Welcome!

And thank you for purchasing our **AZ-Delivery Logic Analyzers**! On the following pages, we will take you through the first steps of the installation to the first signal readout. We wish you a lot of fun!

http://flyt.it/LogicAnalyzer

The **AZ-Delivery Logic Analyzer** is based on the original, so that recorded signals can also be displayed and analyzed using the original software.

Overview of the most important information

» Data connection via mini USB-B cable
» 8 parallel measurable signal outputs
» Measuring range 0-5V
» Maximum sampling rate of 24 million steps per second
» Measuring digital low: < 0,8 V
» Measuring digital high: > 2,0 V

On the following pages, you will find information about

» **Hardware-Installation**

And instructions for

» **reading an I²C data record between Arduinos**.

It is assumed by this tutorial that you are familiar with uploading sketches to an Arduino and use the Serial Monitor!

Overview of all Links

Saleae Logic Analysis Software:

» *https://www.saleae.com/downloads*
» *http://support.saleae.com/hc/en-us/articles/210245583-saleae-logic-beta-software*
» Documentation: *http://support.saleae.com/hc/en-us/categories/201256716-Users-Guide-Documentation*
» ASCII-Table: *https://de.wikipedia.org/wiki/American_Standard_Code_for_Information_Interchange#ASCII-Tabelle*

sigrok – OpenSource Signal Analysis Software:

» Windows: *https://sigrok.org/wiki/Windows*
» MacOS: *https://sigrok.org/wiki/Mac_OS_X*
» Linux: *https://sigrok.org/wiki/Linux*

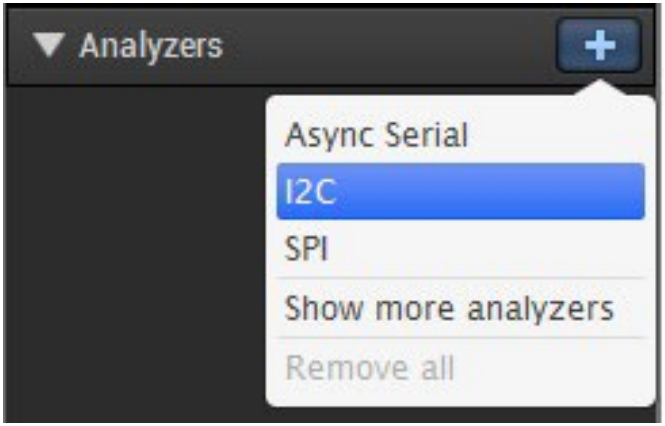
Interesting information from AZ-Delivery

» Arduino-compatible boards: *https://az-delivery.de/collections/arduino-kompatible-boards*
» Arduino accessories: *https://az-delivery.de/collections/arduino-zubehor*
» AZ-Delivery G+Community: *https://plus.google.com/communities/115110265322509467732*
» AZ-Delivery on Facebook: *https://www.facebook.com/AZDeliveryShop/*

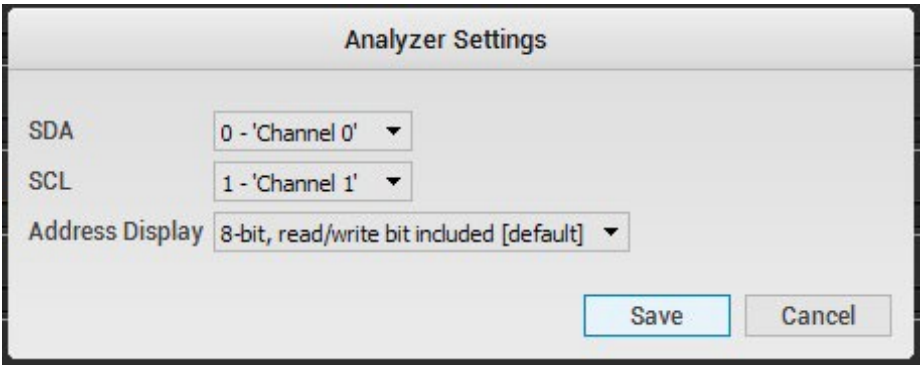
Installation of the Logic Analyzer

Visit this webpage *https://www.saleae.com/downloads* and download the Saleae Logic software for your system. The right device drivers will be immediately delivered with the installation.
When you connect the Logic Analyzer to your computer and you start the "**Saleae Logic**" program, the top line should read "**[Connected]**".
You are now ready to begin!

