# Serial communication

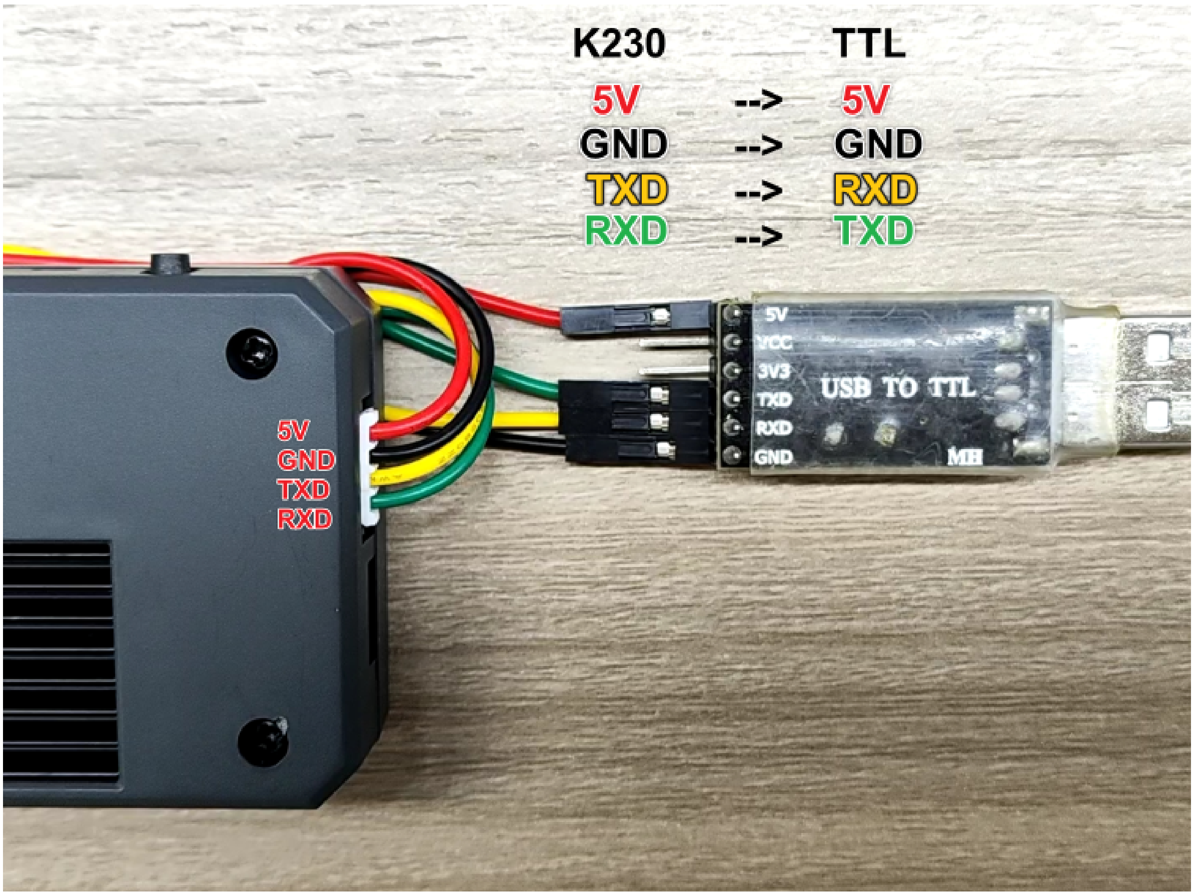
## Introduction

In this section, we introduce how to communicate with the host computer through the serial port of K230

Here we need to use a [USB to TTL module] and a [PH2.0 4Pin to DuPont loose head 2.54 (male to female, 200mm)]

> Because in this tutorial, we will use a computer as a device to communicate with K230, and modern computers usually only have USB interfaces and do not directly provide TTL serial ports, so we need to use a USB to TTL module for "translation"

