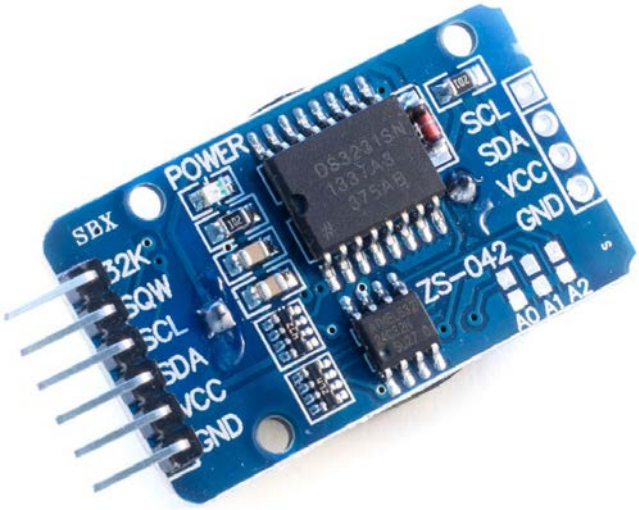
Welcome!

And thank you for purchasing our **AZ-Delivery Real Time Clock ZS-042**! On the following pages, we will take you through the first steps of the installation process to the first script. We wish you a lot of fun!



http://flyt.it/RTC

The **AZ-Delivery Real Time Clock ZS-042**, abbreviated as **RTC**, is an external timer based on the **DS3231** chipset, for example, to be used with Arduino & Co. Thanks to its own rechargeable power source, it runs autonomously; and once set up, it has the ability to run other controllers with one provided absolute value. The data transfer takes places via the **I²C** interface.

Overview of the most important information

» **Chipset:** DS3231
» **Dimensions:** 37x21,5x11mm (without pins)
» **Power supply:** 3V CR2032 lithium battery / VCC
» **Data interface:** I^2C
» **Temperature sensor**: ± 3°C accuracy

» **Watch-related functions:**
  » Year, month (name & number), day, hour, minute, second
  » Leap year calculations up to 2100
  » 2 daily alarms
  » ±0.0002% deviation at 0-40°C
  » ±0.00035% deviation at -40-85°C

On the following pages, you will find information about
» **connection with the Arduino**
And instructions for
» **a time controlled Blink-Event**.

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

Overview of all Links

DS3231:

» Datasheet:
https://datasheets.maximintegrated.com/en/ds/DS3231.pdf

» Arduino library with example sketches:
https://github.com/StephanFink/RTClib

Application programming interfaces:

» Arduino IDE: *https://www.arduino.cc/en/Main/Software*

» Web-Editor: *https://create.arduino.cc/editor*

» Arduino extension for SublimeText:
https://github.com/Robot-Will/Stino

» Arduino extension "Visual Micro" for Atmel Studio or Microsoft Visual Studio:
http://www.visualmicro.com/page/Arduino-for-Atmel-Studio.aspx

Arduino Tutorials, Examples, Reference, Community:

» *https://www.arduino.cc/en/Tutorial/HomePage*

» *https://www.arduino.cc/en/Reference/HomePage*

Interesting information from AZ-Delivery

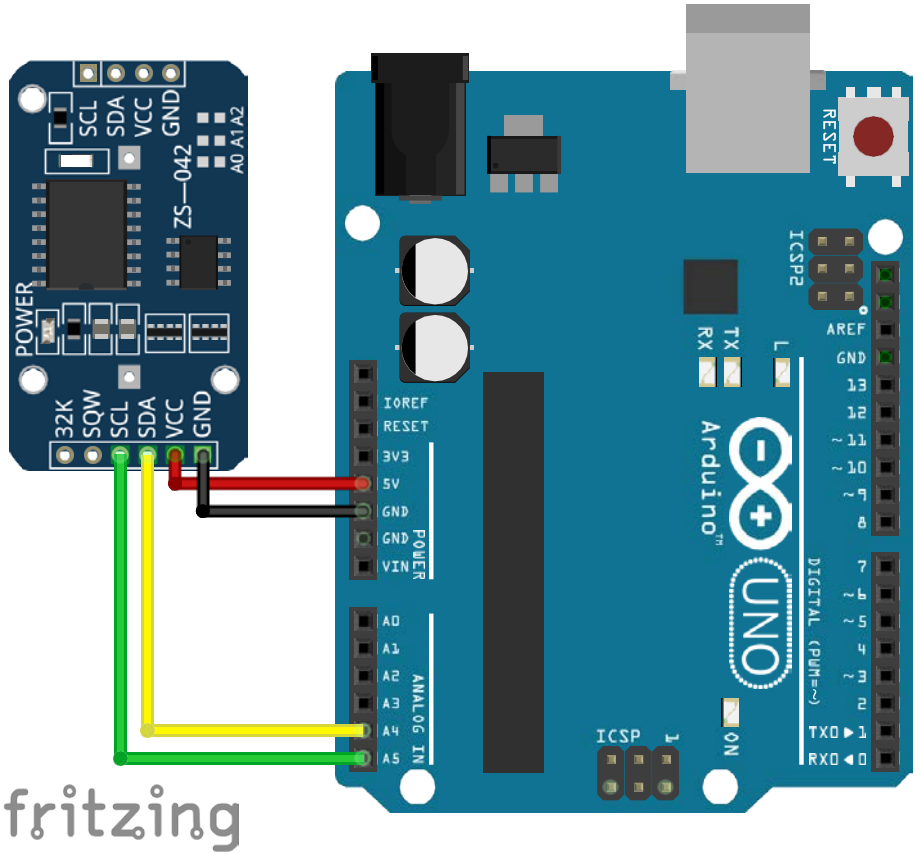
» Arduino accessories:
https://az-delivery.de/collections/arduino-zubehor

