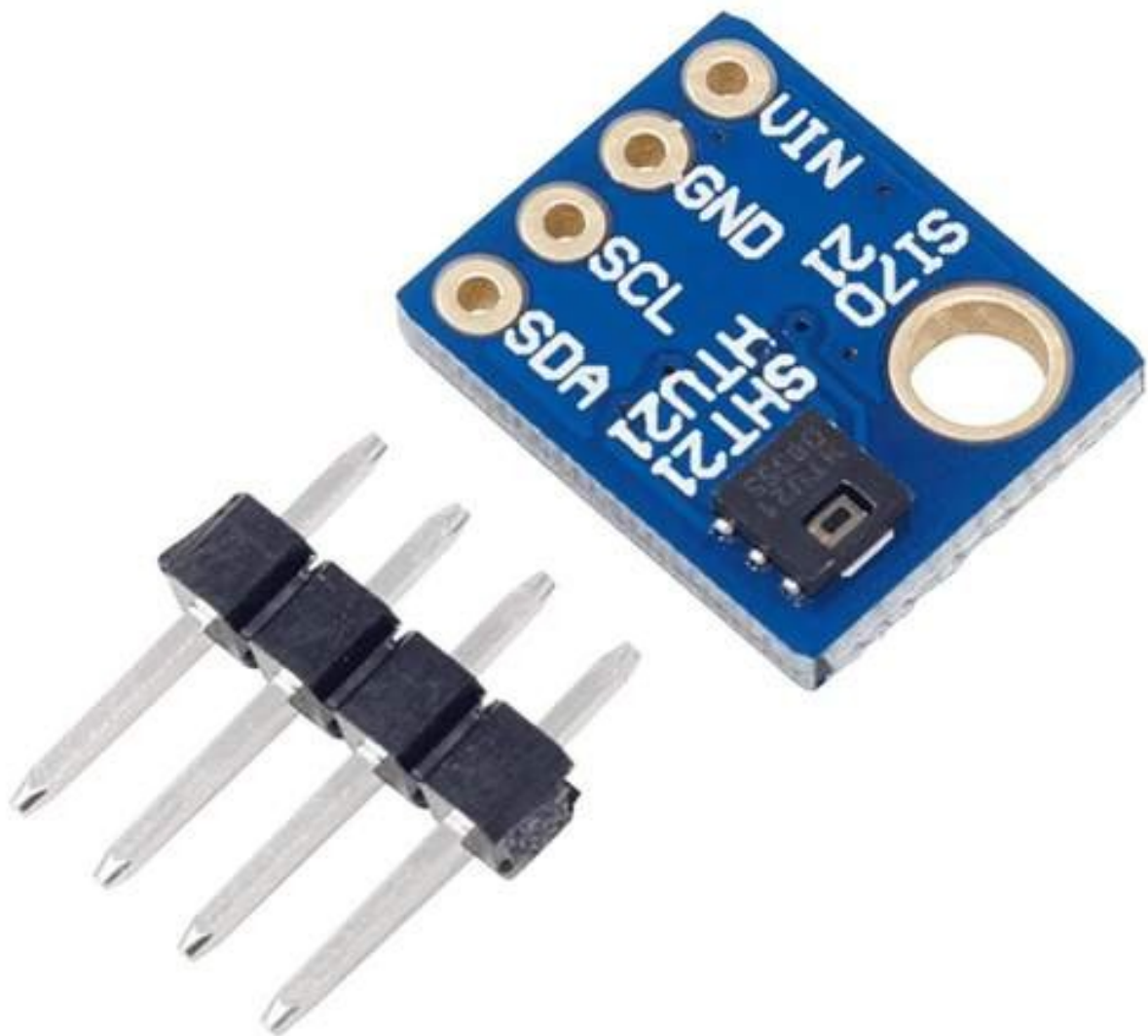


# AZ-Delivery

## Willkommen!

Vielen Dank, dass sie sich für unser Temperatur- und Luftfeuchtesensor-Modul vom Typ "GY-21" von AZ-Delivery entschieden haben. In den nachfolgenden Seiten werden wir Ihnen erklären wie Sie das Gerät einrichten und nutzen können.

**Viel Spaß!**





## Inhaltsverzeichnis

Einführung.....	3
Technische Daten.....	4
Pinbelegung.....	5
Einrichtung der Arduino IDE.....	6
Einrichtung des Raspberry Pi und Python.....	10
Verbindung des Moduls mit dem Microcontroller.....	11
Sketch-Beispiel.....	12
Verbindung des Moduls mit dem Raspberry Pi.....	18
Aktivieren der I2C-Schnittstelle.....	19
Libraries und Tools für Python.....	20
Python-Skript.....	22



## Einführung

Das Herzstück des GY-21-Moduls ist der Sensorchip "HTU21D(F)" von MEAS. Der Sensor ist ein einfach zu bedienender, hochgenauer digitaler Feuchte- und Temperatursensor. Dieser Sensor liefert kalibrierte, linearisierte Signale in digitalem Format.

Er kann vielfach verwendet werden, z.B. in Innen- und Außen-Wetterstationen, Umwelt-/Datenzentren, Kfz-Klimaanlagen und Entfeuchtungsgeräten, Haushaltsgeräten, medizinischen Geräten, Druckern, Luftbefeuchtern, Mobiltelefonen, Tablets und Vieles mehr.

Die Auflösung des digitalen Feuchtesensors kann auf Befehl geändert werden. Die Auflösung der digitalen Temperatursausgabe von 12 bis 14 Bit und die Auflösung der digitalen Luftfeuchtigkeitsausgabe von 8 bis 12 Bit.

Ein Feature des Moduls ist der energiesparende Betrieb, der einen sehr geringen Stromverbrauch ermöglicht.



