Lesson 5 UART Communication Configuration and Programming

1. UART Communication Introduction

Communication protocols play a crucial role in facilitating communication between devices. They are designed in different ways to meet distinctive system requirements and feature specific rules for devices to successfully communicate with each other.

In embedded systems, microcontrollers, and computers, UART is widely used as a hardware communication protocol between devices. Among the applicable communication protocols, UART only uses two wires as its sender and receiver.

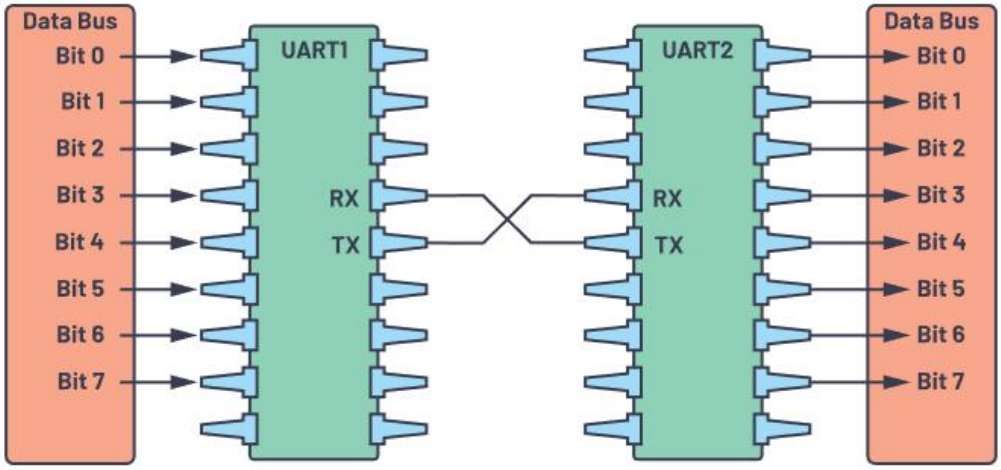
Despite being a widely used hardware communication protocol, UART has not always been fully optimized. When used in microcontrollers, it often overlooks the correct implementation of the frame protocol.

According to its definition, UART is a hardware communication protocol that uses asynchronous serial communication with configurable speed. Asynchronous means that UART has no shared clock signal to synchronize the output data bits transmitted from the sender to the receiver.

The two signals of each UART device are named as:

- Transmitter (Tx)
- Receiver (Rx)

The primary purpose of the transmitter and receiver wires of each device is to send and receive the serial data for serial communication.



UART TX is connected to the control data bus for sending data in parallel form. As a result, the data is transmitted bit by bit serially to UART RX with the transmission wire. Conversely, this action converts serial data into parallel data of the receiving device.

UART wires are used as communication media to send and receive data. Please note that UART devices have dedicated transmitting and receiving pins for sending or obtaining data.

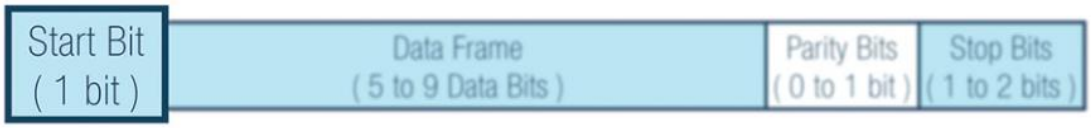
For UART and most serial communication, it is necessary to set the same baud rate on both the sending and receiving devices. Baud rate refers to the rate at which information is transmitted to the communication channel. In the context of the serial port, the set baud rate will serve as the maximum number of bits transmitted per second.

In UART, data is transmitted in the form of data packets. The part connecting the transmitter and receiver includes creating serial data packets and controlling the physical hardware lines. Data packets consist of a start bit, data frame, parity bit, and stop bit.

Start Bit (1 bit)	Data Frame (5 to 9 Data Bits)	Parity Bits (0 to 1 bit)	Stop Bits (1 to 2 bits)

- Start Bit

UART data transmission wires usually remain at a high voltage level when no data is being transmitted. To initiate data transmission, UART TX pulls the transmission wires low for one clock cycle. When the voltage transition from high to low is detected by UART RX, it starts to read the bits in the data frame at the baud rate frequency.



- Data Box

A data frame contains the actual data being transmitted. If parity bits are used, the data frame can be 5 to 8 bits long. If parity bits are not used, the data frame can be 9 bits long. In most cases, the least significant bit is transmitted first.

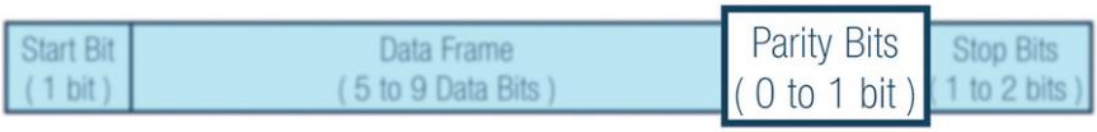
- Par Value

Parity describes whether a number is even or odd. The parity bit is a method that UART RX can determine if the data has changed during the process of transmission, which can be caused by electromagnetic radiation, mismatched baud rates, or long-distance data transmission.

After the UART RX reads the data frame, it calculates the number of bits with a value of “1”, and checks whether the total is even or odd. If the parity bit is “0” (even parity), the total number of bit 1 or logical high bits in the data frame should be even. If the parity bit is “1” (odd parity), the total number of bit 1 or logical high bits in the data frame should be odd.

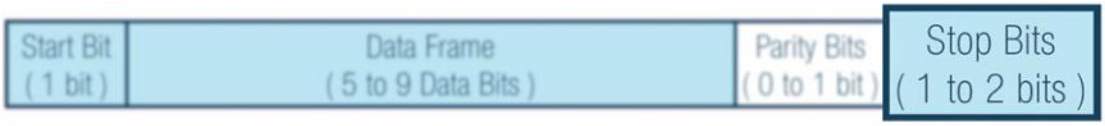
When the parity bit matches the data, UART knows that the transmission

is error-free. However, if the parity bit is “0” and the total is odd, or if the parity bit is “1” and the total is oven, UART knows that there is a change in the bits of the data frame.



- Stop Bit

To send a signal at the end of a data packet, UART TX drives the data transmission wire from a low voltage level to a high voltage level for a duration of 1 to 2 bits.

