

Arduino Uhr

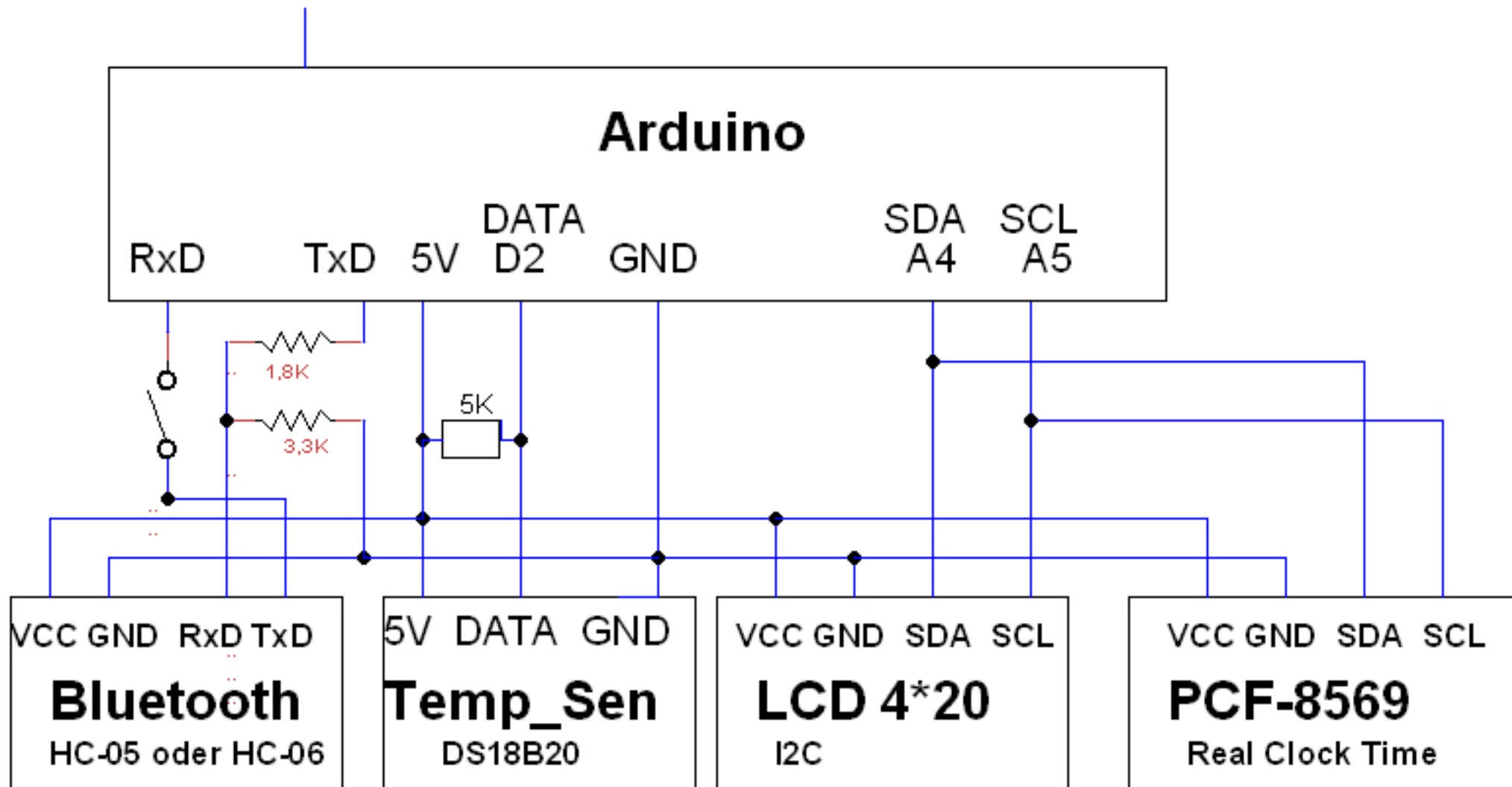


Warum dieses Projekt

- Wie war noch mal die UTC Zeit ? Eine Stunde vor oder zurück? Und im Sommer 2 Stunden ?
- Wieviel Uhr ist es in Peru oder Tokyo ?
- aktuelle „Küchen“ und die UTC-Zeit
- Temperaturanzeige

