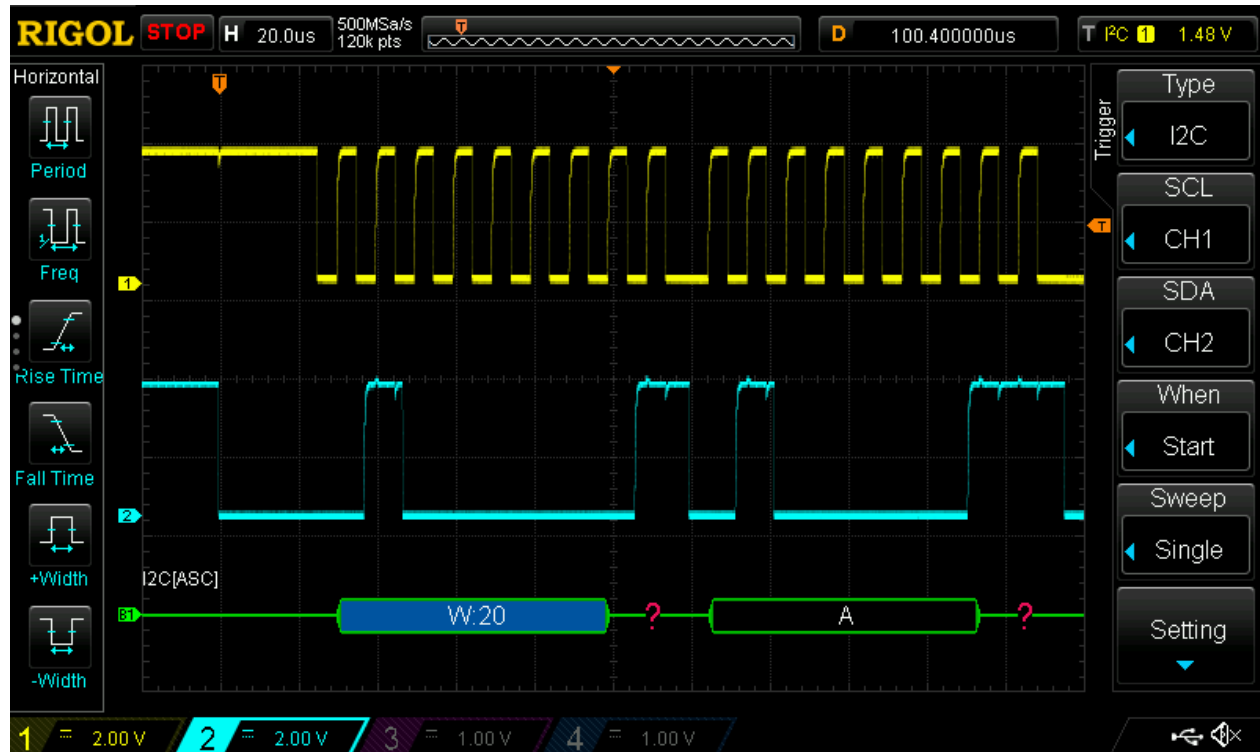
One can see that a start condition is issued before the clock signal starts the transfer. Data may change if the clock is low and it is read if clock is high.

<sup>3</sup> [http://www.neff.site/courses/nwes/nwes\\_5/2.4\\_i2c.pdf](http://www.neff.site/courses/nwes/nwes_5/2.4_i2c.pdf)

## Decoding

To decode the protocol the trigger as well the decoder has to be configured. The trigger is configured as follows:

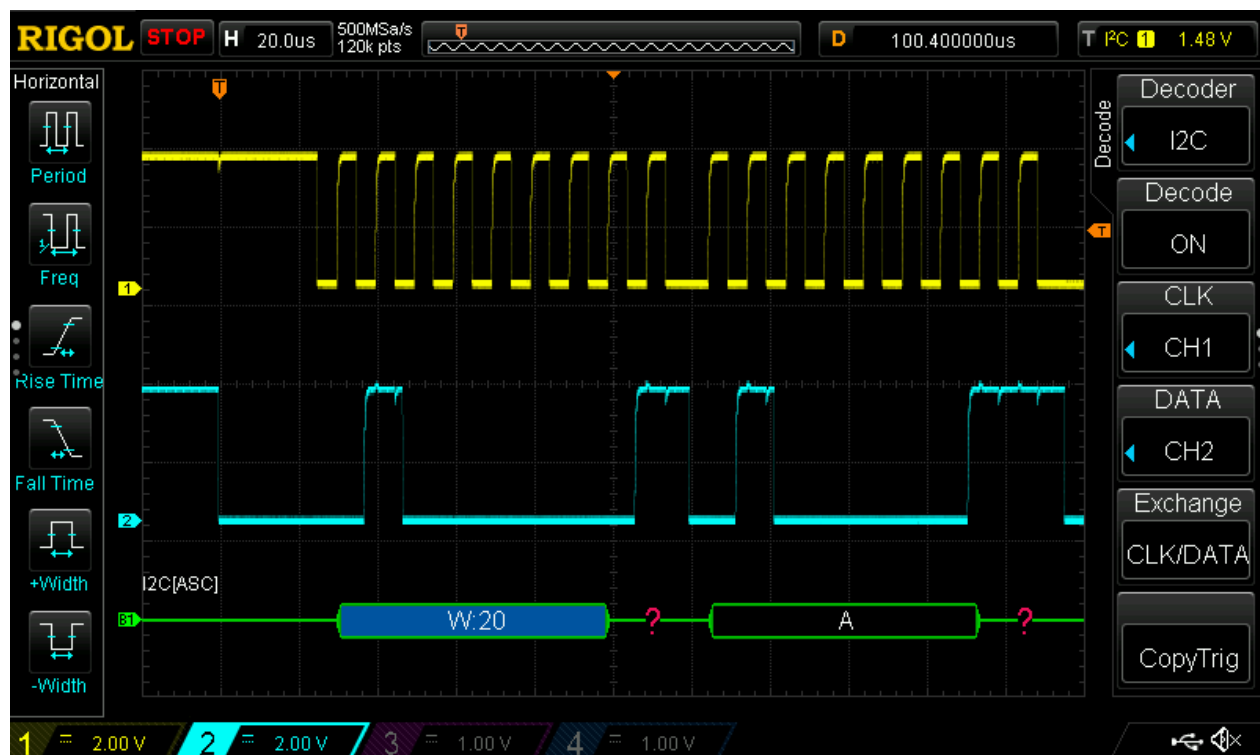
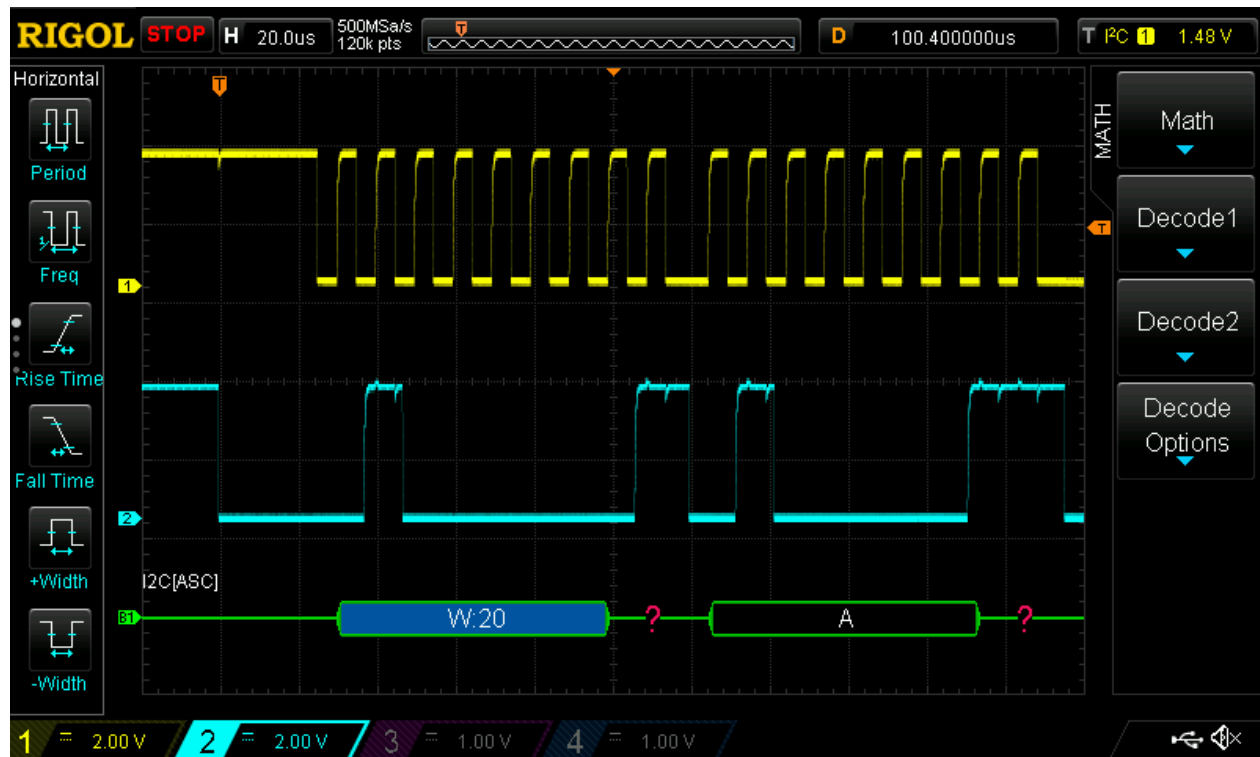
- Use button 14 to select trigger type I2C.
- Select one which channel SCL and SDA is connected.
- Use knob 11 to set the trigger level to 1.5 V.