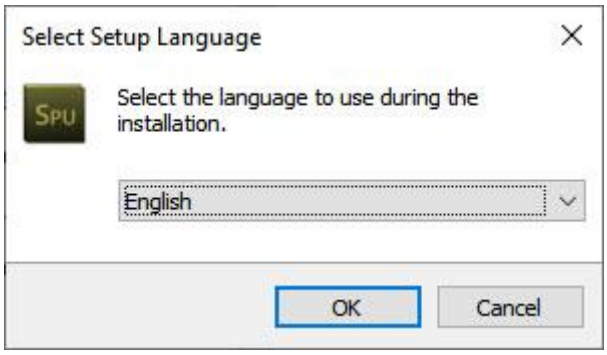


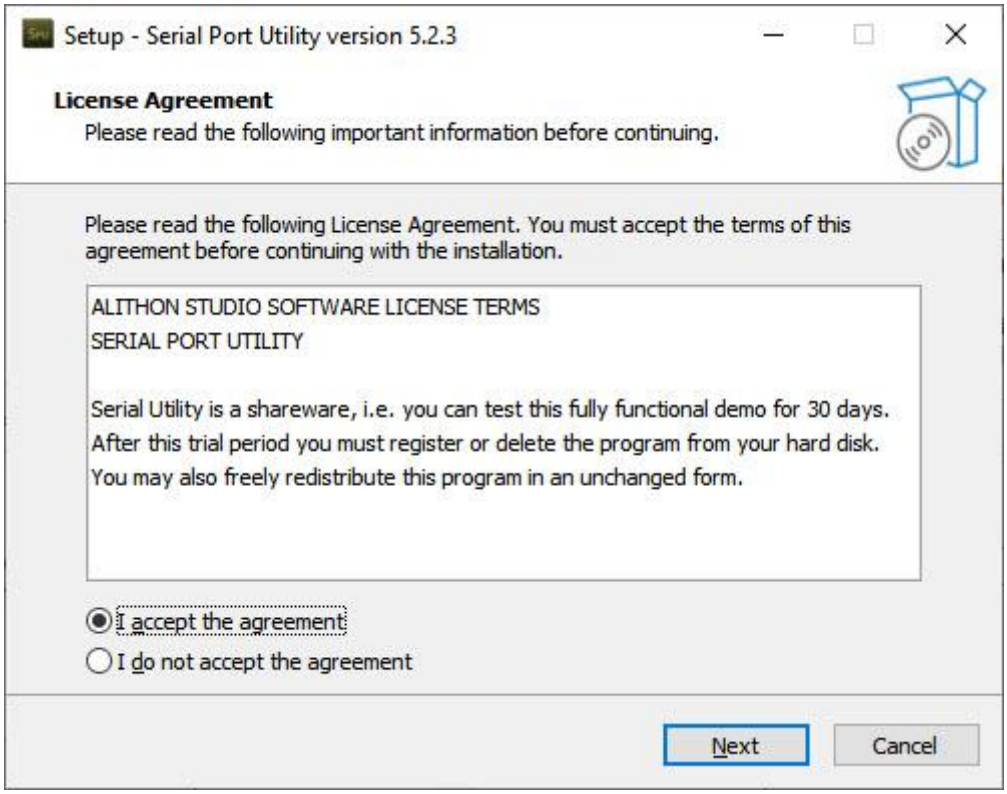
2. Getting Ready

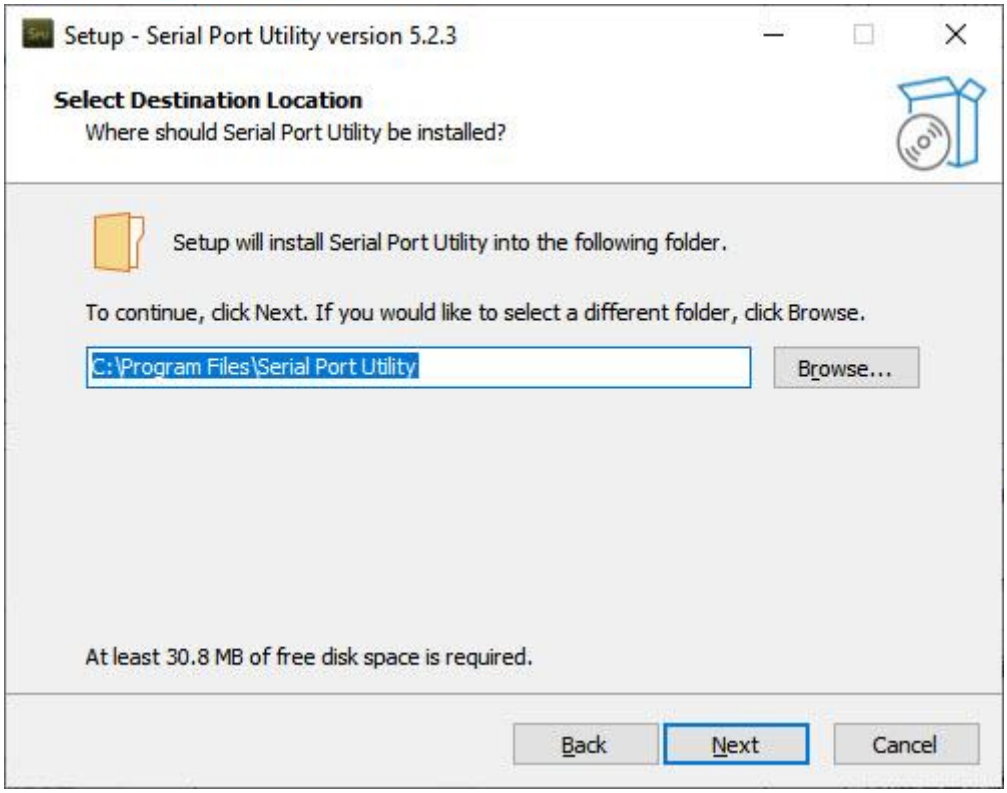
2.1 Serial Port Utility Installation

In this section, Serial Port Utility is used as an example to explain the process.

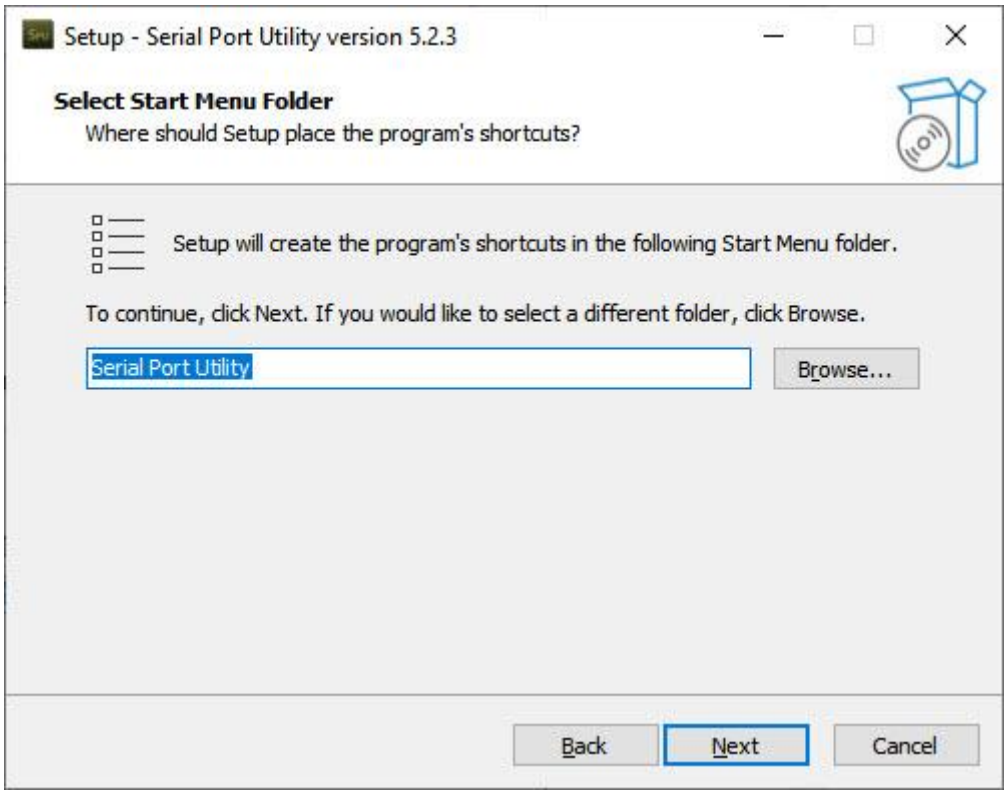
Double-click to open the “serial5.2.3.exe” installation program in this directory. Then, refer to the steps shown in the diagram below to complete the installation.

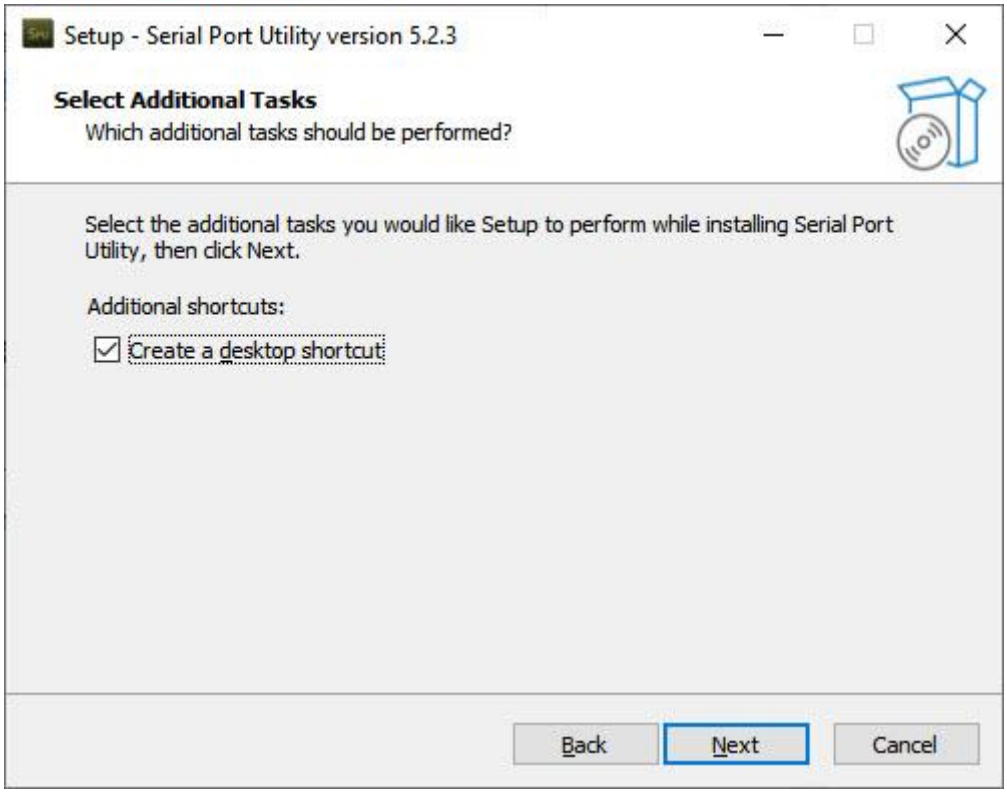

Select Setup Language
Select the language to use during the installation.
English
OK
Cancel

