

TTL ZU RS485 KONVERTER

COM-TTL-RS485

1. ALLGEMEINE INFORMATIONEN

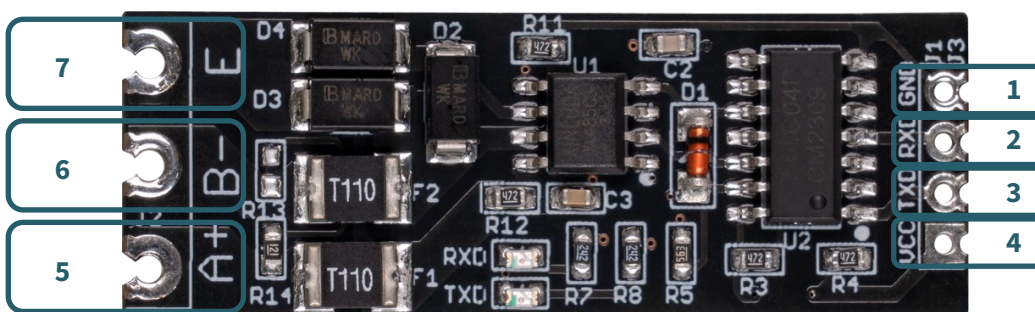
Sehr geehrter Kunde*in,
vielen Dank, dass Sie sich für unser Produkt entschieden haben. Im Folgenden zeigen wir Ihnen, was bei der Inbetriebnahme und der Verwendung zu beachten ist.

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

2. GRUNDLEGENDE INFORMATIONEN

Dieser UART TTL zu RS485 Konverter ermöglicht es Ihnen, Ihren Singleboardcomputer um eine RS485 Schnittstelle zu erweitern. Die Steuerung des Moduls ist leicht zu bedienen und erfolgt über die UART-Schnittstelle. Mit bis zu 1 km Übertragungreichweite sind auch Datenübertragungen mit weit auseinanderstehenden Geräten kein Problem. Seine Bauform und Größe machen das Modul hoch integrierbar. Aufgrund seiner hohen Störunanfälligkeit wird es besonders oft in industriellen Bereichen eingesetzt.

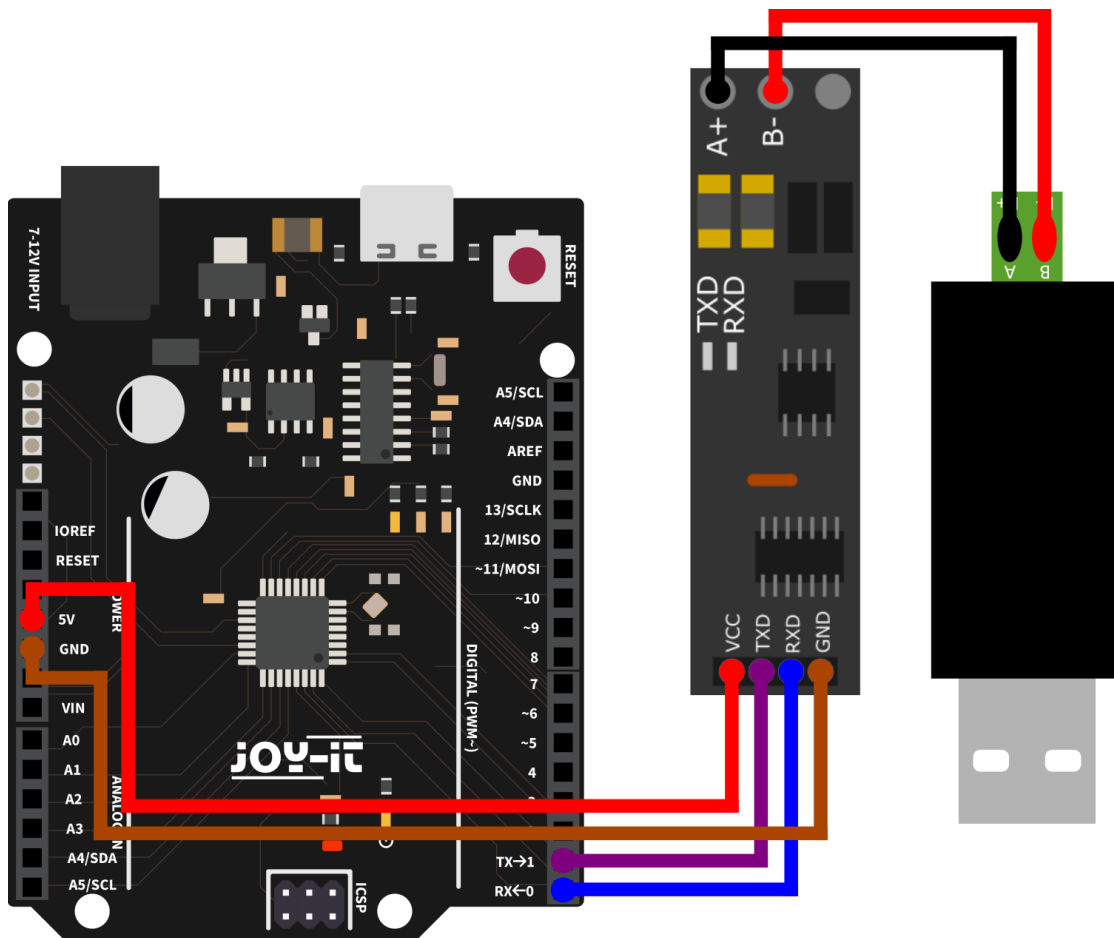
Die mit RX und TX gekennzeichneten LEDs auf dem Produkt sind Status-LEDs, die die serielle Kommunikation an dem entsprechenden Pin visuell anzeigen.



