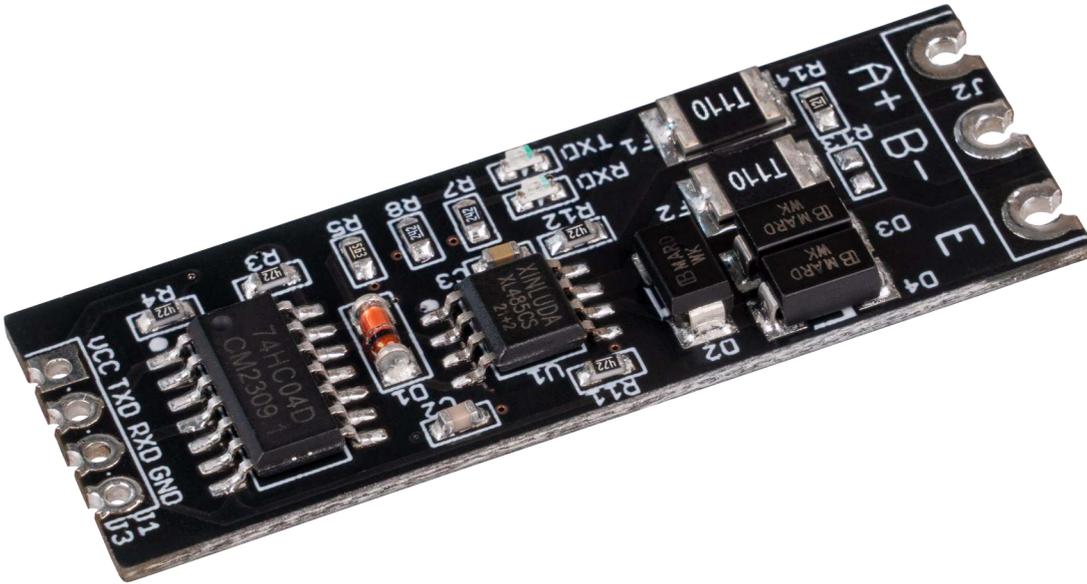


TTL ZU RS485 KONVERTER

COM-TTL-RS485



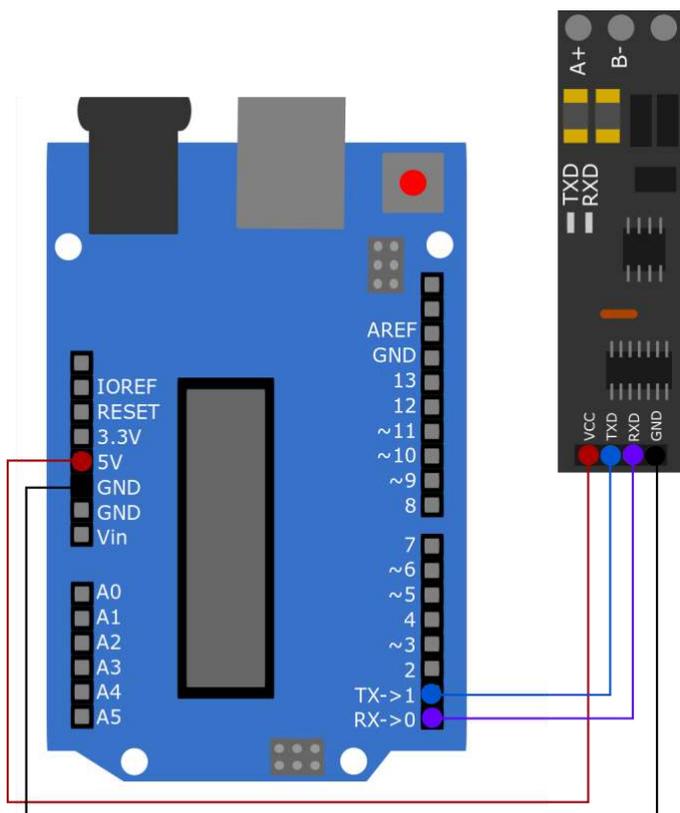
1. ALLGEMEINE INFORMATIONEN

Sehr geehrter Kunde*in,
vielen Dank, dass Sie sich für unser Produkt entschieden haben. Im Folgenden zeigen wir Ihnen, was bei der Inbetriebnahme und der Verwendung zu beachten ist.

Sollten Sie während der Verwendung unerwartet auf Probleme stoßen, so können Sie uns selbstverständlich gerne kontaktieren.