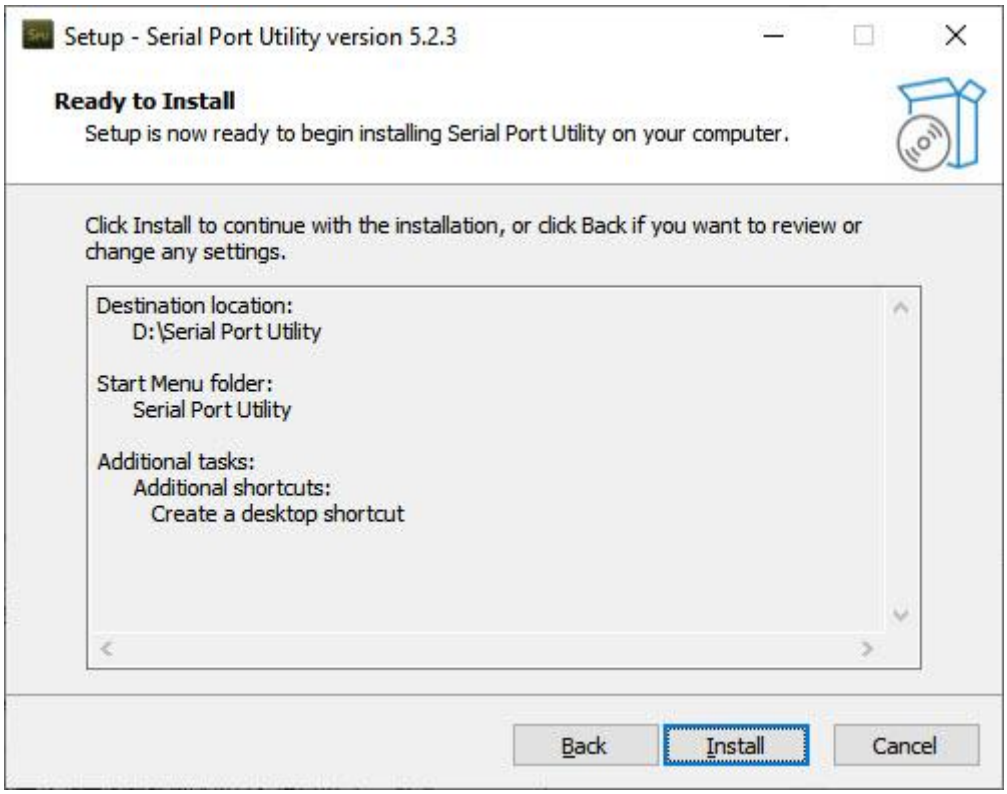

Setup - Serial Port Utility version 5.2.3
License Agreement
Please read the following important information before continuing.
Please read the following License Agreement. You must accept the terms of this agreement before continuing with the installation.
ALITHON STUDIO SOFTWARE LICENSE TERMS
SERIAL PORT UTILITY
Serial Utility is a shareware, i.e. you can test this fully functional demo for 30 days.
After this trial period you must register or delete the program from your hard disk.
You may also freely redistribute this program in an unchanged form.
I accept the agreement
I do not accept the agreement
Next
Cancel

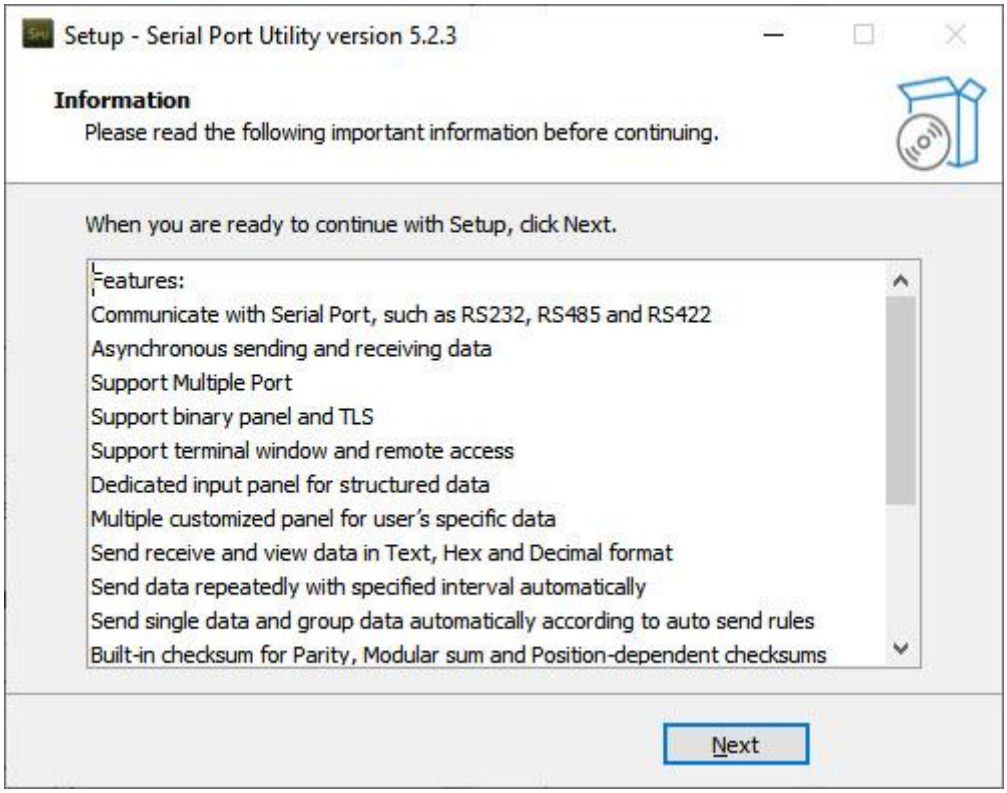


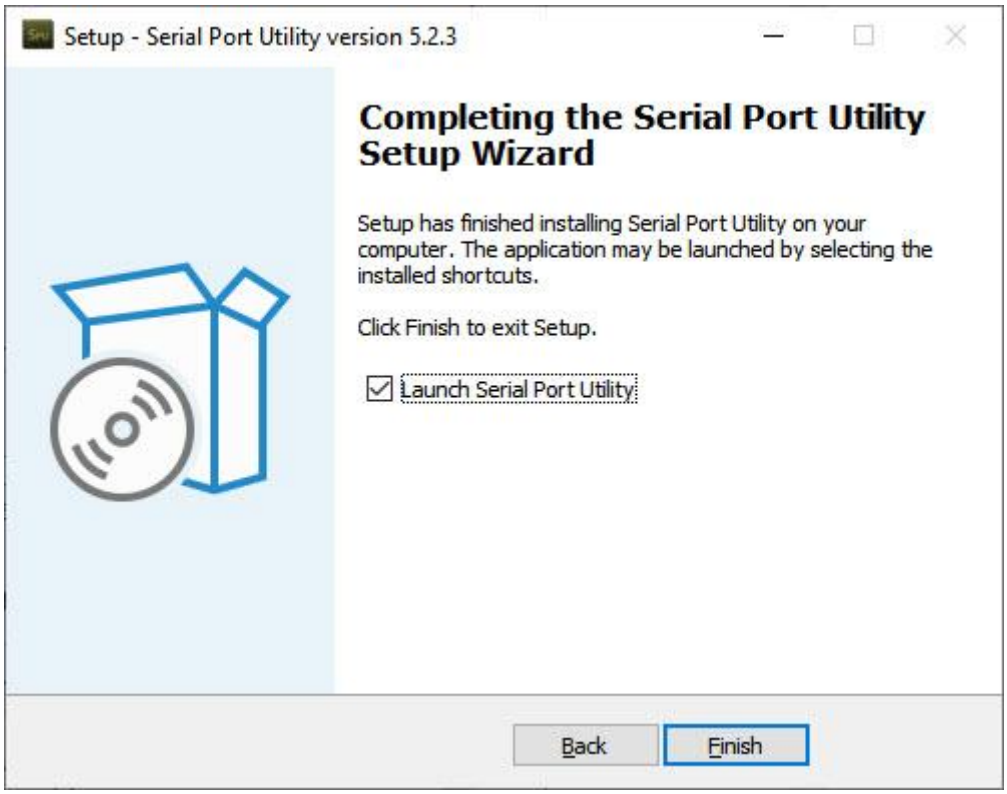
Suggest to select other path required installation.