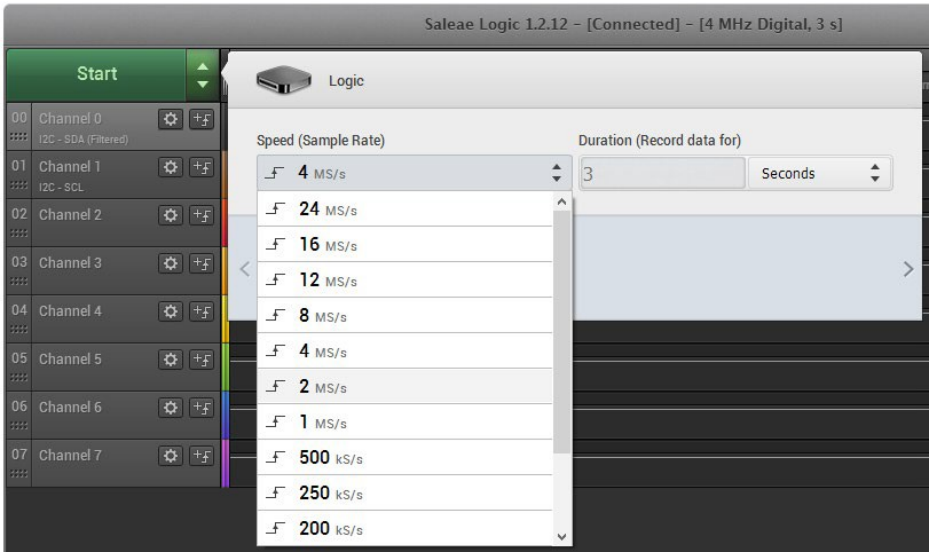
The program can now read signals. In order for the program to be able to interpret these, you will have to make some configurations. As, in this tutorial, you would be monitoring the communication of two Arduinos over I^2C, you must inform the **Saleae Logic** Software, that this protocol must be used for interpretation. On the right side click on the + next to "**Analyzers**" and select "**I2C**".



In the window that appears next, you can make fine adjustments. "**SDA**" and "**SCL**" are the data interfaces of I^2C and you can specify here, which pins from the Logic Analyzers you would like to connect with them. For our project, we would take the standard specifications.



After you click on the *save* button, you should see on the left side that channels **0** and **1** have been assigned the "**SDA**" and "**SCL**" data lines. You can now set the query speed via the two triangles, located on the right, next to "**Start**".



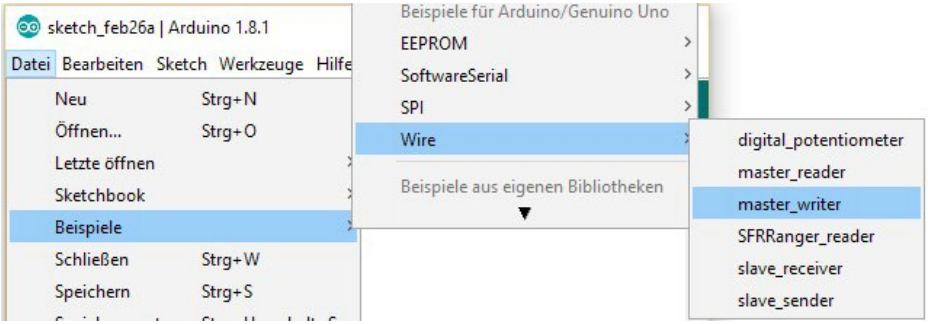
I²C requires a minimum sampling rate of two million steps per second. The **AZ-Delivery Logic Analyzer** can manage up to 24 million, but the speed also depends on the computer you are using. With a speed of more than 4M/s, errors may occur during the readout process.
For the duration of the recording, you should define a suitable time span for your project, so that it can record one or several program runs. The following project sends out a signal every 500 ms, giving three to five interpretable data sets.