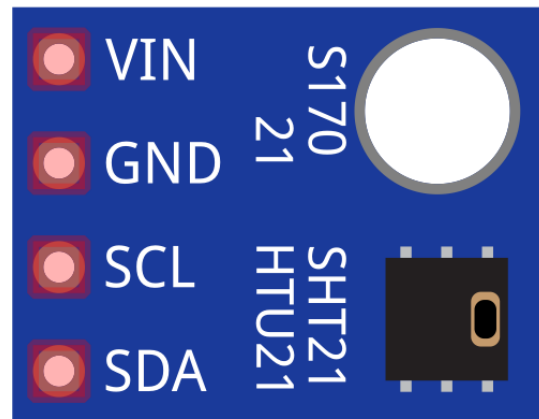
## Technische Daten

» Betriebsspannungsbereich:	3.3V bis 5V DC
» Stromverbrauch:	150µA
» Temperaturbereich:	-40°C bis 85°C
» Temperaturgenauigkeit:	±1.0°C
» Feuchtigkeitsbereich:	0 bis 100% RH
» Feuchtigkeitsgenauigkeit:	±3%
» Kommunikationsschnittstelle:	I2C
» Dimensionen:	9 x 11 x 2mm [0.35 x 0.43 x 0.078inch]

## Pinbelegung

Das GY-21 Feuchte- und Temperatursensormodul hat vier Pins. Die Pinbelegung ist wie folgt:

**Power Supply - VIN**  
**Ground - GND**  
**I2C Serial Clock Link - SCL**  
**I2C Serial Data Link - SDA**



## Einrichtung der Arduino IDE

Falls Sie die Arduino IDE noch nicht installiert haben, gehen Sie auf:

<https://www.arduino.cc/en/Main/Software>

und laden Sie die Installationsdatei für Ihre Betriebssystemplattform herunter.

### Download the Arduino IDE



The screenshot shows the Arduino IDE download page. On the left, there is a teal circle with a white infinity symbol containing a minus and a plus sign. To its right, the text reads: **ARDUINO 1.8.9**, followed by a description of the IDE as open-source software written in Java, and a note that it can be used with any Arduino board. On the right side of the page, there are links for downloading the IDE for different operating systems: Windows (installer and ZIP file), Windows app (with a 'Get' button), Mac OS X, and Linux (32 bits, 64 bits, ARM 32 bits, and ARM 64 bits). At the bottom right, there are links for Release Notes, Source Code, and Checksums (sha512).

Für *windows-User*: Doppelklicken Sie auf die heruntergeladene ".exe"-Datei und folgen den Anweisungen im Installationsfenster.

# Az-Delivery

Für *Linux*-User, laden Sie eine Datei mit der Erweiterung `".tar.xz"` herunter, die Sie entpacken müssen. Wenn Sie sie extrahieren, gehen Sie in das extrahierte Verzeichnis und öffnen Sie das Terminal in diesem Verzeichnis. Sie müssen zwei `".sh"`-Skripte ausführen, das erste heißt `"arduino-linux-setup.sh"` und das zweite `"install.sh"`.

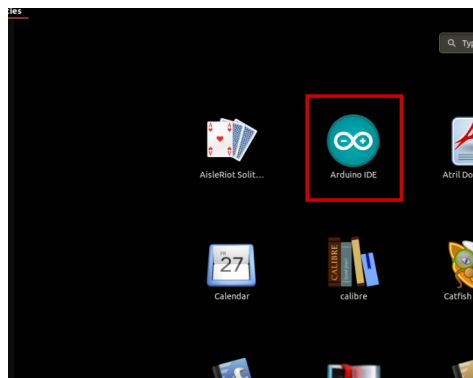
Um das erste Skript im Terminal (im extrahierten Verzeichnis) auszuführen, führen Sie den folgenden Befehl aus:

**`sh arduino-linux-setup.sh user_name`**

**`user_name`** - ist der Name des Superusers im Linux-Betriebssystem. Danach werden Sie aufgefordert, das Passwort für den Superuser einzugeben. Warten Sie einige Minuten, bis das Skript alles abgeschlossen hat.

Nach der Installation des ersten Skripts führen Sie das zweite Skript mit dem Namen `"install.sh"` aus. Führen Sie im Terminal (im extrahierten Verzeichnis) den folgenden Befehl aus: **`sh install.sh`**

Nach der Installation dieser Skripte, gehen Sie zu *All Apps*, um die installierte *Arduino IDE* zu finden.



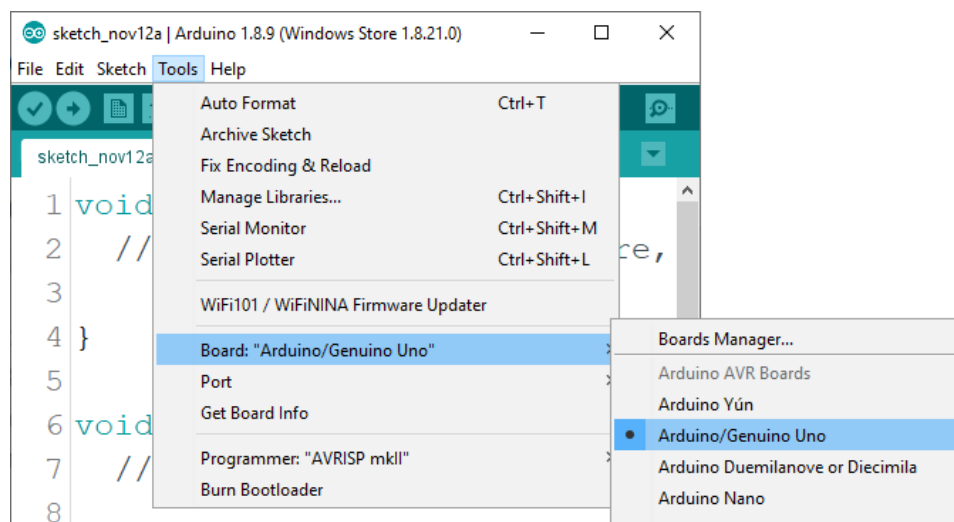
# Az-Delivery

Fast alle Betriebssysteme werden mit einem vorinstallierten Texteditor ausgeliefert (z.B. Windows mit Notepad, Linux Ubuntu mit Gedit, Linux Raspbian mit Leafpad usw.). Alle diese Texteditoren sind für den Zweck des eBooks vollkommen in Ordnung.