Setup - Serial Port Utility version 5.2.3
Select Additional Tasks
Which additional tasks should be performed?
Select the additional tasks you would like Setup to perform while installing Serial Port Utility, then click Next.
Additional shortcuts:
Create a desktop shortcut
Back
Next
Cancel

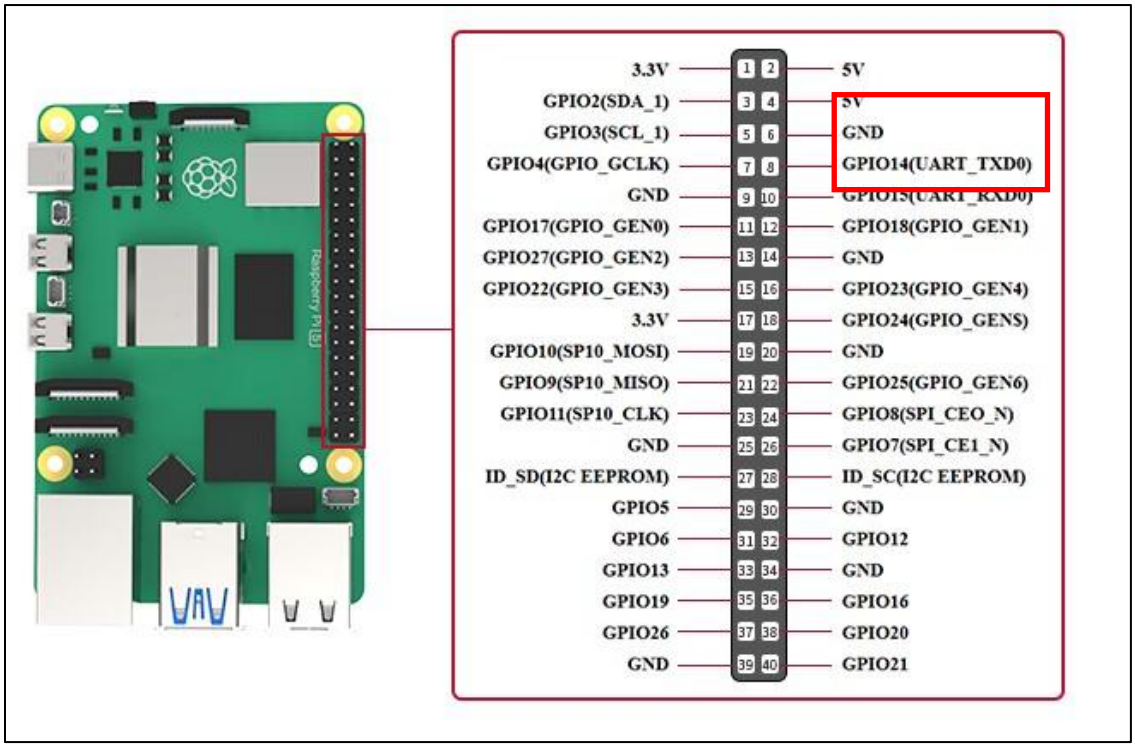

Setup - Serial Port Utility version 5.2.3
Ready to Install
Setup is now ready to begin installing Serial Port Utility on your computer.
Click Install to continue with the installation, or click Back if you want to review or change any settings.
Destination location:
D:\Serial Port Utility
Start Menu folder:
Serial Port Utility
Additional tasks:
Additional shortcuts:
Create a desktop shortcut
Back
Install
Cancel





2.2 Hardware Wiring

According to the Raspberry Pi pin diagram, you will use the following pins:



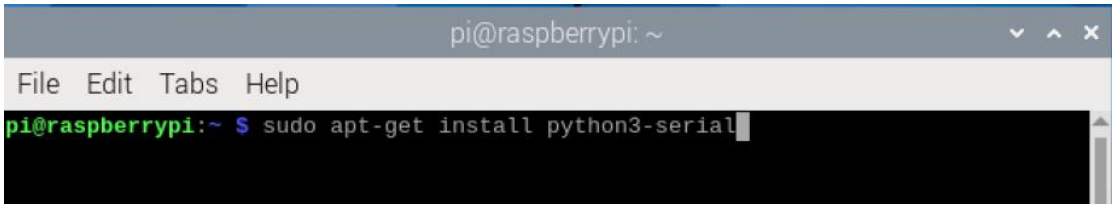
Use a USB to TTL module to connect the Raspberry Pi 5 to the PC with female-to-female DuPont wires, as shown in the following wiring program:

Pin8 on the Raspberry Pi 5（TXD） <--> USB to TTL module RXD
Pin10 on the Raspberry Pi 5（RXD） <--> USB to TTL module TXD
Pin6 on the Raspberry Pi 5（GND） <--> USB to TTL module GND

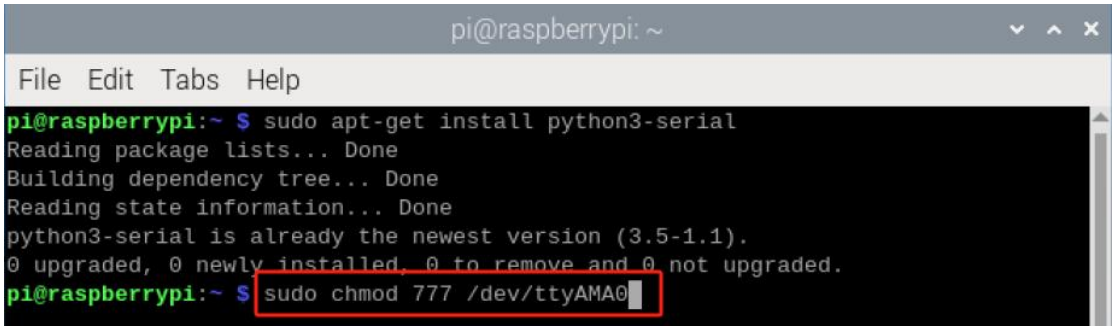
3. Raspberry Pi 5 Preparation

3.1 Library File Installation

1) Power the Raspberry Pi 5 on, and press “Ctrl+Alt+T” to open the command line terminal. Then enter the command “sudo apt-get install python3-serial” to install the serial function library.

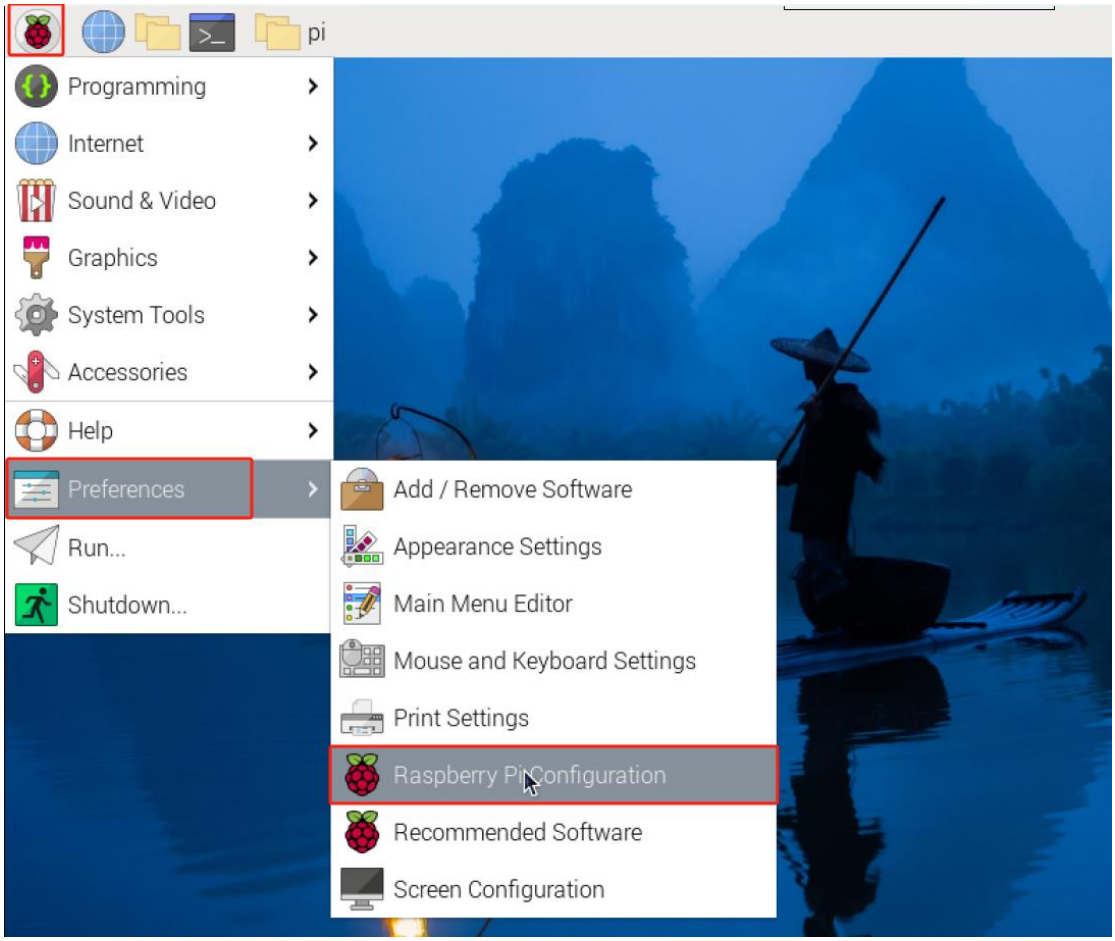


2) Next, enter the command “sudo chmod 777 /dev/ttyAMA0” to grant the serial access permission.

