

## 优秀不够，你是否无可替代

知识从未如此性感。烂程序员关心的是代码,好程序员关心的是数据结构和它们之间的关系 --QQ群: 607064330 --本人QQ:946029359 --淘宝 <https://shop411638453.taobao.com/>

随笔 - 834, 文章 - 0, 评论 - 329, 阅读 - 205万

### 导航

[博客园](#)  
[首页](#)  
[新随笔](#)  
[联系](#)  
[订阅](#)   
[管理](#)

### 公告

渡我不渡她 -  
Not available  
00:00 / 00:00

- 渡我不渡她
- 小镇姑娘
- PDD洪荒之力

加入QQ群

昵称：杨奉武  
 园龄：6年3个月  
 粉丝：693  
 关注：1

### 搜索

### 我的标签

8266(88)  
 MQTT(50)  
 GPRS(33)  
 SDK(29)  
 Air202(28)  
 云服务器(21)  
 ESP8266(21)  
 Lua(18)  
 小程序(17)  
 STM32(16)  
 更多

### 随笔分类

Air724UG学习开发(8)  
 Android(22)  
 Android 开发(8)  
 C# 开发(4)  
 CH395Q学习开发(17)  
 CH573F学习开发(2)  
 CH579M物联网开发(12)  
 CH579M学习开发(11)  
 CH581/2/3 学习开发(2)  
 ESP32学习开发(30)  
 ESP8266 AT指令开发(基于STC89C52单片机)(3)  
 ESP8266 AT指令开发(基于STM32)(1)  
 ESP8266 AT指令开发基础入门篇备份(12)

## 202-CH579M学习开发-以太网例程-DNS域名解析

<p><iframe name="ifd" src="https://mnifdv.cn/resource/cnblogs/LearnCH579M" frameborder="0" scrolling="auto" width="100%" height="1500"></iframe></p>

### 单片机CH579M学习开发(带蓝牙和以太网口的ARM M0内核的单片机)

#### 替代STM32,替代串口转以太网DTU

开发板链接:<https://item.taobao.com/item.htm?ft=t&id=648634562877>

芯片购买链接:<https://item.taobao.com/item.htm?ft=t&id=649533679749>

#### 开发板原理

图:<https://mnifdv.cn/resource/cnblogs/LearnCH579M/CH579M.PDF>

#### 资料源码下载链

接:<https://github.com/yangfengwu45/LearnCH579M.git>

资料源码下载链接:<https://pan.baidu.com/s/1u9TN-EOIJk5K0Ak-PqRw4A> 提取码：uw05

点击链接加入群聊【网络学习开发】： 加入QQ群

- [学习Android](#)  
教程中搭配的Android，C#等教程如上，各个教程正在整理。
- [001-硬件使用说明,下载和运行第一个程序](#)
- [002-官方资料学习说明,开发板蓝牙\(蓝牙定位\),网口通信测试](#)
- -----基本外设-----
- [003-新建工程说明](#)
- [100-基本外设-GPIO输入输出](#)
- [101-基本外设-定时器](#)
- [102-基本外设-串口](#)
- [103-基本外设-引脚中断](#)
- -----以太网-----
- [200-网络指示灯GPIO选择,检测网线连接状态](#)
- [201-DHCP](#)
- [202-DNS域名解析](#)
- [210-TCP客户端\(单路,和电脑直接连接测试\)](#)
- 
- 
- 
-

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(33)  
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(37)  
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(65)  
STM32+W5500+AIR202/302基本控制方案(25)  
STM32+W5500+AIR202/302远程升级方案(6)  
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)  
单片机(STC89C51)单片机开发板学习入门篇(3)  
单片机(STM32)基础入门篇(3)  
更多

#### 阅读排行榜

1. ESP8266使用详解(AT,LUA,SDK)(174634)
2. 1-安装MQTT服务器(Windows),并连接测试(106380)
3. 用ESP8266+android,制作自己的WIFI小车(ESP8266篇)(68536)
4. ESP8266刷AT固件与node mcu固件(67333)
5. 有人WIFI模块使用详解(39646)
6. (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)(37333)
7. C#中public与private与static(36720)
8. 关于TCP和MQTT之间的转换(35711)
9. android 之TCP客户端编程(33259)
10. android服务端+eps8266+单片机+路由器之远程控制系统(31777)

#### 推荐排行榜

## DNS是什么



最开始的网络都是直接一个IP地址访问,后来呢出现了域名访问,比如**baidu.com**

然后记住哈,咱使用域名访问,其实网络设备会先根据这个域名找到对应的IP地址

然后再通过IP地址访问.

