

# Wortuhr

- Patent
- Idee
- Aufbau der Anzeige
- Teile
- Digitaler LED-Streifen
- Anordnung der LED Streifen
- LED-Adressen & Buchstaben
- Worte & LED-Adressen
- Arduino ProMini
- Arduino ProMini programmieren
- Arduino IDE
- Schaltplan LED-Streifen
- Schaltplan RTC
- Schaltplan HC 06 Bluetooth
- Schaltplan Taster
- Bluetooth
- Smartphone-App
- Taster Minuten
- Sketch

# Patent

Link <https://patents.google.com/patent/DE202009017703U1/de>

Diese Präsentation darf nicht aktuell, z.B. vom DARC, veröffentlicht werden.  
Bitte nur an Interessierte weiter geben.

## Application DE202009017703U events ⓘ

- 2009-12-29 • Application filed by Biegert & Funk & Co KG GmbH
- 2009-12-29 • Priority to DE202009017703U
- 2010-04-01 • Publication of DE202009017703U1
- 2019-12-30 • Anticipated expiration
- 2020-01-10 • Application status is Active

**Info:** [Patent citations \(12\)](#), [Non-patent citations \(1\)](#), [Cited by \(2\)](#),  
[Legal events](#), [Similar documents](#), [Priority and Related Applications](#)

**External links:** [Espacenet](#), [Global Dossier](#), [DPMA](#), [Discuss](#)

# Idee

## Wortuhr

- Mit Zeitangaben in englischer Sprache.
- Wechsel der Anzeige alle 5 Minuten.
- Leds in der unteren LED-Leiste zeigen Minuten an.

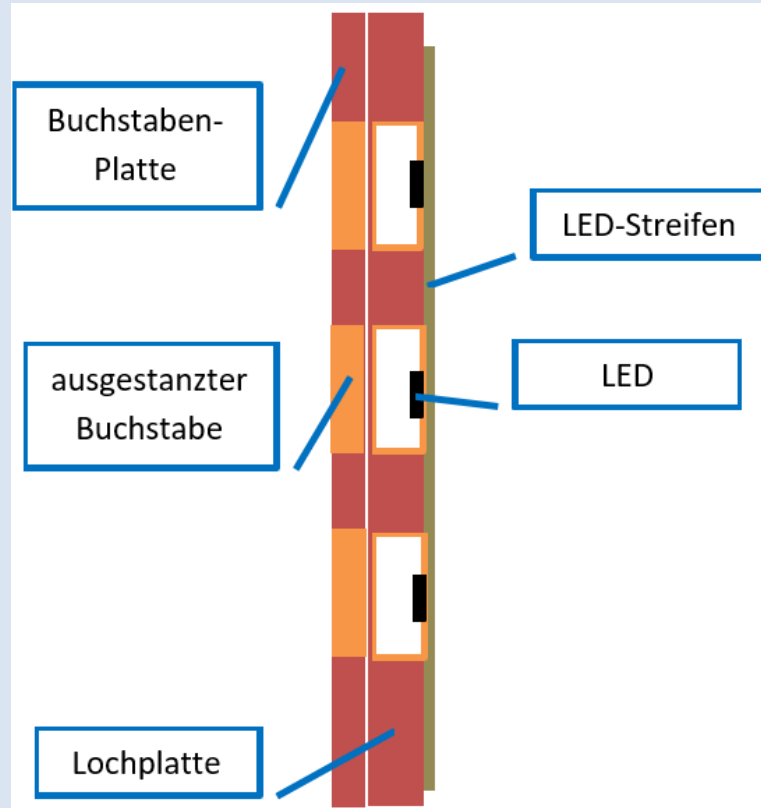
Hier:

„IT's FIVE PAST TWELVE“  
und eine LED ergeben 7:06.



# Aufbau der Anzeige

Blickrichtung  
➔



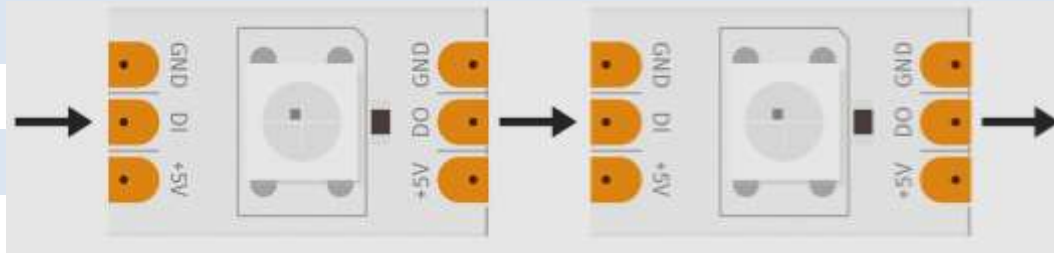
# Teile

Anzeige	Wörter in einer Faserplatte ausgestanzt (Herstellung unbekannt). Faserplatte mit Löchern. Aufgeklebte LED-Streifen.	
LED-Streifen	Digitaler LED-Streifen	(Neopixel kompatibel), 99 Leds
	Digitaler LED-Streifen	(Neopixel kompatibel), 4 Leds <a href="https://www.amazon.de/Adafruit-NeoPixel-Digital-RGB-Strip/dp/B00SLYAHSW">https://www.amazon.de/Adafruit-NeoPixel-Digital-RGB-Strip/dp/B00SLYAHSW</a>
Mikrokontroller	Arduino	(Arduino Pro Mini)
Zeitgeber	Realtime Clock	(DS3231 RTC)
	Bluetooth Modul	(HC-06 Bluetooth)
	Netzteil	
	Holzrahmen	