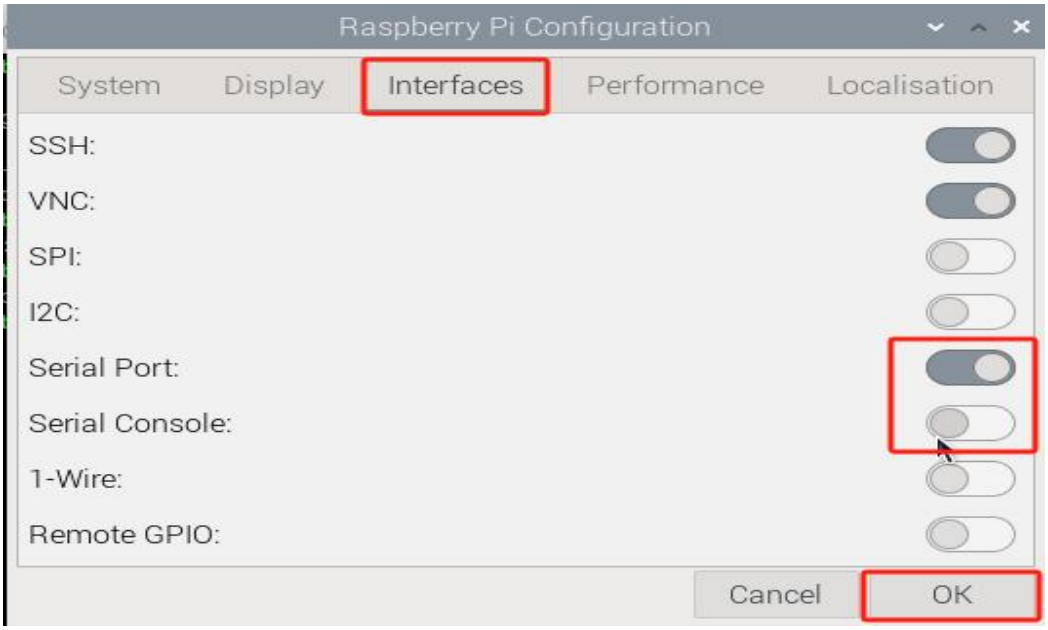


3.2 Open UART Interface

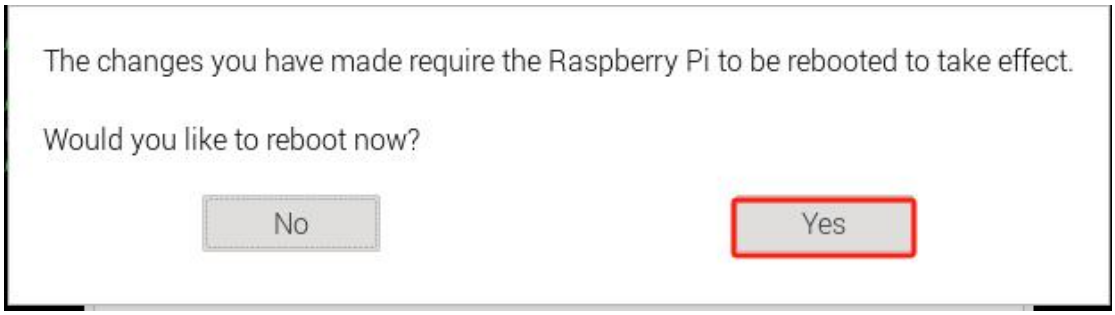
1) Configure the Raspberry Pi to enable the UART interface and open the Serial Port after booting up. Click on the the Raspberry Pi logo at the top left corner of the screen, and select “Preferences” and then “Raspberry Pi Configuration”.



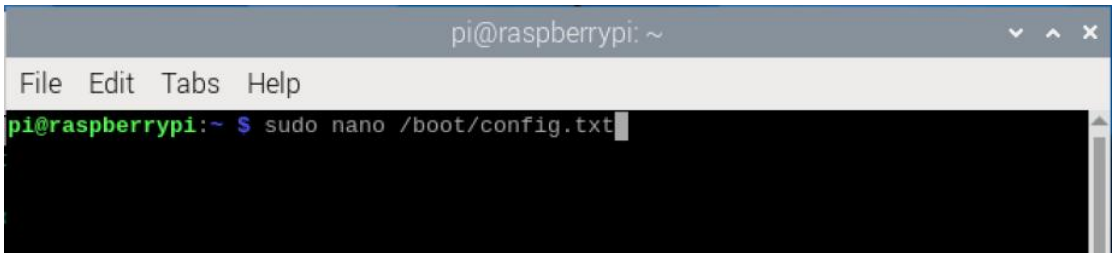
2) Select “Interfaces” to start “Serial Port” and close “Serial Console”, then click “OK”.



3) Click “Yes” to restart the Raspberry Pi. Upon the restart is complete, Raspberry Pi will point the main serial port to the hardware serial port. (If the restarting fails, unplug the USB converter before attempting to restart the Raspberry Pi again.)



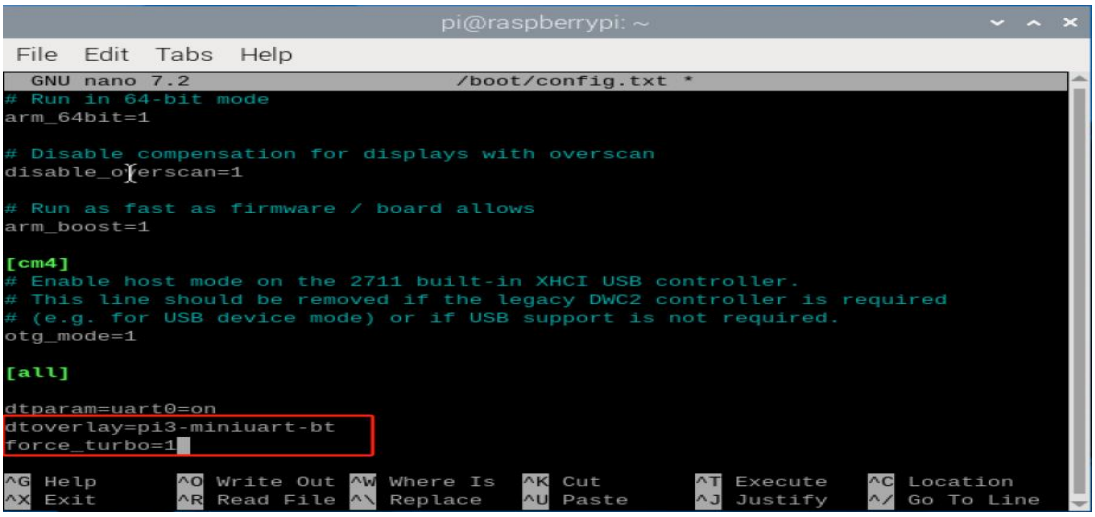
4) Press “Ctrl+Alt+T” to open the command line terminal, and enter “sudo nano /boot/config.txt” to open the configuration file.



5) Scroll to the end of the text to enter the provided code below.

