

ESP8266 LUA脚本语言开发(13)
ESP8266 LUA开发基础入门篇备份(22)
ESP8266 SDK开发(34)
ESP8266 SDK开发基础入门篇备份(30)
GPRS Air202 LUA开发(11)
HC32F460(华大单片机)物联网开发(17)
HC32F460(华大单片机)学习开发(8)
NB-IOT Air302 AT指令和LUA脚本语言开发(27)
PLC(三菱PLC)基础入门篇(2)
STM32+Air724UG(4G模组)物联网开发(43)
STM32+BC26/260Y物联网开发(10)
STM32+CH395Q(以太网)物联网开发(24)
STM32+ESP8266(ZLESP8266A)物联网开发(1)
STM32+ESP8266+AIR202/302远程升级方案(16)
STM32+ESP8266+AIR202/302终端管理方案(6)
STM32+ESP8266+Air302物联网开发(57)
STM32+W5500物联网开发(14)
STM32F103物联网开发(49)
STM32G070物联网开发(8)
UCOSii操作系统(1)
W5500 学习开发(8)
编程语言C#(11)
编程语言Lua脚本语言基础入门篇(6)
编程语言Python(1)
单片机(LPC1778)LPC1778(2)
单片机(MSP430)开发基础入门篇(4)
更多

阅读排行榜

1. ESP8266使用详解(AT,LUA,SDK)(175550)
2. 1-安装MQTT服务器(Windows),并连接测试(109816)
3. 用ESP8266+android,制作自己的WIFI小车(ESP8266篇)(70906)
4. ESP8266刷AT固件与nodemcu固件(68706)
5. 有人WIFI模块使用详解(40205)
6. C#中public与private与static(38957)
7. (一)基于阿里云的MQTT远程控制(Android 连接MQTT服务器,ESP8266连接MQTT服务器实现远程通信控制----简单的连接通信)(38222)
8. 关于TCP和MQTT之间的转换(37069)
9. android 之TCP客户端编程(34119)
10. (一)Lua脚本语言入门(33299)

推荐排行榜

1. C#委托+回调详解(11)
2. 用ESP8266+android,制作自己的WIFI小车(ESP8266篇)(9)
3. 我的大学四年(7)
4. 用ESP8266+android,制作自己的WIFI小车(Android 软件)(6)
5. 关于stm32的正交解码(6)

最新评论

1. Re:Android之TCP服务器编程
客户端发消息给服务器时wireshark能看到服务器回给客户端的ack
--爱运动的机械师
2. Re:Android之TCP服务器编程
我按照您的方法做到现在,服务器能接收到客户端的消息,但是服务器发送不出消息,用wireshark看不到有发送包,能否指点一二

关于学习资料源码

1.打开CH32V30x_List.txt 可以看到所有提供的例程(官方提供的例程相当全面)