# Digitaler LED-Streifen

## Aufbau

Als Erweiterung zu – analogen – LED Streifen hat jede LED einen Chip. Jede LED ist adressierbar und kann einzeln aus- und eingeschaltet werden. Auch können Farbe und weitere Effekte eingestellt werden.



Hier hat der Streifen 3 Kontakte:  
Außen GND und +5V  
Innen DI, Datenleitung Eingang  
Innen DO, Datenleitung Ausgang

Grundsätzlich sollte der LED Streifen eine eigene Stromversorgung haben.

## Quelle

<https://starthardware.org/viele-leds-mit-arduino-steuern-ws2812/>

# Anordnung der LED Streifen

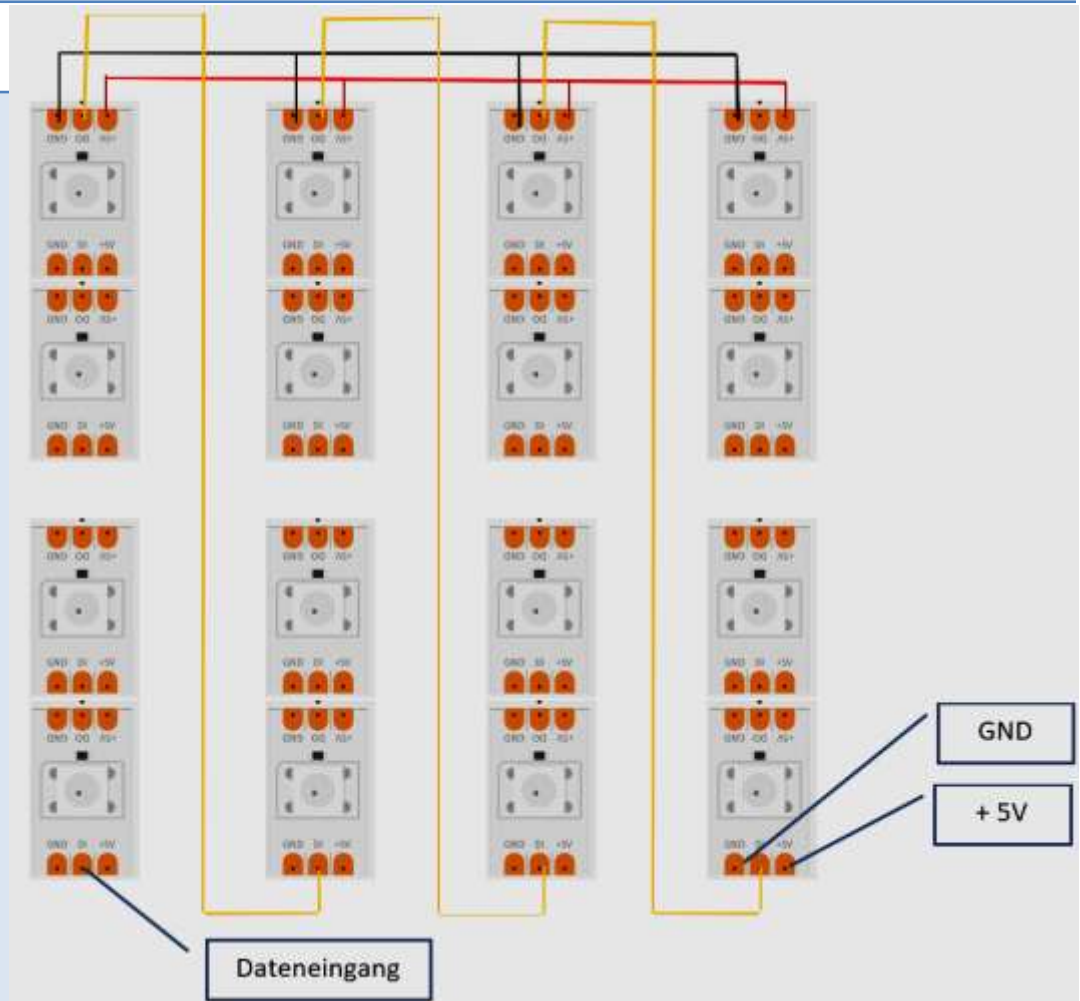
Der LED Streifen kann an den Kontakten aufgetrennt werden. Die Kontakte GND und +5V können parallel verbunden werden.

Der Dateneingang „DI“ wird mit einem digitalem Arduino-Pin verbunden.

In dieser Skizze sind 12 Leds vorhanden.

Diese haben die Adressen von 0 bis 11.

