NUC240 UART接口驱动ESP8266

NuMicro® 32 位系列微控制器范例代码介绍

文件信息

代码简述	本范例代码使用 NUC240 驱动 ESP8266，实现 TCP Server 功能
BSP 版本	NUC230_240_Series_BSP_CMSIS_V3.01.001
开发平台	NuTiny-SDK-NUC240V

The information described in this document is the exclusive intellectual property of Nuvoton Technology Corporation and shall not be reproduced without permission from Nuvoton.

Nuvoton is providing this document only for reference purposes of NuMicro microcontroller based system design. Nuvoton assumes no responsibility for errors or omissions.

All data and specifications are subject to change without notice.

For additional information or questions, please contact: Nuvoton Technology Corporation.

www.nuvoton.com

1 功能介绍

1.1 简介

ESP-01S是Espressif公司生产的一种基于ESP8266的小型模块。

ESP-01S ESP8266 Wi-Fi模块是一个独立的SOC，具有集成的TCP/IP协议栈，可以让任何微控制器访问Wi-Fi网络。ESP8266能够独立运行应用程序，或者从另一个应用程序处理器分担所有Wi-Fi网络功能。每个ESP-01S ESP8266模块都预置有AT命令集软件。

TCP传输控制协议是Internet协议套件的主要协议之一。它起源于最初的网络实现，它补充了互联网协议（IP）。因此，整个套件通常称为TCP/IP。TCP在通过IP网络通信的主机上运行的应用程序之间提供可靠的、有序的和经过错误检查的八位字节流。主要的internet应用程序，如万维网、电子邮件、远程管理和文件传输都依赖于TCP。不需要可靠数据流服务的应用程序可以使用用户数据报协议（UDP），该协议提供无连接数据报服务，强调降低延迟而不是可靠性。

本范例代码使用NUC240系列芯片，通过UART0接口驱动ESP-01S模组，利用ESP8266内置的TCP协议栈实现TCP服务器的功能。当电脑通过Wi-Fi连接ESP8266模块后，用户可以使用电脑工具串口及网络助手工具查看执行结果。

将ESP8266设置为TCP服务器，使能多IP连接，因为可能有多个客户端连接到ESP8266。如下展示了当ESP8266在softAP模式下工作时如何实现TCP服务器。

1. 设置 Wi-Fi 模式

   AT+CWMODE=3　　//　softAP+station 模式

   回复:

   OK

2. 启动多 IP 连接

   AT+CIPMUX=1

   Response:

   OK

3. 创建软件 AP，无密码

   AT+CWSAP_CUR=Nuvoton_AP,,5,0,4,0

   回复:

   OK

4. 创建 TCP 服务器.

   AT+CIPSERVER=1,8080　　//　端口号　= 8080

   回复:

   OK

5. 连接电脑到 ESP8266 softAP，然后使用电脑端网络调试工具创建一个 TCP 客户端并连接到 ESP8266 创建的 TCP 服务器。
6. 当 ESP8266 从 TCP 服务器接收到数据, 它将如下格式输出:

   +IPD,0,n:xxxxxxxxxx　　// 接收 n 字节, 数据=xxxxxxxxxx

2 代码介绍

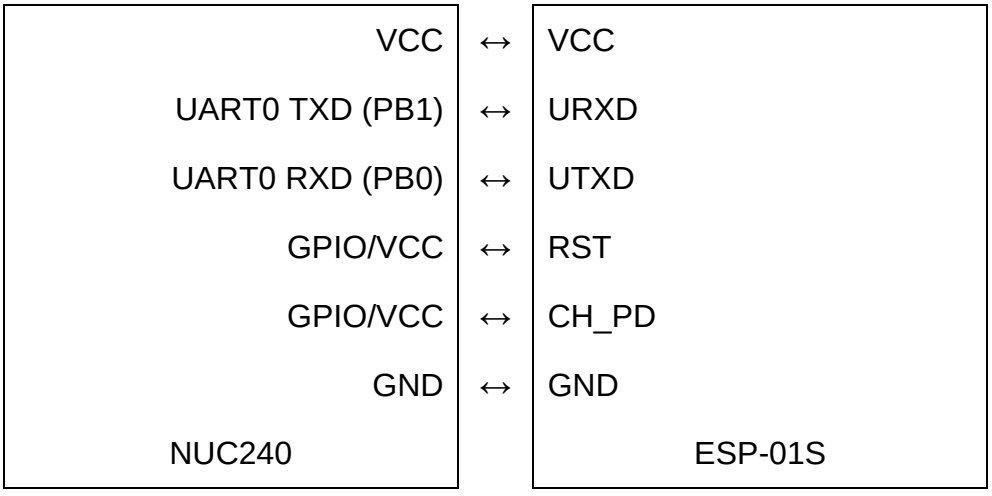
当扫描到按键按下时，ESP8266 被设置为 AP 工作模式并作为 TCP 服务器运行：

```
void Key_Process(uint8_t u8Key)
{
    uint32_t u32cnt;

    if (u8Key == KEY_NONE)
    {
        return;
    }
    switch (u8Key)
    {
        case KEY_SW1:
            // Reset ESP8266
            UART_Write(UART0, atCmdRst.pu8string, atCmdRst.u8CharNum);
            break;
        case KEY_SW2:
            // Set ESP8266 STA+AP mode
            UART_Write(UART0, atCmdCwmode3.pu8string, atCmdCwmode3.u8CharNum);
            for (u32cnt = 0; u32cnt < 0xffff; u32cnt++);
            // Set ESP8266 Multi-connection
            UART_Write(UART0, atCmdCipmux1.pu8string, atCmdCipmux1.u8CharNum);
            for (u32cnt = 0; u32cnt < 0xffff; u32cnt++);
            // Set ESP8266 AP Name, No Password
            UART_Write(UART0, atCmdAp.pu8string, atCmdAp.u8CharNum);
            for (u32cnt = 0; u32cnt < 0xffff; u32cnt++);
            // Set ESP8266 TCP Server mode, Port number 8080
            UART_Write(UART0, atCmdCipserver1.pu8string, atCmdCipserver1.u8CharNum);
            for (u32cnt = 0; u32cnt < 0xffff; u32cnt++);

            u8ReceiveStringArrayHeader = u8ReceiveStringArrayTail = 0;
            memset(ReceiveArray, 0, sizeof(ReceiveArray));
        default:
            break;
    }
}
```

3 软件与硬件环境

- 软件环境
  - BSP 版本
    - NUC230_240_Series_BSP_CMSIS_V3.01.001
  - IDE 版本
    - Keil uVersion 5.29
- 硬件环境
  - 电路组件
    - NuTiny-SDK-NUC240V
    - NuEdu-Basic01 V2.0
    - ESP-01S 模组
  - 示意图

NUC240 通过 UART0 接口与 ESP-01S 模组连接。UART1 用来打印信息。



4 目录信息

EC_NUC240_UART_ESP8266_V1.00	
Document	4a-esp8266_at_instruction_set_en.pdf
Library	Sample code header and source files
CMSIS	Cortex® Microcontroller Software Interface Standard (CMSIS) by Arm® Corp.
Device	CMSIS compliant device header file
StdDriver	All peripheral driver header and source files
SampleCode	
ExampleCode	Source file of example code

5 如何执行范例程序

1. 根据目录信息章节进入 ExampleCode 路径中的 KEIL 文件夹，双击 NUC240_UART_ESP8266.uvprojx。
2. 进入编译模式接口
   a. 编译
   b. 下载代码至内存
   c. 进入 / 离开除错模式
3. 进入除错模式接口
   a. 执行代码

6 修订纪录

Date	Revision	Description
Sept. 17, 2019	1.00	1. 初始发布.

Important Notice

Nuvoton Products are neither intended nor warranted for usage in systems or equipment, any malfunction or failure of which may cause loss of human life, bodily injury or severe property damage. Such applications are deemed, "Insecure Usage".

Insecure usage includes, but is not limited to: equipment for surgical implementation, atomic energy control instruments, airplane or spaceship instruments, the control or operation of dynamic, brake or safety systems designed for vehicular use, traffic signal instruments, all types of safety devices, and other applications intended to support or sustain life.

All Insecure Usage shall be made at customer's risk, and in the event that third parties lay claims to Nuvoton as a result of customer's Insecure Usage, customer shall indemnify the damages and liabilities thus incurred by Nuvoton.

Please note that all data and specifications are subject to change without notice.
All the trademarks of products and companies mentioned in this datasheet belong to their respective owners.