001-CH32V307VCT6核心板硬件
单片机开发相关

CH372DRV.EXE
EVT.rar
SCH.pdf

EXAM
PUB
CH32V30x_List.txt
CH32V30x_List_EN.txt

```

PUB:
-- CH32V30x评估板说明书.pdf: CH32V30x评估板说明及工程创建说明
-- CH32V30xSCH.pdf: CH32V30x评估板原理图
EXAM:
-- SRC
-- Core: 内核系统头文件
-- Debug: 延时函数、串口调试源文件及头文件
-- Ld: ld文件
-- Peripheral: 基本外设驱动源文件及头文件
-- Startup: 启动文件。针对用户使用的芯片型号包含下列相应的启动文件，同时在ch32v30x.h文件中选择匹配芯片型号定义
-- startup_ch32v30x_D8.S: 适用于CH32V303系列
-- startup_ch32v30x_D8C.S: 适用于CH32V307-CH32V305系列
-- ADC
-- ADC_DMA: ADC使用DMA采样例程
-- AnalogWatchdog: 模拟看门狗例程
-- Auto_Injection: 自动注入模式例程
-- Discontinuous_mode: 间断模式例程
-- DualADC_AlternateTrigger: 双ADC交替触发采样例程
-- DualADC_Combined_RegInjectionSimul: 双ADC混合同步规则+注入采样例程
-- DualADC_FastInterleaved: 双ADC快速交叉采样例程
-- DualADC_InjectionSimul: 双ADC同步注入采样例程
-- DualADC_RegSimul: 双ADC同步规则采样例程
-- DualADC_SlowInterleaved: 双ADC慢速交叉采样例程
-- ExtLines_Trigger: 外部触发ADC转换例程
-- Internal_Temperature: 内部温度传感器例程
-- BKP: BKP例程
-- CAN
-- Networking: CAN例程: 正常模式、标准帧和扩展帧数据收发
-- TestMode: 测试模式, 包括静默模式、回环模式、回环静默模式
-- Time-triggered: 时间触发通信模式
-- CRC: CRC例程
-- DAC
-- DAC_DMA: DAC使用DMA转换例程
-- DAC_Exit_9_Trig: 外部触发DAC转换例程
-- DAC_Noise_Generation: 噪声波形输出例程
-- DAC_Normal_OUT: 正常输出例程
-- DAC_Timer_Trig: 定时器触发DAC转换例程
-- DAC_Triangle_Generation: 三角波形输出例程
-- DualDAC_SineWave: 双DAC输出正弦波例程
-- DualDAC_Triangle: 双DAC输出三角波例程
-- DMA
-- DMA_MEM2MEM: 存储器到存储器模式例程
-- DMA_MEM2PERIP: 存储器到外设模式、外设到存储器模式例程见各外设例程
-- DVP
-- 仅适用于CH32V30x_D8C
-- DVP_TFTLCD: DVP操作OV2640摄像头视频模式在LCD显示图像例程
-- DVP_UART: DVP操作OV2640摄像头JPEG模式例程

-- ETH -- 仅适用于CH32V30x_D8C
-- 8_UART_Server: 8串口网络服务器演示例程
-- DHCP: DHCP自动获取ip功能演示例程
-- DNS: DHCP自动获取ip后, 再请求域名解析演示例程
-- ETH_internal_10BASE-T_PHY: 使用内部集成10M物理层收发以太网帧演示例程
-- ETH_MII_RMII: 使用外部MII/RMII接口物理层收发以太网帧演示例程
-- ETH_RGMII: 千兆ETH-数据链路层演示例程
-- NetLib: 网络协议栈库
-- TcpClient: Tcp client连接服务器后接收数据再回传演示例程
-- TcpServer: Tcp Server连接到客户端后接收数据并回传演示例程
-- UdpClient: Udp Client接收数据并回传演示例程
-- UdpServer: Udp Server接收数据并回传演示例程
-- EXTI: 外部中断线例程
-- FLASH: FLASH的擦/读/写, 以及快速编程
-- FPU: FPU硬件浮点运算例程
-- FreeRTOS: FreeRTOS移植例程
-- FSMC
-- LCD: FSMC操作TFTLCD例程
-- NANDFLASH: FSMC操作NANDFLASH例程
-- SRAM: FSMC操作NOR_SRAM例程
-- GPIO: GPIO例程
-- HarmonyOS: HarmonyOS移植例程
-- I2C
-- I2C_7bit_Mode: 7位地址模式, Master/Slave 模式收发例程
-- I2C_10bit_Mode: 10位地址模式, Master/Slave 模式收发例程
-- I2C_DMA: I2C使用DMA, Master/Slave 模式收发例程
-- I2C_EEPROM: I2C接口操作EEPROM外设例程
-- I2C_PEC: 使用PEC错误校验, Master/Slave 模式收发例程
-- I2S
-- HostRx_SlaveTx: I2S主收发例程
-- I2S_DMA: I2S主收发DMA例程
-- I2S_Interrupt: I2S主收发中断例程
-- INT
-- Interrupt_Nest: 中断嵌套例程
-- Interrupt_VTF: 使用VTF_IRQ中断例程
-- IWDG: 独立看门狗例程
-- OPA: OPA4作电压跟随器输出例程
-- PWR
-- Sleep_Mode: 低功耗, 睡眠模式例程
-- Standby_Mode: 低功耗, 待机模式例程
-- Stop_Mode: 低功耗, 停机模式例程
-- Standby_RAM_LV_Mode: 待机模式开启LV时, RAM 2K、30K低功耗数据保持例程
-- Standby_RAM_Mode: 待机模式不开启LV时, RAM 2K、30K低功耗数据保持例程
-- RCC
-- MCO: MCO引脚时钟输出例程
-- HSI_PLL_Source: HSI或HSI/2作为PLL输入时钟例程
-- RTC: 日历例程
-- RT-Thread: RT-Thread移植例程
-- SDIO: SDIO操作SD卡例程
-- SPI
-- 1Lines_half-duplex: 单线半双工模式, Master/Slave 模式数据收发
-- 2Lines_FullDuplex: 双线全双工模式, Master/Slave 模式数据收发
-- FullDuplex_HardNSS: 硬件NSS模式, Master/Slave 模式数据收发
-- SPI_CRC: 使用CRC错误校验, Master/Slave 模式收发例程
-- SPI_DMA: SPI使用DMA, Master/Slave 模式收发例程
-- SPI_FLASH: SPI接口操作FLASH外设例程