Daher müssen die Datenleitungen seriell verdrahtet werden.



## LED-Adressen & Buchstaben

8	I	17	T	26	'	35	S	44	M	53	T	62	W	71	E	80	N	89	T	98	Y
7	I	16	F	25	I	34	V	43	E	52	O	61	Z	70	T	79	E	88	N	97	A
6	Q	15	U	24	A	33	R	42	T	51	E	60	R	69	H	78	A	87	L	96	F
5	T	14	O	23	Z	32	P	41	A	50	S	59	T	68	Z	77	O	86	N	95	E
4	T	13	H	22	R	31	E	40	E	49	Y	58	S	67	E	76	V	85	E	94	N
3	T	12	W	21	O	30	F	39	O	48	U	57	F	66	F	75	I	84	V	93	E
2	N	11	I	20	N	29	E	38	I	47	G	56	H	65	T	74	E	83	N	92	H
1	T	10	W	19	E	28	L	37	V	46	E	55	L	64	E	73	V	82	E	91	N
0	S	9	I	18	X	27	L	36	O	45	'	54	C	63	L	72	O	81	C	90	K



## Worte & LED-Adressen

Wort	LED-Adressen
IT's	8, 17, 26, 35
HALF	69, 78, 87, 96
TO	5, 14
EIGHT	29, 38, 47, 56, 65

### C++ Array

```
byte some_words[5][8]={
    {0, 0, 0, 0, 0, 0, 0, 0},           // index row=0, nothing
    {4, 8, 17, 26, 35, 0, 0, 0},        // index row=1, IT'S
    {4, 32, 41, 50, 59, 0, 0, 0},       // index row=2, PAST
    {2, 5, 14, 0, 0, 0, 0, 0},          // index row=3, TO
    {7, 36, 45, 54, 63, 72, 81, 90}     // index row=4, O'CLOCK
};
```

Das Array „some\_words[5][8]“ ist zweidimensional.

Die erste Dimension steht für ein Wort.

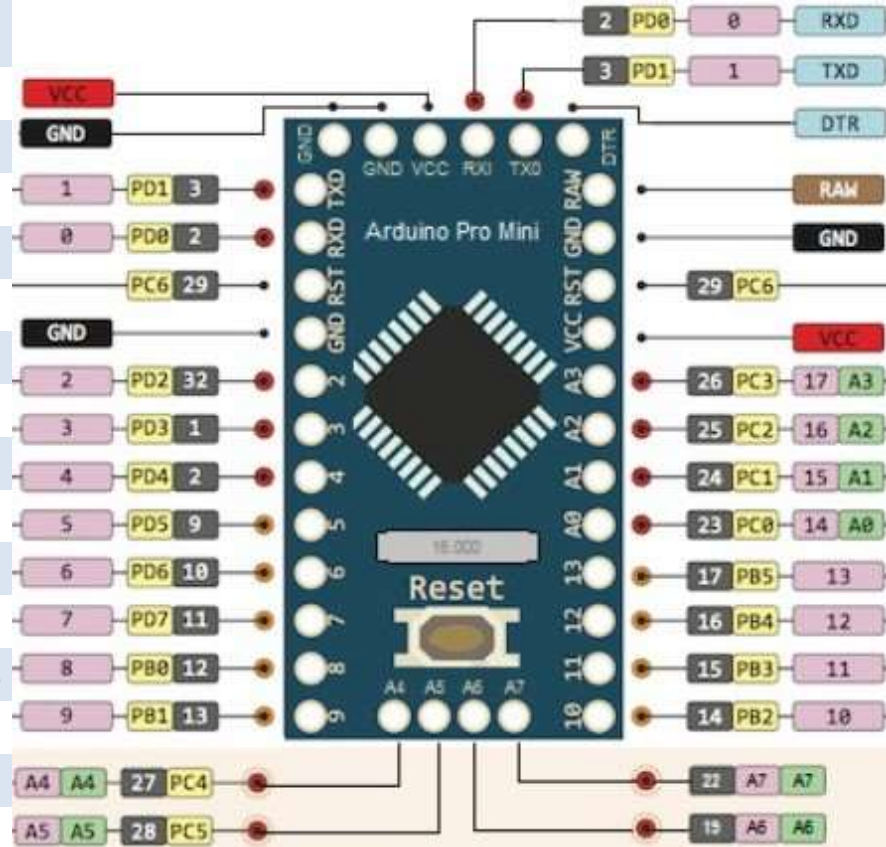
Die zweite Dimension enthält die Adressen der Leds.

Darin der Wert mit Index 0 die Anzahl der Buchstaben des Wortes.