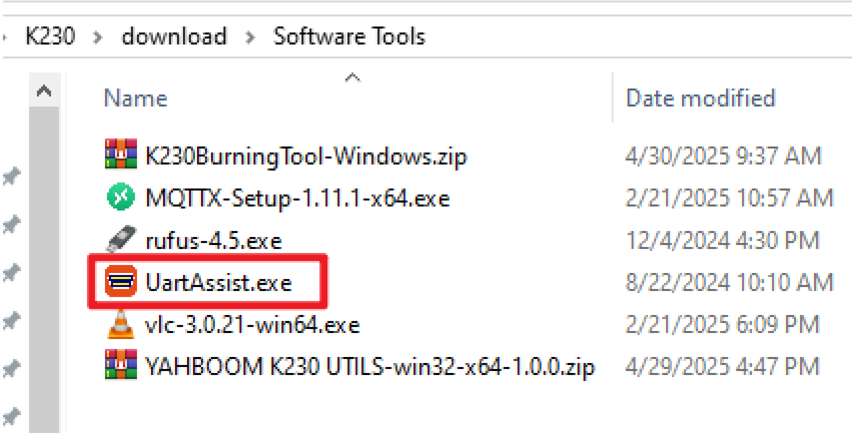
## wiring



# Quick Start

## Open the serial port assistant

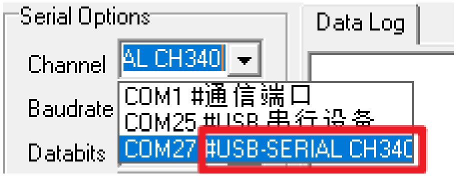
Click 【Software Tools】 in the download area of the tutorial website to get the software 【UartAssist.exe】. After downloading, double-click to run it.