2. VERWENDUNG MIT DEM ARDUINO

1. Anschluss



Arduino	Konverter
5V	VCC
Pin 1	TXD
Pin 0	RXD
GND	GND

2. Codebeispiel

Im Folgenden finden Sie ein Codebeispiel, mit dem Sie das Gerät testen können. Hierzu benötigen Sie jedoch ein Endgerät für den RS485 Anschluss, um eine Kommunikation zu ermöglichen. Wir haben in diesem Beispiel den USB-RS485 Adapter (SBC-TTL-RS485) von Joy-it verwendet. Beachten Sie dabei, dass A+ an A und B- an B angeschlossen wird.

Verwenden Sie für das Codebeispiel die Arduino IDE. Beachten Sie dabei, dass unter **Werkzeuge** der richtige **Port** und das richtige **Board** ausgewählt sind.

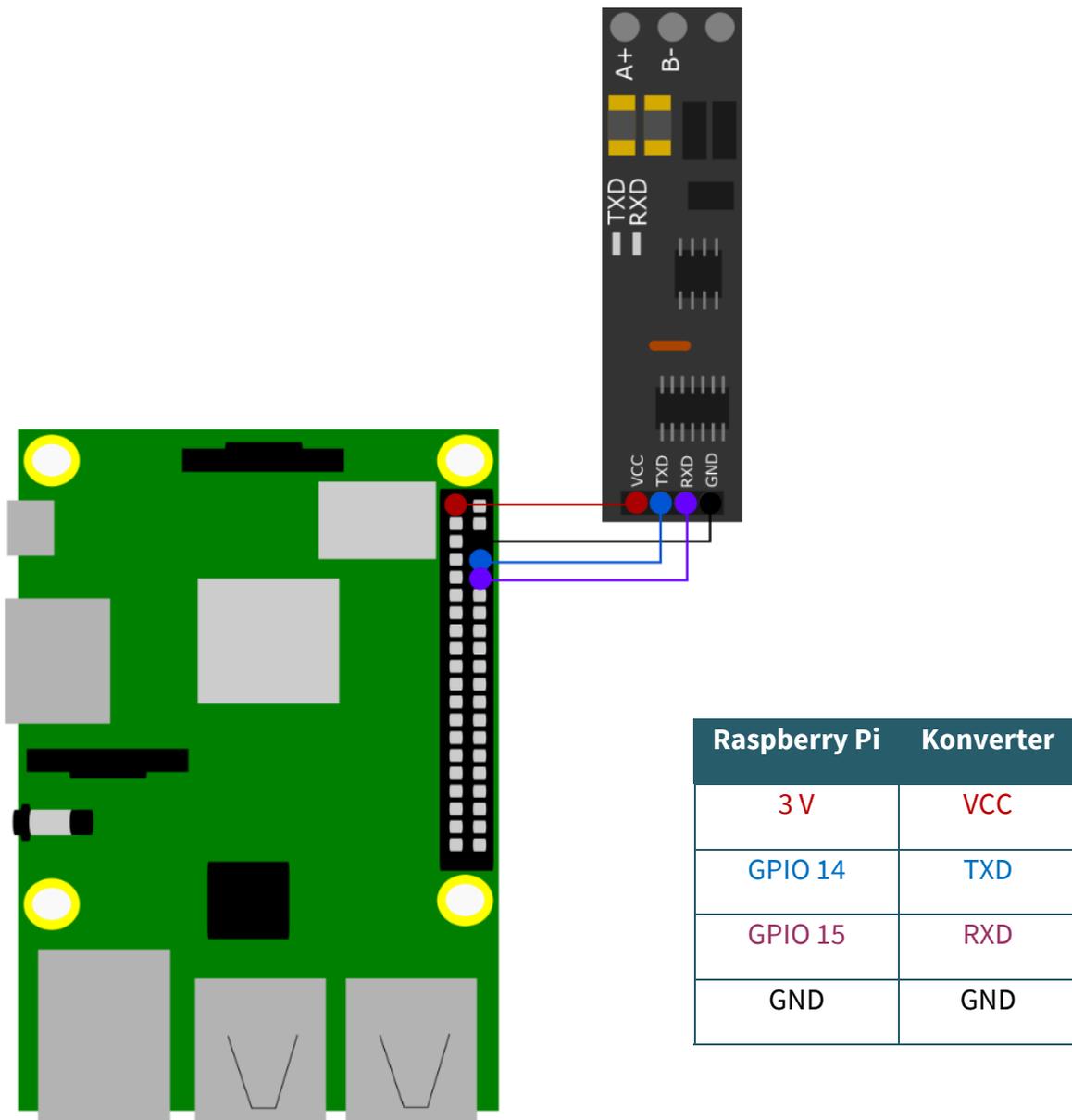
Kopieren Sie den folgenden Code und klicken Sie auf **Hochladen**.

```
String testString; //Deklarieren einer Variable
void setup() {
  Serial.begin(9600); // Aktivieren der seriellen Kommunikation
}
void loop() {
  Serial.println("Please enter Test-String: ");
  while (Serial.available()==0) {} // Warten auf Eingabe
  testString=Serial.readString(); //Auslesen der Eingabe
  Serial.println("The Test-String received is: " + testString);
  Serial.println("-----");
  delay(2000);
}
```