# Arduino ProMini

VCC: Externe Spannung

GND: Gemeinsame Masse



Pin 7: DATA\_PIN\_MATRIX

Pin 8: DATA\_PIN\_MINUTES

Pin 9: pin\_button\_up

Pin A4, SDA: RTC

Pin A5, SDL: RTC

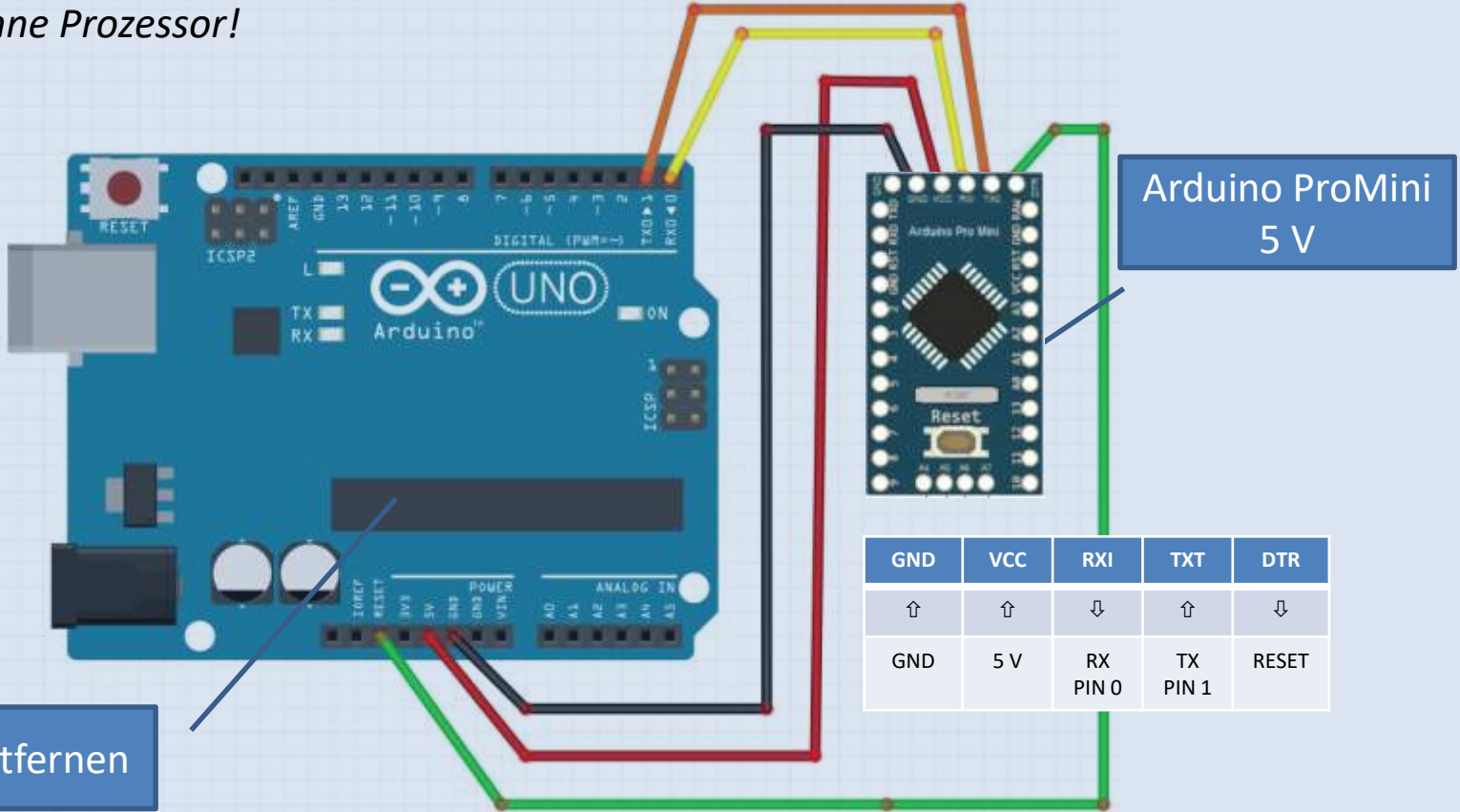
Pin 12: PIN\_SEND (Bluetooth)

Pin 11: PIN\_RECEIVE (Bluetooth)