怎么通过域名找到IP地址呢? 就是访问DNS服务器.

DNS服务器上记录了所有域名对应的IP地址,访问方式呢是UDP方式.端口号默认是53

- 1. 用ESP8266+android,制作自己的WIFI小车(ESP8266篇)(9)
- 2. C#委托+回调详解(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. ESP8266使用详解(AT,LUA,SDK)(6)

最新评论

- 1. Re:用ESP8266+android,制作自己的WIFI小车(Android 软件)可以重新发一遍源代码吗 --evakzxx
- 2. Re:103-CH579M学习开发-基本外设-引脚中断 esp32 gpio 中断支持 chang 类型，即只要是上升沿或者下降沿就会触发，ch579 不支持吗？ --yh251512

公共DNS服务器IP地址大全

名称	各省公共DNS服务器IP大全	
114 DNS	114.114.114.114	114.114.115.115
阿里 AliDNS	223.5.5.5	223.6.6.6
百度 BaiduDNS	180.76.76.76	
DNSPod DNS+	119.29.29.29	182.254.116.116
CNNIC SDNS	1.2.4.8	210.2.4.8
oneDNS	117.50.11.11	52.80.66.66
DNS 派 电信/移动铁通	101.226.4.6	218.30.118.6
DNS 派 联通	123.125.81.6	140.207.198.6
Cloudflare DNS	1.1.1.1	1.0.0.1
Google DNS	8.8.8.8	8.8.4.4
IBM Quad9	9.9.9.9	
DNS.SB	185.222.222.222	185.184.222.222
OpenDNS	208.67.222.222	208.67.220.220
V2EX DNS	199.91.73.222	178.79.131.110

假设我使用上面阿里的DNS服务器,我想知道域名 **mnif.cn**对应的IP地址是多少

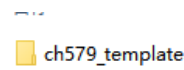
我就把**mnif.cn**这个字符串封装成DNS协议数据,然后通过UDP把数据发送到**223.5.5.5 端口:53**

然后服务器找到对应的IP并返回.

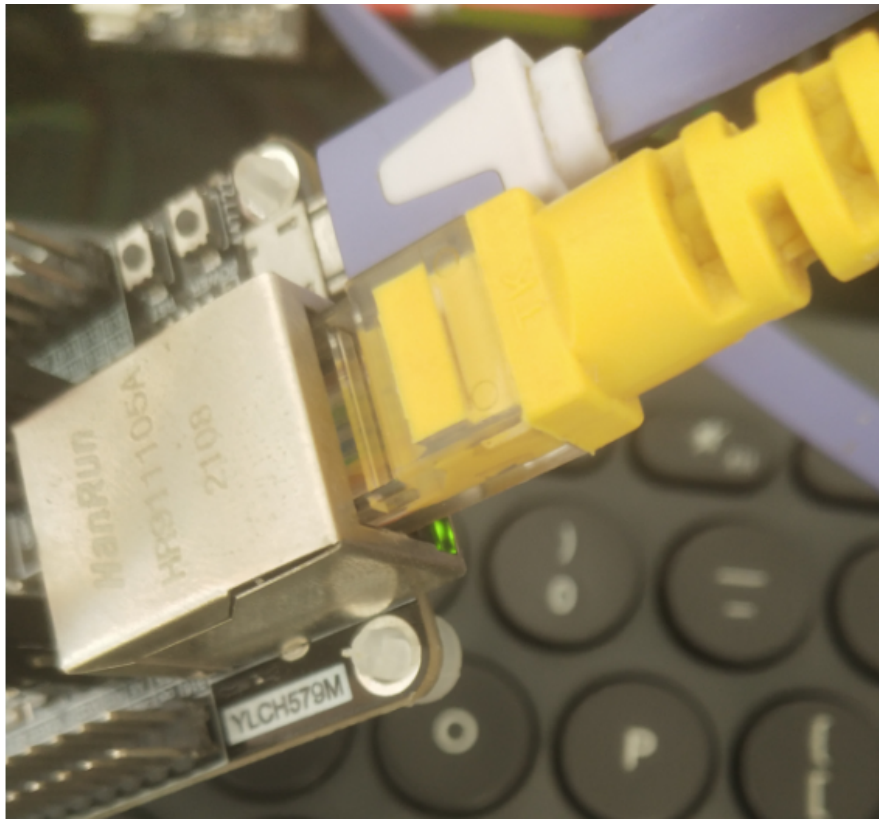
协议啥的自己去百度下.