Ziffer	Funktion
1	GND → Masse Potential
2	RXD → UART Datenausgang
3	TXD → UART Dateneingang
4	VCC → Versorgungsspannung
5	A+ → RS485 nicht invertiertes Signal
6	B- → RS485 invertiertes Signal
7	E → RS485 Masse Potential

3. VERWENDUNG MIT DEM ARDUINO

1. Anschluss



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

2. Codebeispiel

Wir stellen ein Codebeispiel für die Verwendung mit dem Arduino zur Verfügung, das Sie [hier](#) herunterladen können. 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);
}
```

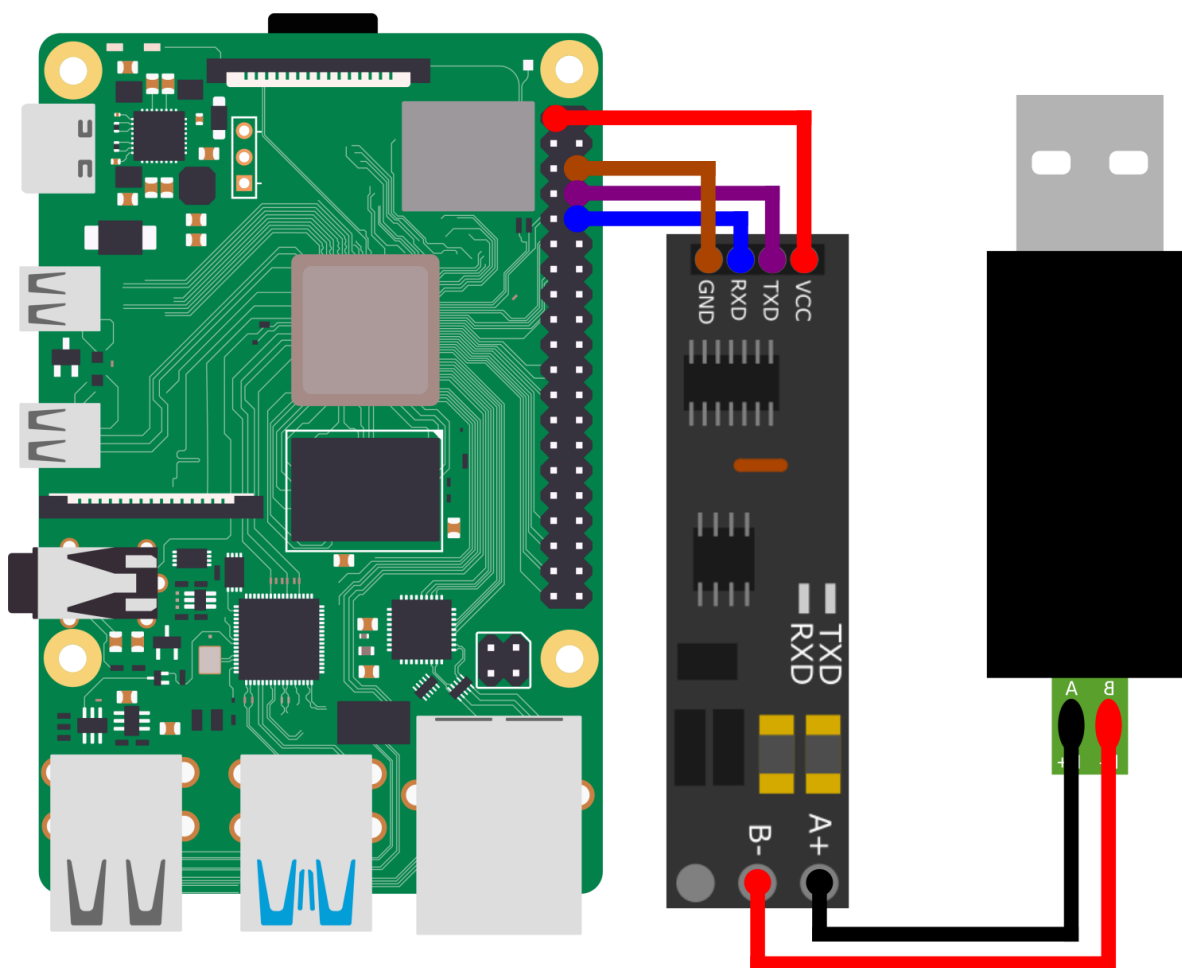
Nun können Sie über ein Terminalprogramm, wie zum Beispiel HTerm, mit Ihrem Arduino kommunizieren. Dabei müssen Sie darauf achten, dass Sie die Baudrate auf dieselbe Geschwindigkeit stellen wie bei Ihrem Arduino, in diesem Beispiel 9600. Die vom Arduino empfangenen Daten werden Ihnen im seriellen Monitor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

4. VERWENDUNG MIT DEM RASPBERRY PI