I²C analysis between Arduinos

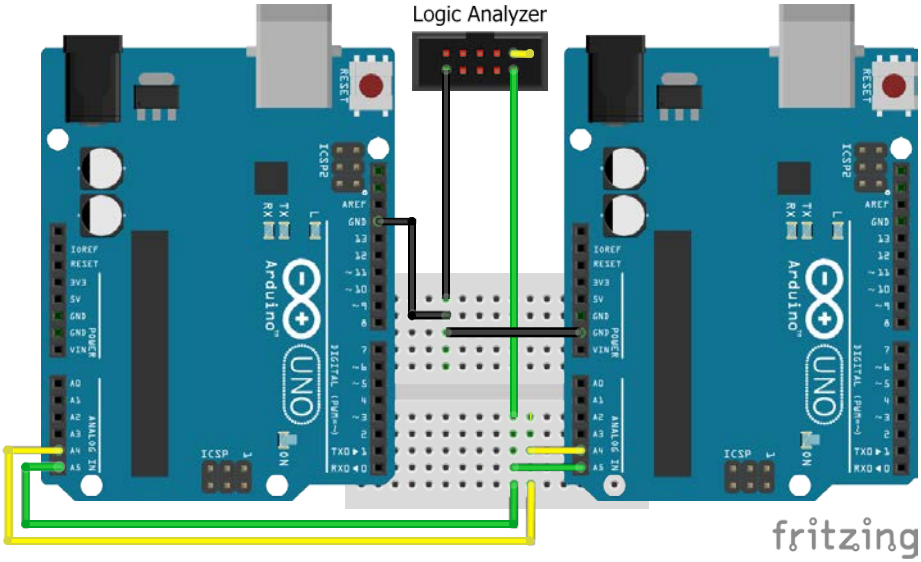
There are many data signals. The most common for Arduinos and their peripherals are the **serial connection**, **SPI** and **I²C**. Arduinos can easily communicate with each other via the latter, and in this tutorial, with the help of the Logic Analyzers, we will be able to listen to them.
In order to do that, you would need two controllers that can be programmable with the Arduino Code. One example can be two UNOs, which you can purchase here from **AZ-Delivery**: *flyt.it/unor3*
Two example sketches are used for the Wire library.

» **master_writer**
» **slave_receiver**

Install the sketches on the controller, preferably the slave_receiver after the master_writer, so that later you would have the correct connection settings.



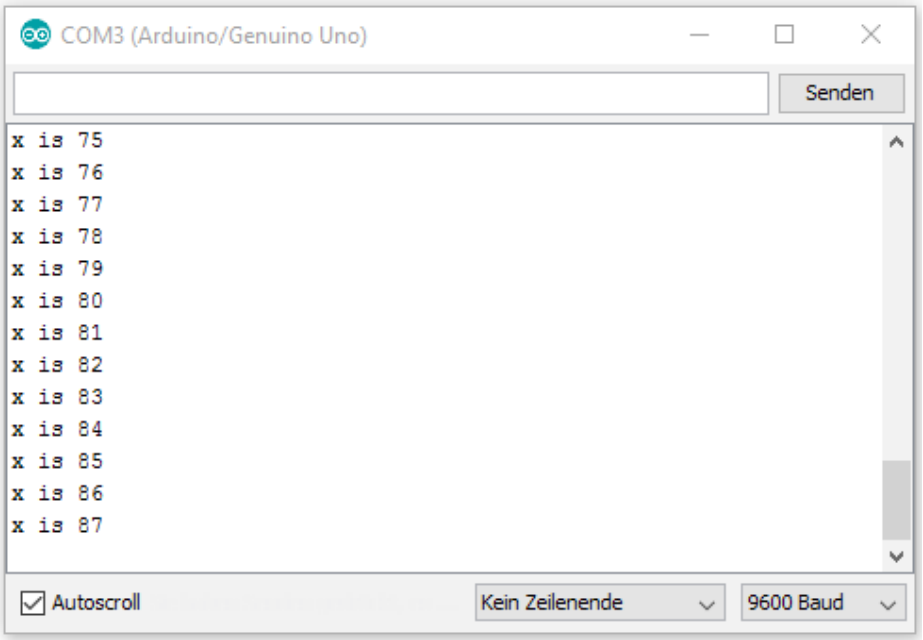
Disconnect the controller from the USB port and then connect it accordingly to the circuit diagram.