Nun können Sie über ein Terminalprogramm wie zum Beispiel HTerm, mit Ihrem Arduino kommunizieren. Dabei müssen Sie darauf achten, dass Sie die Baudrate auf dieselbe Geschwindigkeit stellen wie bei Ihrem Arduino, in diesem Beispiel 9600. Die vom Arduino empfangenen Daten werden Ihnen im seriellen Monitor angezeigt.

3. VERWENDUNG MIT DEM RASPBERRY PI

1. Anschluss

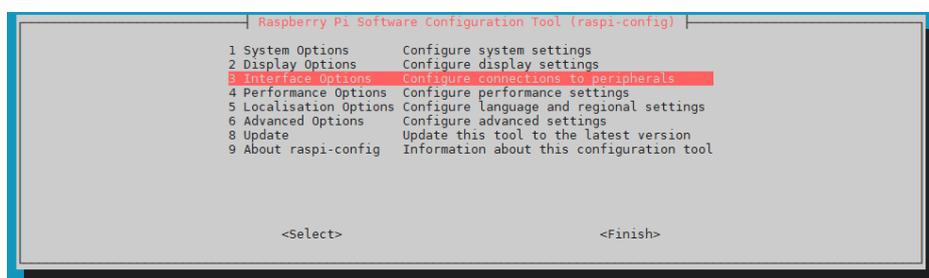


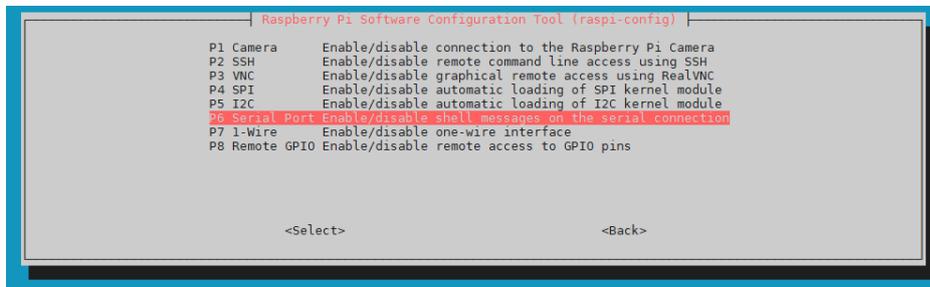
2. Codebeispiel

Um das folgende Codebeispiel nutzen zu können, müssen Sie die serielle Kommunikation des Raspberry Pis aktivieren. Führen Sie dazu den folgenden Befehl aus, um die Konfiguration aufzurufen.

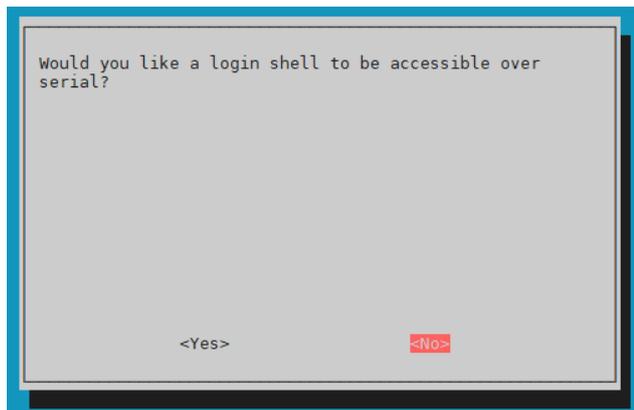
```
sudo raspi-config
```

Wählen Sie dort unter **3 Interface Options** → **P6 Serial Port** aus.

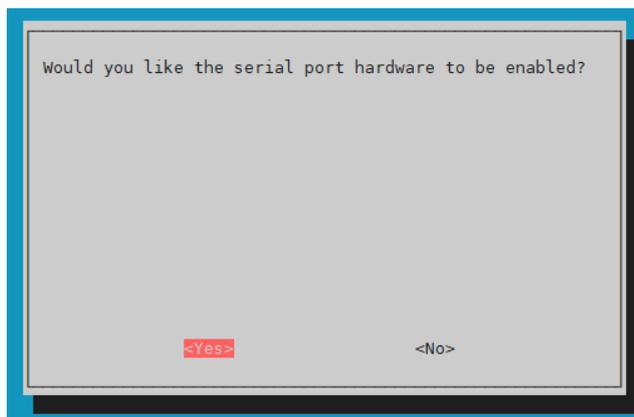




Verneinen Sie dort die Frage *Would you like a login shell to be accessible over serial?* .