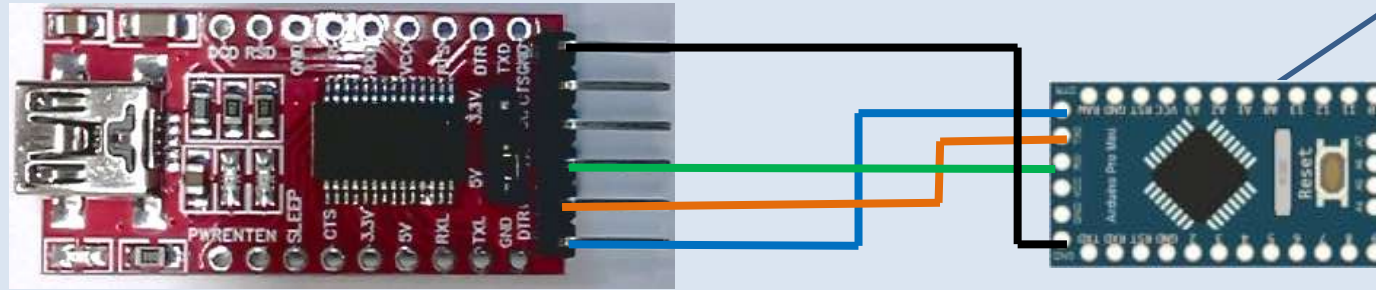
Pin 10: pin\_button\_down

# Arduino ProMini programmieren

*Mit Arduino ohne Prozessor!*



# Arduino ProMini programmieren FTDI232



GND			RXD	TXT	DTR	ProMini
↑			↓	↑	↓	
GND	CTS	VCC	TX	RX	DTR	FTDI232

# Arduino IDE

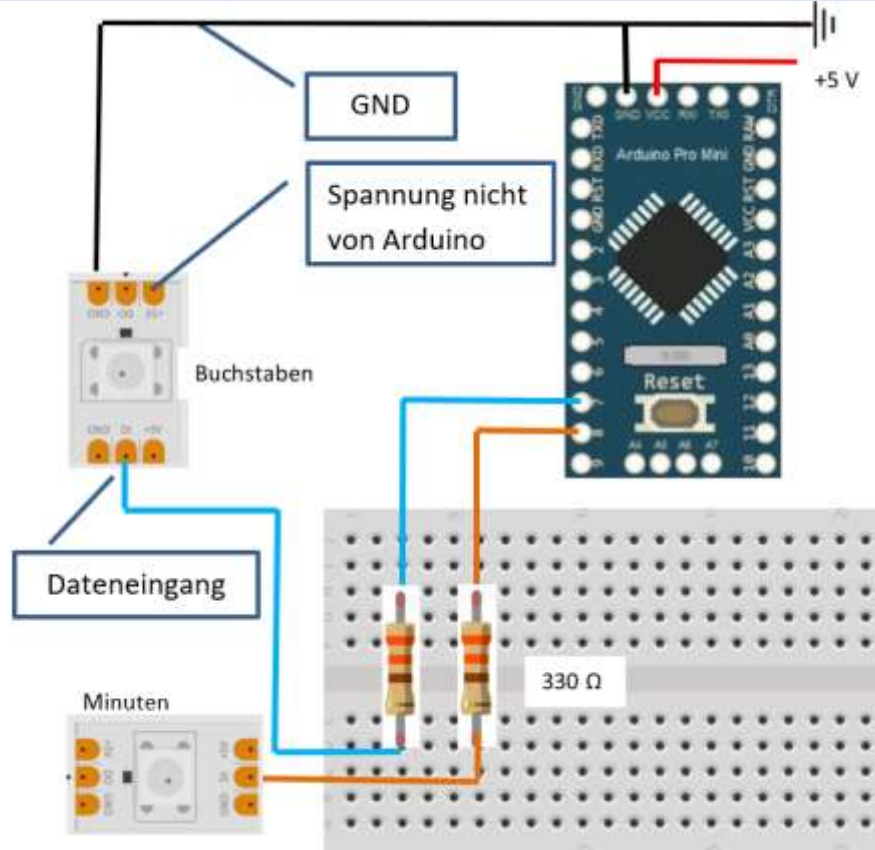
## Einstellungen

Werkzeuge	Board:	Arduino Pro oder Pro Mini
	Prozessor:	Atmega328P (5V, 16 MHz)
	Port:	COMxx (Arduino/Genuino Uno)

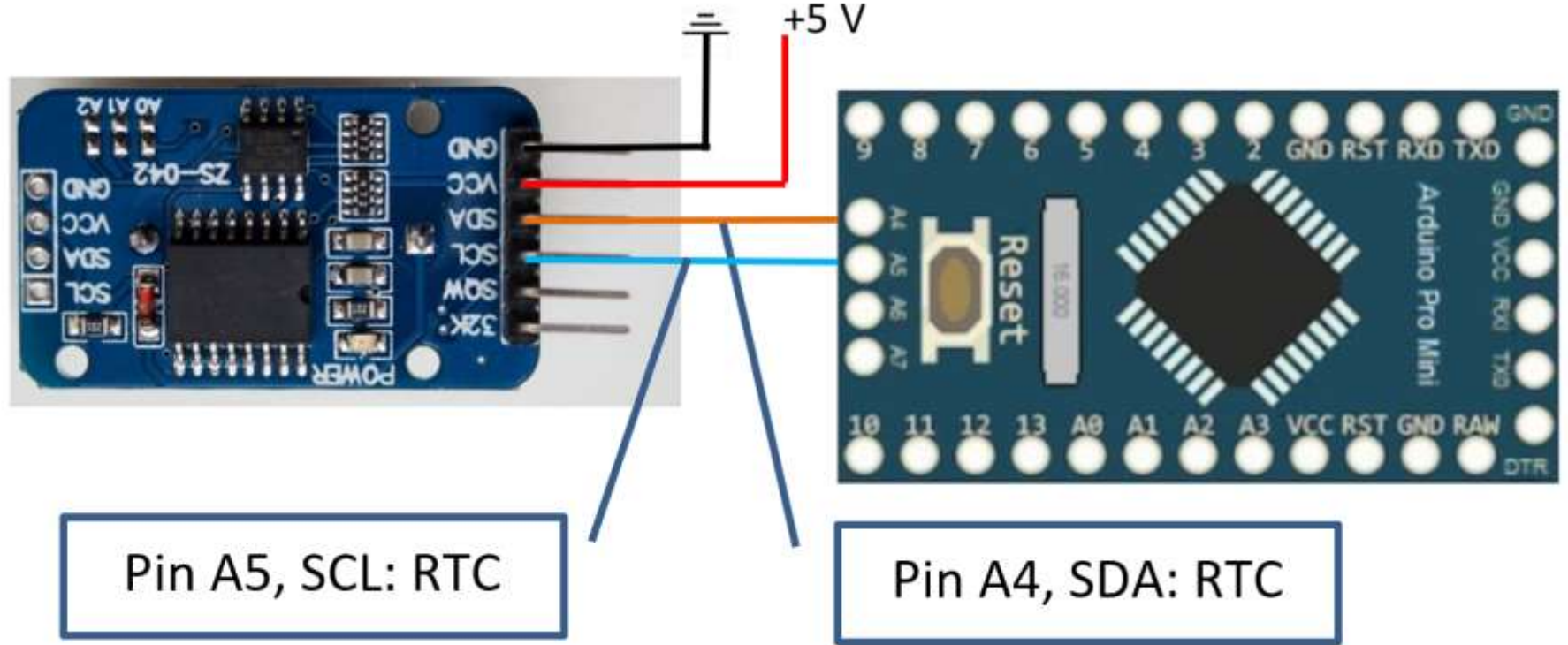
Libraries	FastLED library	<a href="https://github.com/fastled/fastled">https://github.com/fastled/fastled</a>
-----------	-----------------	---