Diese Anleitung wurde unter Raspberry Pi OS Bookworm für den Raspberry Pi 4 und 5 geschrieben. Es wurde keine Überprüfung mit neueren Betriebssystemen oder Hardware durchgeführt.

1. Anschluss



Raspberry Pi

3V

GPIO 14

GPIO 15

GND

Konverter

VCC

TXD

RXD

GND

2. Codebeispiel

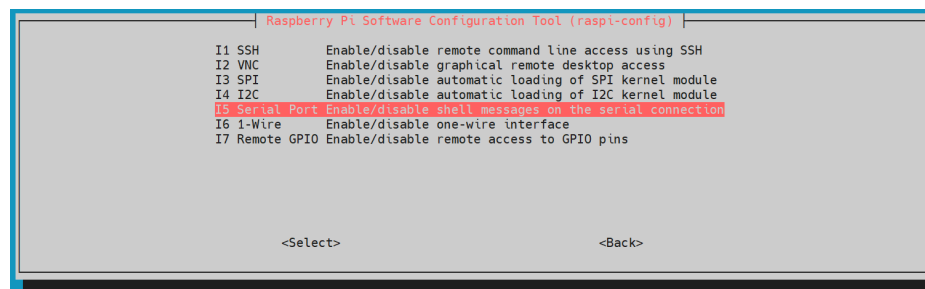
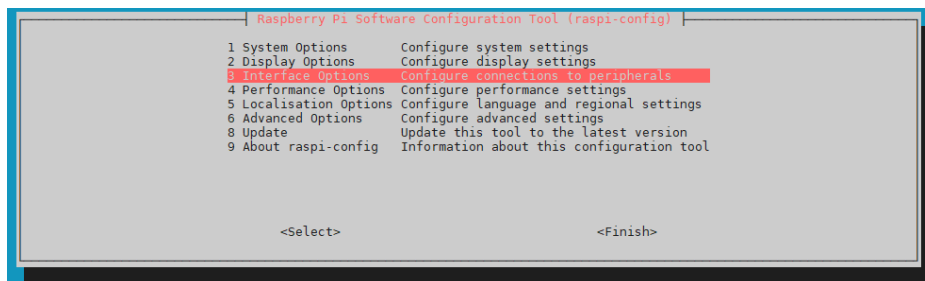
Wir stellen ein Codebeispiel für die Verwendung mit dem Raspberry Pi zur Verfügung, das Sie [hier](#) herunterladen können. Im Folgenden finden Sie ein Codebeispiel, mit dem Sie das Gerät testen können. Hierzu benötigen Sie jedoch ein Endgerät für den RS485 Anschluss, um eine Kommunikation zu ermöglichen. 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.

Um das folgende Codebeispiel nutzen zu können, müssen Sie die serielle Kommunikation des Raspberry Pis aktivieren.

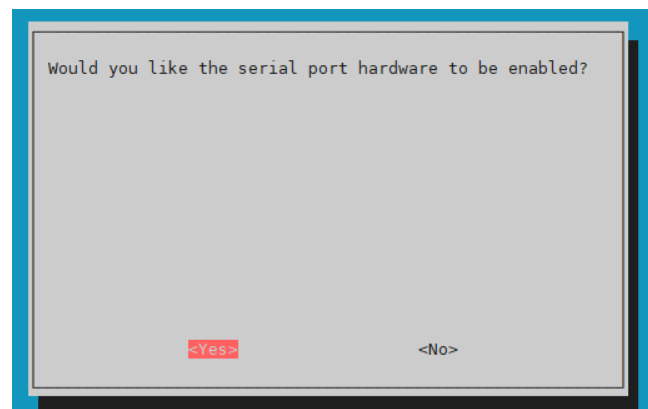
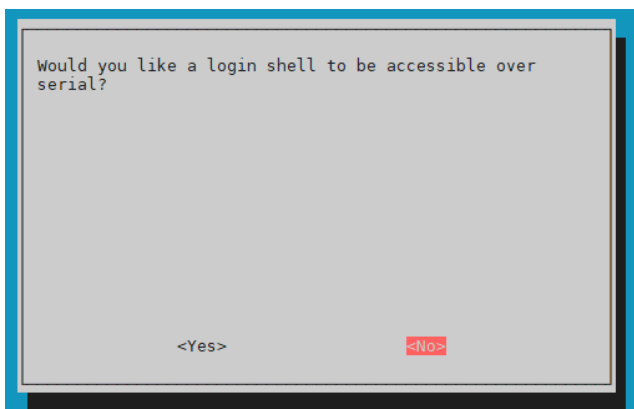
Führen Sie dazu den folgenden Befehl aus, um die Konfiguration aufzurufen.