Then the decoder has to be configured:

- Use knob 12 to set 2 V per division.
- Use knob 13 to set 20.0  $\mu$ s per division.
- Press button 10 and select Decode1.
- Select Decoder I2C.
- Switch Decode on.
- Select the clock and data line.
- Press button 5 to capture the signal.
- Use knob 8 to move the signals vertically. Use button 3 to select the signal to be moved.
- Use knob 9 to move the signals horizontally.
- Press button 5 again to capture the adjusted signal.

The signal will be decoded, now. After the start condition which is not indicated the address token follows. There a write to address 20<sub>hex</sub> is performed. Then the acknowledgment of the slave (represented by a question mark) and the first data byte (letter A) follow.



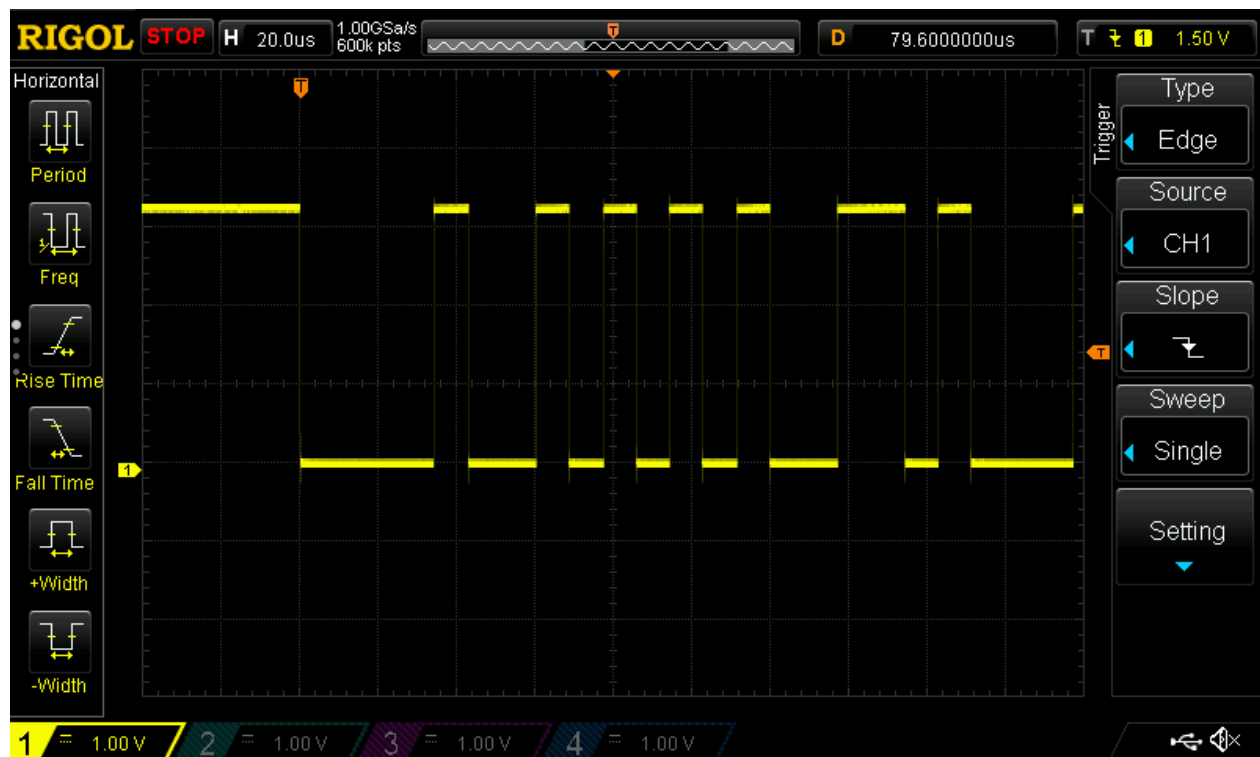
## RS-232

RS-232 is an asynchronous point-to-point protocol that uses a transmit and a receive line but no clock line. This is why the transmission speed has to be specified. If there is nothing to transmit an idle line state is indicated by a high signal. Therefore, a trigger on the negative edge is required. Genuine RS-232 uses negative logic and high voltage levels (0: +12 V, 1: -12 V) but in this case standard logic levels (0: 0.0 V, 1: 3.3 V). Please see Module 5.3.4<sup>4</sup> for more information or consult the Internet.