Echtzeituhr DS3231	<a href="https://arduino-projekte.webnode.at/meine-libraries/rtc-ds3231/">https://arduino-projekte.webnode.at/meine-libraries/rtc-ds3231/</a> Die Dateien: MyDS3231.cpp.txt MyDS3231.h.txt keywords.txt herunterladen. Umbenennen in MyDS3231.cpp und MyDS3231.h. Im Ordner „libraries“ den Ordner „RTC3231_Lib“ anlegen. Die o.g. Dateien in den Ordner kopieren
--------------------	--

# Schaltplan LED-Streifen

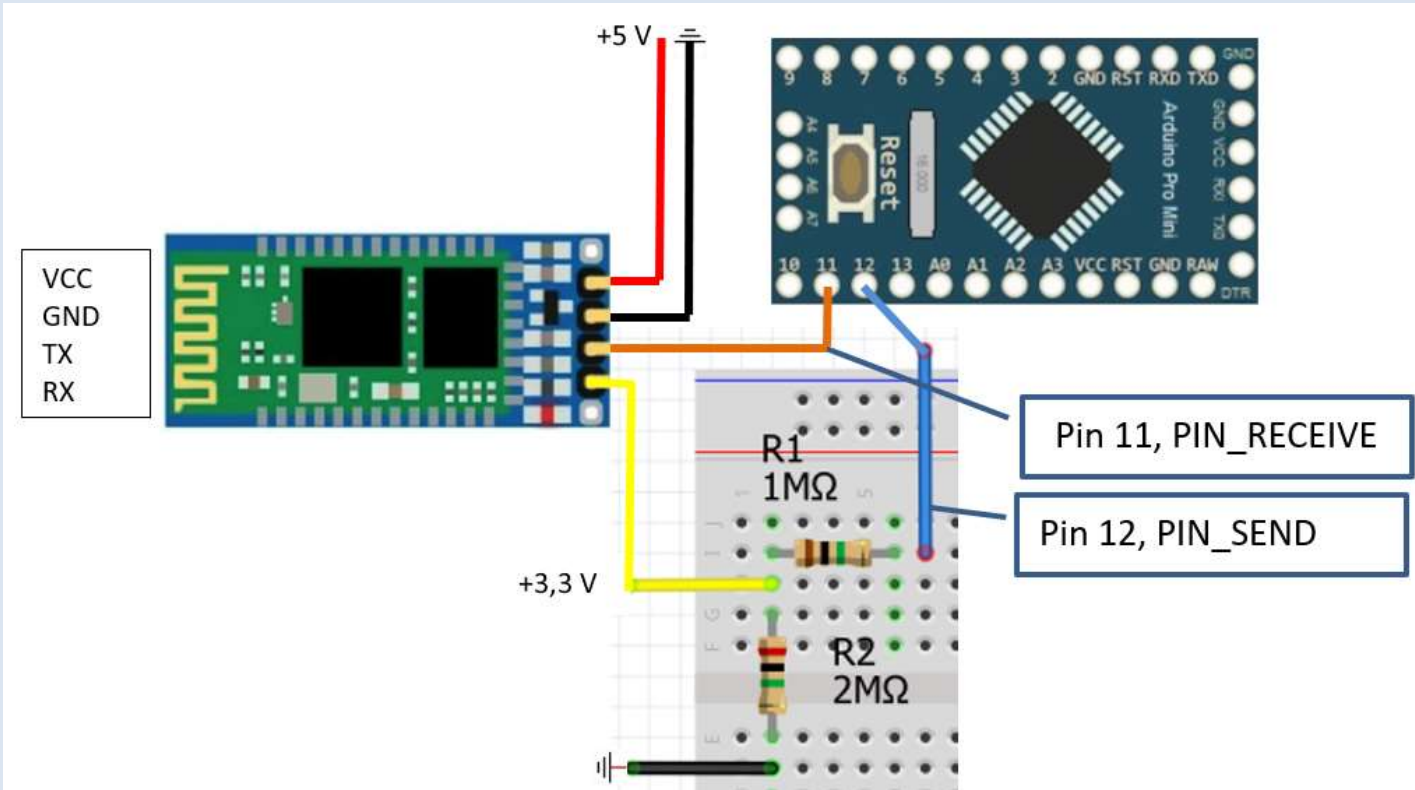