```

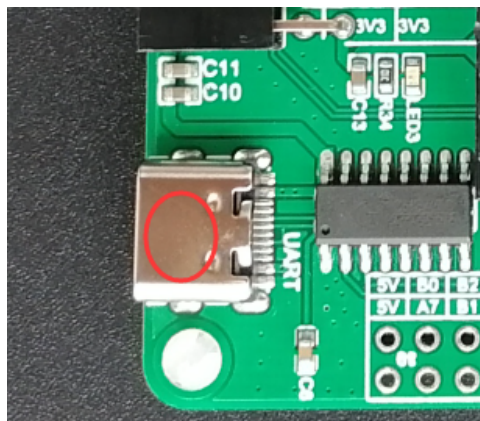
```

-- TencentOS: TencentOS移植例程
-- TIM
-- Clock_Select: 时钟源选择例程
-- ComplementaryOutput_DeadTime: 互补输出和死区插入模式例程
-- ExtTrigger_Start_Two_Timer: 外部触发同步启动两个定时器例程
-- Input_Capture: 输入捕获例程
-- One_Pulse: 单脉冲输出例程
-- Output_Compare_Mode: 输出比较模式例程
-- PWM_Output: PWM输出例程
-- Synchro_ExtTrigger: 从模式例程, 包括复位模式、门控模式、触发模式
-- Synchro_Timer: 定时器同步模式
-- TIM_DMA: 定时器使用DMA例程
-- TOUCHKEY: TouchKey检测例程
-- RNG: 随机数发生器例程
-- USART
-- USART_DMA: USART使用DMA, Master/Slave 模式收发例程
-- USART_HalfDuplex: 单线半双工模式, Master/Slave 模式收发例程
-- USART_HardwareFlowControl: 硬件流控制模式, Master/Slave 模式收发例程
-- USART_Interruption: USART中断例程, Master/Slave 模式收发例程
-- USART_MultiProcessorCommunication: 多处理器通信模式例程
-- USART_Polling: 轮询收发模式, Master/Slave 模式收发例程
-- USART_Printf: 串口打印调试例程
-- USART_SynchronousMode: 同步模式, Master/Slave 模式收发例程
-- USB
-- USBFS
-- DEVICE
-- --CH372Device: 模拟自定义USB设备 (CH372设备) 例程
-- --HIDKeyboardDevice: 模拟Keyboard例程
-- HOST: USBFS设备的简易枚举过程例程
-- USBHS --仅适用于CH32V30x_D8C
-- DEVICE
-- --CH372Device: 模拟自定义USB设备 (CH372设备) 例程
-- --HIDKeyboardDevice: 模拟Keyboard例程
-- HOST: USBFS设备的简易枚举过程例程
-- HOST_Udisk: C示例程序, 以字节为单位读写文件, 包括文件创建、删除、修改文件属性, 修改文件名
-- Udisk_Lib: U盘文件系统库文件
-- VoiceRcgExam: 独立词语音识别例程
-- WWDG: 窗口看门狗例程