To ensure pure data connection, both controllers can also be directly connected to each other, but since the Logic Analyzer also requires access, the use of a breadboard is advisable. The reader is also connected via the data ports "**A4 (SDA)**" to "**Ch1**", "**A5 (SCL)**" to "**CH2**" and "**GND**" to "**GND**" according to the pin label:

At this point, you would have to modify something, because the Saleae Logic program will begin to count from 0, e.g. **CH1 = Channel 0** and **CH2 = Channel 1**.

When you are done, make sure that all devices are supplied with power, or at least Logic Analyzer and the Controller with the slave-sketch via the USB outputs of your computer. Now, if you open the Serial Monitor in the Arduino IDE, you should be able to see that it counts from 0-255 continuously in a loop. If you do not see it, or if you see something else, you should recheck if the correct port for the controller was set with the slave_sketch and if the baud rate was set to 9600.



Return to the **Saleae Logic** and check, firstly, if "**[Connected]**" is still on the top line. Click on "**Start**" and wait for the three already set seconds, until the program detects the I^2C signal.

If you do not see anything in the first two lines, you should zoom out completely with the mouse wheel. By doing so, you should see five or six blocks of data, on which you can now carefully zoom in.
On the right side, in the "**Decoded Protocols**" section, you can see what the program has recorded and interpreted:

X

..

I

S

‚162‘

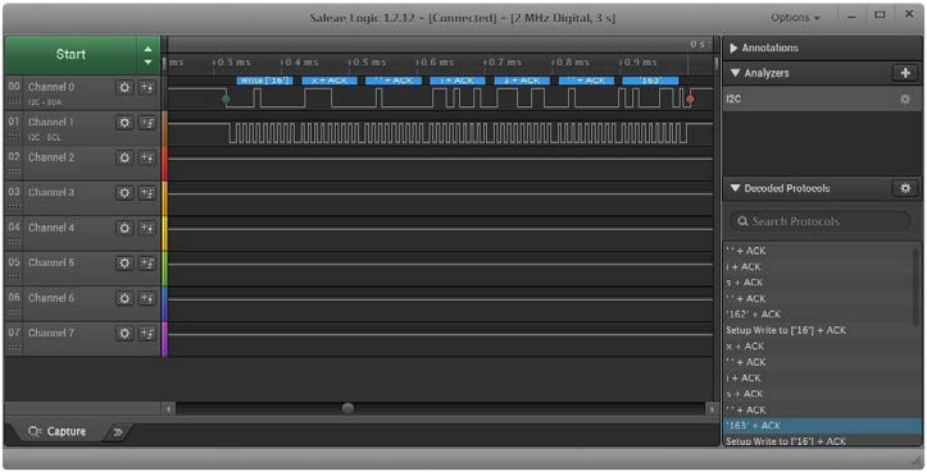
The numbers 32 to 126 are displayed as special characters and letters because the number transmitted as a byte is also interpreted as such by the Saleae Logic and the corresponding ASCII characters are displayed. Which number (DEZ) corresponds to which character (ASCII) you can find out and read more about here:

» *https://de.wikipedia.org/wiki/American_Standard_Code_for_Information_Interchange#ASCII-Tabelle*

If you are still interested and wondering how long does it take your controller to send a byte, or how accurately it has kept the pre-setted break of 500ms, you can quickly figure it out by simply hovering your mouse over the space between two alterations.

—

Now you know how to attach your Logic Analyzer to a data line, which emits signals and allows those signals to be interpreted.



Now it is time to try it out! For example, you can easily read out the serial connection of your slave-controller, located in your circuit. In order to do this, connect "**CH3/Channel 2**" to the "**TX-Pin**" (D1 on the UNO) of your controller and add the analyzer "**Async Serial**" in the program (select "**Channel 2**" and your chosen baud rate of "**9600**"). After you click on "**Start**", you can, for example, see which signal was sent first and for which one more or less time was needed.

If you would like to learn more about **Saleae Logic** and how to use it, check out the manufacturer's support page here:

» *http://support.saleae.com/hc/en-us/categories/201256716-Users-Guide-Documentation*

And for more hardware to analyze, our online store is always at your disposal:

https://az-delivery.de

Enjoy!

Imprint

https://az-delivery.de/pages/about-us