## Schaltplan RTC





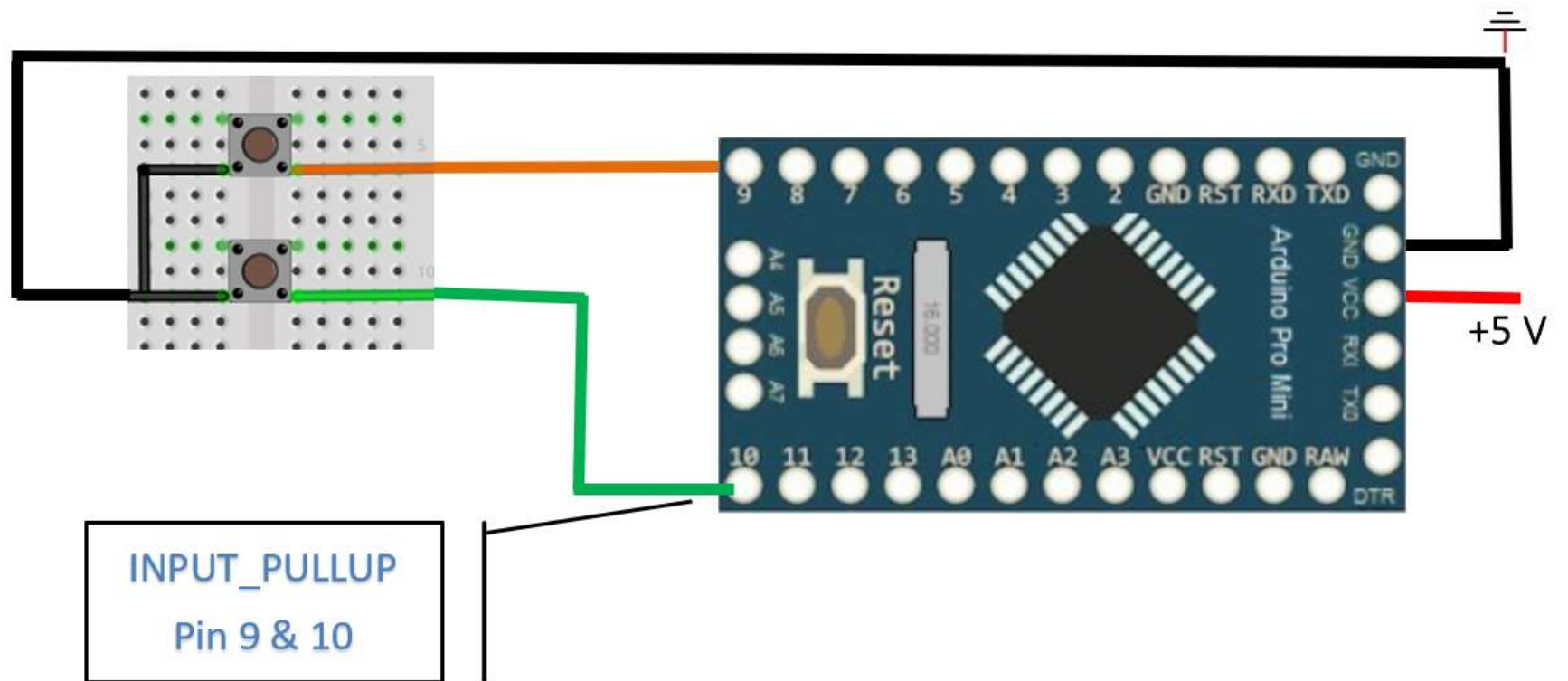
# Schaltplan HC 06 Bluetooth



Äußere Pins „Key und State“ unbenutzt



# Schaltplan Taster



# Bluetooth

Modul	HC-06 Bluetooth
-------	-----------------

Einrichtung	Das Bluetooth Modul kann mit Hilfe eines Arduino und einem Sketch konfiguriert werden. Dazu stehen AT-Befehle zur Verfügung.
-------------	--

<http://www.martyncurrey.com/hc-05-fc-114-and-hc-06-fc-114-part-2-basic-at-commands/>

