

stm_ K210 QR code recognition

stm_ K210 QR code recognition

1. Communication between k210 and stm32

1.1 Experimental premises

1.2 Experimental wiring

1.3 Main code explanation

1.4 experimental phenomena

1. Communication between k210 and stm32

1.1 Experimental premises

This tutorial uses stm32C8T6, and k210 requires running the program in * * K210-AI (stm32_pico_arduino) * * to start the experiment

Stm32 * 1

K210 perspective module * 1 (requires SD card (with AI model inside) and camera)

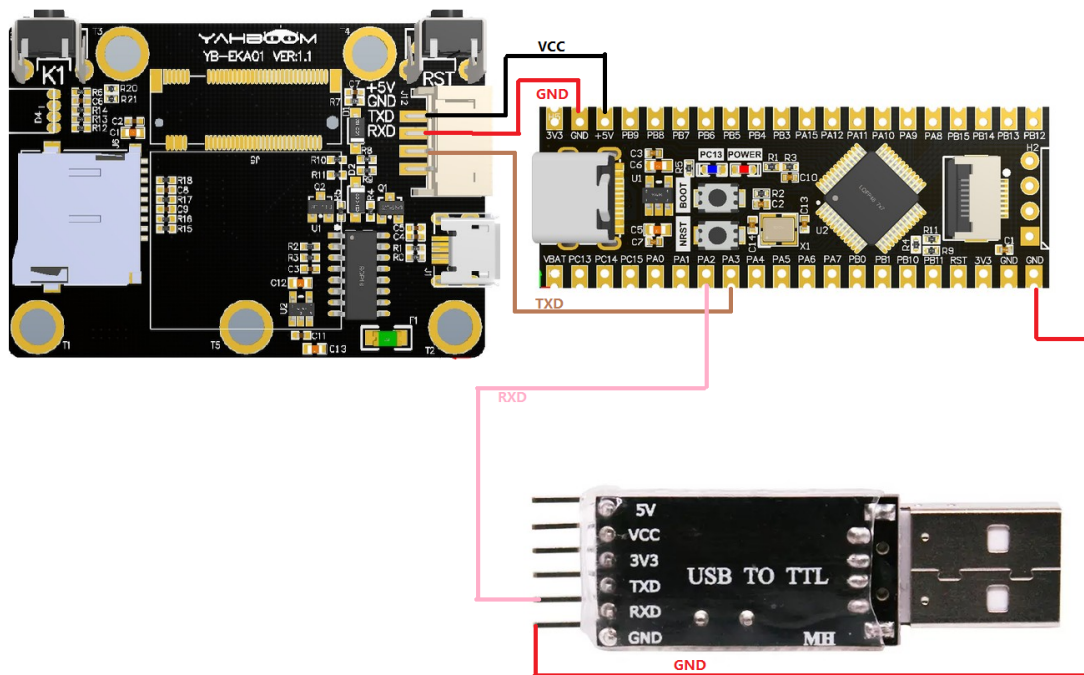
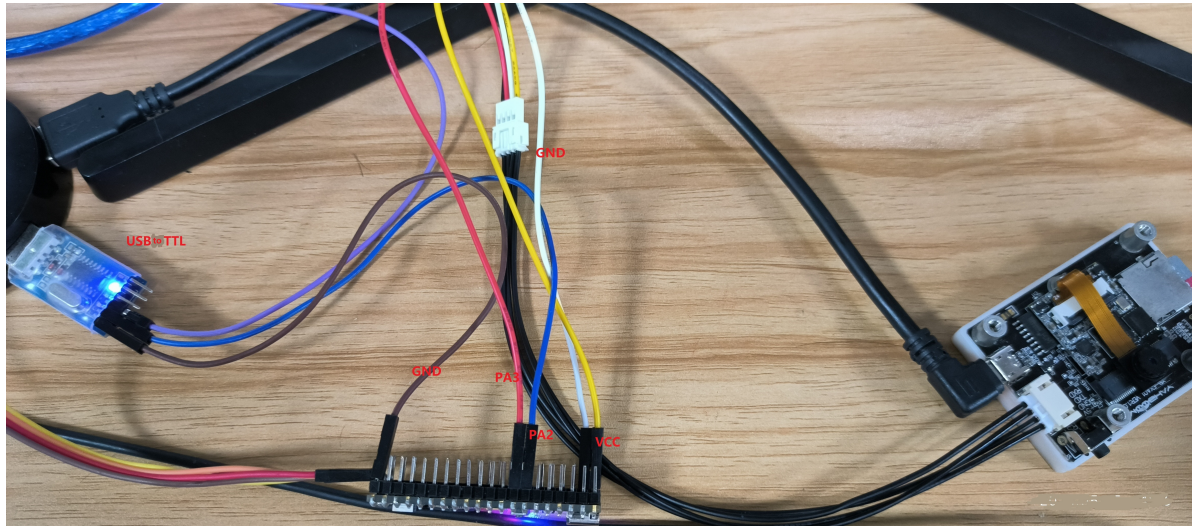
USB to TTL module * 1