Als nächstes sollten Sie überprüfen, ob Ihr PC das Microcontroller board erkennen kann. Öffnen Sie die frisch installierte *Arduino IDE* und gehen Sie zu:

*Tools > Board > {your board name here}*

*{your board name here}* sollte das *Arduino/Genuino Uno* sein, wie Sie dem Bild unten entnehmen können:



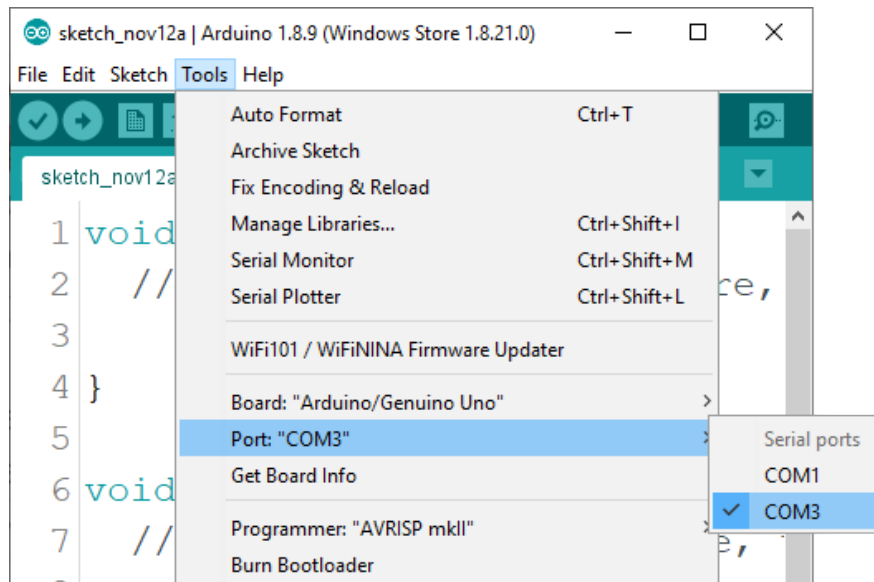
Der Port, an den das Microcontroller-Board angeschlossen ist, muss ausgewählt werden. Gehe zu: *Tools > Port > {port name goes here}* und wenn das Microcontroller-Board an den USB-Port angeschlossen ist, ist der Portname im Drop-down Menü wie auf dem





vorherigen Bild zu sehen.

Wenn die Arduino-IDE unter Windows verwendet wird, lauten die Portnamen wie folgt:



Für *Linux* Benutzer, ist zum Beispiel der Portname `/dev/ttyUSBx`, wobei x für eine ganze Zahl zwischen 0 und 9 steht.



## Einrichtung von Raspberry Pi und Python

Für den Raspberry Pi müssen Sie zuerst das Betriebssystem installieren und dann alles so einrichten, dass Sie es im "*Headless*"-Modus verwenden können. Im "*Headless*"-Modus können Sie eine Verbindung zur Raspberry Pi herstellen, ohne dass Sie einen PC-Bildschirm, eine Maus oder eine Tastatur benötigen. Das Einzige, was Sie für diesen Modus benötigen, sind ein Raspberry Pi, eine Stromversorgung und eine Internetverbindung. All dies wird ausführlich in dem kostenlosen eBook "*Raspberry Pi Quick Startup Guide*" erklärt.

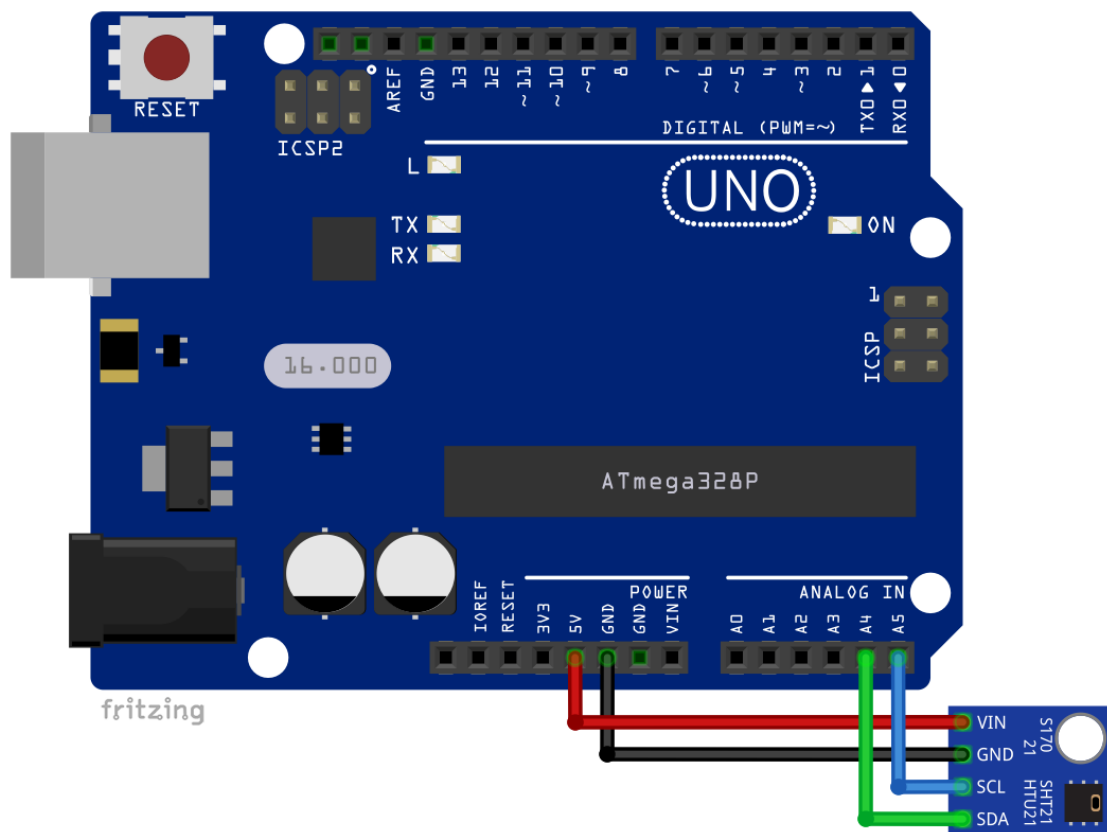
Es kann unter folgendem Link gefunden werden:

<https://www.az-delivery.de/products/raspberry-pi-kostenfreies-e-book?ls=en>

Das Betriebssystem *Rasbian* wird vorinstalliert mit *Python* ausgeliefert.

## Verbindung des Moduls mit dem Microcontroller

Verbinden Sie das GY-21 Feuchte- und Temperatursensormodul mit dem Microcontroller wie unten abgebildet:



Modul Pin	Microcontroller Pin	Farbe des Drahts
VIN	5V	<b>Roter Draht</b>
GND	GND	<b>Schwarzer Draht</b>
SCL	A5	<b>Blauer Draht</b>
SDA	A4	<b>Grüner Draht</b>

## Sketch-Beispiel

```
#include <Wire.h>
const int ADDRESS = 0x40;
double temperature, humidity;
void sensor_init(const int addr) {
    Wire.begin(); delay(100);
    Wire.beginTransmission(addr);
    Wire.endTransmission();
}
double read_temperature(const int addr) {
    double temperature;
    int low_byte, high_byte, raw_data;
    /**Send command of initiating temperature measurement**/
    Wire.beginTransmission(addr);
    Wire.write(0xE3);
    Wire.endTransmission();
    /**Read data of temperature**/
    Wire.requestFrom(addr, 2);
    if (Wire.available() <= 2) {
        high_byte = Wire.read();
        low_byte = Wire.read();
        high_byte = high_byte << 8;
        raw_data = high_byte + low_byte;
    }
    temperature = (175.72 * raw_data) / 65536;
    temperature = temperature - 46.85;
    return temperature;
}
```

# Az-Delivery

```
double read_humidity(const int addr) {
    double humidity, raw_data_1, raw_data_2;
    int low_byte, high_byte, container;
    /**Send command of initiating relative humidity measurement**/
    Wire.beginTransaction(addr);
    Wire.write(0xE5);
    Wire.endTransmission();
    /**Read data of relative humidity**/
    Wire.requestFrom(addr, 2);
    if(Wire.available() <= 2) {
        high_byte = Wire.read();
        container = high_byte / 100;
        high_byte = high_byte % 100;
        low_byte = Wire.read();
        raw_data_1 = container * 25600;
        raw_data_2 = high_byte * 256 + low_byte;
    }
    raw_data_1 = (125 * raw_data_1) / 65536;
    raw_data_2 = (125 * raw_data_2) / 65536;
    humidity = raw_data_1 + raw_data_2;
    humidity = humidity - 6;
    return humidity;
}

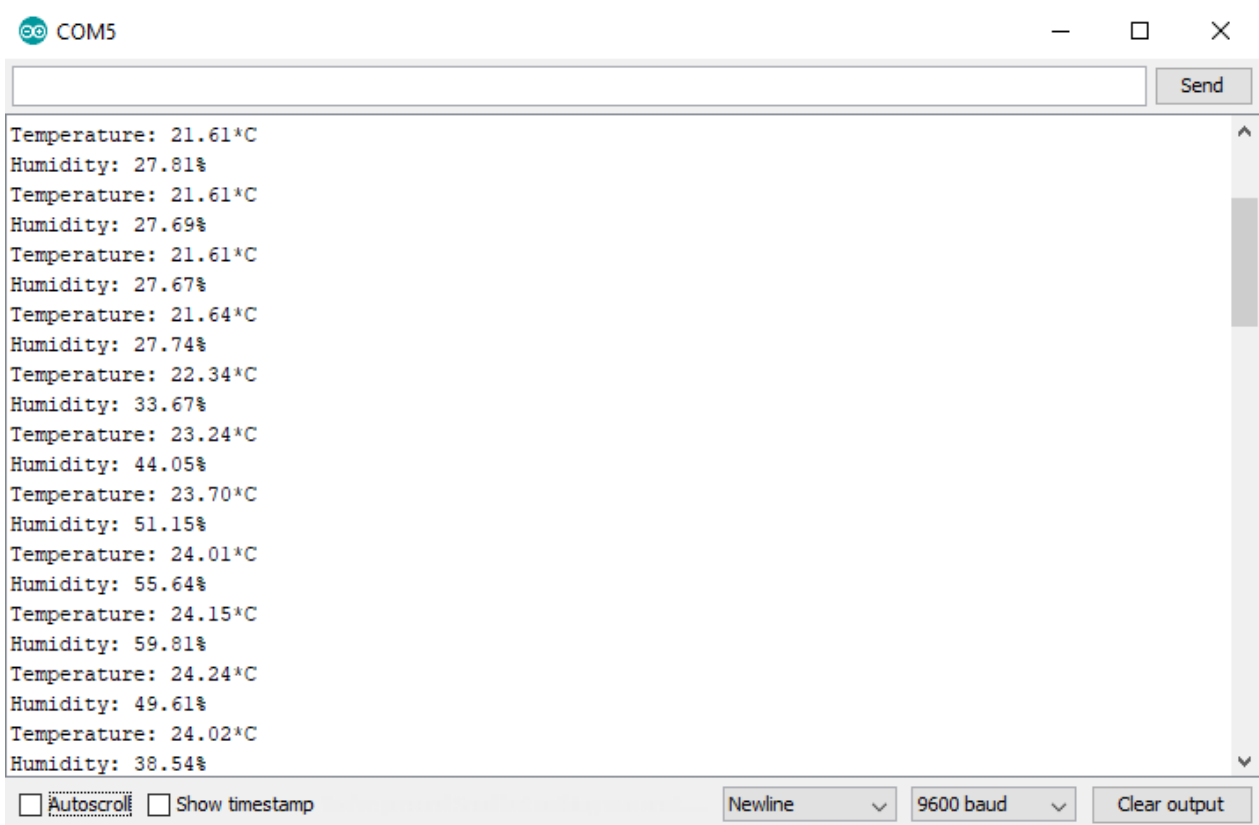
void setup() {
    Serial.begin(9600);
    sensor_init(ADDRESS);
}
```

