

# TTL TO RS485 CONVERTER

## COM-TTL-RS485

## 1. GENERAL INFORMATION

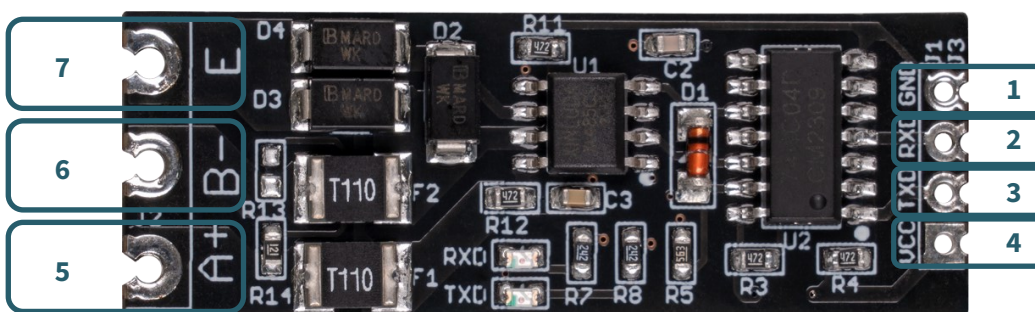
Dear Customer,  
Thank you for choosing our product. In the following, we will show you what to consider during commissioning and use.

Should you encounter any unexpected problems during use, please feel free to contact us.

## 2. BASIC INFORMATION

This UART TTL to RS485 converter allows you to add an RS485 interface to your single board computer. The module is easy to operate and is controlled via the UART interface. With up to 1 km transmission range, data transfers with widely spaced devices are no problem. Its design and size make the module highly integrable. Due to its high immunity to interference, it is particularly often used in industrial areas.

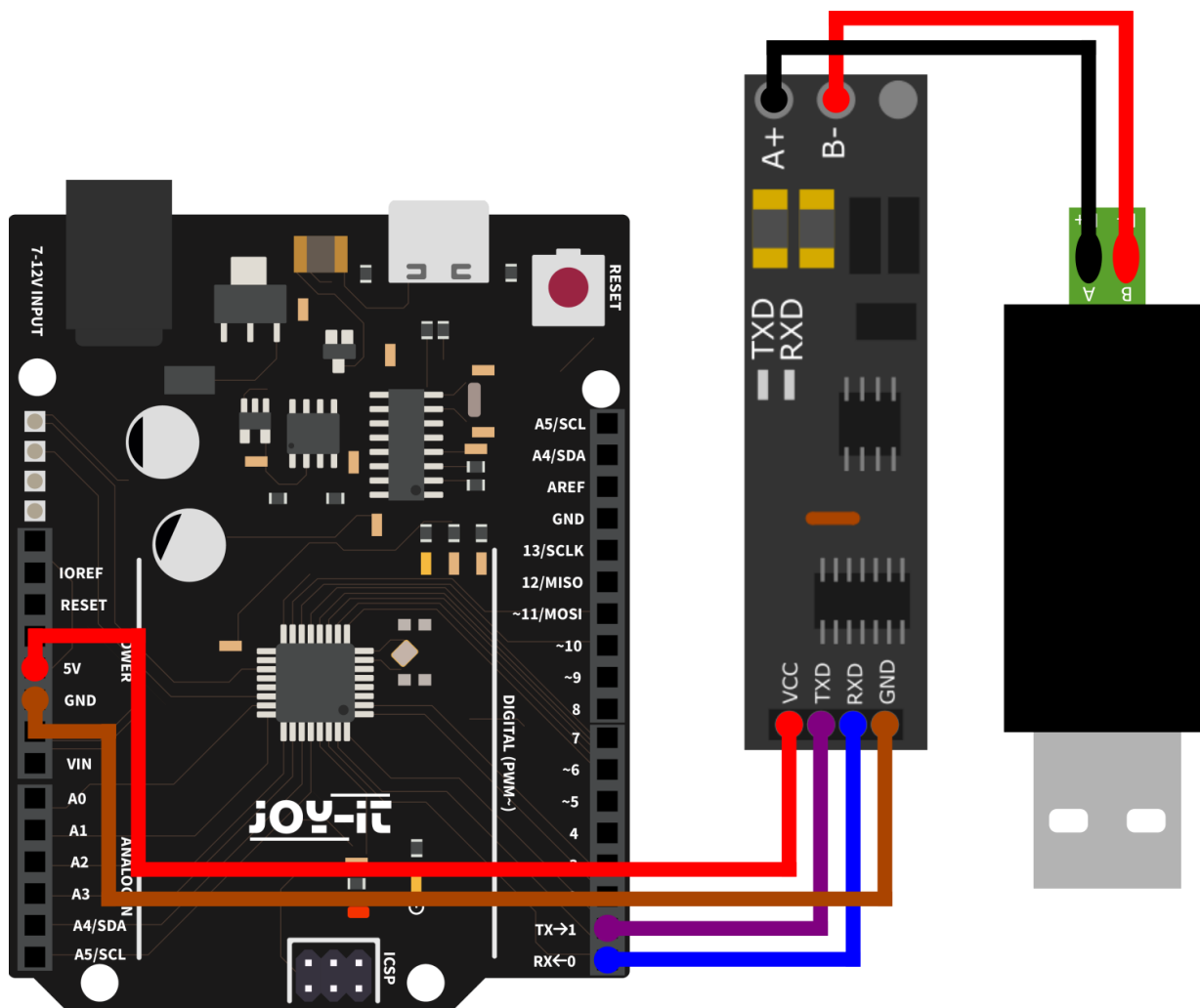
**The LEDs labeled RX and TX on the product are status LEDs that visually indicate serial communication on the corresponding pin.**