测试本节代码

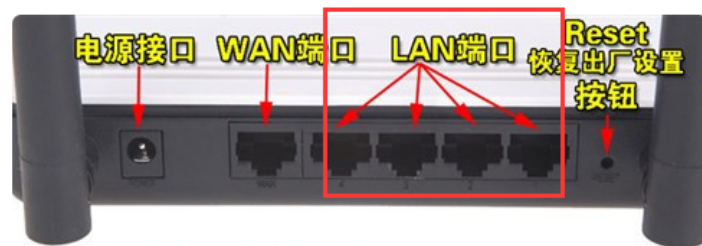
1.把这节的程序下载到开发板



2.连接路由器或者交换机



注意哈,连接的是 LAN端口



**WAN端口: 连接网线**

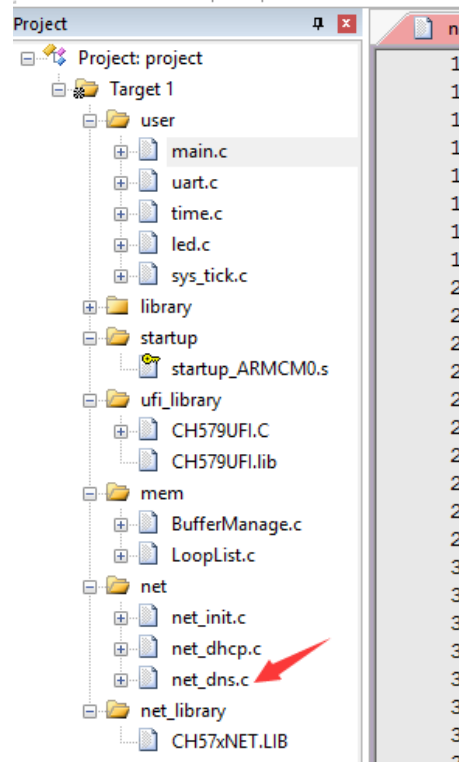
**LAN端口: 连接电脑 ( 任选一个端口就行 )**

### 3.查看串口打印的日志

```
ATK XCOM V2.0
start
GINT_STAT_PHY_CHANGE 02net_dns_query
net_dns_query
net_dns_query
net_dns_ip=47.92.31.46
```

## 程序说明

1.为便于大家移植使用,我封装了下



## 2.移植的话把下面的文件放到自己的工程里面

- net\_dhcp.c
- net\_dhcp.h
- net\_dns.c
- net\_dns.h
- net\_init.c
- net\_init.h

