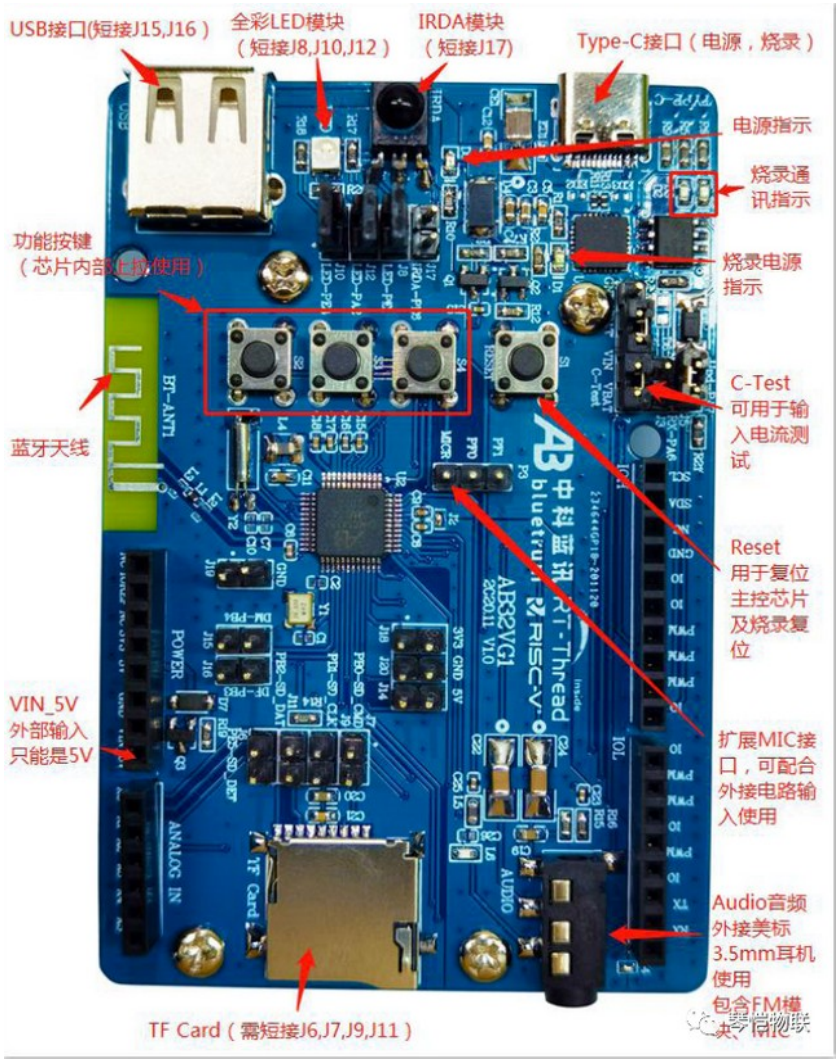
Bluetrum AB32VG1 RISC-V development board trial notes (1) Blink that LED

Original post: https://mp.weixin.qq.com/s/YPbBtZMiqWx3z348mWjuIg

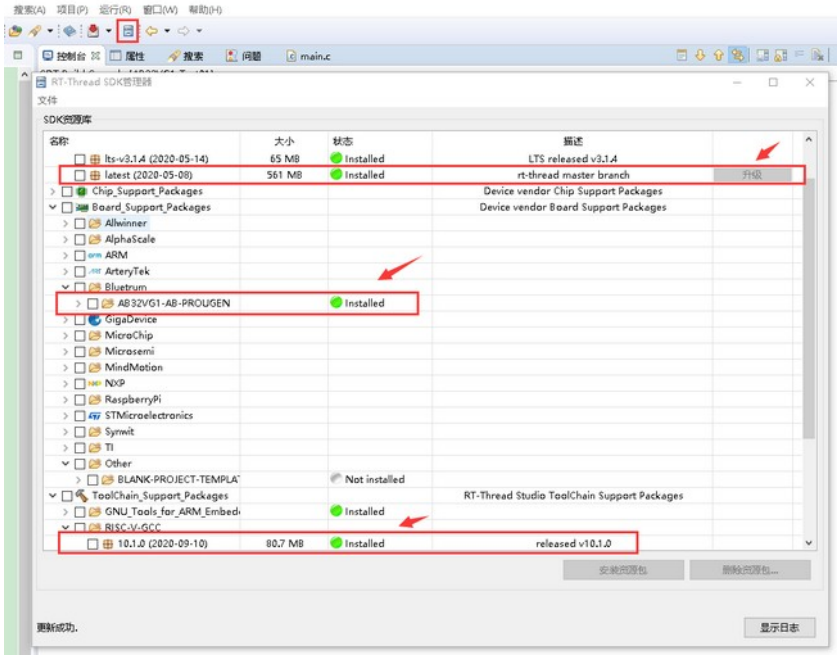
The AB32VG1 development board is based on Bluetrum's AB5301A RISC-V MCU.