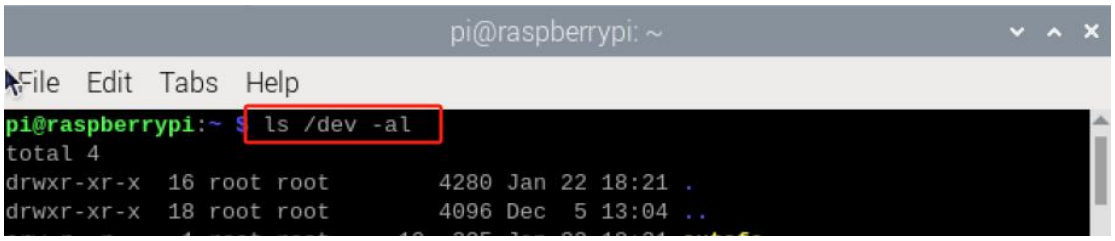
```
dtoverlay=pi3-miniuart-bt

Force_turbo=1
```

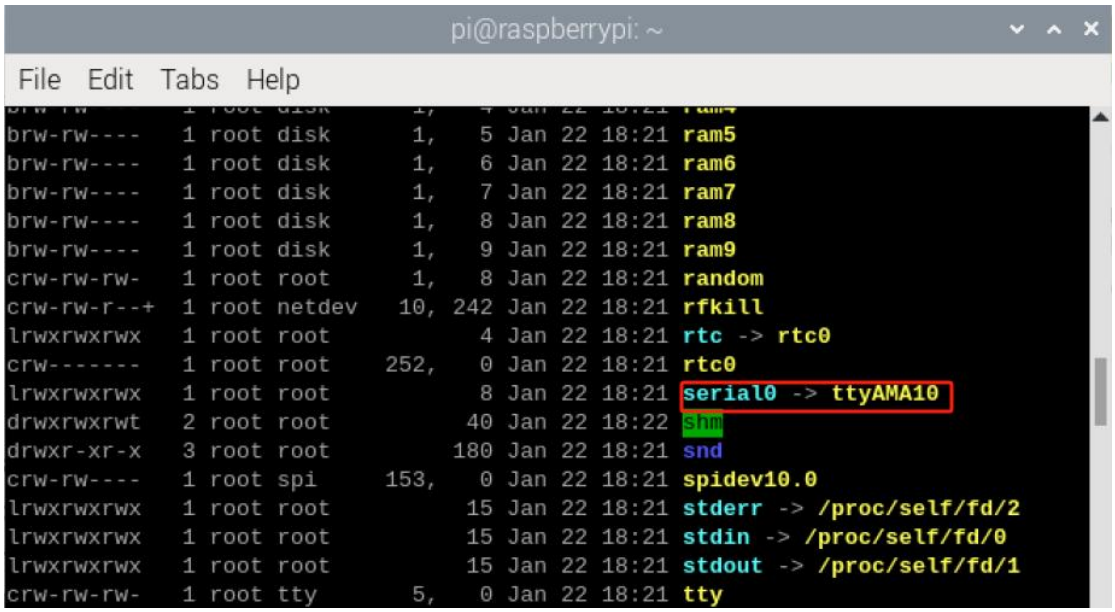


6) Press “Ctrl+S” to save it and “Ctrl+X” to return, then enter “sudo reboot” to restart the Raspberry Pi.

7) Enter “ls /dev -al” to check the serial port assignment.



8) Scroll down to locate the content highlighted in the red box shown in the following image. This confirms that the modification is successful.



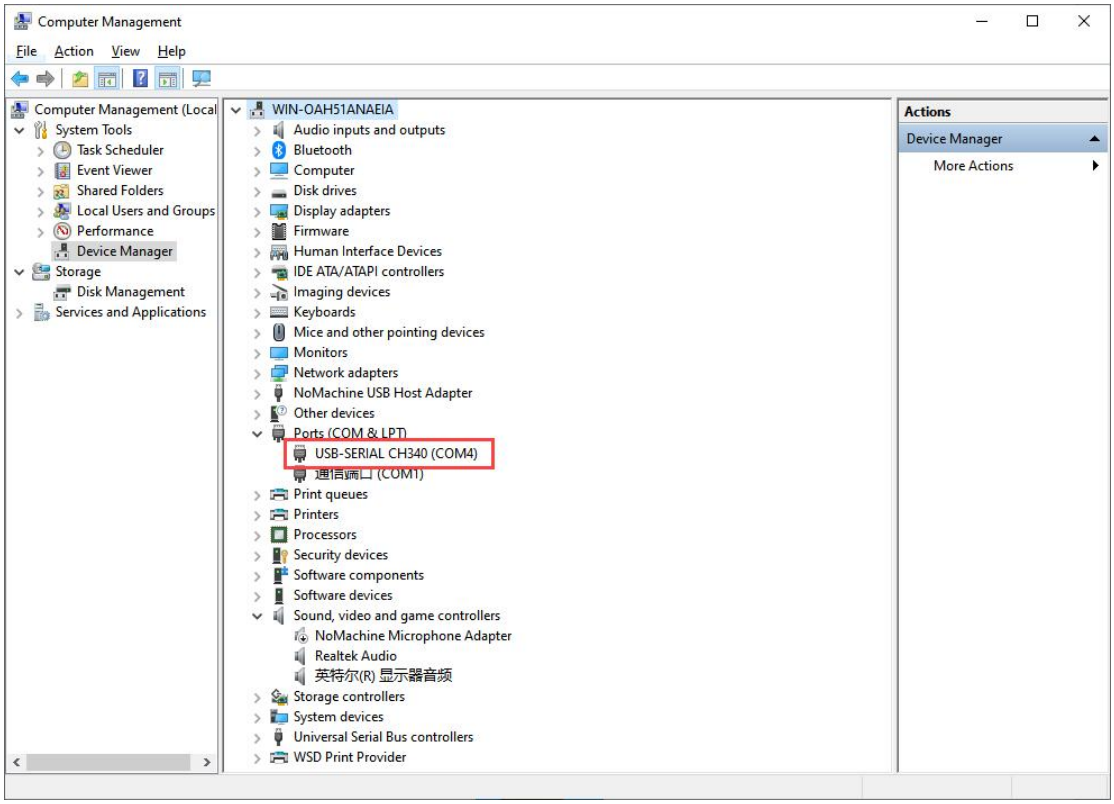
4. Example Program - Sending and Receiving

This section provides an example of using the serial port utility to display the instruction character string sent by Raspberry Pi 5 on a PC.

The character on the PC sent by the serial port utility can also be displayed on the Raspberry Pi 5 terminal.

4.1 Starting Serial Port Utility

Plug the USB to TTL converter tool into any USB port on the PC, then open the device manager to view if the port is recognized, as below:

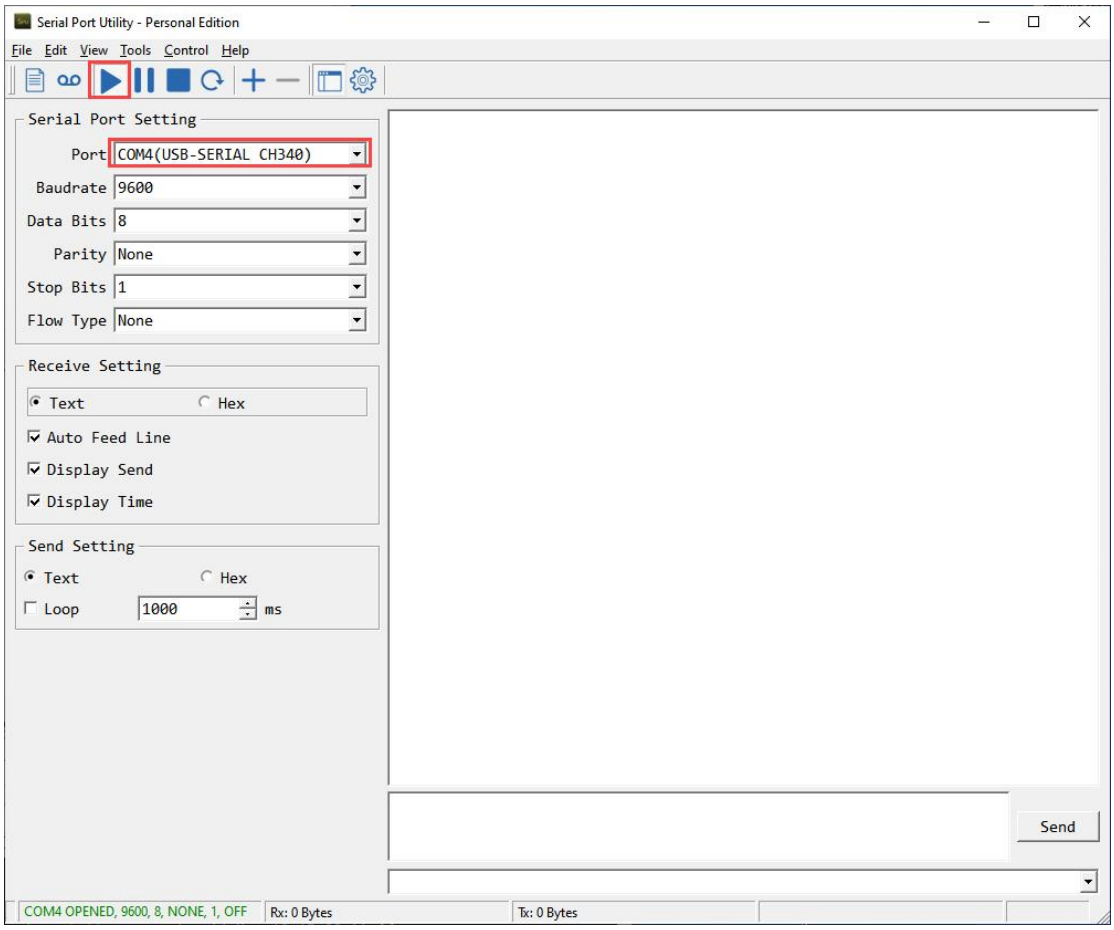


If the port with the CH340 label does not appear, you can check if your PC has installed the CH340 driver (the drive packet is located in this section's directory). if the driver is installed but the port cannot be recognized, try changing the USB port to troubleshoot the issue.