## 3.把超时检测函数放到1ms定时器里面轮行执行

```
Project: project
Target 1
├── user
│   ├── main.c
│   ├── uart.c
│   ├── time.c
│   ├── led.c
│   └── sys_tick.c
├── library
├── startup
│   └── startup_ARMCM0.s
├── ufi_library
│   ├── CH579UFI.C
│   └── CH579UFI.lib
├── mem
│   ├── BufferManage.c
│   └── LoopList.c
├── net
│   ├── net_init.c
│   ├── net_dhcp.c
│   └── net_dns.c
└── net_library
    └── CH57xNET.LIB
```

```
355 /**
356 char net_dns_socket_close(unsigned char socketid)
357 {
358     if(socketid!=255){//创建了socket
359         CH57xNET_SocketClose( socketid,TCP_CLOSE_NORMAL );
360     }
361 }
362
363
364 /**
365 * @brief    DNS超时检测,放到定时器里面进行轮训
366 * @param    None
367 * @param    None
368 * @param    None
369 * @param    None
370 * @retval    None
371 * @warning   推荐放到1ms定时器, 超时时间200ms就可以
372 * @example
373 */
374 void net_dns_loop(void)
375 {
376     if(net_dns_start==1)
377     {
378         net_dns_time_out_cnt++;
379         if(net_dns_time_out_cnt>200)//超时200ms
380         {
381             net_dns_time_out_cnt=0;
382             net_dns_start=0;
383
384             net_dns_socket_close(net_dns_socket_index);
385         }
386     }
387     else
388     {
389         net_dns_time_out_cnt=0;
390     }
391 }
```

```
time.c net_dhcp.c net_dns.c main.c net_dns.h
1  #define TIME_C_
2
3  #include <stdio.h>
4  #include "time.h"
5
6  #include "net_init.h"
7  #include "net_dns.h"
8
9  int timer0_value;
10
11 void time0_init(void)
12 {
13     /*配置定时器0,定时器的时钟是FREQ_SYS*/
14     TMR0_TimerInit( 1*(FREQ_SYS/1000) ); //定时1ms
15     TMR0_ITCfg(ENABLE, TMR0_3_IT_CYC_END); //使能定时器周期中断
16     // NVIC_SetPriority(TMR0_IRQn,18); //设置中断优先级(也可以不设置,默认3)
17     NVIC_EnableIRQ( TMR0_IRQn );
18 }
19
20
21 /*TMR0周期中断*/
22 void TMR0_IRQHandler(void)
23 {
24     if( TMR0_GetITFlag( TMR0_3_IT_CYC_END ) )
25     {
26         TMR0_ClearITFlag( TMR0_3_IT_CYC_END );
27         timer0_value++;
28         CH57xNET_TimeIsr(CH57xNETTIMEPERIOD); /* 定时器中断服务函数 */
29         net_dns_loop();
30     }
31 }
32
```

#### 4.把网络接收的数据丢给 net\_dns\_socket\_data 函数

```

time.c net_dhcp.c net_dns.c main.c net_dns.h
39 void CH57xNET_HandleGlobalInt(void)
40 {
41     UINT8 initstat;
42     UINT8 i;
43     UINT8 socketinit;
44     initstat = CH57xNET_GetGlobalInt(); /* 读全局中断状态并清除 */
45
46     if(initstat & GINT_STAT_UNREACH) /* 不可达中断 */
47     {
48         printf("UnreachCode : %ld\n", CH57xInf.UnreachCode); /* 查看不可达代码 */
49         printf("UnreachProto : %ld\n", CH57xInf.UnreachProto); /* 查看不可达协议类型 */
50         printf("UnreachPort : %ld\n", CH57xInf.UnreachPort); /* 查询不可达端口 */
51     }
52     if(initstat & GINT_STAT_IP_CONFLI) /* IP冲突中断 */
53     {
54         if(initstat & GINT_STAT_PHY_CHANGE) /* PHY改变中断 */
55         {
56             i = CH57xNET_GetPHYStatus(); /* 获取PHY状态 */
57             printf("GINT_STAT_PHY_CHANGE %02x\n", i);
58         }
59     }
60     if(initstat & GINT_STAT_SOCKET) /* Socket中断 */
61     {
62         for(i = 0; i < CH57xNET_MAX_SOCKET_NUM; i++){
63             socketinit = CH57xNET_GetSocketInt(i); /* 读socket中断并清零 */
64             if(socketinit) /* 有事件需要处理 */
65             {
66                 net_dns_socket_data(i, socketinit); //DNS
67             }
68         }
69     }
70 }
71
72 int main()
73 {
74     int i;
75     PWR_UnitModCfg(ENABLE, UNIT_SYS_PLL); /* 使能PLL(给以太网提供时钟) */
76     DelayMs(3);
77 }