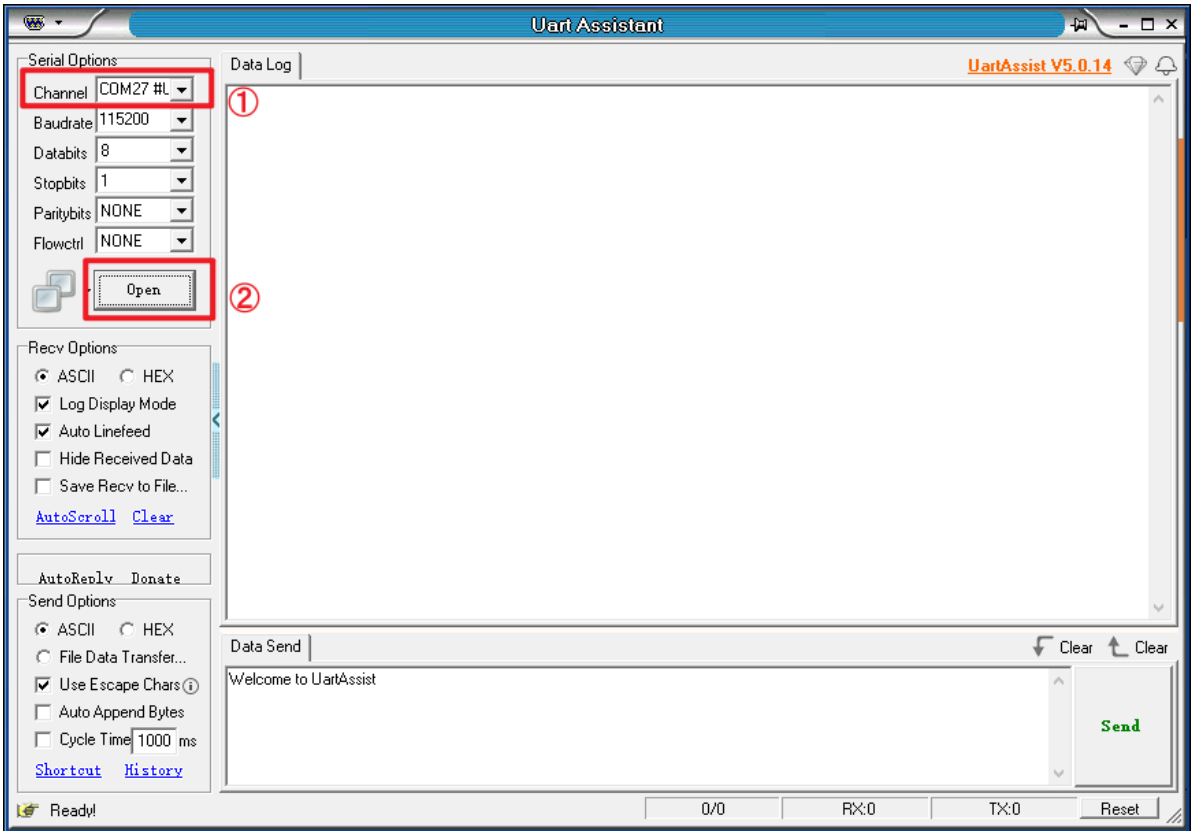


K230 › download › Software Tools

| Name | Date modified |
|---|---|
| K230BurningTool-Windows.zip | 4/30/2025 9:37 AM |
| MQTTX-Setup-1.11.1-x64.exe | 2/21/2025 10:57 AM |
| rufus-4.5.exe | 12/4/2024 4:30 PM |
| UartAssist.exe | 8/22/2024 10:10 AM |
| vlc-3.0.21-win64.exe | 2/21/2025 6:09 PM |
| YAHBOOM K230 UTILS-win32-x64-1.0.0.zip | 4/29/2025 4:47 PM |

After opening, we select the serial port number displayed on the computer by the USB to TTL module at ①

> Normally, this serial port number will end with CH340
>
> 
>
> Serial Options
>
> Channel AL CH340
>
> Baudrate
>
> Databits
>
> Data Log
>
> COM1 #通信端口
> COM25 #USB 串行设备
> COM27 #USB-SERIAL CH340
>
> The quickest way to judge is to unplug the USB to TTL converter and plug it in again. **The serial port number that disappears after unplugging is the serial port of the USB to TTL converter.**



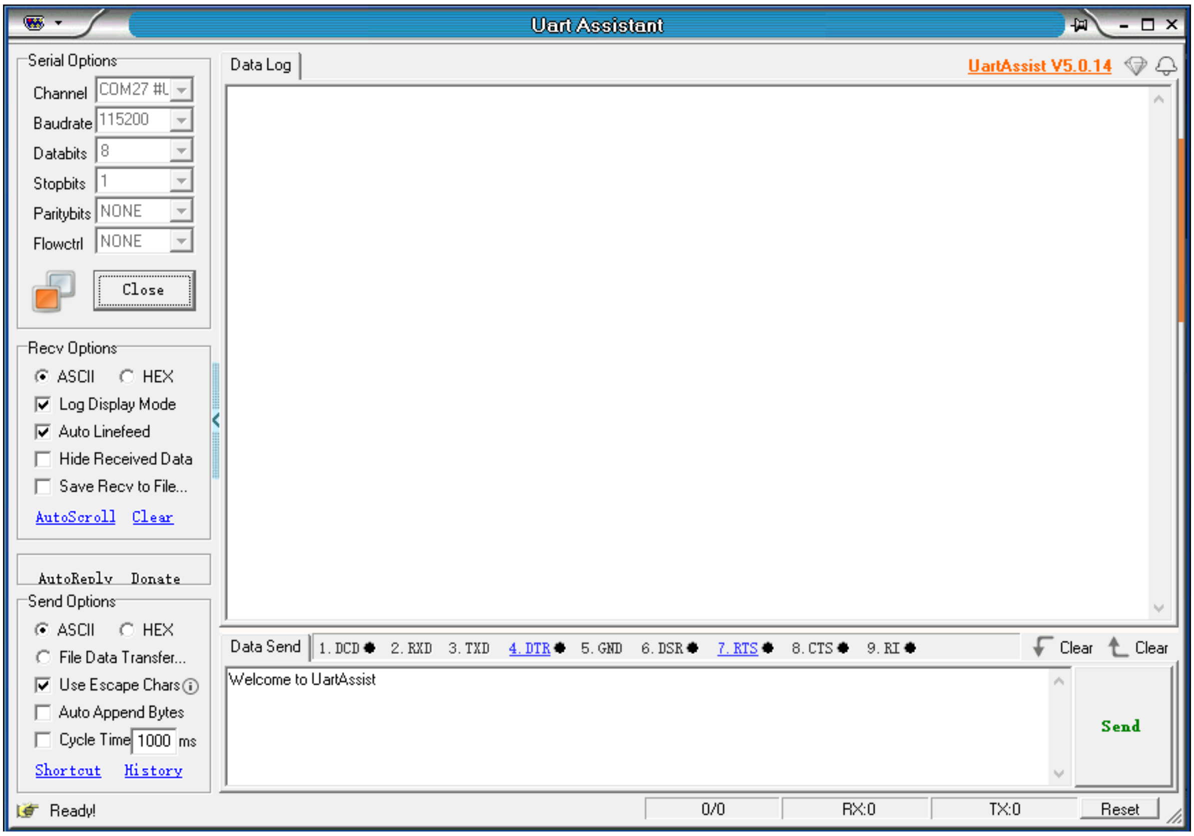
Channel COM27 #L
Baudrate 115200
Databits 8
Stopbits 1
Paritybits NONE
Flowctrl NONE

Recv Options: ● ASCII ○ HEX
☑ Log Display Mode
☑ Auto Linefeed
□ Hide Received Data
□ Save Recv to File...