```

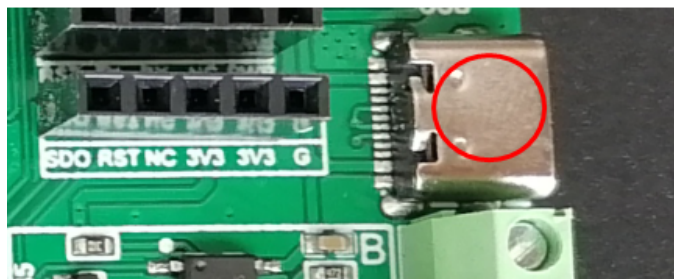
测试开发板上的串口

只要是可下载程序,说明串口是没问题的,就不测试了



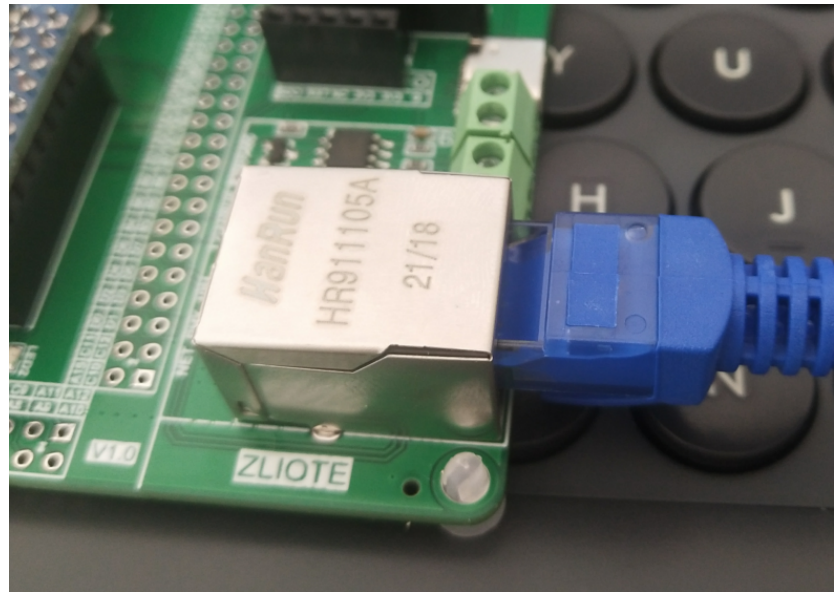
测试开发板上的USB

只要是可下载程序,说明串口是没问题的,就不测试了



测试开发板上的自带PHY的以太网接口

1.使用网线连接路由器或者交换机



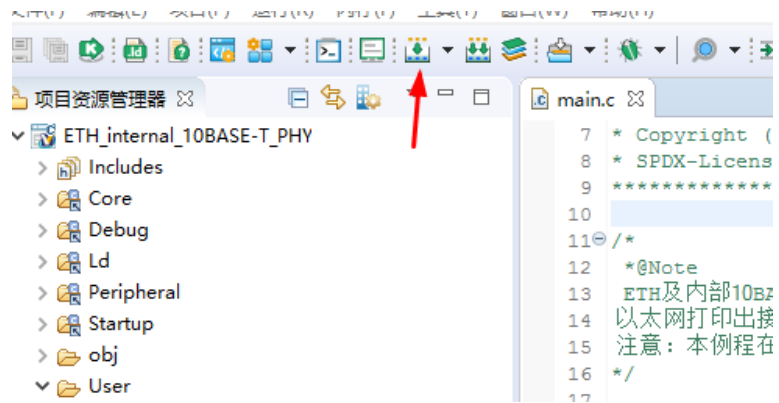
2.双击打开工程

| | |
|------|--------------|
| ADC | 2022/1/26 15 |
| BKP | 2022/1/26 15 |
| CAN | 2022/1/26 15 |
| CRC | 2022/1/26 15 |
| DAC | 2022/1/26 15 |
| DMA | 2022/1/26 15 |
| DVP | 2022/1/26 15 |
| ETH | 2022/1/26 15 |
| EXTI | 2022/1/26 15 |

| | |
|---------------------------|-----|
| 8_UART_Server | 20: |
| DHCP | 20: |
| DNS | 20: |
| ETH_internal_10BASE-T_PHY | 20: |
| ETH_MII_RMII | 20: |
| ETH_RGMII | 20: |
| NetLib | 20: |
| TcpClient | 20: |
| TcpServer | 20: |
| UdpClient | 20: |
| UdpServer | 20: |

| | |
|----------------------------------|---------------|
| .settings | 2022/1/26 15: |
| obj | 2022/4/12 20: |
| User | 2022/1/26 15: |
| .cproject | 2022/1/11 9:1 |
| .project | 2022/4/4 17:5 |
| .template | 2022/4/12 20: |
| ETH_internal_10BASE-T_PHY.wvproj | 2022/1/11 9:1 |

