

## **Function description for RS485 communication Raspberry Pi 3 B to Teensy4.0, V1.00**

2022-04-18, Armin Rehberger

Example programs for RS485 communication between Raspberry Pi 3 B and Teensy4.0

Communication is started by typing a character in the serial monitor of the Arduino software (Setting with no line end).

### **ASCII Example:**

End character = LineFeed \n

Raspberry: [A\\_RS485.py](#)

Teensy: [A\\_RS485\\_Teensy.ino](#)

### **Decimal Example:**

Communicates a fixed amount of Bytes.

Last two Bytes are used for CRC16 checksum.

Raspberry: [B\\_RS485.py](#)

Teensy: [B\\_RS485\\_Teensy.ino](#)

## Teensy4.0 with Arduino V1.8.15

The image shows the Arduino IDE interface. The top toolbar includes icons for opening files, saving, compiling, uploading, and monitoring. The main editor displays a C++ sketch for an LCD display. The sketch includes the `LiquidCrystal_I2C` library and configures an I2C LCD module. It also initializes a USB serial port and sets up digital outputs for an LED. The serial monitor at the bottom shows the compilation process, including the use of the `avr-gcc` compiler and the `avr-libc` library. The output indicates that the sketch was compiled successfully and the resulting binary was uploaded to the Arduino board.

```
1 // ##### Include
2 #include <LiquidCrystal_I2C.h> // For LCD display HD44780 2004 LCD, 4x20 characters
3
4 // ##### Configure LCD display, address 0x27 (39), 4x20 Zeichen
5 LiquidCrystal_I2C lcd(0x27, 20, 4);
6
7 // ##### Setup
8 void setup()
9 {
10 // ##### Initialize USB Serial port
11 Serial.begin(9600);
12 Serial.println("Startup");
13
14 // ##### Initialize serial 1, UART port, RS485
15 // UART, Universal Asynchronous Receiver / Transmitter
16 // RX default pin 0
17 // TX default pin 1
18 // Transmitter Enable could be any pin
19 #define HWSERIAL_SERIAL1
20 HWSERIAL.setRX(0); // Pin 0 = RX
21 HWSERIAL.setTX(1); // Pin 1 = TX
22 HWSERIAL.transmitterEnable(2); // Pin 2 = Transmitter enable
23 HWSERIAL.setTimeout(1000); // Read timeout value in ms
24 HWSERIAL.begin(115200); // 115200 9600 baud
25
26 // ##### Digital outputs
27 pinMode(LED_BUILTIN, OUTPUT); // LED on board
28
29 // ##### Initialize LCD display, 4x20 digits
30 lcd.init();
31 lcd.backlight();
32 }
33
```