1.2 Experimental wiring

stm32	usb to ttl
PA2	RXD
GND	GND

STM32	k210
PA3	TXD
GND	GND
VCC	5V

Wiring as shown in the diagram:



This type of wiring is not necessary for the RXD of k210 and the TXD of USB to TTL, as it was not used in the experiment.

1.3 Main code explanation

```
int main()
{
    //.....
    while(1)
    {
        if (k210_msg.class_n != 0)
        {
            if(k210_msg.class_n == 3)
            {

sprintf(buff_com, "x=%d,y=%d,w=%d,h=%d\r\n", k210_msg.x, k210_msg.y, k210_msg.w, k210_msg.h);

                USART2_Send_ArrayU8((uint8_t*)buff_com, strlen(buff_com));
            }
        }
    }
}
```

```

        sprintf(buff_com, "str = %s\r\n", k210_msg.msg_msg);
        USART2_Send_ArrayU8((uint8_t*)buff_com, strlen(buff_com));

        k210_msg.class_n = 0;
    }

}

}

}

```

After the above program, if you are running this routine, k210_ The members of the msg structure have corresponding values and are processed through serial port printing

K210_Msg: is a structure that receives information, and its main members are

- X: is the horizontal coordinate of the top left corner of the recognized box (range: 0-240)
- Y: is the vertical coordinate of the upper left corner of the identified box (range: 0-320)
- W: is the width of the recognized box (range: 0-240)
- H: The length of the recognized box (range: 0-320)
- ID: is the recognized label
- Class_n: Routine number
- Msg_Msg [20]: Valid data

After receiving and processing data, k210_ Each member of the msg will store valid information. If you want to develop it again, call K210 directly_ Members of msg are sufficient

1.4 experimental phenomena

1. After connecting the cable, the K210 perspective module runs offline. Please check 【6.2 K210 as coprocessor】 -- 【ReadMe】

2. Set the serial port assistant to the interface shown in the figure

The image shows a serial port configuration window. The 'COM Configs' section includes dropdown menus for Channel (COM1), Baudrate (115200), Paritybits (NONE), Databits (8), Stopbits (1), and Flowctrl (NONE). Below these is a 'Close' button with a red circular icon. The 'Recv Options' section has radio buttons for ASCII (selected) and HEX, and checkboxes for Log display mode, Auto linefeed, Hide received data, and Recv save to file... (all unchecked). It also has 'AutoScroll' and 'Clear' links. The 'Send Options' section has radio buttons for ASCII (selected) and HEX, and checkboxes for Enable escape chars, AT CMD auto CRLF, Auto append bytes, Send from file ..., and Period (1000 ms). It also has 'Shortcut' and 'History' links.

3. Then run the QR code recognition routine, and the serial assistant will print out the important information transmitted from k210 to stm32, as shown in the following figure

```
x=35, y=0, w=198, h=195
str = hello world
x=34, y=1, w=195, h=195
str = hello world
x=32, y=7, w=195, h=197
str = hello world
x=37, y=8, w=192, h=191
str = hello world
x=39, y=9, w=193, h=192
str = hello world
x=42, y=11, w=188, h=192
str = hello world
x=43, y=10, w=189, h=190
str = hello world
x=43, y=12, w=189, h=187
str = hello world
x=55, y=19, w=187, h=189
str = hello world
x=60, y=22, w=187, h=188
str = hello world
x=61, y=23, w=186, h=189
str = hello world
x=61, y=26, w=184, h=185
str = hello world
x=60, y=24, w=185, h=187
str = hello world
x=56, y=21, w=190, h=185
```

QR code recognition only transmits the five Member variable of k210msg, namely, x, y, w, h, and msg.