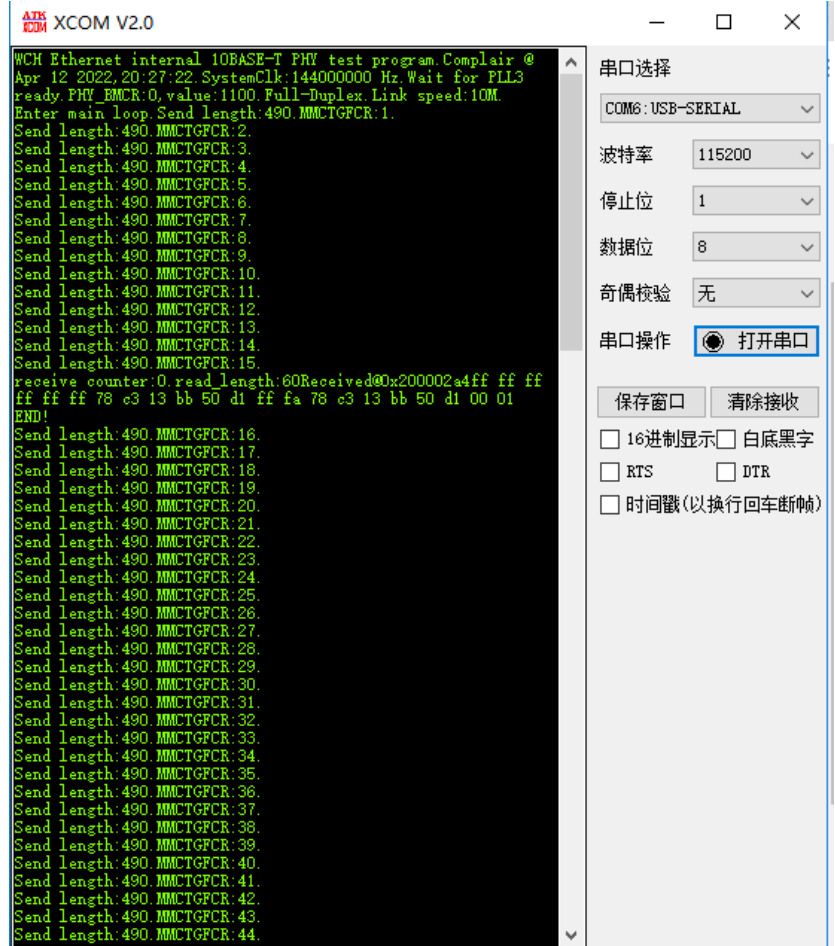
3.编译



4.下载程序到板子以后,使用串口调试助手看下开发板打印的数据

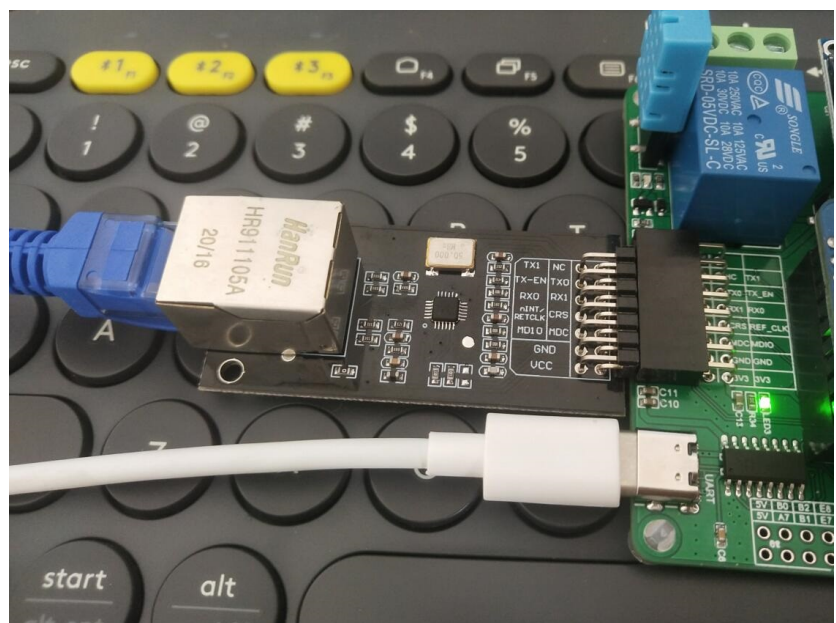


提供的测试例程是不停的发送ARP数据包, 和接收数据包



测试开发板上的外置的PHY以太网接口 (LAN8720)