# Az-Delivery

```
void loop() {  
    temperature = read_temperature(ADDRESS);  
    humidity = read_humidity(ADDRESS);  
  
    Serial.print("Temperature: ");  
    Serial.print(temperature);  
    Serial.println("*C");  
  
    Serial.print("Humidity: ");  
    Serial.print(humidity);  
    Serial.println("%");  
  
    delay(1000);  
}
```

# Az-Delivery

Laden Sie den Sketch in den Microcontroller hoch und öffnen Sie den Serial Monitor (*Tools > Serial Monitor*). Die Ausgabe sollte wie folgt aussehen:



The screenshot shows the Serial Monitor window for COM5. The window has a title bar with standard Windows controls. Below the title bar is a text input field and a 'Send' button. The main area displays a list of sensor readings. At the bottom, there are checkboxes for 'Autoscroll' and 'Show timestamp', and dropdown menus for 'Newline' and '9600 baud', along with a 'Clear output' button.

```
Temperature: 21.61°C  
Humidity: 27.81%  
Temperature: 21.61°C  
Humidity: 27.69%  
Temperature: 21.61°C  
Humidity: 27.67%  
Temperature: 21.64°C  
Humidity: 27.74%  
Temperature: 22.34°C  
Humidity: 33.67%  
Temperature: 23.24°C  
Humidity: 44.05%  
Temperature: 23.70°C  
Humidity: 51.15%  
Temperature: 24.01°C  
Humidity: 55.64%  
Temperature: 24.15°C  
Humidity: 59.81%  
Temperature: 24.24°C  
Humidity: 49.61%  
Temperature: 24.02°C  
Humidity: 38.54%
```



# Az-Delivery

Der Sketch beginnt mit der Einbeziehung einer Library, die für die I2C-Kommunikation verwendet wird.

Als nächstes wird eine Konstante namens *ADDRESS* erstellt. Diese Konstante stellt die I2C-Adresse des Moduls dar.

Danach werden zwei doppelte Variablen erstellt: *temperature* und *humidity*. Diese beiden Variablen werden zum Speichern und Anzeigen der aus dem Modul gelesenen Daten verwendet.

Dann werden drei Funktionen erstellt: *sensor\_init()*, *read\_temperature()* und *read\_humidity()*. Alle drei Funktionen haben ein Argument, das die I2C-Adresse des Moduls darstellt.

Die erste Funktion, *sensor\_init()*, gibt keinen Wert zurück. Sie wird verwendet, um die I2C-Kommunikation zwischen Microcontroller und dem Modul zu initialisieren.

Die zweite Funktion, *read\_temperature()*, gibt einen Doppelwert zurück. Der Rückgabewert stellt die aus dem Modul gelesenen Temperaturdaten dar. Alles, was diese Funktion tut, ist den Sensor auszulesen und die Temperaturdaten zurückzugeben. Der Algorithmus dieser Funktion wird in diesem eBook nicht näher behandelt.

# Az-Delivery

Die dritte Funktion, `read_humidity()`, gibt einen doppelten Wert zurück. Der Rückgabewert stellt die aus dem Modul gelesenen Feuchtigkeitsdaten dar. Diese Funktion liest lediglich den Sensor und gibt die Feuchtigkeitsdaten aus. Der Algorithmus dieser Funktion wird in diesem eBook nicht näher behandelt.