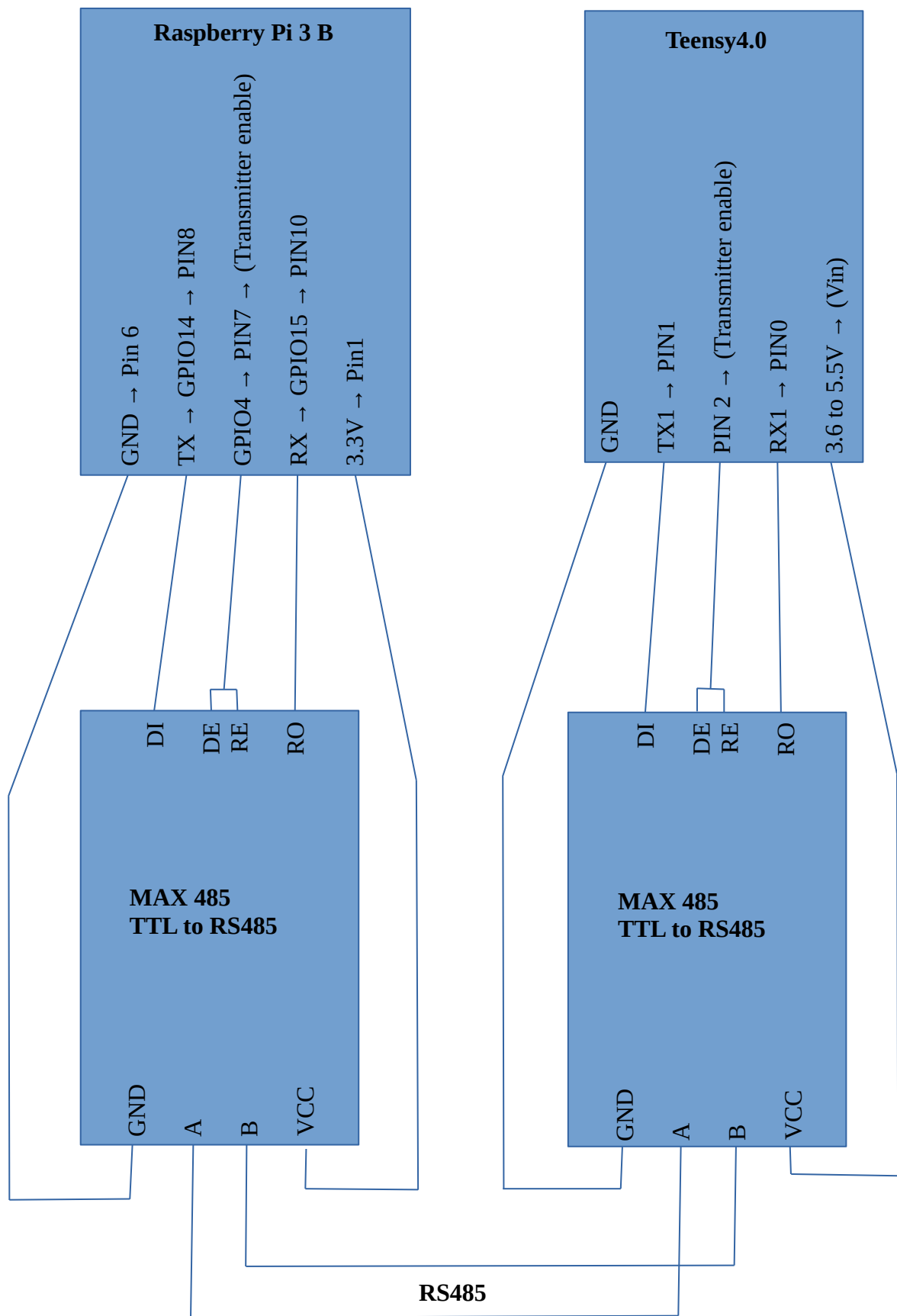
Serial Monitor Output:

```
Zuvor kompilierte Datei wird verwendet: C:\Users\armin\AppData\Local\Temp\arduino_build_47037\libraries\teensy\
Zuvor kompilierte Datei wird verwendet: C:\Users\armin\AppData\Local\Temp\arduino_build_47037\libraries\teensy\
Compiling core...
Using precompiled core: C:\Users\armin\AppData\Local\Temp\arduino_cache_165247\core\core_B2424dfcf9509c3677a
Linking everything together...
"C:\Program Files (x86)\Arduino\hardware\teensy\tools/arm/bin/arm-none-eabi-gcc" -O2 -Wl,--gc-sections
"C:\Program Files (x86)\Arduino\hardware\teensy\tools/arm/bin/arm-none-eabi-objcopy" -O ihex -j .eep
"C:\Program Files (x86)\Arduino\hardware\teensy\tools/arm/bin/arm-none-eabi-objcopy" -O ihex -R .eep
"C:\Program Files (x86)\Arduino\hardware\teensy\tools/ld/ld-elf.sh" -T linker.ld -o teensy.hex
"C:\Program Files (x86)\Arduino\hardware\teensy\tools/teensy_post_compile" -file=RS485_Teensy.ino -path=C:\
Memory Usage on Teensy 4.0:
FLASH: code:32648, data:5228, headers:8200 free for files:1985540
RAM: variables:12992, code:30984, padding:1784 free for local variables:478528
BSP: LiquidCrystal_I2C free for malloc:511904
BSP: LiquidCrystal_I2C in Version 1.1.2 in Order: D:\Arduino\libraries\LiquidCrystal_I2C wird verwendet
Bibliothek teensy_i2c-master in Version 1.1.0 in Order: D:\Arduino\libraries\teensy_i2c-master wird verwendet
C:\Program Files (x86)\Arduino\hardware\teensy\tools\teensy_post_compile -file=RS485_Teensy.ino -path=C:\
```