Send Options: ● ASCII ○ HEX
○ File Data Transfer...
☑ Use Escape Chars
□ Auto Append Bytes
□ Cycle Time 1000 ms

Welcome to UartAssist

Ready! 0/0 RX:0 TX:0



Channel COM27 #L
Baudrate 115200
Databits 8
Stopbits 1
Paritybits NONE
Flowctrl NONE

Recv Options: ● ASCII ○ HEX
☑ Log Display Mode
☑ Auto Linefeed
□ Hide Received Data
□ Save Recv to File...

Send Options: ● ASCII ○ HEX
○ File Data Transfer...
☑ Use Escape Chars
□ Auto Append Bytes
□ Cycle Time 1000 ms

Welcome to UartAssist

Ready! 0/0 RX:0 TX:0

## Run the code

We open CanMV IDE and connect K230 to the computer.

Press Ctrl + N to create a new code and delete all automatically generated code content

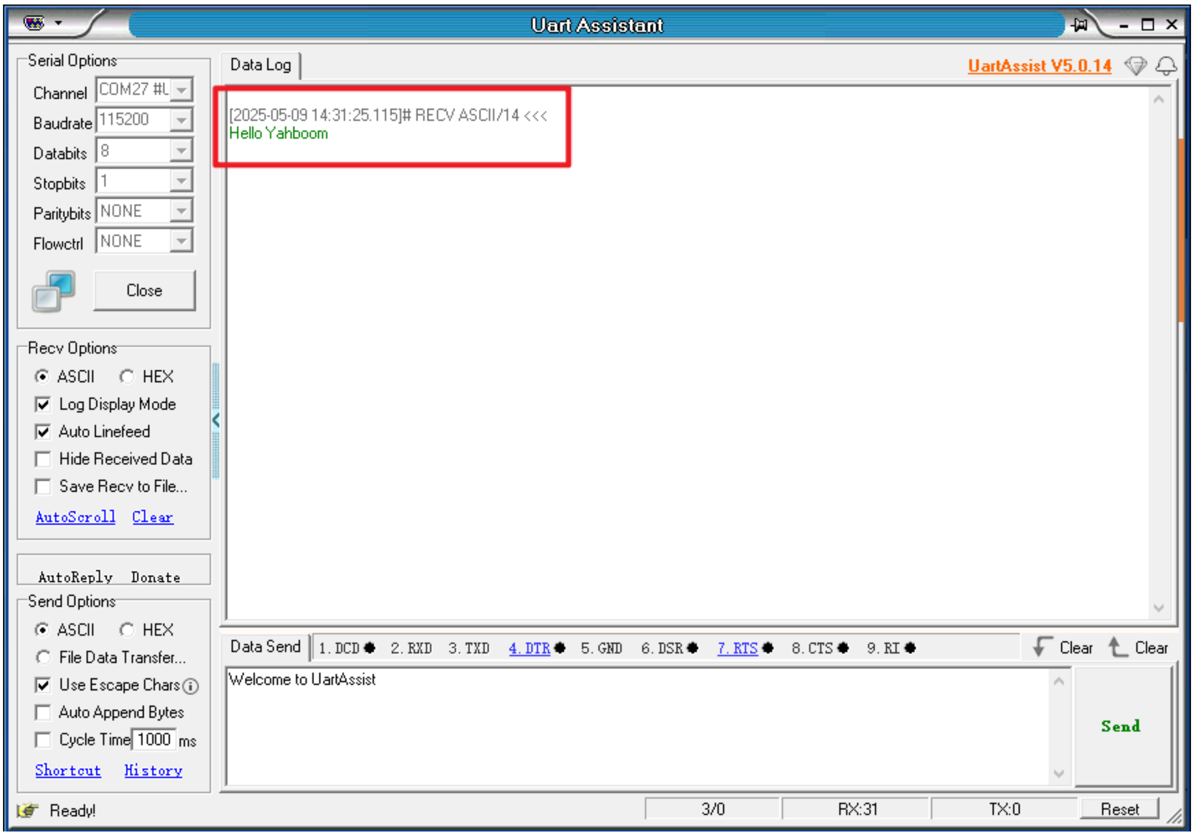
Copy the following code and paste it into the IDE [Source Code/02.Basic/05.uart.py]

```
# 导入亚博智能串口通信库
# (Import Yahboom UART communication library)
```

```
from ybUtils.YbUart import YbUart

# 创建串口实例，设置波特率为115200
# (Create UART instance with baud rate set to 115200)
# 波特率是指每秒传输的比特数，115200是常用的高速通信速率
# (Baud rate refers to bits per second, 115200 is a commonly used high-speed
communication rate)
uart = YbUart(baudrate=115200)

# 发送数据到连接的设备
# (Send data to the connected device)
# 发送字符串"Hello Yahboom"并附加换行符
# (Send the string "Hello Yahboom" with a newline character)
uart.send("Hello Yahboom\n")

# 无限循环，持续监听串口数据
# (Infinite loop to continuously monitor serial port data)
while True:
    # 读取串口接收到的数据
    # (Read data received from the serial port)
    # 如果没有数据，将返回None或空字符串
    # (If no data is available, it will return None or an empty string)
    data = uart.read()

    # 检查是否接收到数据
    # (Check if data was received)
    if data:
        # 将接收到的数据打印到控制台
        # (Print the received data to the console)
        print(data)

# 注意：由于上面的无限循环，这段代码实际上永远不会执行
# (Note: Due to the infinite loop above, this code will never execute)
# 关闭串口，释放资源
# (Close the UART and release resources)
uart.deinit()
```