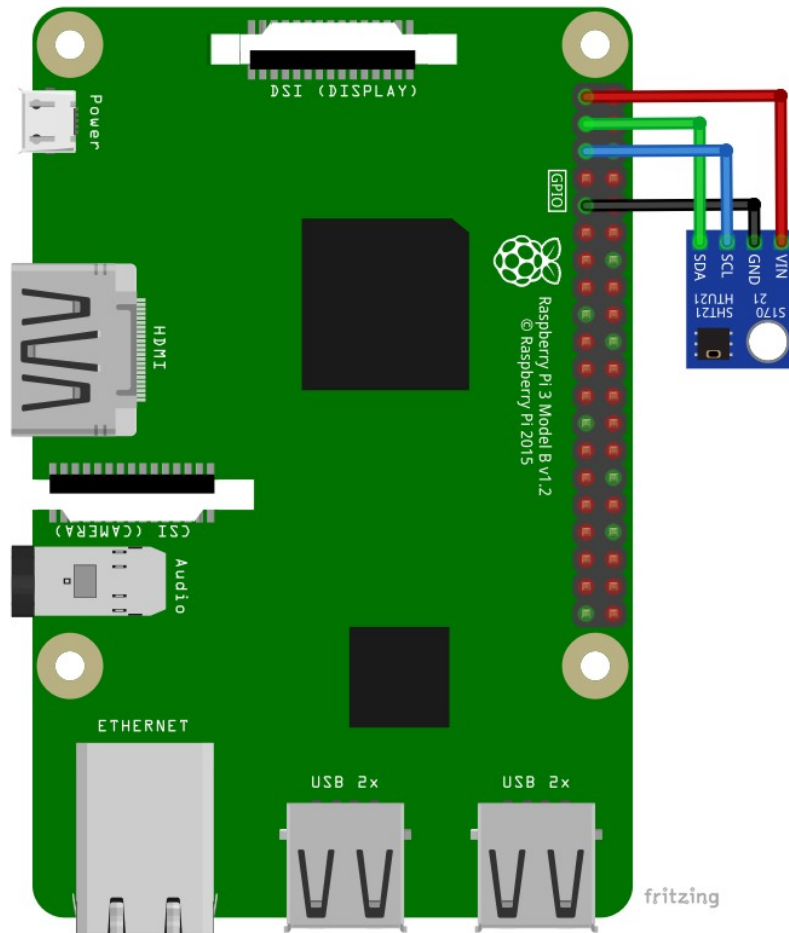
In der `setup()` Funktion, wird die serielle Kommunikation mit einer Baudrate von `9600bps` gestartet und die `sensor_init()` Funktion wird ausgeführt.

In der `loop()` Funktion, werden Sensorendaten gelesen und in den Variablen `temperature` and `humidity` gespeichert. Danach werden die Daten im Serial Monitor angezeigt.

Zwischen zwei loops der `loop()` Funktion liegt eine Verzögerung von einer Sekunde.

## Verbindung des Moduls mit dem Raspberry Pi

Verbinden Sie das GY-21 Sensormodul mit dem Raspberry Pi, wie unten abgebildet:

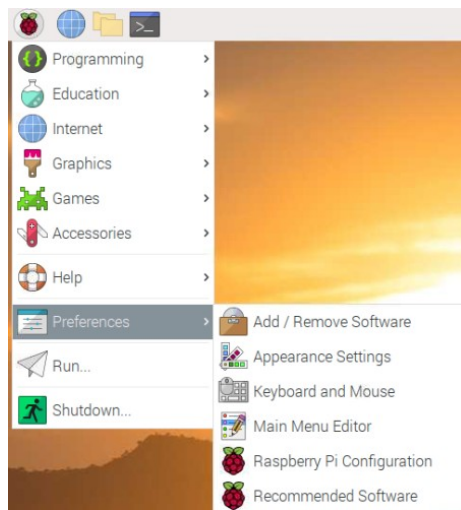


Modul Pin	Raspberry Pi Pin	Physischer Pin	Farbe des Drahts
VIN	3.3V	1	Roter Draht
SDA	GPIO2	3	Grüner Draht
SCL	GPIO3	5	Blauer Draht
GND	GND	9	Schwarzer Draht

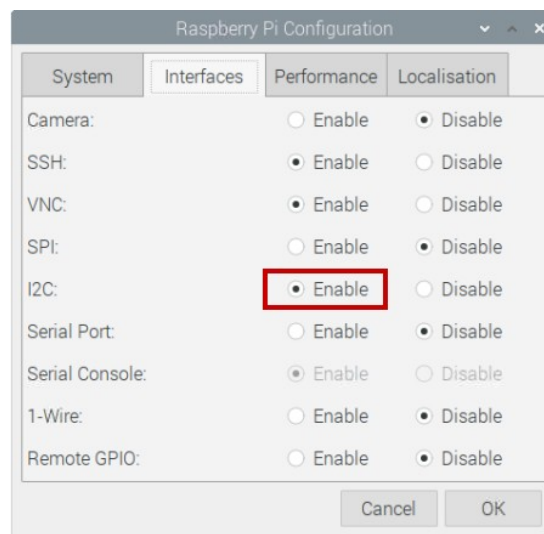
## Aktivieren der I2C-Schnittstelle

Um den Bildschirm mit einer Raspberry Pi zu verwenden, muss zunächst die SPI-Schnittstelle in Raspbian aktiviert werden. Gehen Sie dafür zu:

*Application Menu > Preferences > Raspberry Pi Configuration*



Als nächstes öffnen Sie die Registerkarte *Interfaces/Schnittstelle*, stellen Sie die SPI-Radiobuttons auf *Ok* und aktivieren Sie es, wie auf dem folgenden Bild gezeigt:





## Libraries und Tools für Python

Um das Modul mit dem Raspberry Pi zu verwenden, wird empfohlen, eine externe Library dafür herunterzuladen. Die Library, die in diesem eBook verwendet wird, heißt `luma.led_matrix`. Um die Library zu benutzen, ist es notwendig, die Abhängigkeiten zu installieren. Öffnen Sie dazu das Terminal und führen Sie den folgenden Befehl aus:

Bevor Sie die Library verwenden können, müssen folgende Tools installiert werden: *i2c-tools*, *setuptools* and *python3-pip*. Sollten diese Libraries nicht installiert sein, öffnen Sie das Terminal und führen nacheinander die folgenden Befehle aus:

```
sudo apt-get update
```

```
sudo apt-get upgrade -y
```

```
sudo apt-get install -y python3-pip setuptools i2c-tools
```

