Using Serial port

RDK X5 enables UART1 on 40pin by default, physical pins 8 and 10, IO voltage 3.3V.
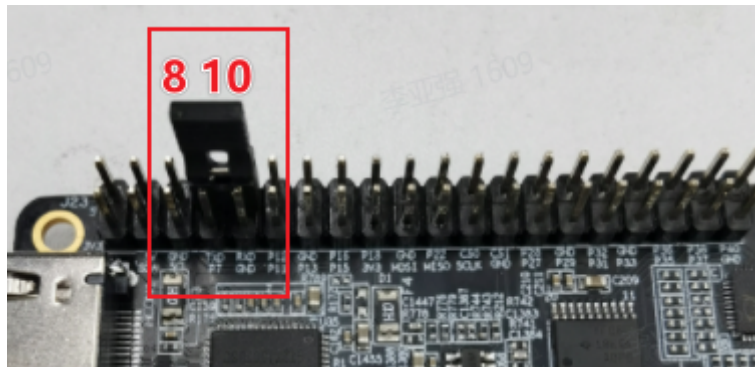
Please refer to `/app/40pin_samples/test_serial.py` for details on how to use the serial port.

Loopback test

Connect TXD and RXD in hardware, then run the test program to perform write and read operations. The expected result is that the read data should be exactly the same as the written data.

Hardware connection

Connect TXD and RXD together directly on the hardware using a jumper cap.

Test procedure

- Run `python3 /app/40pin_samples/test_serial.py`
- From the printed serial devices (where `/dev/ttyS0` is the system debugging port and it is not recommended to test it unless you fully understand its purpose), select the bus number and the chip number as input options. For example, select `/dev/ttyS3` for testing, press Enter to confirm, and enter the baud rate parameter:

```
List of enabled UART:
/dev/ttyS0  /dev/ttyS1  /dev/ttyS3  /dev/ttyUSB0

Please enter the name of the serial port to be tested:/dev/ttyS3
Please enter the baud rate (9600,19200,38400,57600,115200,921600):921600
Serial<id=0x7f819dcac0, open=True>(port='/dev/ttyS3', baudrate=921600,
bytesize=8, parity='N', stopbits=1, timeout=1, xonxoff=False, rtscts=False,
dsrdtr=False)
```

- After the program runs correctly, it will continuously print `Send: AA55` and `Recv: AA55`:

```
Starting demo now! Press CTRL+C to exit
Send: AA55
Recv: AA55
```

Test code

```python
#!/usr/bin/env python3

import sys
import os
import time

# Import Python serial library
import serial
import serial.tools.list_ports

def serialTest():
    print("List of enabled UART:")
    os.system('ls /dev/tty[a-zA-Z]*')
    uart_dev = input("Please enter the name of the serial port device to test:")

    baudrate = input("Please enter the baud rate
(9600,19200,38400,57600,115200,921600):")
    try:
        ser = serial.Serial(uart_dev, int(baudrate), timeout=1) # 1s timeout
    except Exception as e:
        print("Open serial failed!\n")

    print(ser)

    print("Starting demo now! Press CTRL+C to exit")

    while True:
        test_data = "AA55"
        write_num = ser.write(test_data.encode('UTF-8'))
        print("Send: ", test_data)

        received_data = ser.read(write_num).decode('UTF-8')
        print("Recv: ", received_data)

        time.sleep(1)

    ser.close()
    return 0

if __name__ == '__main__':
    if serialTest() != 0:
        print("Serial test failed!")
    else:
        print("Serial test success!")
```