Resources on board:

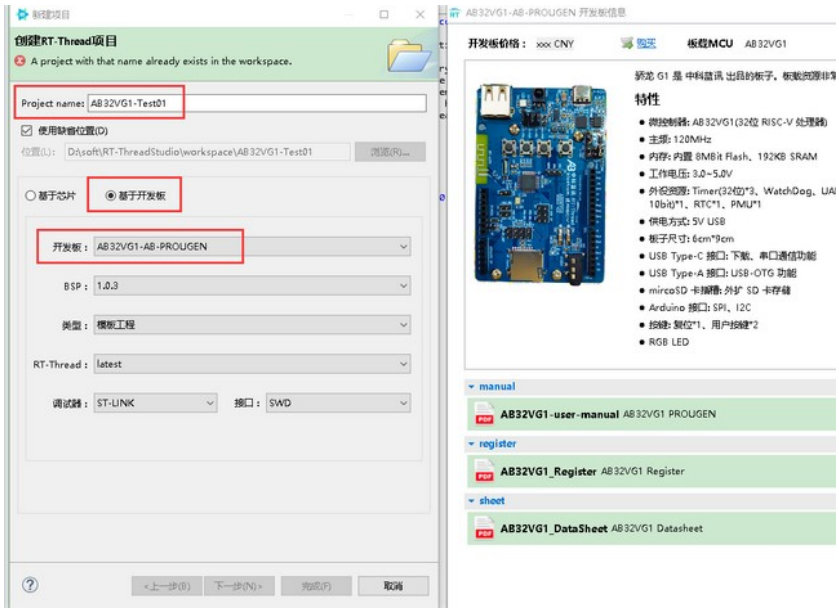
- CPU: AB5301A (LQFP48 package, 120MHz, 192KB RAM, 8 Mbit flash, ADC, PWM, USB, UART, I2C and other resources)
- Bluetooth module
- FM module
- One TF Card interface
- One USB interface
- One IIC interface
- One audio interface (CTIA US standard)
- Six ADC channels broken out on female header
- Six PWM channels broken out on female headers
- One RGB LED
- One infrared remote-control receiver
- One reset button, three push buttons
- Board size: 6cm*9cm
- The 2.54mm I/O connectors are compatible with the Arduino Uno expansion interface.



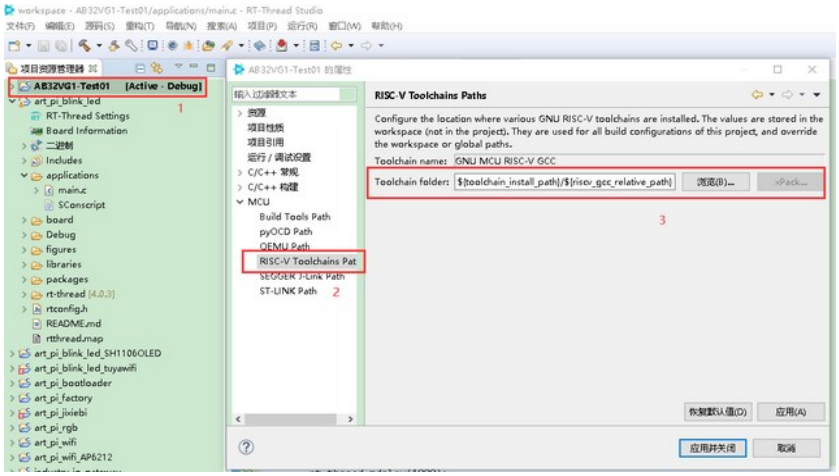
Start the program RT-Thread Studio-SDK Manager, and make sure that the following are installed:



"File" – "New" – "rt-thread project":



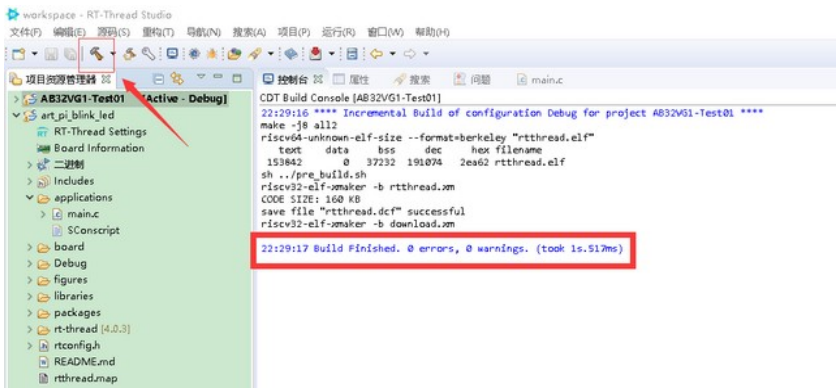
Click to build the project – right-click – "Properties", make sure that the following settings are correct, otherwise it will fail to compile:



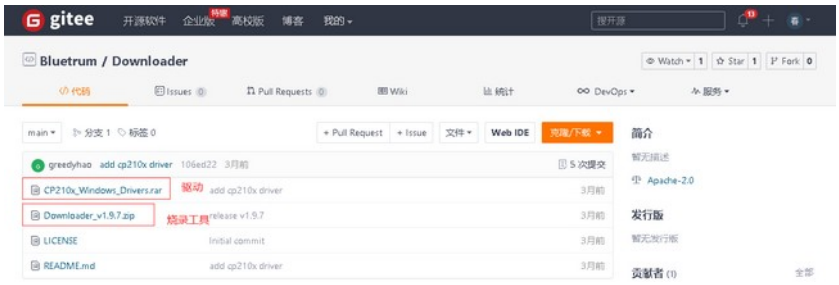
The path in the screenshot above is:

C:\RT-ThreadStudio\repo\Extract\ToolChain_Support_Packages\RISC-V\RISC-V-GCC\10.1.0\bin

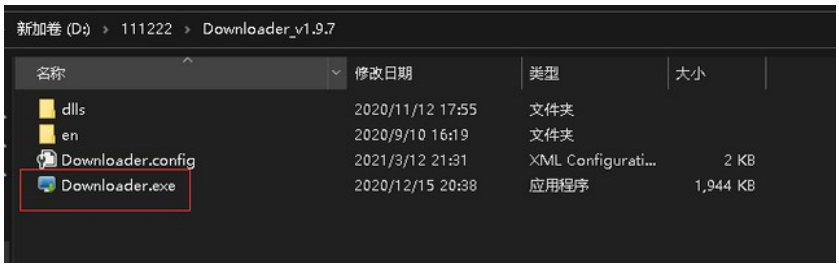
Start compiling:



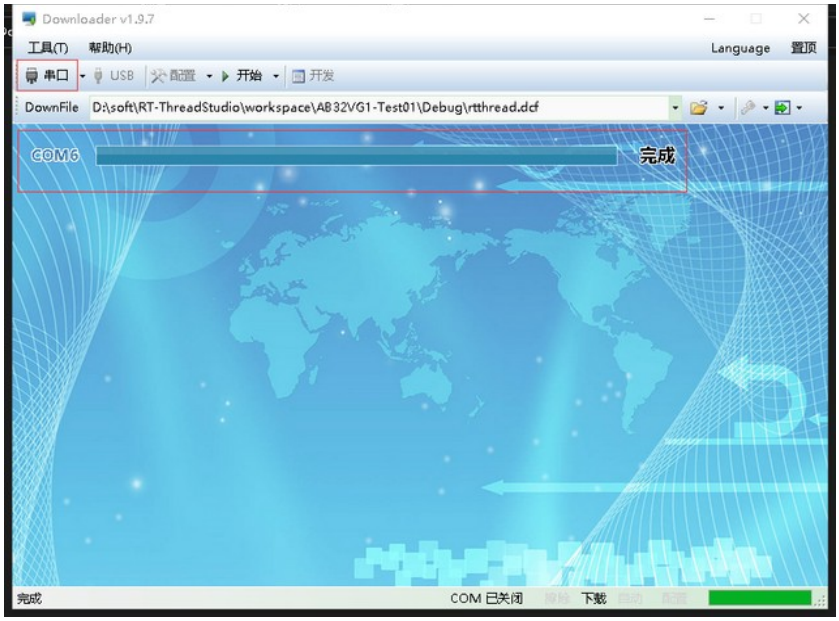
The board cannot be flashed with RT-Thread Studio, the official tool should be used (and the serial port driver must be installed): https://github.com/BLUETRUM/Downloader