Die folgenden Python-Libraries sollten ebenfalls installiert sein: *RPI.GPIO* und *adafruit-blinka*. Sollten diese Libraries nicht installiert sein, öffnen Sie das Terminal und führen nacheinander die folgenden Befehle aus:

```
sudo pip3 install RPI.GPIO
```

```
sudo pip3 install adafruit-blinka
```

Nun kann die Library für das GY-21 installiert werden. Öffnen Sie das Terminal und führen Sie folgenden Befehl aus:

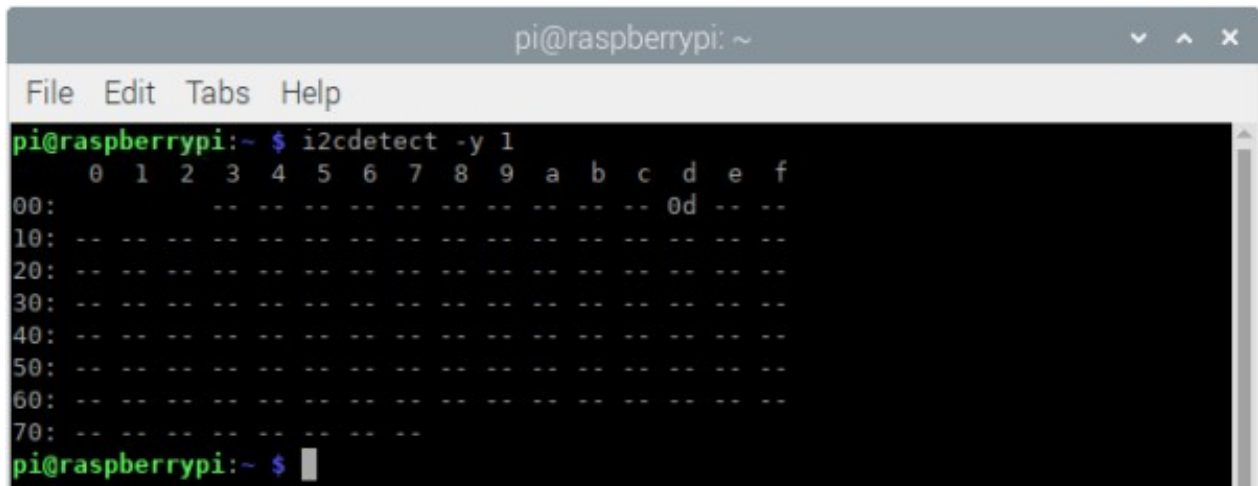
```
sudo pip3 install adafruit-circuitpython-htu21d
```

# Az-Delivery

Überprüfen Sie die I2C-Adresse des Moduls, indem Sie folgenden Befehl ausführen:

```
i2c detect -y 1
```

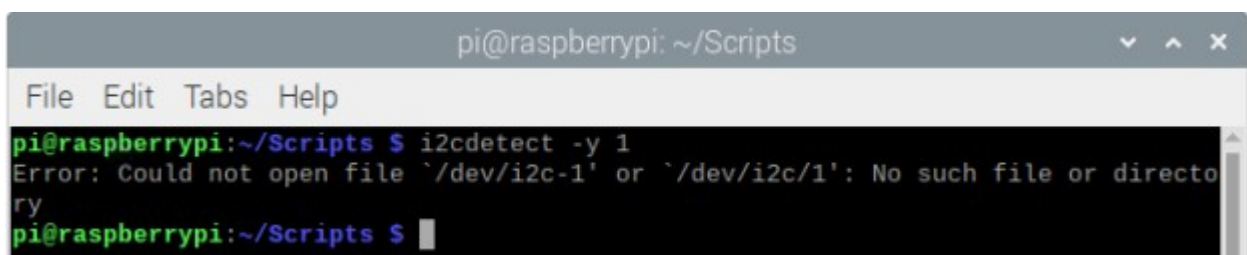
Die Ausgabe sollte wie folgt aussehen:



```
pi@raspberrypi: ~  
File Edit Tabs Help  
pi@raspberrypi:~ $ i2cdetect -y 1  
    0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f  
00:  -- -- -- -- -- -- -- -- -- -- -- 0d -- --  
10:  -- -- -- -- -- -- -- -- -- -- -- -- -- --  
20:  -- -- -- -- -- -- -- -- -- -- -- -- -- --  
30:  -- -- -- -- -- -- -- -- -- -- -- -- -- --  
40:  -- -- -- -- -- -- -- -- -- -- -- -- -- --  
50:  -- -- -- -- -- -- -- -- -- -- -- -- -- --  
60:  -- -- -- -- -- -- -- -- -- -- -- -- -- --  
70:  -- -- -- -- -- -- -- -- -- -- -- -- -- --  
pi@raspberrypi:~ $
```

Wobei *0x0d* die I2C-Adresse des Moduls ist.