Double-click the installed "Serial Port Utility".

After opening the utility, select the port with the CH340 label and set the attributes such as baud rate and data bit according to the diagram below. Then,

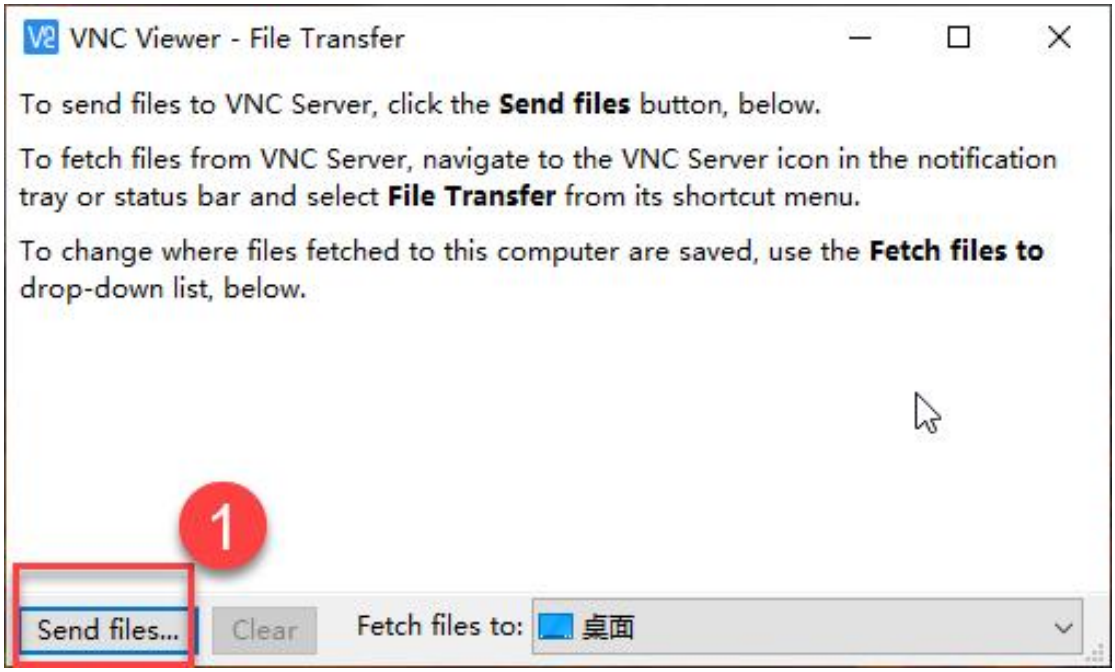
click the button within the red box to proceed.

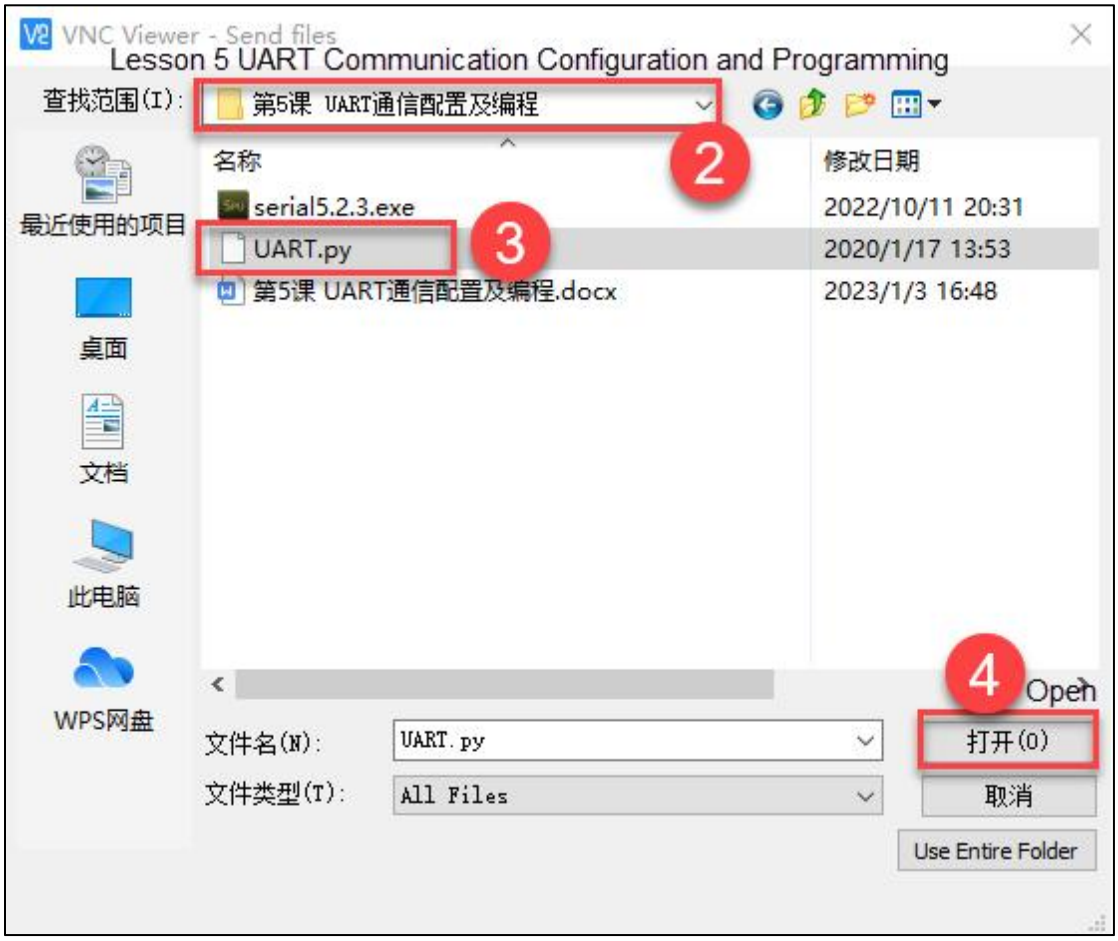


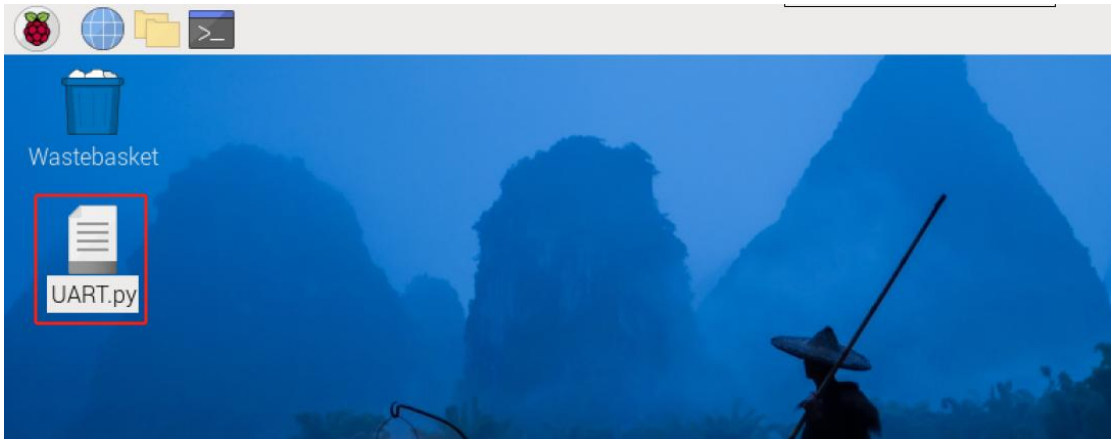
4.2 Import Example Program

1) Click the floating box located at the top of the system desktop to select the file transmission icon.

2) Click “Send files” in the popup window, then select “UART.py” in the same path as this document in the next pop-up window. Next, click “Open” to import the file into the Raspberry Pi system desktop.