Digit	Function
1	GND → Ground potential
2	RXD → UART data output
3	TXD → UART data input
4	VCC → Supply voltage
5	A+ → RS485 non-inverted signal
6	B- → RS485 inverted signal
7	E → RS485 Ground Potential

### 3. USE WITH THE ARDUINO

#### 1. Connection



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

## 2. Code example

We provide a code example for use with the Arduino that you can download [here](#). Below is a code example that you can use to test the device. However, for this you will need a terminal device for the RS485 port to allow communication. In this example we used the USB-RS485 adapter ([SBC-TTL-RS485](#)) from Joy-IT. Please note that A+ is connected to A and B- to B.

Use the Arduino IDE for the code example. Make sure that the correct **port** and **board** are selected under **Tools**. Copy the following code and click Upload.

```
String testString; //Declaring a variable
void setup() {
  Serial.begin(9600); // Enable serial communication
}
void loop() {
  Serial.println("Please enter Test-String: ");
  while (Serial.available()==0) {} // Waiting for input
  testString=Serial.readString(); //Read out the input
  Serial.println("The Test-String received is: " + testString);
  Serial.println("-----");
  delay(2000);
}
```

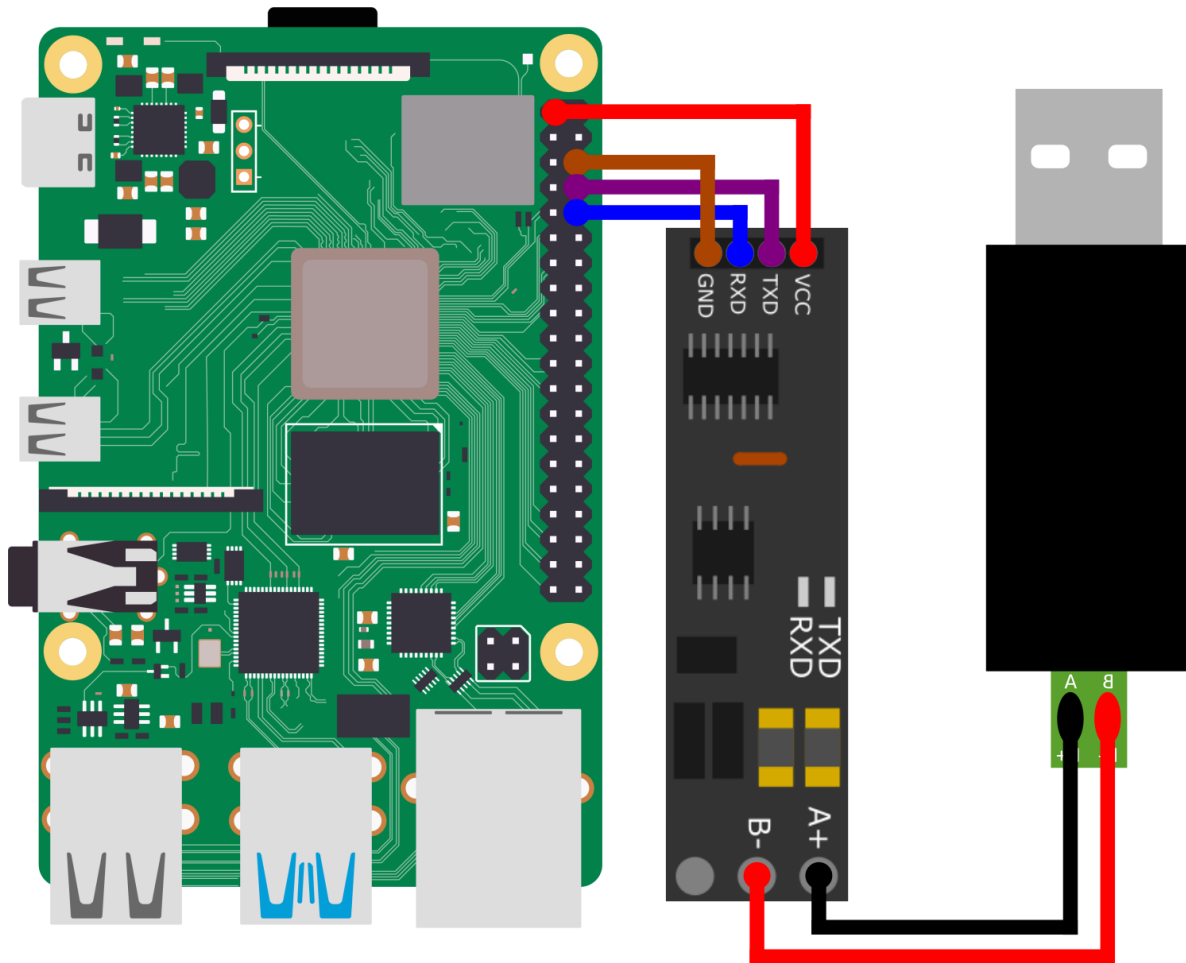
Now you can communicate with your Arduino via a terminal program, such as HTerm. You have to make sure that you set the baud rate to the same speed as your Arduino, in this example 9600. The data received from the Arduino will be displayed to you in the serial monitor.