Schaltplan

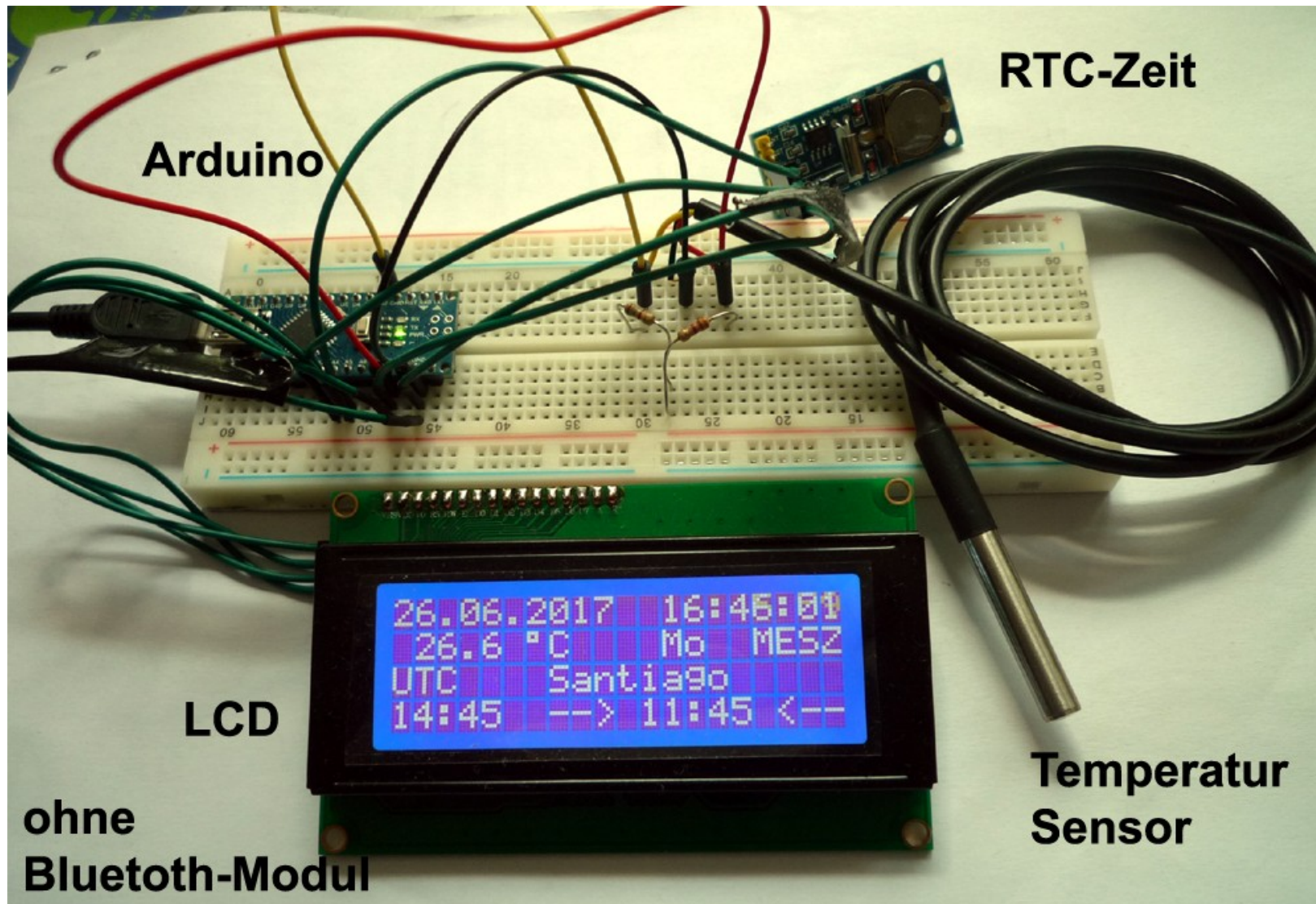
USB-Schnitt zum Rechner.