```
sudo raspi-config
```

Wählen Sie dort unter **3 Interface Options** → **I5 Serial Port** aus.



Deaktivieren Sie nun die serial login shell und aktivieren Sie die serielle Hardware.



Sie haben nun die serielle Kommunikation auf Ihrem Raspberry Pi ermöglicht. Starten Sie jetzt 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 install python3-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. Das Codebeispiel ist für den Pi5 ausgelegt, wenn Sie ein älteres Modell verwenden müssen Sie die Initialisierung für den Pi5 in Zeile 8 entfernen und die für den Pi4 in Zeile 5 hinzufügen.

```
nano ttlToRS485.py
```

```
import serial
import time

# Opening the serial port for Pi4
#port = serial.Serial("/dev/ttyS0", baudrate=9600, timeout=3.0)

# Opening the serial port for Pi5
port = serial.Serial("/dev/ttyAMA0", baudrate=9600, timeout=3.0)

print ("Starting test...")
port.write(bytes("Please enter Test-String: \n",'ascii'))
# Message is sent
while True:
    testString = port.readline() # Something is read out
    testString = testString.decode('utf-8') # Convert to string

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

Speichern können Sie die Datei mit **STRG+O** und **Enter** und anschließend mit **STRG+X** schließen.

Ausführen können Sie die Datei in der Konsole mit dem Befehl

```
python3 ttlToRS485.py
```

und stoppen können Sie die Datei mit **STRG+C**.

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.

5. 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.

6. 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: <https://support.joy-it.net>

Telefon: +49 (0)2845 9360-50 (Mo - Do: 09:00 - 17:00 Uhr,
Fr: 09:00 - 14:30 Uhr)

Für weitere Informationen besuchen Sie unsere Website:

www.joy-it.net