## 4. USE WITH THE RASPBERRY PI



This guide was written under Raspberry Pi OS Bookworm for the Raspberry Pi 4 and 5. It has not been checked with newer operating systems or hardware.

### 1. Connection



Raspberry Pi	Converter
3V	VCC
GPIO 14	TXD
GPIO 15	RXD
GND	GND

## 2. Code example

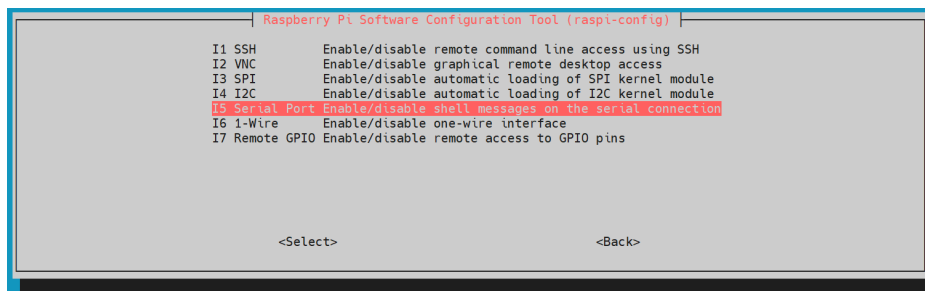
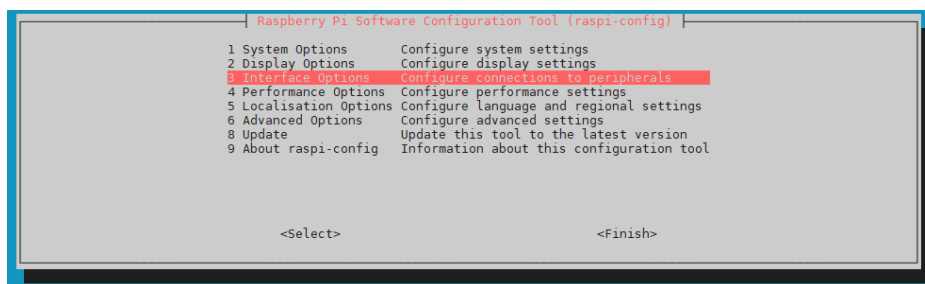
We provide a code example for use with the Raspberry Pi that you can download [here](#). Below is a code example that you can use to test the device. However, for this you will need a terminal device for the RS485 port to allow communication. In this example we used the USB-RS485 adapter ([SBC-TTL-RS485](#)) from Joy-IT. Please note that A+ is connected to A and B- to B.

To use the following code example, you need to enable serial communication on the Raspberry Pi.