```

## 5.假设要解析 mnif.cn这个域名

```

time.c net_dhcp.c net_dns.c main.c net_dns.h
12
13 #include "net_init.h"
14 #include "net_dhcp.h"
15 #include "net_dns.h"
16
17 __align(4) UINT8 RxBuffer[ MAX_PACKET_SIZE ]; /* IN, must even address */
18 __align(4) UINT8 TxBuffer[ MAX_PACKET_SIZE ]; /* OUT, must even address */
19
20 //缓存数据使用
21 unsigned char main_buffer[main_buffer_len]; //缓存数据,全局通用
22 uint32_t main_len; /* 全局通用变量 */
23 unsigned char *main_str; /* 全局通用变量 */
24
25
26 void uart_data(void); //串口数据处理
27
28
29 char domain_name[20] = "mnif.cn"; //要解析的域名
30 unsigned char domain_ip[4] = {0,0,0,0}; //解析后的IP地址存储数组
31
32
33 /*****
34 * Function Name : CH57xNET_HandleGloableInt
35 * Description : 事件处理函数
36 * Input : None
37 */

```

```
time.c net_dhcp.c net_dns.c main.c net_dns.h
66
67
68
69 int main()
70 {
71     int i;
72     PWR_UnitModCfg(ENABLE, UNIT_SYS_PLL); //使能PLL(给以太网提供时钟)
73     DelayMs(3);
74     // SetSysClock(CLK_SOURCE_HSE_32MHz); /* 外部晶振 PLL 输出32MHz */
75     // SysTick_Config(FREQ_SYS/1000); //系统定时器,1ms
76     uart_init(); //初始化串口
77     net_init(); //初始化以太网
78     time0_init(); //初始化定时器
79     printf("start\r\n");
80     while(1)
81     {
82         if(domain_ip[0]==0) //没有获取到IP地址
83         {
84             net_dns_query(domain_name, domain_ip); //执行DNS
85         }
86
87         CH57xNET_MainTask(); /* CH57xNET库主任务函数,需要在主循环中不断调用 */
88
89         /* 查询事件,如果有事件,则调用处理函数 */
90         if(CH57xNET_QueryGlobalInt()) CH57xNET_HandleGlobalInt();
91
92         uart_data();
93     }
94 }
95
96
```

## 其它

### 如果要更改其它设置

```
time.c net_dhcp.c net_dns.c main.c net_dns.h
1 #define net_dns_c
2
3 #include "net_dns.h"
4
5 #if 1 //是否打印日志
6 #define debug_printf printf
7 #else
8 #define debug_printf(...)
9 #endif
10
11
12 unsigned char net_dns_socket_index=255; /* 保存socket索引*/
13
14 unsigned char dns_server_ip[4]={114, 114, 114, 114}; /*DNS服务器,需根据实际DNS服务器修改*/
15 unsigned short dns_server_port = 53; /*DNS默认端口*/
16 unsigned short dns_local_port = 1000; /*Socket 本地端口 */
17
18 unsigned short nds_msg_id = 0x1100; /*标识*/
19
20 /*缓存DNS包协议数据*/
21 #define net_dns_buf_len 512
```

分类: CH579M学习开发

好文要顶

关注我

收藏该文



杨奉武

关注 - 1

粉丝 - 693

0

0

« 上一篇: 002-CH573F学习开发-官方资料学习说明,开发板USB,蓝牙通信测试

posted on 2022-01-09 13:46 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】发布 vscode 插件 Cnblogs Client For VSCode 预览版
- 【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】参与华为 HarmonyOS 开发者创新大赛，一起创造无限可能

#### 编辑推荐：

- 全链路压测(4)：全链路压测的价值是什么？
- 神奇的滤镜！巧妙实现内凹的平滑圆角
- 实践剖析 .NET Core 如何支持 Cookie 和 JWT 混合认证、授权
- 探索 dotnet core 为何在 Windows7 系统需要补丁的原因
- 技术管理进阶——精要主义设计人生，对混乱的工作说不

#### 最新新闻：

- 三星永久关闭Tizen应用商店：手机、手表彻底放弃该系统
  - 新氧金星：六大战略布局新医美，赋能机构降本增效
  - NASA完成韦伯望远镜展开部署 今夏开始执行科学任务
  - 华为P50 Pocket和P50 Pro将于1月12日全球发布：搭载骁龙888 4G
  - 全球最富500人去年财富增加1万亿美元 马斯克增加超千亿
- » 更多新闻...

#### 历史上的今天：

2017-01-09 有人WIFI模块使用详解

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,...

单片机,物联网,上位机,...

扫一扫二维码, 加入群聊。