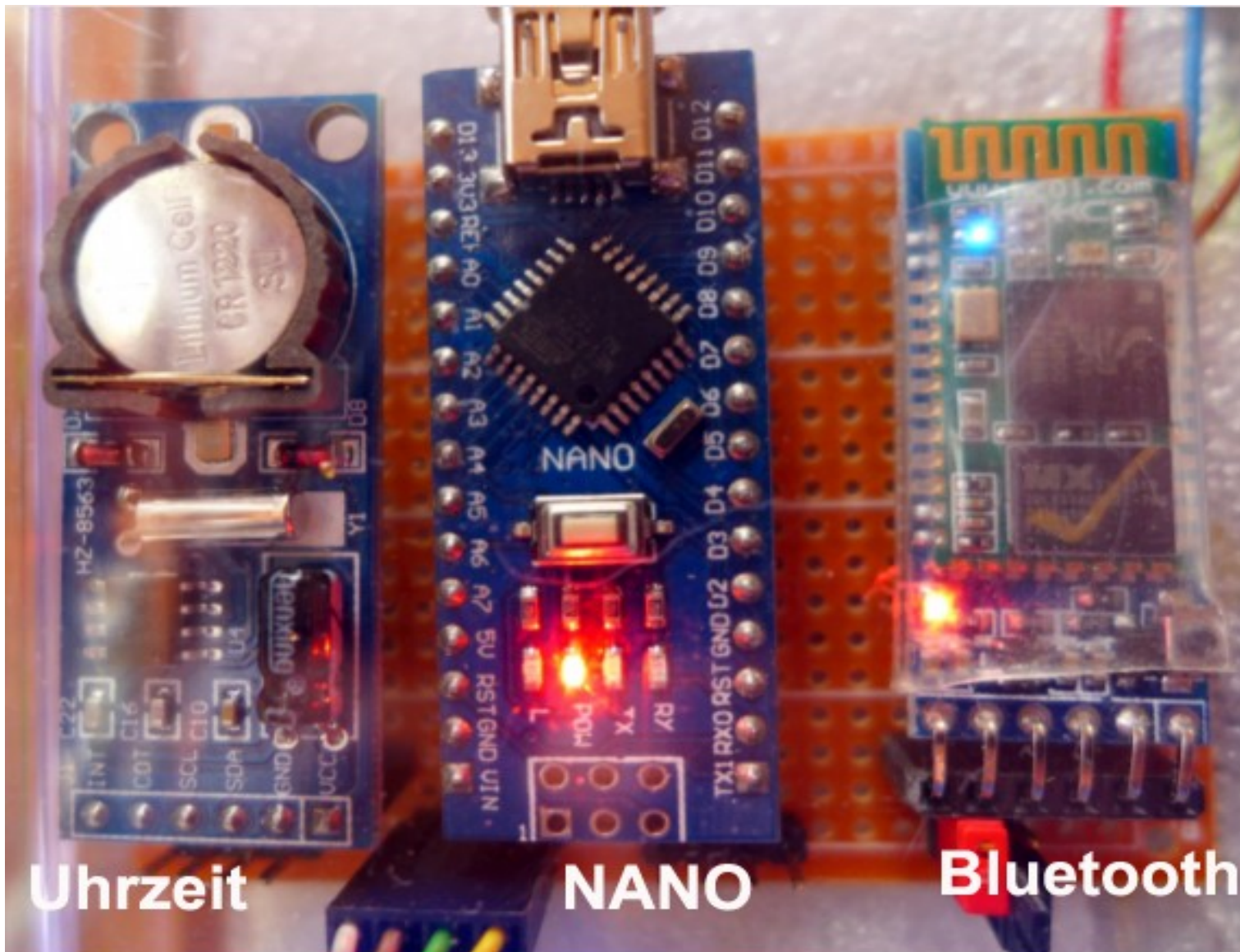
Das Bluetooth-Modul und der Temperatursensor kann natürlich entfallen, falls diese Funktionen nicht benötigt werden.

Beim Hochladen des Programmes muß der Schalter RxD-Bluetooth offen sein

Aufbau ohne BT



Aufbau mit BT-Modul



Abspeichern und Aufrufen des Programmes

- In den Ordner Sketchbook, den Ordner *SFB_I2C_LCD_PCF8563* erstellen und die Dateien: *SFB_I2C_LCD_PCF8563* und *SFB_I2C_LCD_PCF8563_Bluetooth* hinein kopieren.
- Wichtig: Der Ordnername und der Programmname müssen gleich sein.
- Im IDE via Sketchbook den Ordner *SFB_I2C_LCD_PCF8563* wählen



Programmstruktur

Bibliotheken einbinden, hier: LCD; RTC, Temperatursensor

Adresszuweisungen, **Variablen definieren** wie: int led=13 oder int IstMonat, auch Tabellen für die Zeitzonen und 41 Städte

void setup(). Hier werden Grundeinstellungen vorgenommen. Serielle Schnittstelle, LCD Temperatursensor, Infoausgabe auf LCD

Die **void loop()** Methode wird im Gegensatz zum Setup ständig wiederholt. Hier wird der eigentliche Programmablauf geschrieben.