VNC Viewer - File Transfer
To send files to VNC Server, click the **Send files** button, below.
To fetch files from VNC Server, navigate to the VNC Server icon in the notification tray or status bar and select **File Transfer** from its shortcut menu.
To change where files fetched to this computer are saved, use the **Fetch files to** drop-down list, below.
1
Send files...
Clear
Fetch files to:
桌面


VNC Viewer - Send files
Lesson 5 UART Communication Configuration and Programming
查找范围(I):
第5课 UART通信配置及编程
2
名称
修改日期
serial5.2.3.exe
2022/10/11 20:31
UART.py
2020/1/17 13:53
3
第5课 UART通信配置及编程.docx
2023/1/3 16:48
最近使用的项目
桌面
文档
此电脑
WPS网盘
4
Open
文件名(N):
UART.py
打开(O)
文件类型(T):
All Files
取消
Use Entire Folder



3) Input the command “chmod a+x /home/pi/Desktop/UART.py” to grant the executing permission for the program.

```
pi@raspberrypi:~ $ chmod a+x /home/pi/Desktop/UART.py
pi@raspberrypi:~ $
```

“pi” is just an example created by this PC, you need to rewrite the command above according to the actual execution.

4.3 Program Execution

1) Start the Raspberry Pi, and connect it to the remote control soft VNC.

2) Press "Ctrl+Alt+t" to open the command line terminal, and enter the command "cd Desktop/" to switch to the desktop.

```
pi@raspberrypi:~ $ cd Desktop/
```

3) Input the “python3 UART.py” command to run the program.

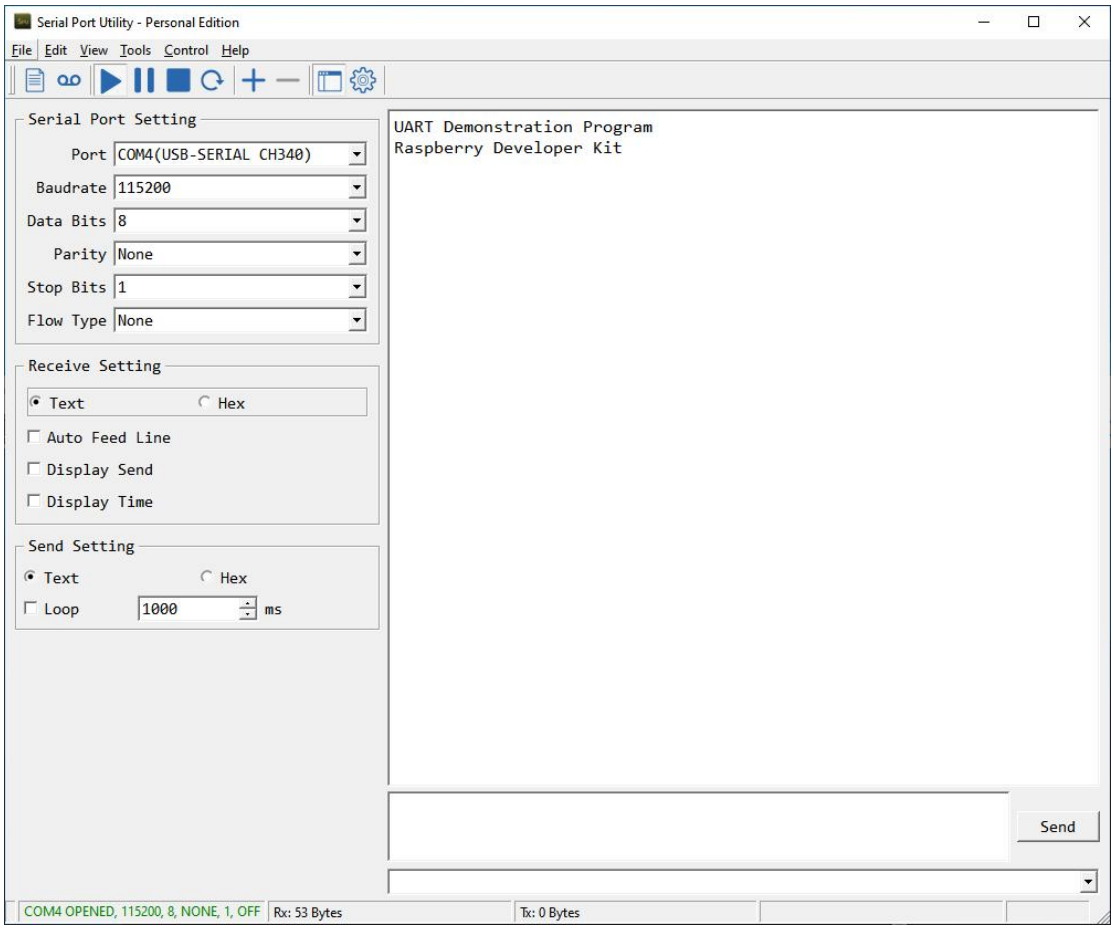
```
pi@raspberrypi:~/Desktop $ python3 UART.py
UART Demonstration Program
Raspberry Developer Kit
```

4.4 Program Display

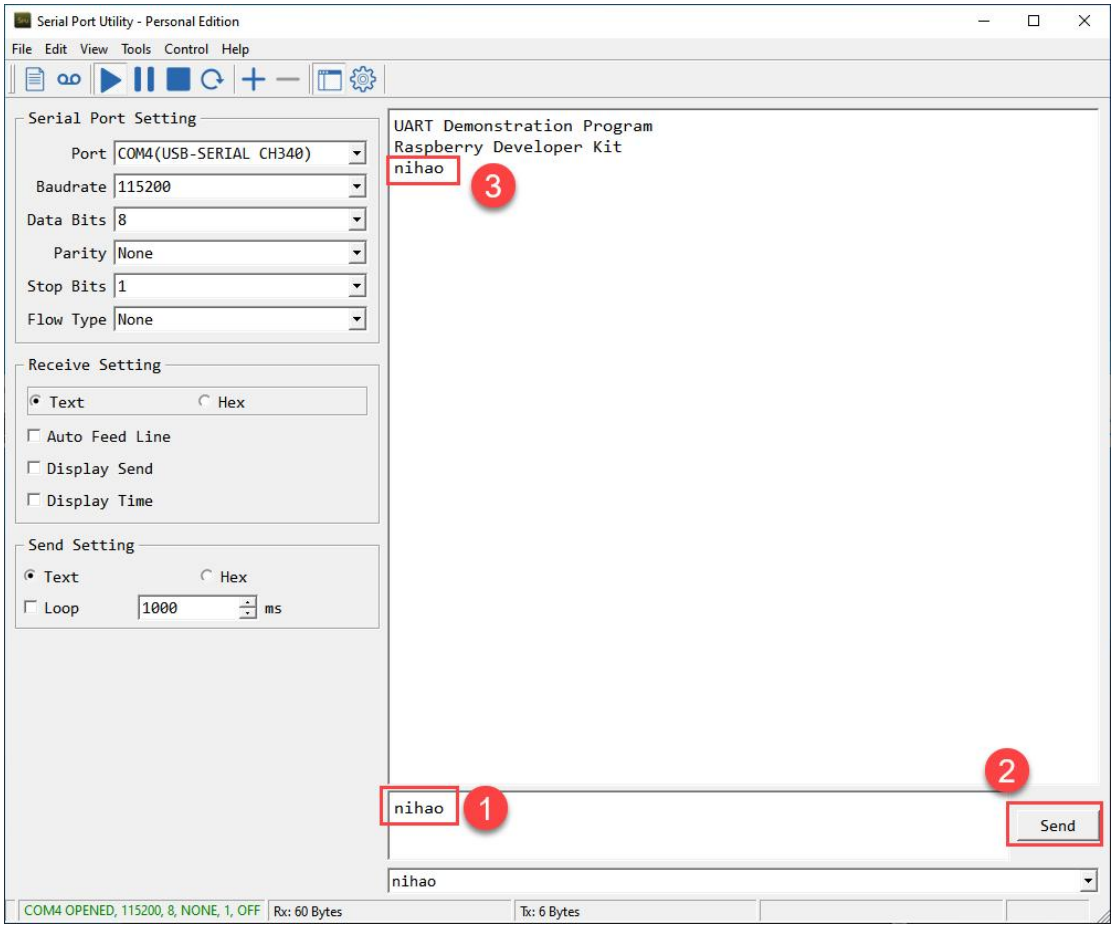
4.4.1 Serial Port Utility

The message sent by Raspberry Pi 5 can be received in the Serial Port

Utility. Make sure to edit the baud rate to “115200” as follows:



For example, you can enter text “nihao” in the field below, then click “Send”.



4.4.2 Raspberry Pi 5

On the command line interface of the Raspberry Pi 5, you can receive the message sent from the PC side.