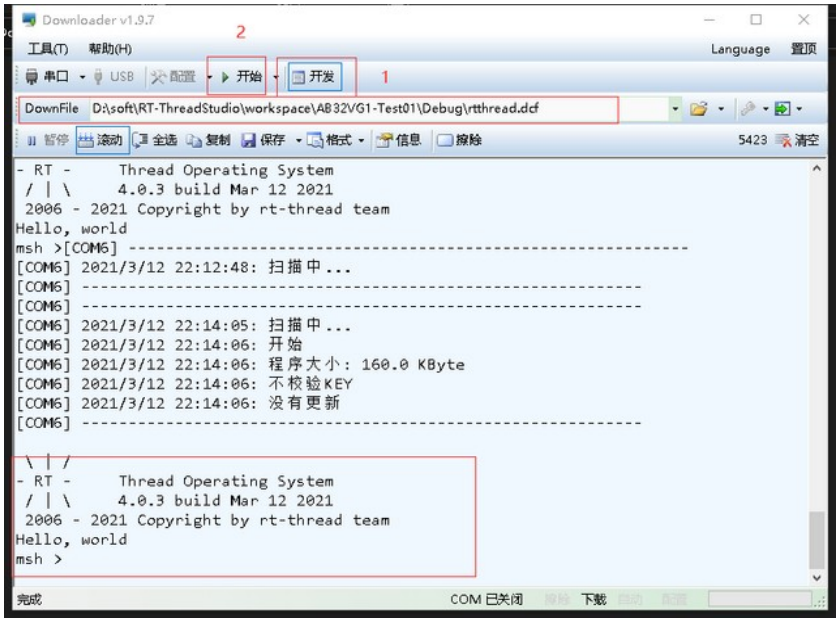
Unzip tool:



Double-click to run:



Pay attention to the sequence of operations in the screenshot below, otherwise the rt-thread command line will not appear.



Again, remember to install the serial port driver. It is best to connect the board with type-C first, and connect it to the computer with USB to install the driver. If the driver is installed without connecting the board in advance and the driver is abnormal, uninstall and install it again. For example "Scanning" keeps appearing in the picture below:

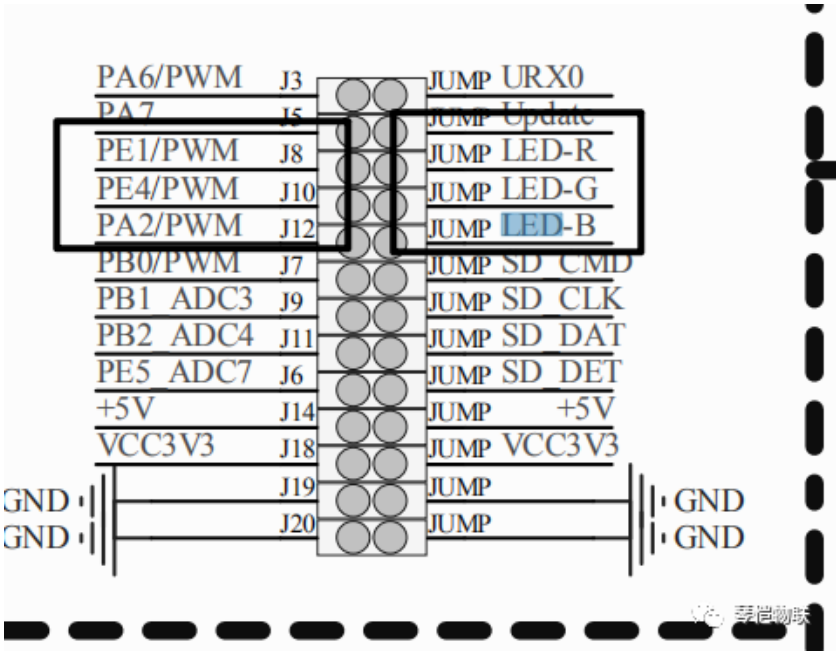
At this point, the LED blinks: video.

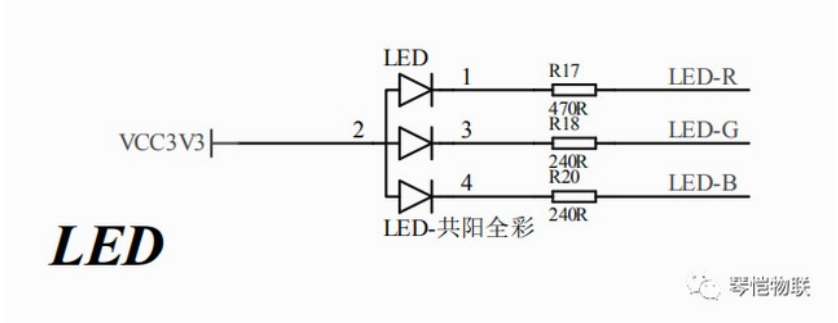
Bluetrum AB32VG1 RISC-V development board trial notes (2) RGB LED

Original post: https://club.rt-thread.org/ask/article/5bc04b7f8455c6e5.html

There is an RGB LED on the board:

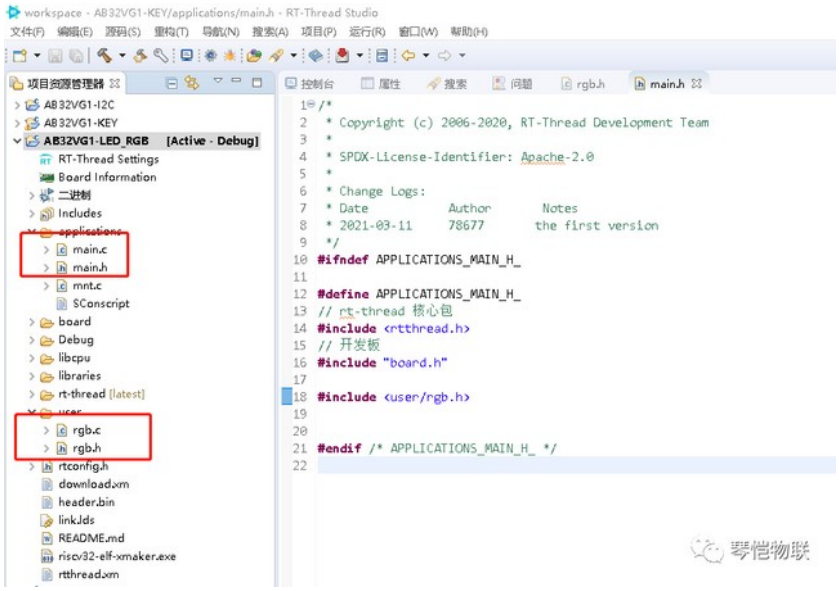
The schematic diagram is as follows:





Pin PA2 corresponds to the blue light, PE1 to the red light, and PE4 to the green light. The RGB LED is common anode.

RT-Thread Studio creates a new project, and the following files are added or modified:



main.c

```
#include <rtthread.h>
#include "board.h"
#include "main.h"

static void rgb_thread_entry(void* p) {
    RGB_Init();
    while (1) {
      rt_thread_mdelay(1000);
      RGB_Blue(1);
      rt_thread_mdelay(1000);
      rt_thread_mdelay(1000);
      RGB_Green(1);
      rt_thread_mdelay(1000);
      RGB_Red(1);
    }
}

static int Thread_RGB(void) {
    rt_thread_t thread = RT_NULL;
    thread = rt_thread_create("rgb", rgb_thread_entry, RT_NULL, 512, 10, 10);
    if (thread == RT_NULL) {
        rt_kprintf("Thread_GRB Init ERROR");
        return RT_ERROR;
    }
    rt_thread_startup(thread);
}

INIT_APP_EXPORT(Thread_RGB); // 线程自动初始化

int main(void) {
    while (1) {
        rt_thread_mdelay(1000); //空延时
    }
    return 0;
}
```

main.h

```
#ifndef APPLICATIONS_MAIN_H_
```

```
#define APPLICATIONS_MAIN_H_
// rt-thread 核心包
#include <rtthread.h>
// 开发板
#include "board.h"
#include <user/rgb.h>

#endif /* APPLICATIONS_MAIN_H_ */
```

rgb.c

```
#include "main.h"

struct Led_s Led;

void RGB_Init(void) {
    // 获得 led 成员
    Led.LED_R = rt_pin_get("PE.1");
    Led.LED_G = rt_pin_get("PE.4");
    Led.LED_B = rt_pin_get("PA.2");
    // 设置引脚输出方式
    rt_pin_mode(Led.LED_R, PIN_MODE_OUTPUT);
    rt_pin_mode(Led.LED_G, PIN_MODE_OUTPUT);
    rt_pin_mode(Led.LED_B, PIN_MODE_OUTPUT);
}

void RGB_Red(rt_bool_t on) {
    if (on) {
        rt_pin_write(Led.LED_R, PIN_LOW);
    }else {
        rt_pin_write(Led.LED_R, PIN_HIGH);
    }
    rt_pin_write(Led.LED_G, PIN_HIGH);
    rt_pin_write(Led.LED_B, PIN_HIGH);
}

void RGB_Blue(rt_bool_t on) {
    if (on) {
        rt_pin_write(Led.LED_B, PIN_LOW);
    }else {
        rt_pin_write(Led.LED_B, PIN_HIGH);
    }
    rt_pin_write(Led.LED_G, PIN_HIGH);
    rt_pin_write(Led.LED_R, PIN_HIGH);
}

void RGB_Green(rt_bool_t on) {
    if (on) {
        rt_pin_write(Led.LED_G, PIN_LOW);
    }else {
        rt_pin_write(Led.LED_G, PIN_HIGH);
    }
    rt_pin_write(Led.LED_R, PIN_HIGH);
    rt_pin_write(Led.LED_B, PIN_HIGH);
}
```

rgb.h

```
#ifndef USER_INC_RGB_H_
#define USER_INC_LED_H_

struct Led_s {
    uint8_t LED_R;
    uint8_t LED_B;
    uint8_t LED_G;
};

void RGB_Init(void);
void RGB_Red(rt_bool_t on);
void RGB_Blue(rt_bool_t on);
void RGB_Green(rt_bool_t on);
#endif /* USER_INC_RGB_H_ */
```