Click the green run button in the lower left corner of CanMV IDE

It can be observed that the serial port assistant received the message sent by K230



Serial Options
Channel COM27 #L
Baudrate 115200
Databits 8
Stopbits 1
Paritybits NONE
Flowctrl NONE

Recv Options
● ASCII ○ HEX
☑ Log Display Mode
☑ Auto Linefeed
□ Hide Received Data
□ Save Recv to File...

Send Options
● ASCII ○ HEX
○ File Data Transfer...
☑ Use Escape Chars
□ Auto Append Bytes
□ Cycle Time 1000 ms

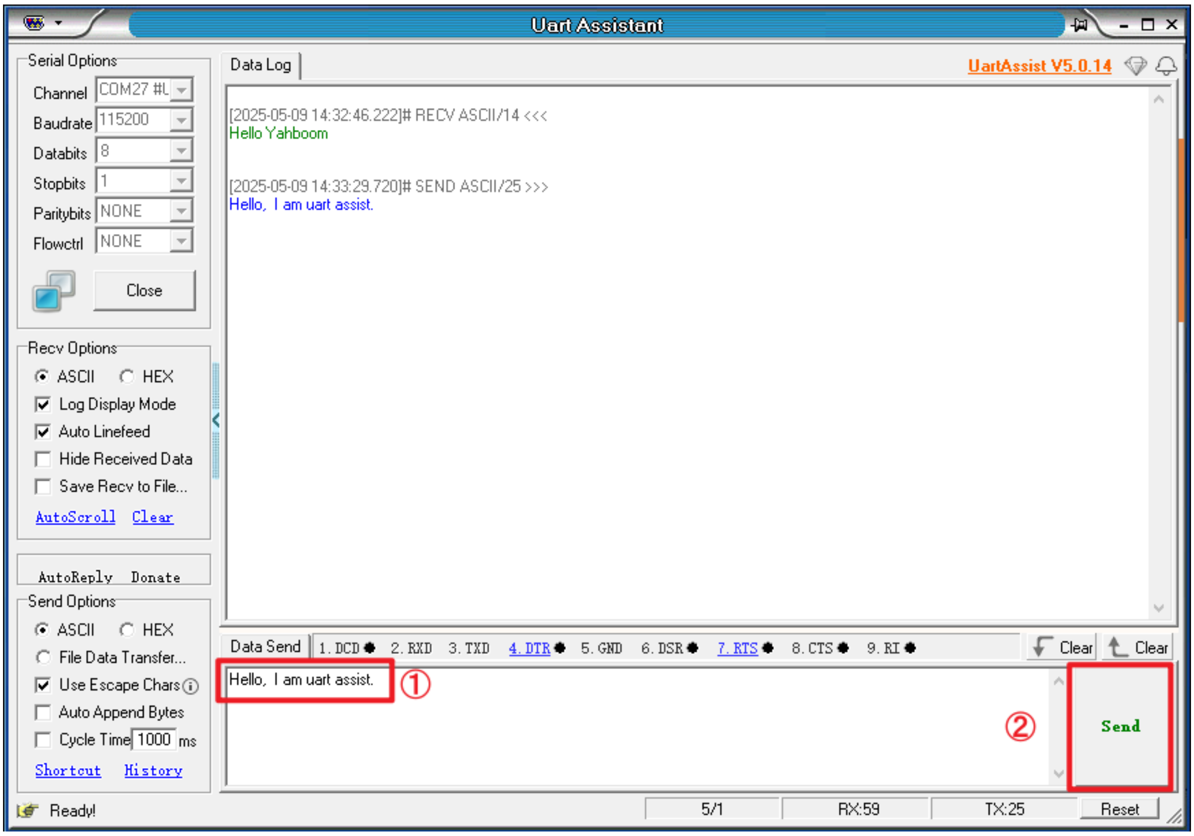
```
[2025-05-09 14:31:25.115]# RECV ASCII/14 <<<
Hello Yahboom
```

