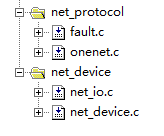
网络设备层移植说明

张继瑞 2016-11-24

1. 设计说明

此版本代码将平台协议层和网络设备层进行了分离，开发者无需关心平台的具体协议，只要按照以下规定(三)提供网络设备层的数据交互函数即可接入ONENET平台。

1. 目录结构说明



1. net\_protocol-协议应用层

onenet.c：平台连接、数据发送、心跳、命令解析应用。

fault.c：网络错误处理。

1. net\_device-网络设备层

net\_io.c：网络设备底层驱动，数据输入、输出的IO驱动层。

net\_device.c：网络设备应用层，包括初始化、发送命令、发送数据、获取数据等。

1. 需要移植的网络设备功能

**net\_io.c**

1. IO驱动初始化---void NET\_IO\_Init(void)

单片机与网络设备的通信方式的初始化。

例如ESP8266和M6311是通过串口与单片机通信，所以这个函数只需要初始化串口即可。

1. 数据发送---void NET\_IO\_Send(unsigned char \*str, unsigned short len)

单片机发送数据到网络设备的函数。

1. 数据接收完成指示---\_Bool NET\_IO\_WaitRecive(void)

判断接收缓存区的数据是否接收完整。此函数必须是非阻塞函数。

1. 清空缓存区---void NET\_IO\_ClearRecive(void)

清空接收缓存区即可。

1. 数据接收---void USART2\_IRQHandler(void)

将收到的数据缓存进接收缓存区即可。

**net\_device.c**

1. IO引脚初始化--- void NET\_DEVICE\_IO\_Init(void)

调用NET\_IO\_Init完成IO层的初始化。

1. 网络接入初始化--- \_Bool NET\_DEVICE\_Init(void)

调用此函数后，网络设备必须具备接入外网的能力。

采用状态机方式对网络设备进行初始化。需要有步骤控制。

1. 网络设备复位--- void NET\_DEVICE\_Reset(void)

当发生未知错误、长时间无法接入网络时，需要复位重新配置。

1. 平台重连--- \_Bool NET\_DEVICE\_ReLink(void)

调用此函数，能连接ONENT的IP地址。

1. 发送命令到网络设备--- \_Bool NET\_DEVICE\_SendCmd(char \*cmd, char \*res)

发送一个命令到网络设备，阻塞式调用，发送命令，检查命令执行状态，最后返回结果。

1. 发送数据到ONENET平台---

void NET\_DEVICE\_SendData(unsigned char \*data, unsigned short len)

发送一帧数据到ONENET。

1. 获取ONENT下发的数据---

unsigned char \*NET\_DEVICE\_GetIPD(unsigned short timeOut)

在timeOut这段时间内查询、获取ONENET下发的

数据，成功之后返回平台原始数据缓存区指针。

1. 清空缓存区---void NET\_DEVICE\_ClrData(void)

清空接收缓存区。

1. 网络设备状态检查---unsigned char NET\_DEVICE\_Check(void)

网络设备连接状态检查，返回状态结果。

1. 网络设备初始化重配置---void NET\_DEVICE\_ReConfig(unsigned char step)

在初始化中的配置步骤。

1. 设置数据传输模式---

void NET\_DEVICE\_Set\_DataMode(unsigned char mode)

以区分是网络设备指令、平台数据、心跳、平台命令等。

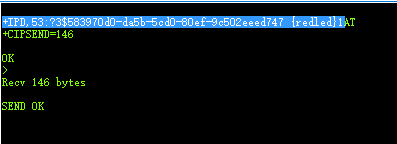
1. 获取数据传输模式---

unsigned char NET\_DEVICE\_Get\_DataMode(void)

解决数据发送之间的冲突。

比如在发送心跳过程中，由于网络情况较差，系统会切换任务，此时有可能会进行数据发送或者状态检查，为了避免数据冲突，只有当为空闲模式时，可以进行相应的数据收发。

1. 如何解析网络模块的IPD头---ESP8266

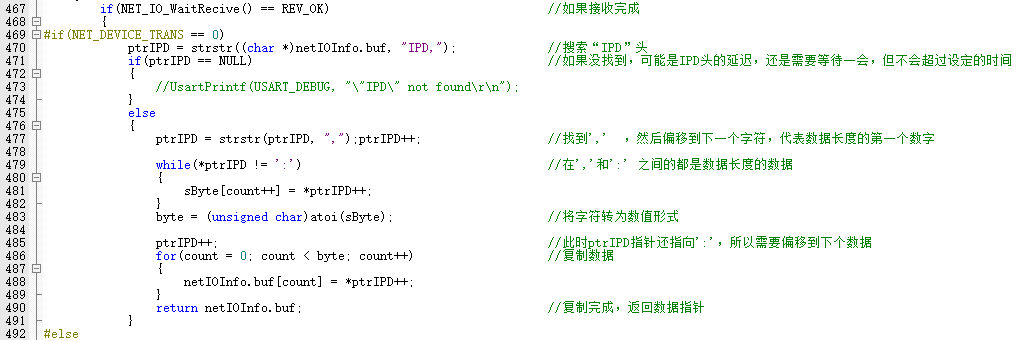


图中蓝色高亮部分为平台下发命令的数据(其中有4个停止符)。

使用USB转TTL，将接收引脚接到开发板的PA3引出脚上，等到开发板开始请求连接时，会出现“IPD”头的格式。

NET\_DEVICE\_GetIPD函数需要做的事情就是提取出“：”后边的原始数据。

1. 找到“IPD”头：使用strstr库函数进行搜索。
2. 找到“，”，紧接着就是数据长度，直到“：”为止。
3. 根据数据长度提取出原始数据。
4. 在例程里边也有相关的注释



1. 后续

可参考已提供的ESP8266、M6311的驱动。

只需要提供以上功能，即可接入ONENET！