### Capturing

With RS-232 only the transmit line (TxD) will be captured. This line will therefore be used as trigger. The necessary settings are:

- Use knob 12 to set 1 V per division.
- Use knob 13 to set 20.0  $\mu$ s per division.
- Use button 14 to select trigger type Edge falling Slope.
- Use knob 11 to set the trigger level to 1.5 V.
- Press button 5 to capture the signal.
- Use knob 8 to move the signals vertically. Use button 3 to select the signal to be moved.
- Use knob 9 to move the signals horizontally.
- Press button 5 again to capture the adjusted signal.

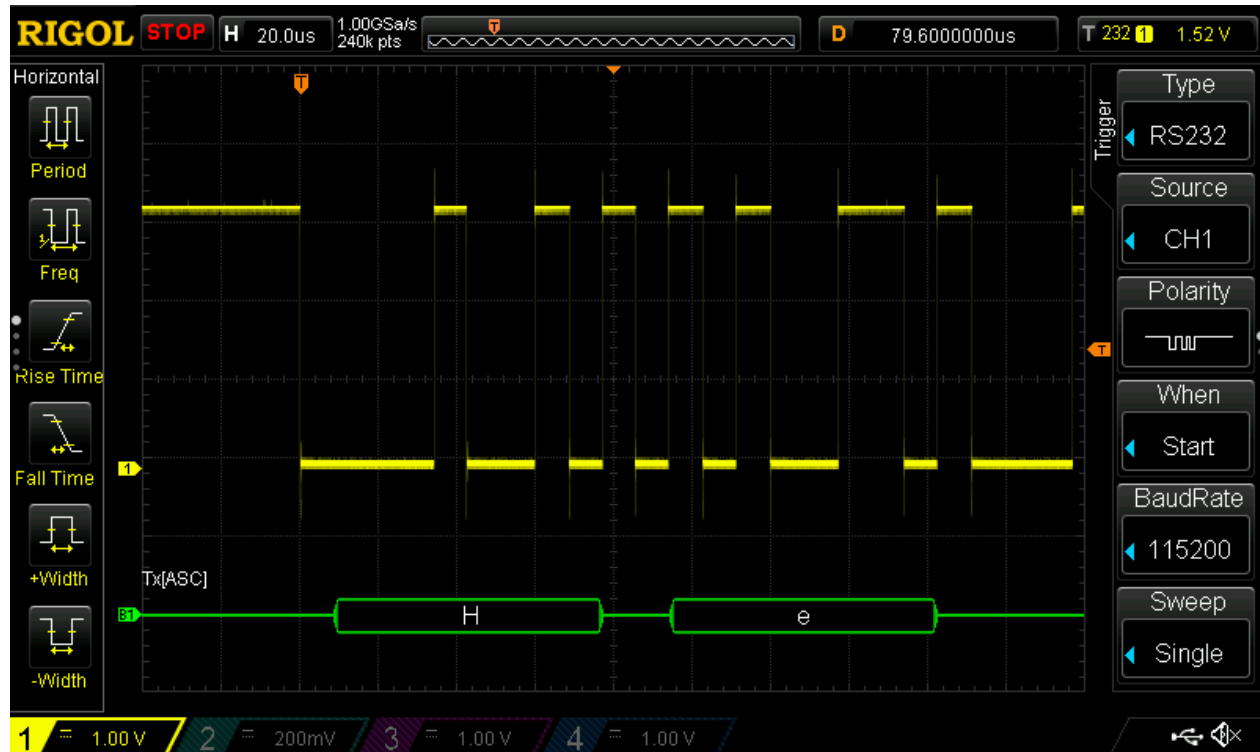


<sup>4</sup> [http://www.neff.site/courses/nwes/nwes\\_5/2.4\\_rs232.pdf](http://www.neff.site/courses/nwes/nwes_5/2.4_rs232.pdf)

## Decoding

To decode the protocol the trigger as well the decoder has to be configured. The trigger is configured as follows:

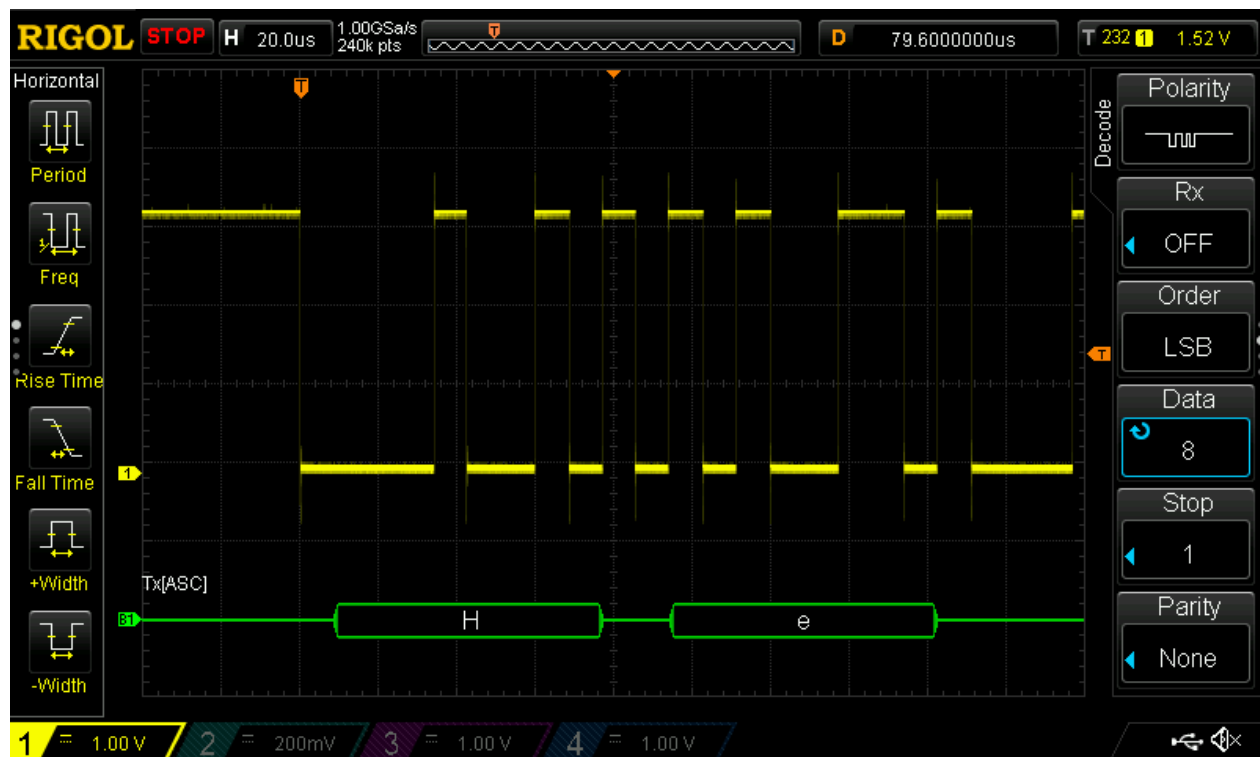
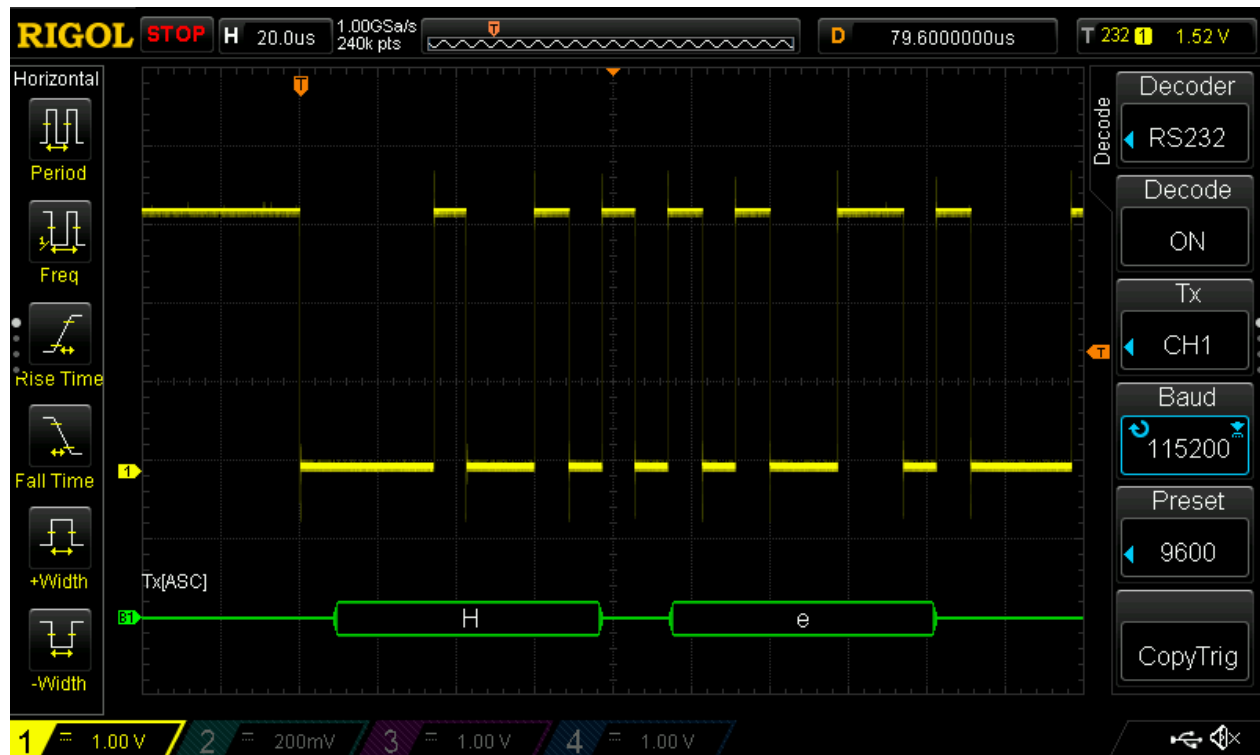
- Use button 14 to select trigger type RS232.
- Use knob 11 to set the trigger level to 1.5 V.
- Select Source, Polarity and Baud Rate appropriately.