## Raspberry Pi 3 B with Spyder V4.2.1 Python 3.9.2

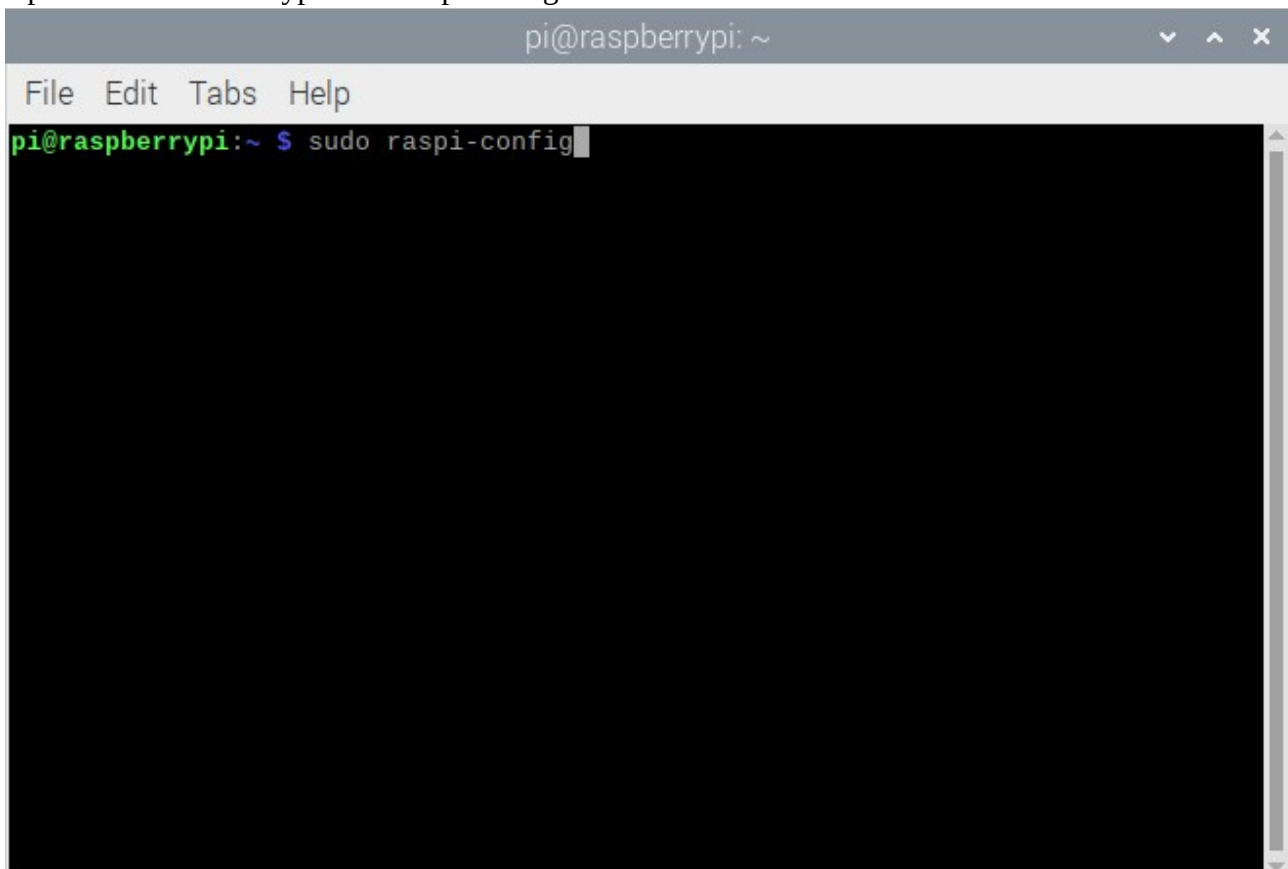
[illegible]