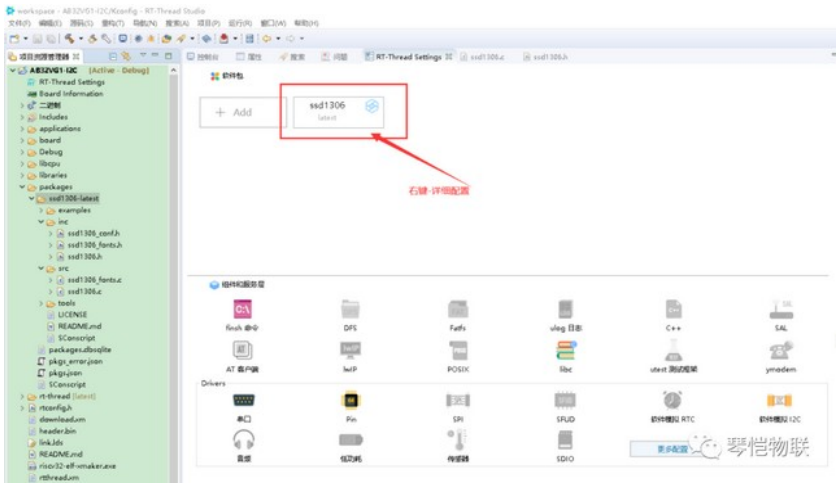
Note: If the three LEDs are on at the same time, the RGB is white.

To compile and flash, please refer to **Bluetrum AB32VG1 RISC-V Development Board Trial Notes (1) Blink that LED**.

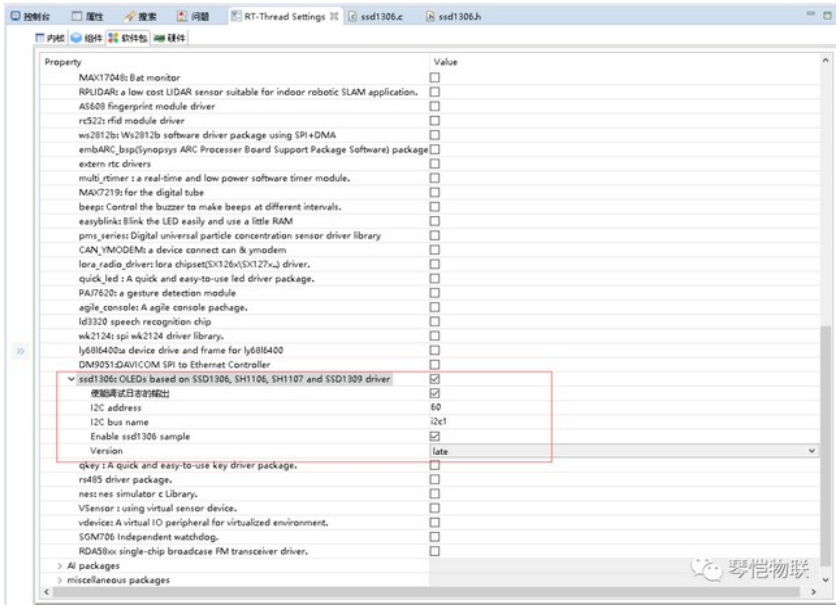
Bluetrum AB32VG1 RISC-V development board trial notes (3) I2C OLED display with SSD1306 controller

Original post: https://mp.weixin.qq.com/s/bSFhKHAZ0VJ8C8yppcgLVA

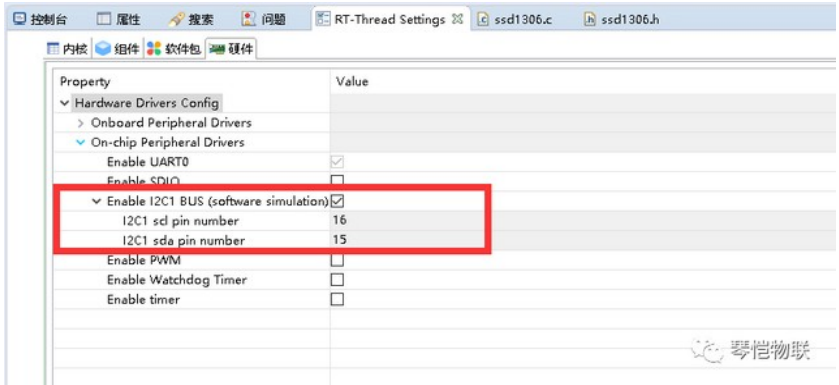
Create a new RT-Thread Studio project and add components:



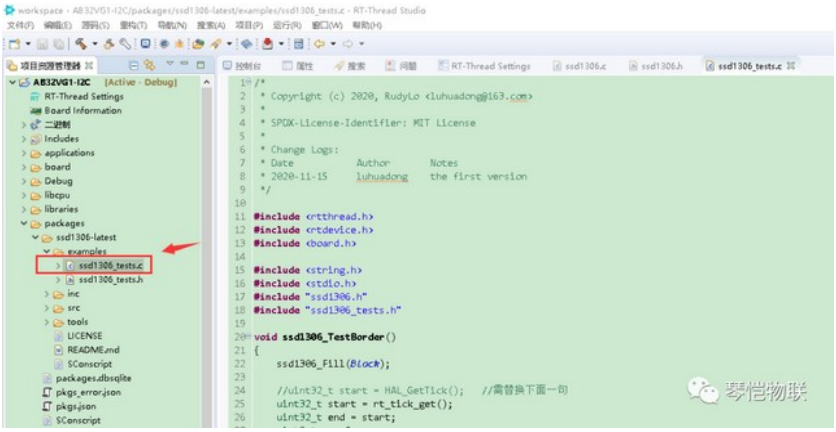
Configure the details again:



The following configuration should be the default and has no effect:



The sample code needs to be modified a bit:



ssd1306_tests.c

```
#include <rtthread.h>
#include <rtdevice.h>
#include <board.h>

#include <string.h>
#include <stdio.h>
#include "ssd1306.h"
#include "ssd1306_tests.h"

void ssd1306_TestBorder() {
    ssd1306_Fill(Black);

    //uint32_t start = HAL_GetTick();   //需替换下面一句
    uint32_t start = rt_tick_get();
    uint32_t end = start;
    uint8_t x = 0;
    uint8_t y = 0;
    do {
        ssd1306_DrawPixel(x, y, Black);

        if((y == 0) && (x < 127))
            x++;
        else if((x == 127) && (y < 63))
            y++;
        else if((y == 63) && (x > 0))
            x--;
        else
            y--;

        ssd1306_DrawPixel(x, y, White);
        ssd1306_UpdateScreen();

        //HAL_Delay(5);
        rt_thread_mdelay(5);
        //end = HAL_GetTick();
        end = rt_tick_get();
    } while((end - start) < 8000);

    //HAL_Delay(1000);   //需替换下面一句
    rt_thread_mdelay(5000);
}

void ssd1306_TestFonts() {
    ssd1306_Fill(Black);
    ssd1306_SetCursor(2, 0);
```