Then the decoder has to be configured:

- Use knob 12 to set 1 V per division.
- Use knob 13 to set 20.0  $\mu$ s per division.
- Press button 10 and select Decode1.
- Select Decoder RS-232.
- Switch Decode on.
- Select the channel for the TX line.
- Select the baud rate.
- Select data bytes, stop bits and parity appropriately.
- Press button 5 to capture the signal.
- Use knob 8 to move the signals vertically. Use button 3 to select the signal to be moved.
- Use knob 9 to move the signals horizontally.
- Press button 5 again to capture the adjusted signal.

At first an idle line is indicated by a high level. When the transmission starts a start bit (low level) is put on the line. Between the first data byte (letter *H*) and the second one (letter *e*) there is a stop bit (high level) and another start bit.





## SPI

SPI is a serial bus protocol with a master-slave architecture. It uses three lines for data transfer:

- MISO: Master Input, Slave Output
- MOSI: Master Output, Slave Input
- SCLK: Serial Clock

And an extra line to select the addressee:

- SS: Slave Select

SPI is used on a circular buffer and data transfer is rather a data exchange between master and slave. Please see Module 5.3.4<sup>5</sup> for more information or consult the Internet.

## Capturing

For clearness we will capture MOSI and SCLK, only. This is done by the following settings:

- Use knob 12 to set 2 V per division.
- Use knob 13 to set 50.0  $\mu$ s per division.
- Use knob 11 to set the trigger level to 1.5 V.
- Press button 5 to capture the signal.
- Use knob 8 to move the signals vertically. Use button 3 to select the signal to be moved.
- Use knob 9 to move the signals horizontally.
- Press button 5 again to capture the adjusted signal.