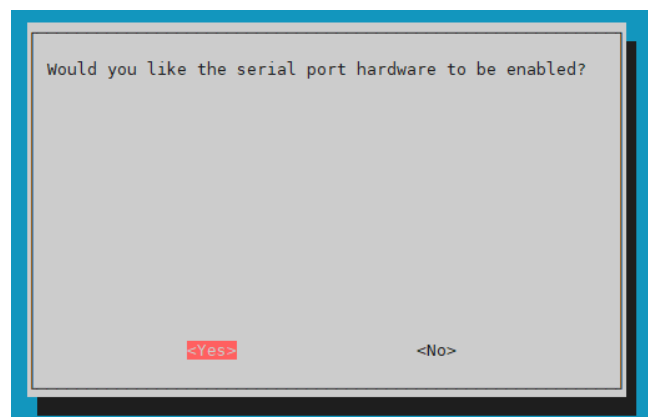
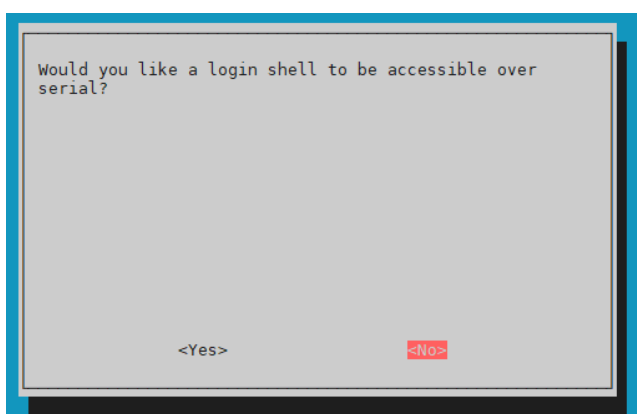
To do this, run the following command to invoke the configuration.

```
sudo raspi-config
```

Select there under **3 Interface Options** → **I5 Serial Port**.



Now deactivate the serial login shell and activate the serial hardware.



You have now enabled serial communication on your Raspberry Pi. Now restart your device once to apply the changes. Make sure that the serial library is installed on your Raspberry Pi. To do this, run the following command in your console.

```
sudo apt install python3-serial
```

You can create a file with the following command. Paste the following code there. The code example is designed for the Pi5, if you use an older model you have to remove the initialization for the Pi5 in line 8 and add the one for the Pi4 in line 5.

```
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'))
```

You can save the file with **CTRL+O** and **Enter** and then close it with **CTRL+X**.

You can execute the file in the console with the command

```
python3 ttlToRS485.py
```

and stop the file with **CTRL+C**.

Now you can communicate with your Raspberry Pi via a terminal program, such as HTerm. You have to make sure that you set the baud rate to the same speed as in your Python code, in this example 9600. The data received from the Raspberry Pi will be displayed to you in the console.

## 5. OTHER INFORMATION

### Our Information and Take-back Obligations according to the Electrical and Electronic Equipment Act (ElektroG)



#### **Symbol on Electrical and Electronic Products:**

This crossed-out bin means that electrical and electronic products do not belong into the household waste. You must hand over your old appliance to a registration place. Before you can hand over the old appliance, you must remove used batteries and replacement batteries which are not enclosed by the device.

#### **Return Options:**

As the end user, you can hand over your old appliance (which has essentially the same functions as the new one bought with us) free of charge for disposal with the purchase of a new device. Small devices, which do not have outer dimensions bigger than 25 cm can be handed in for disposal independently of the purchase of a new product in normal household quantities.

#### **1. Possibility of return at our company location during our opening hours**

Simac Electronics Handel GmbH, Pascalstr. 8, D-47506 Neukirchen-Vluyn

#### **2. Possibility of return nearby**

We will send you a parcel stamp with which you can send us your old appliance free of charge. For this possibility, please contact us via e-mail at [service@joy-it.net](mailto:service@joy-it.net) or via telephone.

#### **Information about Package:**

Please package your old appliance safe for transport. Should you not have suitable packaging material or you do not want to use your own material, you can contact us and we will send you an appropriate package.

## 6. SUPPORT

If any questions remained open or problems may arise after your purchase, we are available by e-mail, telephone and ticket support system to answer these.

Email: [service@joy-it.net](mailto:service@joy-it.net)

Ticket system: <https://support.joy-it.net>

Telephone: +49 (0)2845 9360-50 (Mon - Thur: 09:00 - 17:00 o'clock,  
Fri: 09:00 - 14:30 o'clock)

For further information please visit our website:

[www.joy-it.net](http://www.joy-it.net)