1,安装上LAN8720模块(DP83848也可以), 使用网线连接路由器或者交换机



2. 双击打开工程

| | |
|------|--------------|
| ADC | 2022/1/26 15 |
| BKP | 2022/1/26 15 |
| CAN | 2022/1/26 15 |
| CRC | 2022/1/26 15 |
| DAC | 2022/1/26 15 |
| DMA | 2022/1/26 15 |
| DVP | 2022/1/26 15 |
| ETH | 2022/1/26 15 |
| EXTI | 2022/1/26 15 |

| | |
|---------------------------|--------------|
| 8_UART_Server | 2022/1/26 15 |
| DHCP | 2022/1/26 15 |
| DNS | 2022/1/26 15 |
| ETH_internal_10BASE-T_PHY | 2022/1/26 15 |
| ETH_MII_RMII | 2022/1/26 15 |
| ETH_RGMII | 2022/1/26 15 |
| NetLib | 2022/1/26 15 |
| TcpClient | 2022/1/26 15 |
| TcpServer | 2022/1/26 15 |
| UdpClient | 2022/1/26 15 |
| UdpServer | 2022/1/26 15 |

| | |
|---------------------|--|
| .settings | |
| obj | |
| User | |
| .cproject | |
| .project | |
| .template | |
| ETH_MII_RMII.wvproj | |

3. 在main.h里面打开使用 RMII 通信方式的宏定义

The screenshot shows an IDE with two panes. The left pane displays the project structure, and the right pane displays the content of the main.h file.

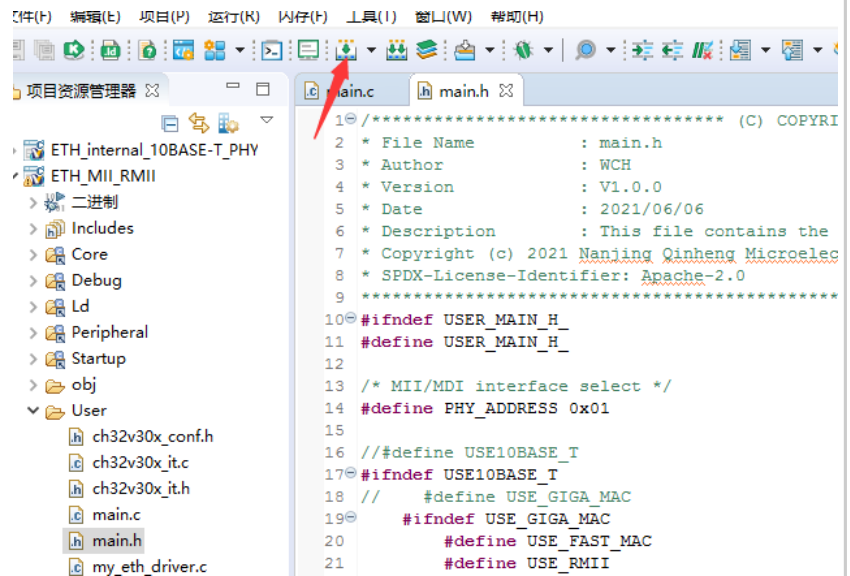
Project Structure (Left Pane):

- ETH_internal_10BASE-T_PHY
 - ETH_MII_RMII
 - main.h (highlighted with a red arrow)

main.h File Content (Right Pane):

```
1  /* ***** (C) COPYRIGHT *****  
2  * File Name      : main.h  
3  * Author         : WCH  
4  * Version        : V1.0.0  
5  * Date           : 2021/06/06  
6  * Description    : This file contains the header file for the RMII interface.  
7  * Copyright (c) 2021 Nanjing Qinheng Microelectronic Co., Ltd.  
8  * SPDX-License-Identifier: Apache-2.0  
9  *****  
10 #ifndef USER_MAIN_H_  
11 #define USER_MAIN_H_  
12  
13 /* MII/MDI interface select */  
14 #define PHY_ADDRESS 0x01  
15  
16 // #define USE_10BASE_T  
17 #ifndef USE_10BASE_T  
18 // #define USE_GIGA_MAC  
19 #ifndef USE_GIGA_MAC  
20 #define USE_FAST_MAC  
21 #define USE_RMII (highlighted with a red arrow)  
22 #endif  
23 #endif  
24  
25 #endif
```