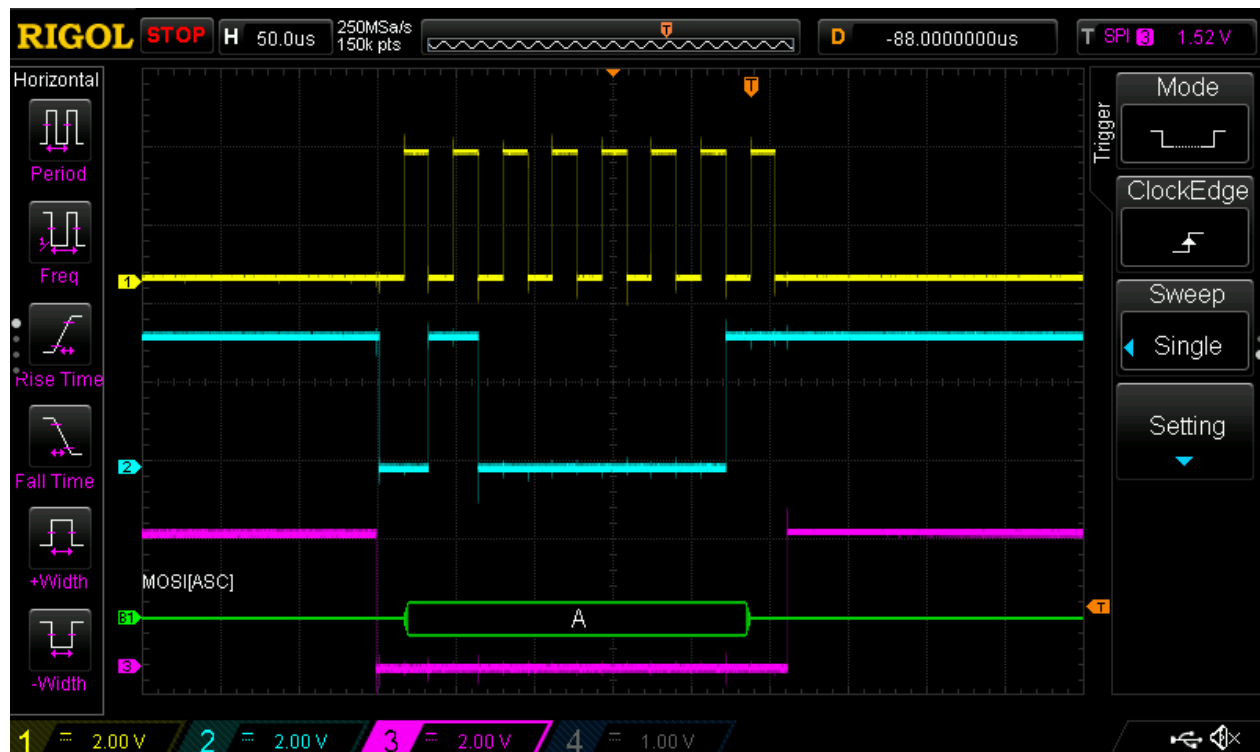
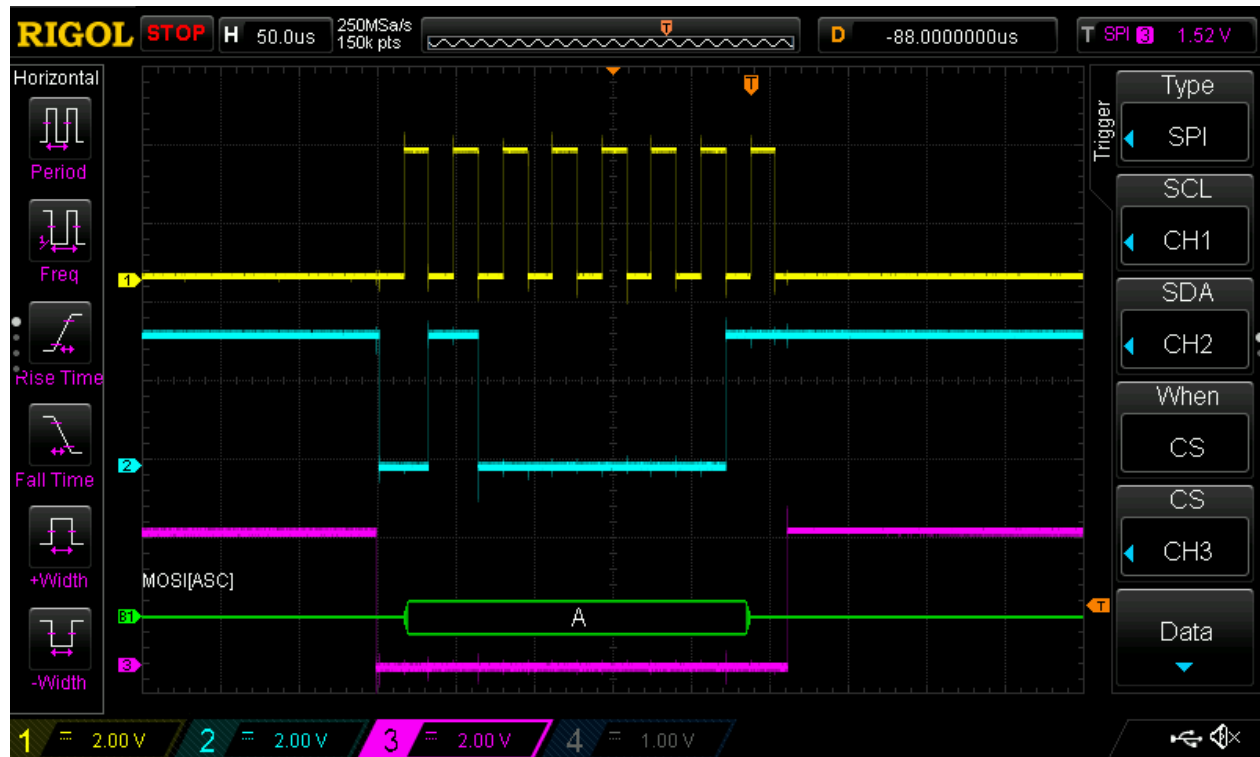