Welcome to UartAssist

Ready! 3/0 RX:31 TX:0

> Corresponding to the code
>
> ```
> uart.send("Hello Yahboom\n")
> ```

Then we try to use the serial port assistant to send information to K230



Serial Options
Channel COM27 #L
Baudrate 115200
Databits 8
Stopbits 1
Paritybits NONE
Flowctrl NONE

Recv Options
● ASCII ○ HEX
☑ Log Display Mode
☑ Auto Linefeed
□ Hide Received Data
□ Save Recv to File...

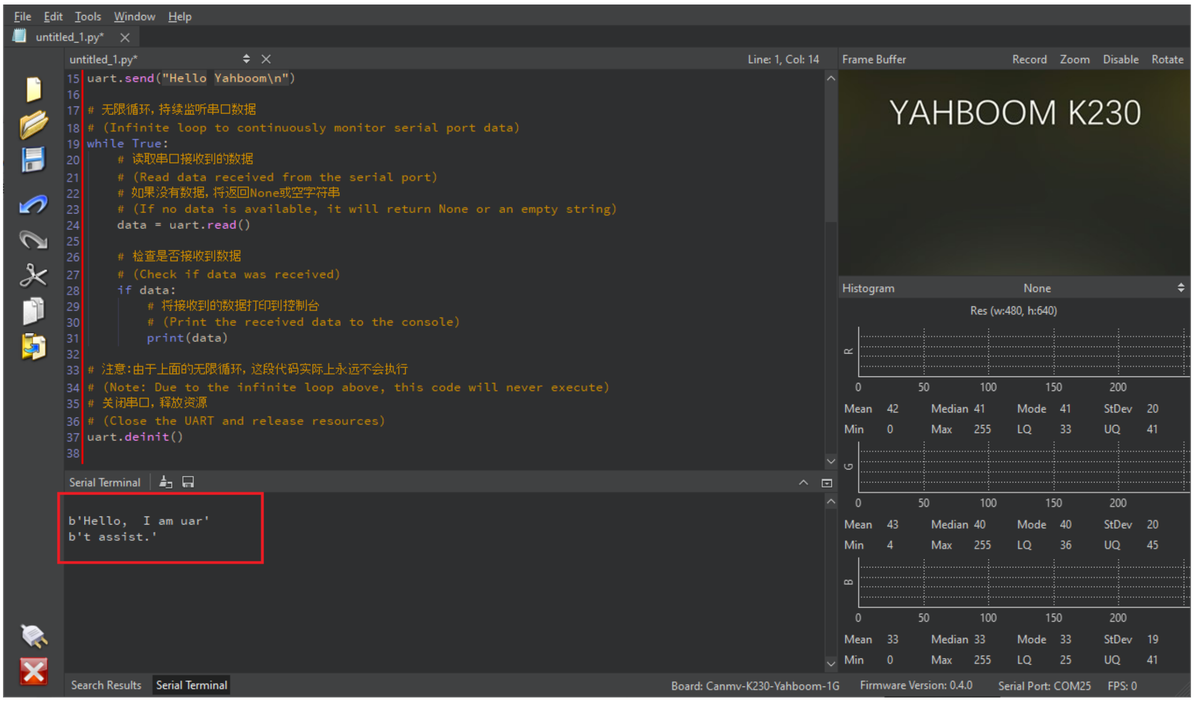
Send Options
● ASCII ○ HEX
○ File Data Transfer...
☑ Use Escape Chars
□ Auto Append Bytes
□ Cycle Time 1000 ms

```
[2025-05-09 14:32:46.222]# RECV ASCII/14 <<<
Hello Yahboom

[2025-05-09 14:33:29.720]# SEND ASCII/25 >>>
Hello, I am uart assist.
```

Hello, I am uart assist.