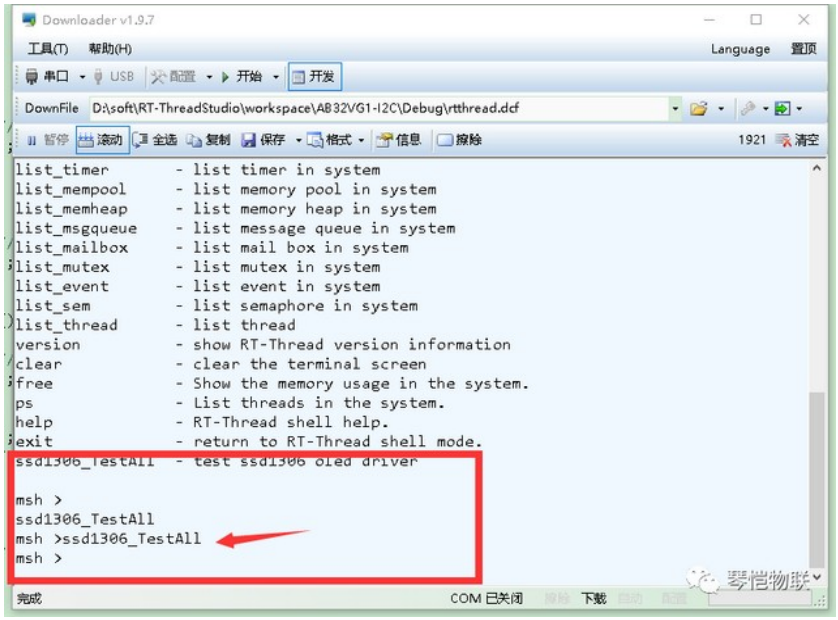
```
    ssd1306_WriteString("Font 16x26", Font_16x26, White);
    ssd1306_SetCursor(2, 26);
    ssd1306_WriteString("Font 11x18", Font_11x18, White);
    ssd1306_SetCursor(2, 26+18);
    ssd1306_WriteString("Font 7x10", Font_7x10, White);
    ssd1306_SetCursor(2, 26+18+10);
    ssd1306_WriteString("Font 6x8", Font_6x8, White);
    ssd1306_UpdateScreen();
}

void ssd1306_TestFPS() {
    ssd1306_Fill(White);

    //uint32_t start = HAL_GetTick();   //需替换下面一句
    uint32_t start = rt_tick_get();

    uint32_t end = start;
    int fps = 0;
    char message[] = "ABCDEFGHIJK";

    ssd1306_SetCursor(2,0);
    ssd1306_WriteString("Testing...", Font_11x18, Black);

    do {
        ssd1306_SetCursor(2, 18);
        ssd1306_WriteString(message, Font_11x18, Black);
        ssd1306_UpdateScreen();

        char ch = message[0];
        memmove(message, message+1, sizeof(message)-2);
        message[sizeof(message)-2] = ch;

        fps++;
        //end = HAL_GetTick();   //需替换下面一句
        end = rt_tick_get();
    } while((end - start) < 5000);

    //HAL_Delay(1000);
    rt_thread_mdelay(1000);

    char buff[64];
    fps = (float)fps / ((end - start) / 1000.0);
    snprintf(buff, sizeof(buff), "~%d FPS", fps);

    ssd1306_Fill(White);
    ssd1306_SetCursor(2, 18);
    ssd1306_WriteString(buff, Font_11x18, Black);
    ssd1306_UpdateScreen();
}

void ssd1306_TestLine() {
    ssd1306_Line(1,1,SSD1306_WIDTH - 1,SSD1306_HEIGHT - 1,White);
    ssd1306_Line(SSD1306_WIDTH - 1,1,1,SSD1306_HEIGHT - 1,White);
    ssd1306_UpdateScreen();
    return;
}

void ssd1306_TestRectangle() {
    uint32_t delta;

    for(delta = 0; delta < 5; delta ++) {
        ssd1306_DrawRectangle(1 + (5*delta),1 + (5*delta) ,SSD1306_WIDTH-1 -
(5*delta),SSD1306_HEIGHT-1 - (5*delta),White);
    }
```

```
    ssd1306_UpdateScreen();
    return;
}

void ssd1306_TestCircle() {
    uint32_t delta;

    for(delta = 0; delta < 5; delta ++) {
        ssd1306_DrawCircle(20* delta+30, 30, 10, White);
    }
    ssd1306_UpdateScreen();
    return;
}

void ssd1306_TestArc() {
    ssd1306_DrawArc(30, 30, 30, 20, 270, White);
    ssd1306_UpdateScreen();
    return;
}

void ssd1306_TestPolyline() {
    SSD1306_VERTEX loc_vertex[] = {
        {35,40},
        {40,20},
        {45,28},
        {50,10},
        {45,16},
        {50,10},
        {53,16}
    };

    ssd1306_Polyline(loc_vertex,sizeof(loc_vertex)/sizeof(loc_vertex[0]),White);
    ssd1306_UpdateScreen();
    return;
}

void ssd1306_TestAll() {
    ssd1306_Init();

    ssd1306_TestFPS();
    //HAL_Delay(3000);   //需替换下面一句
    rt_thread_mdelay(3000);

    ssd1306_TestBorder();

    ssd1306_TestFonts();
    //HAL_Delay(3000);   //需替换下面一句
    rt_thread_mdelay(3000);

    ssd1306_Fill(Black);
    ssd1306_TestRectangle();
    ssd1306_TestLine();
    //HAL_Delay(3000);   //需替换下面一句
    rt_thread_mdelay(3000);

    ssd1306_Fill(Black);
    ssd1306_TestPolyline();
    //HAL_Delay(3000);   //需替换下面一句
    rt_thread_mdelay(3000);

    ssd1306_Fill(Black);
    ssd1306_TestArc();
    //HAL_Delay(3000);   //需替换下面一句
    rt_thread_mdelay(3000);
```

```
    ssd1306_Fill(Black);
    ssd1306_TestCircle();
    //HAL_Delay(3000);   //需替换下面一句
    rt_thread_mdelay(3000);
}

#ifdef FINSH_USING_MSH
MSH_CMD_EXPORT(ssd1306_TestAll, test ssd1306 oled driver);
#endif
```

To compile and flash, please refer to **Bluetrum AB32VG1 RISC-V Development Board Trial Notes (1) Blink that LED**.

Command line execution:



See video.