<sup>5</sup> [http://www.neff.site/courses/nwes/nwes\\_5/2.4\\_spi.pdf](http://www.neff.site/courses/nwes/nwes_5/2.4_spi.pdf)

## Decoding

For the decoding we will use MOSI, SCLK and SS which is called CS (chip select) in this case. The trigger is configured as follows:

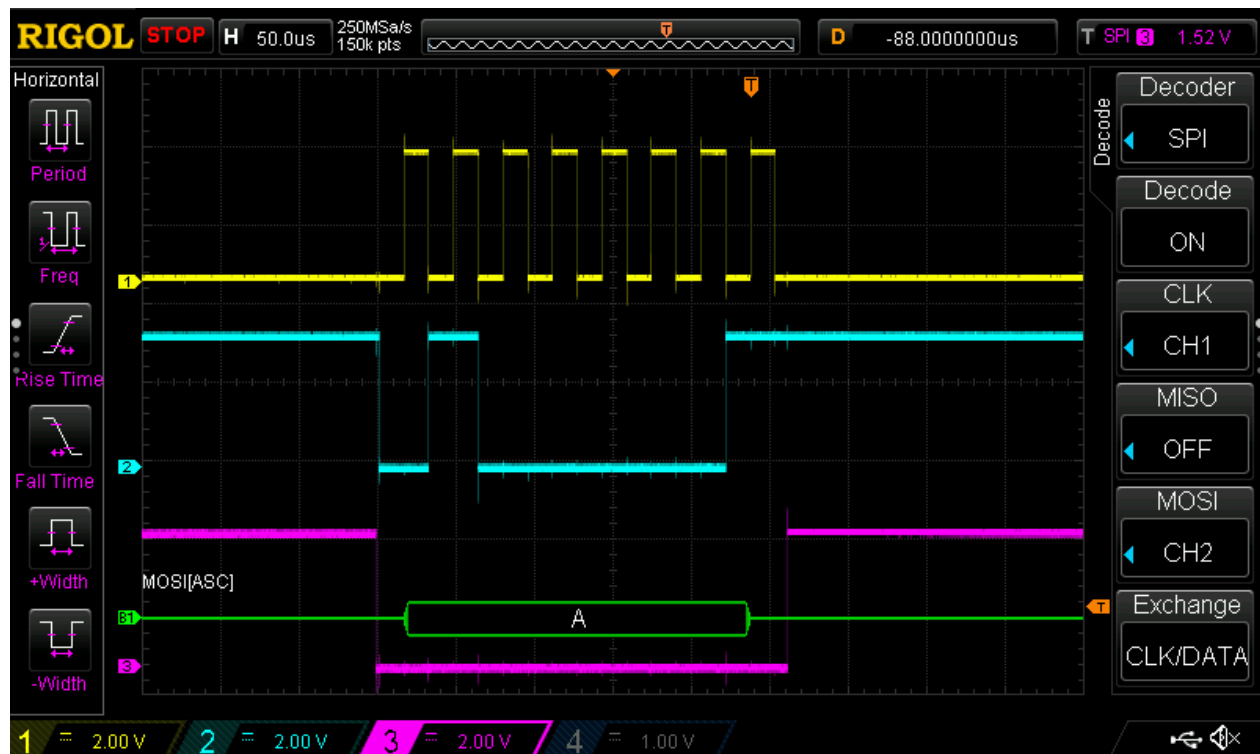
- Use button 14 to select trigger type SPI.



- Select the analog channels for SCL, SDA and CS.
- Use knob 11 to set the trigger level to 1.5 V.
- Select trigger type and Clock Edge appropriately.

To configure the decoder:

- Use knob 12 to set 2 V per division.
- Use knob 13 to set 50.0  $\mu$ s per division.
- Press button 10 and select Decode1.
- Select Decoder SPI.
- Switch Decode on.
- Select the channels for CLK, MISO and MOSI.
- Select the channel for CS and set Mode and Edge appropriately.
- Press button 5 to capture the signal.
- Use knob 8 to move the signals vertically. Use button 3 to select the signal to be moved.
- Use knob 9 to move the signals horizontally.



After the slave is selected by a low signal on the CS line the master puts the data on MOSI. The speed of the data exchange is controlled by the serial clock CLK. The master writes a bit if CLK is low and the slave reads the bit if CLK is high. The data transmitted is the letter A.