4.编译并下载程序到开发板



5.下载程序到板子以后,使用串口调试助手看下开发板打印的数据



提供的测试例程是不停的发送ARP数据包, 和接收数据包

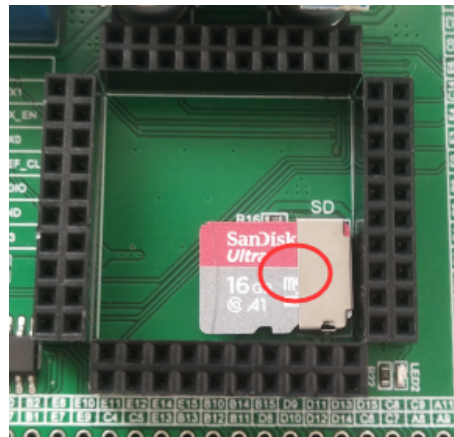

```

WCH Ethernet MII/RMII test program.Complair @ Apr 14 2022, 14:02:24.SystemClk:96000000 Hz.Wait
for PLL3 ready.
PHY_RMCR:0,value:3100,Full-Duplex Link speed:100M.
Enter main loop.Send length:490.MMCTGFCR:1.
Send length:490.MMCTGFCR:2.
Send length:490.MMCTGFCR:3.
Send length:490.MMCTGFCR:4.
Send length:490.MMCTGFCR:5.
Send length:490.MMCTGFCR:6.
Send length:490.MMCTGFCR:7.
Send length:490.MMCTGFCR:8.
Send length:490.MMCTGFCR:9.
Send length:490.MMCTGFCR:10.
Send length:490.MMCTGFCR:11.
Send length:490.MMCTGFCR:12.
Send length:490.MMCTGFCR:13.
Send length:490.MMCTGFCR:14.
Send length:490.MMCTGFCR:15.
Send length:490.MMCTGFCR:16.
Send length:490.MMCTGFCR:17.
Send length:490.MMCTGFCR:18.
Send length:490.MMCTGFCR:19.
Send length:490.MMCTGFCR:20.
Send length:490.MMCTGFCR:21.
Send length:490.MMCTGFCR:22.
Send length:490.MMCTGFCR:23.
Send length:490.MMCTGFCR:24.
Send length:490.MMCTGFCR:25.
Send length:490.MMCTGFCR:26.
Send length:490.MMCTGFCR:27.
Send length:490.MMCTGFCR:28.
Send length:490.MMCTGFCR:29.
Send length:490.MMCTGFCR:30.
Send length:490.MMCTGFCR:31.
Send length:490.MMCTGFCR:32.
Send length:490.MMCTGFCR:33.
Send length:490.MMCTGFCR:34.
Send length:490.MMCTGFCR:35.
Send length:490.MMCTGFCR:36.
Send length:490.MMCTGFCR:37.
Send length:490.MMCTGFCR:38.
receive counter:0 read length:60Received@0x200002a4ff ff ff ff ff ff 78 c3 13 bb 50 d1 ff fa 78
c3 13 bb 50 d1 00 01 END!
Send length:490.MMCTGFCR:39.
Send length:490.MMCTGFCR:40.
Send length:490.MMCTGFCR:41.
Send length:490.MMCTGFCR:42.
Send length:490.MMCTGFCR:43.
Send length:490.MMCTGFCR:44.

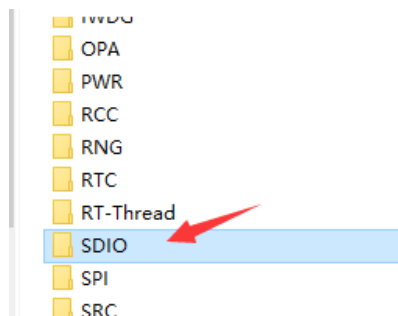
```