» AZ-Delivery G+Community:
https://plus.google.com/communities/115110265322509467732

» AZ-Delivery on Facebook:
https://www.facebook.com/AZDeliveryShop/

Connection with the Arduino

Thanks to the 2,3-5,5 V voltage tolerance of the **DS3231,** the RTC is very controller-friendly and can be used with all Arduino types. You can connect to a UNO, as it is also used in this tutorial, with the help of the image shown below:



Installing the DHT22 library

The **DS3231** RTCs are so popular that several Arduino libraries are available on the internet. **AZ-Delivery** itself provides a library, which was built by the Adafruit Company and has improved the example sketches, also used here in this tutorial. You can download it from the following web address:

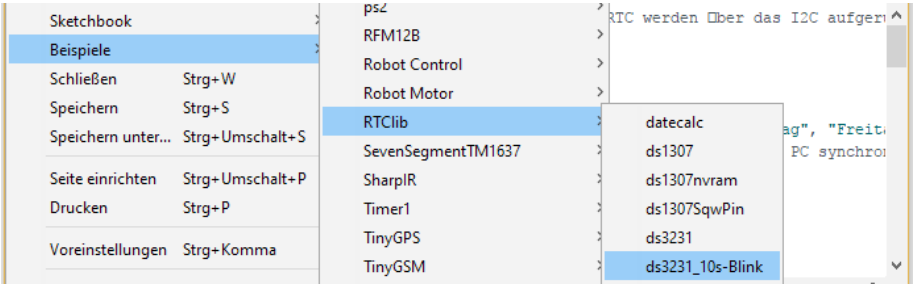
» *https://github.com/StephanFink/RTClib/archive/master.zip*

Unpack the "**RTClib-master**" folder from the zip archives in the libraries' directory of your Arduino Workspace, e.g. "**my documents > Arduino > libraries**" and rename it "**RTClib**". Now if you start the Arduino IDE (new), you can find the subdirectory "**RTClib**" under "**Sketch > Include Library**" and "**File > Examples**".

The first Script

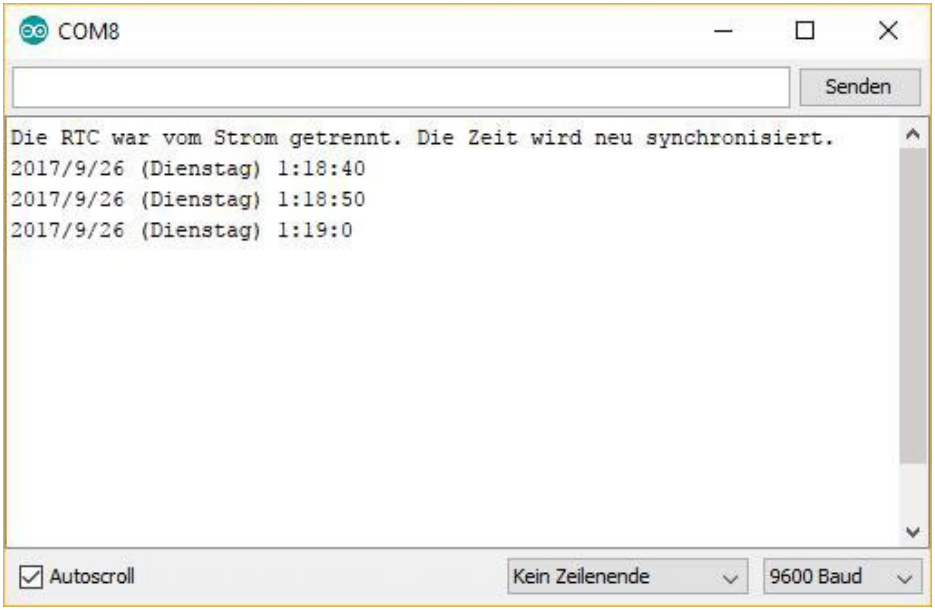
Although the first sign of success in most programming languages is the phrase: "Hello World!", for Arduino, the first sign of success is the blinking of the board's internal LED. With the help of the RTC, we now let it always light up every ten full seconds.

Open the sample sketch "**ds3231_10s-Blink**", under "**File > Examples > RTClib**":



If, in the meantime, the RTC has been left without power supply, then during the upload of the sketch, the current time of your PC will be provided to the RTC. Keep in mind, though, that right from the beginning, it will take a few seconds because the timestamp transferred, is the time when the compilation process had been started. The RTC will not start counting until the upload process had been completed. The difference usually is no more than 15 seconds.
Meanwhile, if you want to manually synchronize the RTC, then set the variable "**synchOnFirstStart**" in line 7 to "**true**" and upload the sketch again.

Start the terminal with the correct baud rate and the current time, incl. date and weekday will be displayed every ten seconds:



Meanwhile, the Arduino board's internal LED will light up at exactly the same time for one second.

You did it! Congratulations!

Now it is time to learn and try it out. It is best advised to look at the sketches and commentary, to find out how you can access each data on the RTC. For example, with this module, you can now activate and control other hardware connected to the Arduino at any time you want. You can find this and much more in your online shop:

https://az-delivery.de

Enjoy!

Imprint

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