Unterprogramme, welche von void Loop() aufgerufen werden

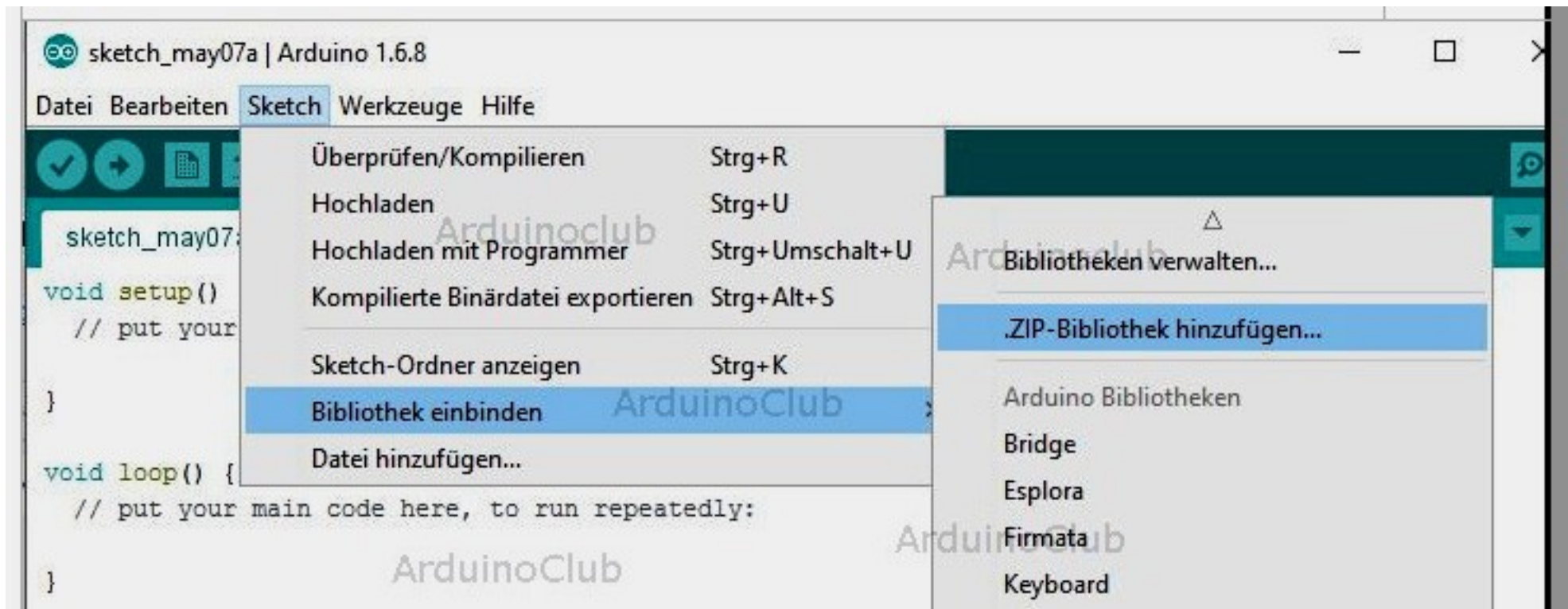
Bibliotheken

Hier werden die Bibliotheken eingebunden. Diese müssen vorher ins IDE kopiert werden.

```
//.....  
// Import Standard Library  
#include <ire.h>  
// Import Fremd Library  
#include <LiquidCrystal_I2C.h>  
#include <Rtc_Pcf8563.h>  
#include <OneWire.h>  
#include <DallasTemperature.h>
```


Einbinden Libraries ins Ardu

Navigieren Sie über das Menü „Sketch->Library importieren...->Add Library...“. Dann.zip importieren



- Oder direkt im Verzeichnis ablegen

- Rangierung von LCD, RTC und Temperatursensor.

Adresszuweisung und Signalleitungen werden hier festgelegt.