Sollte das I2C-interface auf dem Raspberry Pi nicht aktiviert, und der vorherige Befehl ausgeführt sein, wird Ihnen folgende Fehlermeldung angezeigt:



```
pi@raspberrypi: ~/Scripts  
File Edit Tabs Help  
pi@raspberrypi:~/Scripts $ i2cdetect -y 1  
Error: Could not open file `/dev/i2c-1' or `/dev/i2c/1': No such file or directory  
pi@raspberrypi:~/Scripts $
```

# Az-Delivery

## Python-Skript

```
import time
import board
import busio
from adafruit_htu21d import HTU21D

ds = u'\xb0' # degree sign

i2c = busio.I2C(board.SCL, board.SDA)
sensor = HTU21D(i2c)

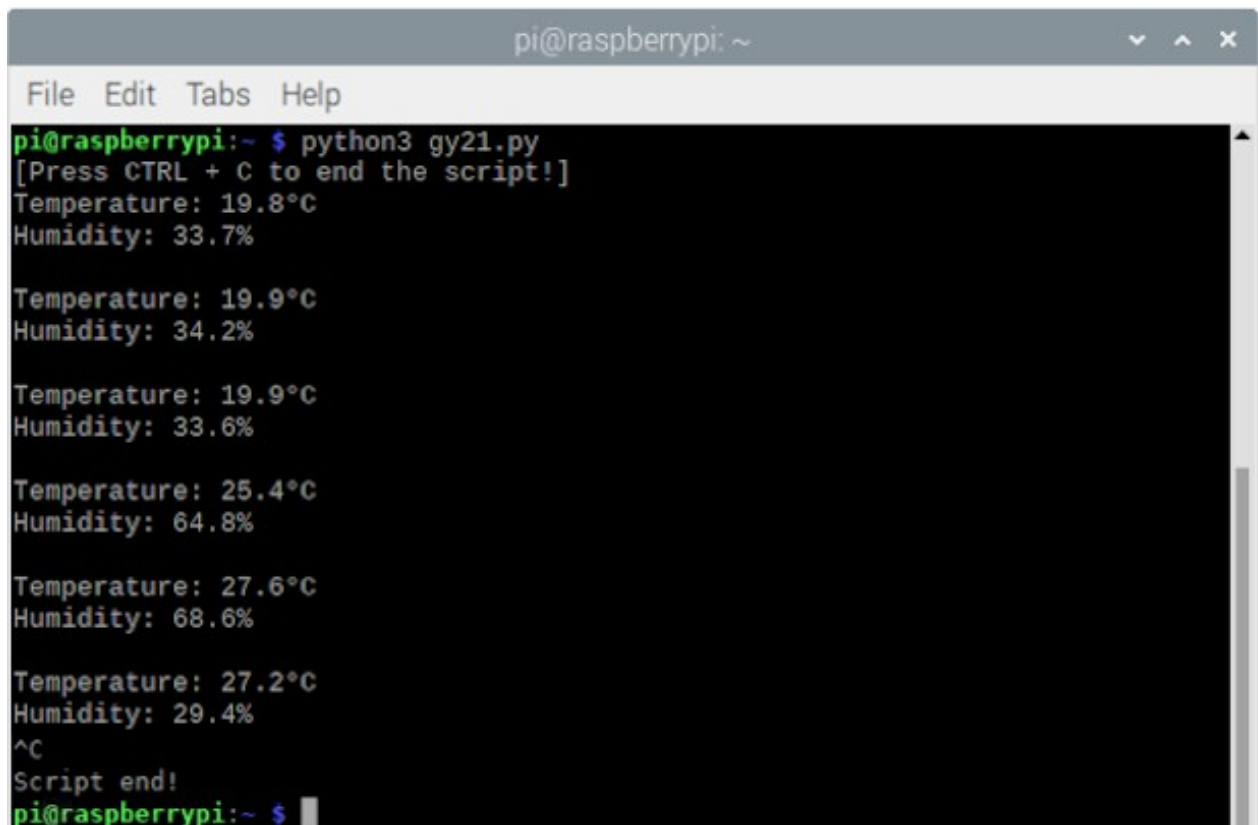
print('Press CTRL + C to end the script!')
try:
    while True:
        temp = sensor.temperature
        humidity = sensor.relative_humidity
        print('Temperature: {:.1f}{}C'.format(temp, ds))
        print('Humidity: {:.1f}%\n'.format(humidity))
        time.sleep(1)

except KeyboardInterrupt:
    print('\nScript end!')
```

# Az-Delivery

Speichern Sie das Skript unter dem Namen `gy21.py`. Um das Skript auszuführen, öffnen Sie das Terminal in dem Verzeichnis, in dem die Skripten gespeichert sind, und führen Sie den folgenden Befehl aus:  
**python3 gy21.py**

Die Ausgabe sollte wie folgt aussehen:



```
pi@raspberrypi:~ $ python3 gy21.py
[Press CTRL + C to end the script!]
Temperature: 19.8°C
Humidity: 33.7%

Temperature: 19.9°C
Humidity: 34.2%

Temperature: 19.9°C
Humidity: 33.6%

Temperature: 25.4°C
Humidity: 64.8%

Temperature: 27.6°C
Humidity: 68.6%

Temperature: 27.2°C
Humidity: 29.4%
^C
Script end!
pi@raspberrypi:~ $
```

To stop the script press CTRL + C on the keyboard.

**Sie haben es geschafft. Sie können jetzt unser Modul  
für Ihre Projekte nutzen.**





Jetzt sind Sie dran! Entwickeln Sie Ihre eigenen Projekte und Smart-Home Installationen. Wie Sie das bewerkstelligen können, zeigen wir Ihnen unkompliziert und verständlich auf unserem Blog. Dort bieten wir Ihnen Beispielskripte und Tutorials mit interessanten kleinen Projekten an, um schnell in die Welt der Mikroelektronik einzusteigen. Zusätzlich bietet Ihnen auch das Internet unzählige Möglichkeiten, um sich in Sachen Mikroelektronik weiterzubilden.

**Falls Sie nach weiteren hochwertigen Produkten für Arduino und Raspberry Pi suchen, sind Sie bei AZ-Delivery Vertriebs GmbH goldrichtig. Wir bieten Ihnen zahlreiche Anwendungsbeispiele, ausführliche Installationsanleitungen, E-Books, Bibliotheken und natürlich die Unterstützung unserer technischen Experten.**

<https://az-delivery.de>

Viel Spaß!

Impressum

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