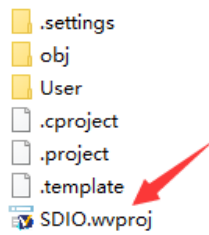
测试开发板上的SDIO

1,安装SD卡

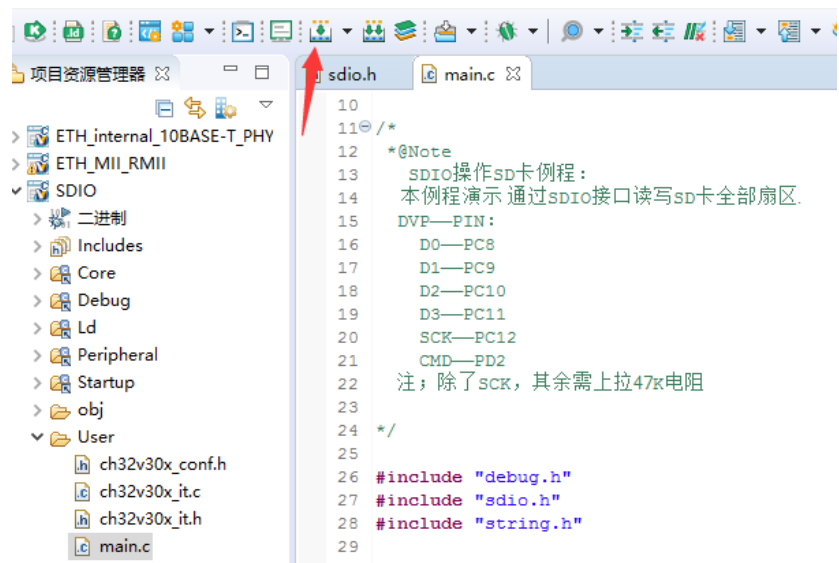


2,打开工程

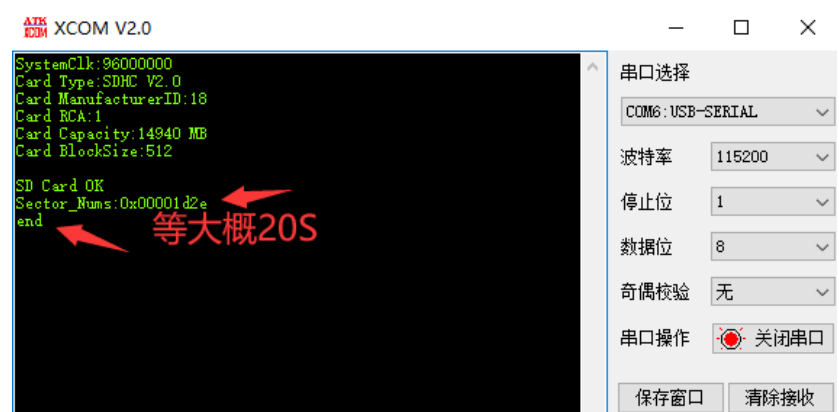




3.编译并下载程序



4.串口会打印



分类: [CH32V307\(WCH单片机\)学习开发](#)



0

0

« 上一篇: 002-CH32V307(WCH单片机)学习开发-CH32V307VCT6开发板硬件使用说明,下载和运行第一个程序

posted on 2022-04-14 14:46 杨奉武 阅读(0) 评论(0) 编辑 收藏 举报

[刷新评论](#) [刷新页面](#) [返回顶部](#)

发表评论

[编辑](#) [预览](#)

[B](#) [🔗](#) [</>](#) [“”](#) [🖼](#)

支持 Markdown

[🔍](#) 自动补全

[提交评论](#) [退出](#)

[Ctrl+Enter快捷键提交]

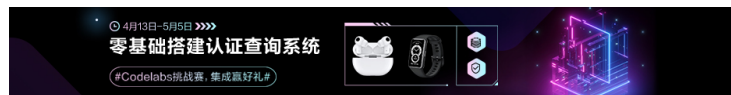
【推荐】百度智能云开发者赋能计划，云服务器4元起，域名1元起

【推荐】华为开发者专区，与开发者一起构建万物互联的智能世界

【推荐】集成赢好礼：华为 Codelabs 挑战赛，搭建认证查询系统

编辑推荐：

- 使用 Harr 特征的级联分类器实现目标检测
- 后端思维之数据库性能优化方案
- ASP.NET Core WebApi 返回结果统一包装实践
- asp.net core启动源码以及监听，到处理请求响应的过程
- ASP.NET Core 高性能服务器 HTTP.SYS



最新新闻：

- Google 将给知名开发者和发行商展示徽章
 - RaidForums 站长被捕
 - 欧洲大力投资火车
 - vivo X Fold 评测：后发，能「先至」吗？
 - 游戏版号重发，教父难知天命
- » 更多新闻...

历史上的今天：

2021-04-14 ESP8266转485通讯模块使用说明

2020-04-14 ESA2GJK1DH1K基础篇: 阿里云物联网平台: 云平台显示单片机采集的温...

Powered by:

博客园

Copyright © 2022 杨奉武

Powered by .NET 6 on Kubernetes



单片机,物联网,上位机,...

扫一扫二维码, 入群聊。