- Verwendete Variablen werden definiert bzw. sinnvoll vorbesetzt wie Orte der Zeitzone Normalzeit und Sommerzeit

-In der Arduino-Funktion „**SETUP**“ erfolgt die einmalige Initialisierung von der Seriellen-Schnittstelle (Terminal / Bluetoothmodul), LCD und des Temperatursensors DS1820. Danach erscheint der Infotext/Starttext (Eigenschaft des Gerat) kurz auf dem Display.

-

Programm testen und hochladen

- Im Sketchbook den Ordner SFB_I2C_LCD_PCF8563 wählen. Das Programm überprüfen (Haken im IDE) und falls keine Fehlermeldung hochladen (Pfeil) auf den Arduino.

Es erscheint ein Infotext/Starttext (Eigenschaft des Gerät) kurz auf dem Display. Datum und Uhrzeit müssen noch eingestellt werden.



Arduino-Funktion „LOOP“

Mit der Arduino-Funktion „LOOP“ wird der zyklische Aufruf der Hauptfunktionen bewerkstelligt.

Im folgenden:

1. Datum und Zeit an den RTC übergeben, also Uhr stellen
2. Aktuelles Datum und Zeit aus den RTC holen
3. Aktuelles Datum und Zeit auf dem LCD ausgeben
4. Temperatur auf dem LCD ausgeben
5. Wochentag und MEZ / MESZ auf dem LCD ausgeben
6. Text „UTC“ und die Zeit in UTC auf dem LCD ausgeben
7. Zeitzone mit Ort und Zeit auf dem LCD ausgeben
8. Steuerung der Ausgabe Zeitzone im Sekundentakt, Raster oder feste Ortauswahl
9. Aktualisierung der Ausgabe auf dem LCD im Sekundentakt

Terminal zum Einstellen, mit Lupe im IDE öffnen

```
E
--- Info Terminal Start ---
Uhr mit UTC und Zeitzonen
h Eingabe Zeit Stunde
m Eingabe Zeit Minute
s Eingabe Zeit Sekunde
T Eingabe Datum Tag
V Eingabe Datum Wochentag (0-6 / So-Sa)
M Eingabe Datum Monat
J Eingabe Datum Jahr
t Ausgabe Temperatur in Â°C
+ Positive Weiterschaltung Ort, wenn Raster = 0
- Negative Weiterschaltung Ort, wenn Raster = 0
R r Einstellung Raster für Anzeige Ort (0 bis 45)
) o Anzeige Ort, direkte Anwahl über Nummer (0-45)
z Aktuelle Zeit und Datum
E e Übernahme Zeit und Datum
P Information Steuerbefehle
-- Info Terminal Ende ---

temperatur = 0.0 Â°C
Uhrzeit einstellen 21.43.22h
h21<enter>E<enter>: 21.00.00
m43<enter>E<enter>: 21.43.00
s22<enter>E<enter>: 21.43.22h
```

Dateneingabe Uhrzeit etc.

Über die folgende Funktion werden Werte mittels des Terminal (Lupe im IDE) oder dem Bluetooth-Modul an das Programm übergeben.

Mittels Buchstaben und Zahlenfolgen werden Übergaben an das Programm gesteuert. Die zulässigen Eingabefolgen können mit dem Zeichen „?“ eingesehen werden.

Zeit einlesen: h Stunde m Minute s Tag

Datum: Tag, W Wochentag M (0=Sonntag, ...6=Samstag)