Eingestellte Parameter:
-------------------------

Device name	ON8TOM
-------------	--------

Pin code	1992
----------	------

Arduino-Sketch	<pre>// Library to allow other digital pins &amp; multiple serial ports (bluetooth) #include &lt;SoftwareSerial.h&gt; #define PIN_RECEIVE 11 #define PIN_SEND 12 #define SPEED_BLUETOOTH 9600 #define SPEED_SERIAL 9600  // Creating Bluetooth serial port "blueSerial" SoftwareSerial blueSerial(PIN_RECEIVE, PIN_SEND);  void setup() {   // Establish connection to Bluetooth &amp; notice via smartphone-app   blueSerial.begin(SPEED_BLUETOOTH);   blueSerial.listen();   blueSerial.println("ON8TOM-CLOCK working"); }</pre>
----------------	--

# Bluetooth

## Sketch

## Bluetooth-Eingaben & Verarbeitung

### Abfrage

```
void loop() {  
  // Calling for serial input done from blueEvent()  
  blueEvent();  
  // Setting time from Bluetooth input  
  if (serialComplete) processCommand();  
  . . .  
}
```

### Funktion

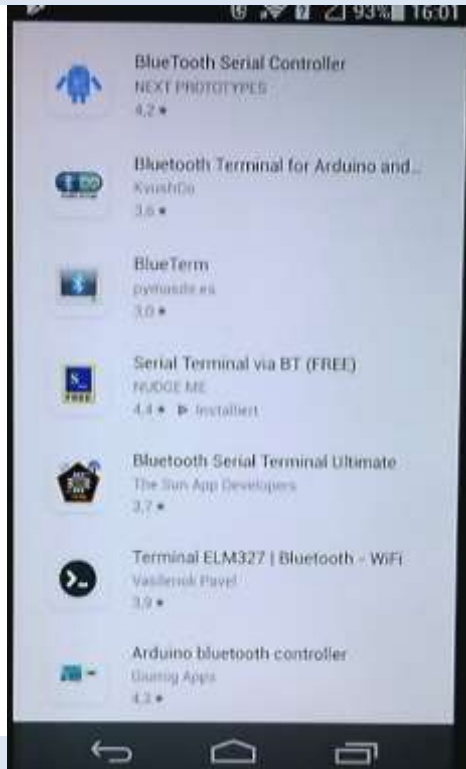
```
void blueEvent() {  
  while (blueSerial.available()) {  
    // Listen for Bluetooth commands like:  
    // "Time:12:35:00"  
    char inChar = (char)blueSerial.read(); // Cast incoming byte to char  
    Serial.print(inChar);                 // Echo the character to monitor  
                                           // Store it in buffer[]; increase index  
                                           // Buffer[] is global  
    . . .  
    serialComplete = true;                // Setting flag input completed for loop()  
                                           // SerialComplete is global  
  }  
}
```

### Funktion

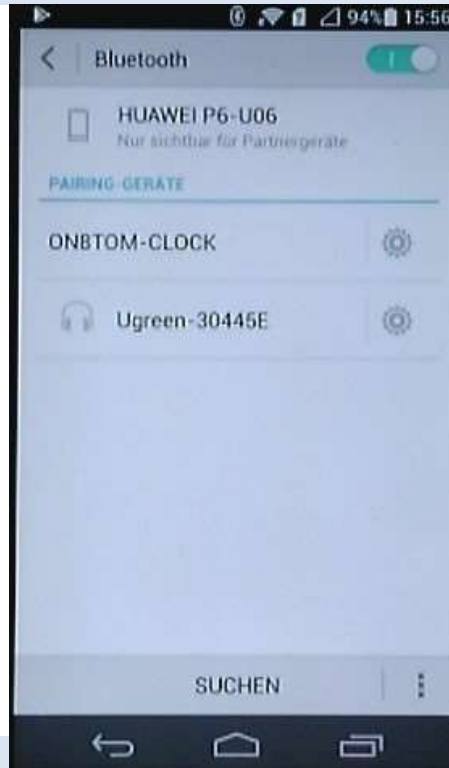
```
void processCommand(){  
  // Evaluate Bluetooth input  
  // Deviding Bluetooth input e.g. "Time:12:35:00" into tokens  
  char *arg0 = strtok(buffer, ":");      // Token "Time"  
  char *arg1 = strtok(NULL, ":");        // Token "12"  
  char *arg2 = strtok(NULL, ":");        // Token "35"  
  char *arg3 = strtok(NULL, ":");        // Token "00,"  
  . . .  
  RTC.setTime(hour_value, minute_value, second_value);  
  serialComplete = false;                // Event is done  
}
```

# Smartphone-App

## Play Store



## Device koppeln



## BT Terminal Verbindung aktivieren



# Smartphone-App

Befehl: „time:15:46:00“

Rückmeldung vom Arduino



---

## Taster Minuten

Idee	Seitlich am Wortuhr-Gehäuse finden sich 2 Taster. Durch kurzes betätigen wird die RTC-Uhrzeit verstellt.
Oberer Taster	Um eine Minute vor.
Unterer Taster	Um eine Minute zurück.

---

# Sketch

Sketch auf Anfrage von: [H39@email.de](mailto:H39@email.de)

---