Ready! 5/1 RX:59 TX:25

The following output is observed in the serial terminal of CanMV IDE



```python
uart.send("Hello Yahboom\n")

# 无限循环，持续监听串口数据
# (Infinite loop to continuously monitor serial port data)
while True:
    # 读取串口接收到的数据
    # (Read data received from the serial port)
    # 如果没有数据，将返回None或空字符串
    # (If no data is available, it will return None or an empty string)
    data = uart.read()

    # 检查是否接收到数据
    # (Check if data was received)
    if data:
        # 将接收到的数据打印到控制台
        # (Print the received data to the console)
        print(data)

# 注意:由于上面的无限循环，这段代码实际上永远不会执行
# (Note: Due to the infinite loop above, this code will never execute)
# 关闭串口，释放资源
# (Close the UART and release resources)
uart.deinit()
```

```
b'Hello,  I am uar'
b't assist.'
```

> Here are a few common questions:
>
> 1. Why is the string in the format of b'xxxxxxxxxx (what is sent)'?
>
>    A: The received information is of bytes data type by default. You can use the .decode method to convert it to a string.
>
>    Note: Chinese characters cannot be accepted here for the time being.
>
> 2. Why does the comma "," become "\xef\xbc\x8c" ?
>
>    A: Because Chinese commas are used when sending, the bytes format of Chinese characters is like this.
>
>    *Update: In the read method, setting decode=True can parse most Chinese characters normally.
>
> 3. Why is it divided into two lines?
>
>    Because the default receive buffer size is 128 bytes, any data exceeding the buffer size will be treated as a new receive.
>
>    - The buffer size can be modified

# FAQ & Supplements

## USB to TTL Module

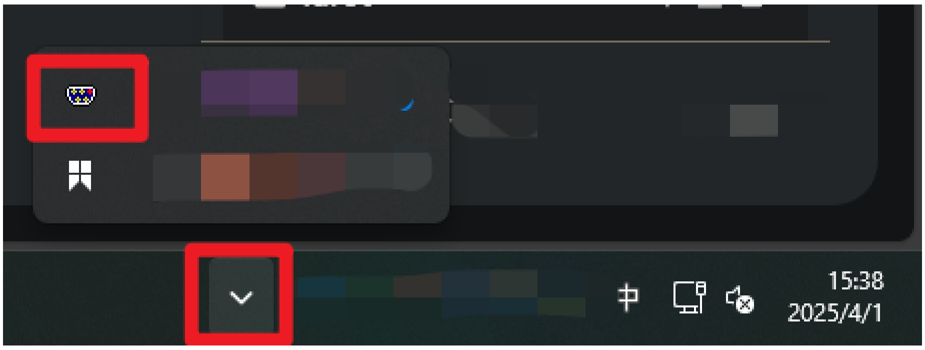
Why does K230 need a USB to TTL module to communicate with a computer?

1. Interface differences: The K230's communication ports use TTL (transistor-transistor logic) serial ports, while modern computers usually only have USB ports and do not directly provide TTL serial ports.
2. Level conversion: TTL serial ports use 0V and 3.3V/5V level standards, while USB uses different level standards. The USB to TTL module can safely convert these levels to prevent hardware damage.
3. Driver support: The USB to TTL module has the necessary circuits and chips (such as CH340, CP2102, FTDI, etc.) built in to enable the computer to recognize the USB connection as a virtual serial port (COM port).

This module is essentially a "translator" that allows devices using different communication standards to exchange data smoothly.

## Close the serial port assistant

By default, when you click the close button in the upper right corner, the serial port assistant will not be completely closed, but hidden in the tray icon in the lower right corner of the system.



If you reopen a serial port assistant software when the serial port assistant is not completely closed and click [Open], an error message will be displayed saying that the serial port is occupied.