M=Monat, J=Jahr, Temperatur= t,

Weiterschaltung Ort + - Raster Einstellung R, Anzeige Ort 0 bis 45

Kontrolle Zeit und Datum K, Aktuelle Zeit/Datum Z

Zeit/Datum übernehmen E

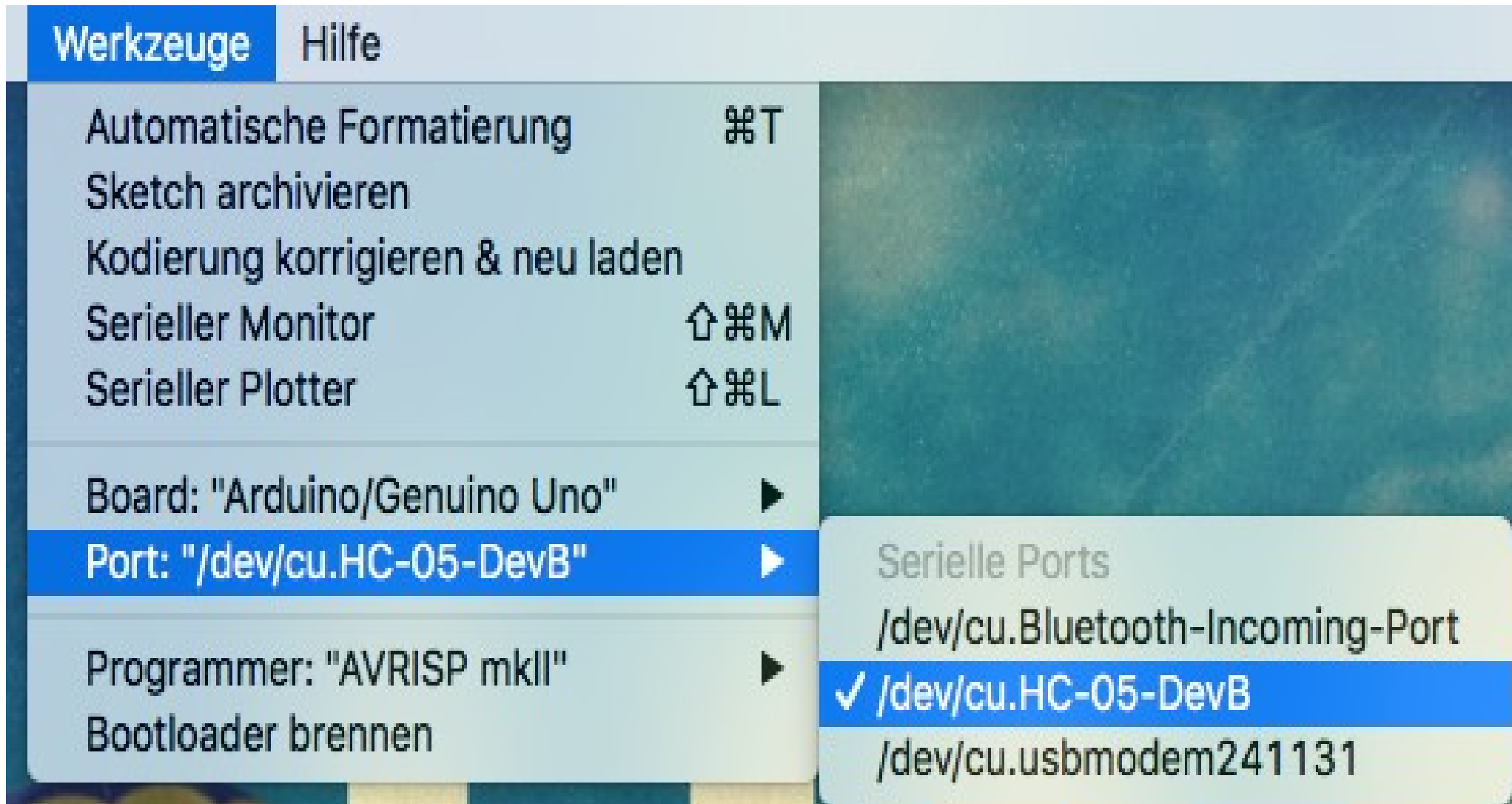
Beispiel: für 21 Uhr: h21<enter> E <enter>, jetzt ist im Display 21 Uhr

43 Minuten: m43<enter>E<enter>, jetzt ist im Display 21.43 Uh

Dateneingabe via Bluetooth, erst via BT-Manager Verbindung zum BT-Modul aufbauen

The image shows a Linux desktop environment. A window titled "Bluetooth-Geräte" is open, displaying a list of Bluetooth devices. The first device is "HC-05", which is unknown ("Unbekannt") and has the MAC address "98:D3:32:10:D7:01". Below the device list, the text "/dev/rfcomm0" is displayed, with a black arrow pointing downwards from it. At the bottom of the screen, a terminal window shows the message "Serieller Anschluß verbunden mit /dev/rfcomm0" and network statistics: "↑ 25.73 KB 16.00 B/s ↓ 50.97 KB 27.00 B/s".

Im IDE den Port zum BT.....wählen Terminal via „Lupe“



BT via Smartphone