Wiring Raspberry Pi 3 B → MAX 485 → MAX 485 → Teensy4.0

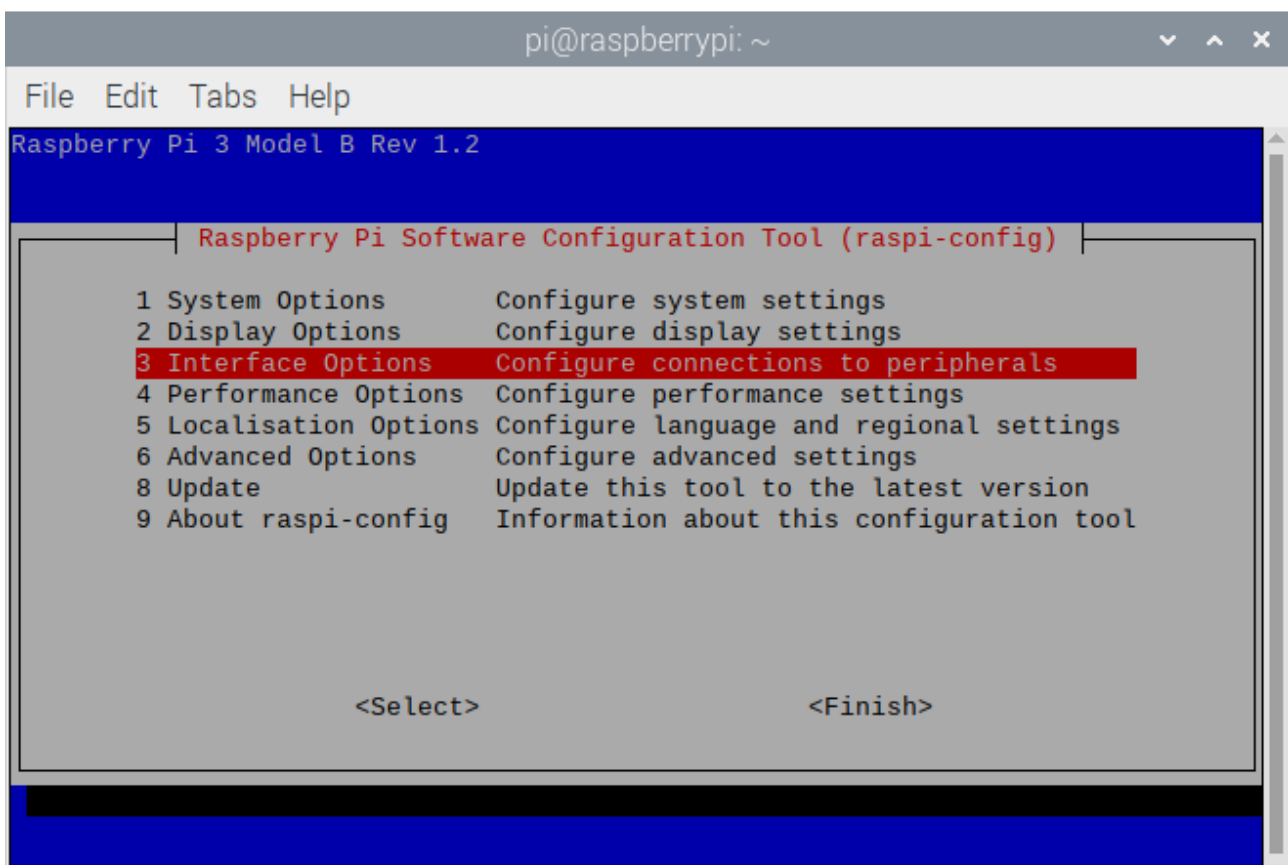


## Settings serial port Raspberry Pi 3 B with Linux11 (bullseye)

Open a terminal and type `sudo raspi-config`



A terminal window titled 'pi@raspberrypi: ~' with a menu bar containing 'File', 'Edit', 'Tabs', and 'Help'. The command prompt shows 'pi@raspberrypi:~ \$' followed by 'sudo raspi-config' with a cursor at the end.



A terminal window titled 'pi@raspberrypi: ~' with a menu bar containing 'File', 'Edit', 'Tabs', and 'Help'. The window displays the 'Raspberry Pi Software Configuration Tool (raspi-config)' menu. The first line is 'Raspberry Pi 3 Model B Rev 1.2'. The menu options are listed in a table-like format, with the third option, '3 Interface Options', highlighted in red. At the bottom, there are two options: '<Select>' and '<Finish>'. The background of the menu is blue.

Raspberry Pi Software Configuration Tool (raspi-config)	
1 System Options	Configure system settings
2 Display Options	Configure display settings
3 Interface Options	Configure connections to peripherals
4 Performance Options	Configure performance settings
5 Localisation Options	Configure language and regional settings
6 Advanced Options	Configure advanced settings
8 Update	Update this tool to the latest version
9 About raspi-config	Information about this configuration tool

<Select>                      <Finish>