Stimmen Sie der folgenden Frage zu.



Sie haben nun erfolgreich die serielle Kommunikation ermöglicht. Starten Sie nun Ihr Gerät einmal neu um die Änderungen zu übernehmen. Vergewissern Sie sich, dass die serielle Bibliothek auf Ihrem Raspberry Pi installiert ist. Führen Sie dazu den folgenden Befehl in Ihrer Konsole aus.

```
sudo apt-get install python-serial
```

Für das Codebeispiel benötigen Sie jedoch ein Endgerät für den RS485 Anschluss, um eine Kommunikation zu ermöglichen. Wir haben in diesem Beispiel den USB-RS485 Adapter (SBC-TTL-RS485) von Joy-it verwendet. Beachten Sie dabei, dass A+ an A und B- an B angeschlossen wird.

Sie können mit dem folgenden Befehl eine Datei erstellen. Fügen Sie dort den darauffolgenden Code ein.

```
nano ttlToRS485.py
```

```

import serial
import time

# Oeffnen des seriellen Ports
port = serial.Serial("/dev/ttyS0", baudrate=9600, timeout=3.0)
print ("Starting test...")
port.write(bytes("Please enter Test-String: \n", 'ascii'))
# Nachricht wird gesendet
while True:
    testString = port.readline() # Es wird etwas ausgelesen
    testString = testString.decode('utf-8') # in String umwandeln

    if testString != "":
        print("The Test-String received is: " ,testString) # Ausgabe in Konsole
        print("-----")
        time.sleep(2)
        port.write(bytes("Please enter Test-String: \n", 'ascii'))

```

Sie können nun das Skript mit dem folgenden Befehl ausführen.

```
python3 ttlToRS485.py
```

Nun können Sie über ein Terminalprogramm wie zum Beispiel HTerm, mit Ihrem Raspberry Pi kommunizieren. Dabei müssen Sie darauf achten, dass Sie die Baudrate auf dieselbe Geschwindigkeit stellen wie in Ihrem Pythoncode, in diesem Beispiel 9600. Die vom Raspberry Pi empfangenen Daten werden Ihnen in der Konsole angezeigt.

4. SONSTIGE INFORMATIONEN

Unsere Informations- und Rücknahmepflichten nach dem Elektroggesetz (ElektroG)



Symbol auf Elektro- und Elektronikgeräten:

Diese durchgestrichene Mülltonne bedeutet, dass Elektro- und Elektronikgeräte **nicht** in den Hausmüll gehören. Sie müssen die Altgeräte an einer Erfassungsstelle abgeben. Vor der Abgabe haben Sie Altbatterien und Altakkumulatoren, die nicht vom Altgerät umschlossen sind, von diesem zu trennen.

Rückgabemöglichkeiten:

Als Endnutzer können Sie beim Kauf eines neuen Gerätes, Ihr Altgerät (das im Wesentlichen die gleiche Funktion wie das bei uns erworbene neue erfüllt) kostenlos zur Entsorgung abgeben. Kleingeräte bei denen keine äußere Abmessungen größer als 25 cm sind können unabhängig vom Kauf eines Neugerätes in haushaltsüblichen Mengen abgeben werden.

Möglichkeit Rückgabe an unserem Firmenstandort während der Öffnungszeiten:

SIMAC Electronics GmbH, Pascalstr. 8, D-47506 Neukirchen-Vluyn

Möglichkeit Rückgabe in Ihrer Nähe:

Wir senden Ihnen eine Paketmarke zu mit der Sie das Gerät kostenlos an uns zurücksenden können. Hierzu wenden Sie sich bitte per E-Mail an Service@joy-it.net oder per Telefon an uns.

Informationen zur Verpackung:

Verpacken Sie Ihr Altgerät bitte transportsicher, sollten Sie kein geeignetes Verpackungsmaterial haben oder kein eigenes nutzen möchten kontaktieren Sie uns, wir lassen Ihnen dann eine geeignete Verpackung zukommen.

5. SUPPORT

Wir sind auch nach dem Kauf für Sie da. Sollten noch Fragen offen bleiben oder Probleme auftauchen stehen wir Ihnen auch per E-Mail, Telefon und Ticket-Supportsystem zur Seite.

E-Mail: service@joy-it.net

Ticket-System: <http://support.joy-it.net>

Telefon: +49 (0)2845 98469 – 66 (10 - 17 Uhr)

Für weitere Informationen besuchen Sie unsere Website:

www.joy-it.net