

文章编号:1007-2780(2014)02-0195-07

# 基于 STM32+ucGUI 的北斗导航定位系统设计

罗卫兵,胡健生\*

(武警工程大学 信息工程系,陕西 西安 710086)

**摘要:**针对当前手持嵌入式产品在导航定位功能实现上存在的局限性,本文提出基于 STM32+ucGUI 构架的北斗导航定位终端。利用 UM220 北斗接收模块实现了导航信息的获取,利用 STM32 的中断处理功能完成信息的处理;为了简便完美地完成人机界面设计,在硬件平台上移植了 ucGUI;利用 ucGUI 的动态设备对象接口,高效地实现了电子罗盘界面设计。实验结果表明,系统冷启动时的定位时间小于 2 min,热启动时的定位时间小于 1 min,经度纬度不受运动速度影响,误差小于 3 m。工程实践表明该系统应用效果良好,具有一定的实际应用价值。

**关键词:**北斗导航;STM32;ucGUI

**中图分类号:**TP941 **文献标识码:**A **doi:**10.3788/YJYXS20142902.0195

## Design of Beidou navigation system based on STM32 & ucGUI

LUO Wei-bing, HU Jian-sheng\*

(Engineering University of CAPF, Xi'an 710086, China)

**Abstract:** Aimed at the limitation of the navigation of the embedded handheld devices, the display terminal of Beidou navigation based on the STM32+ucGUI is put forward. Using the Beidou navigation receiver module-UM220, the navigation information is obtained. The procession of information is completed by the interrupt handling of the STM32. In order to realize the human machine interface friendly, the ucGUI is transplanted on the platform. The electronic compass is completed efficiently by the auto device object of the ucGUI. The experiment result shows that the navigation time of cold boot is less than 2 min while that of hot boot is less than 1 min. Either the longitude or the latitude is affected by the speed and the deviation is less than 3 m. It can satisfy the system requirements of non-contact, online, real time, higher precision and rapid speed, as well as strong anti-jamming and stabilization.

**Key words:** Beidou navigation; STM32; ucGUI computer vision

## 1 引言

随着导航模块的小型化,在手持设备上实现导航定位的应用越来越广泛。目前,对于带有界面

显示功能的手持定位设备,主要都是使用 GPS 系统,并且基于“嵌入式处理器+嵌入式操作系统+GUI”实现的<sup>[1-3]</sup>。然而,随着我国自行研制的“北斗卫星导航定位系统”不断完善,目前已经逐步得到应用,将来应用前景必将更加广阔<sup>[4-6]</sup>。同时对

收稿日期:2013-08-28;修订日期:2013-09-05.

\* 通信联系人,E-mail:hujiansheng121@163.com

于某些只需实现简单显示功能,对人机界面要求不高的应用场合,基于上述复杂嵌入式构架进行设计,不仅是对资源的浪费,也提高了开发难度,同时对于成熟产品的扩展也不太适用。

本文从某款手持设备用户需求出发,只需要提供简单的定位信息,而不需要复杂的导航界面,同时兼顾设备已用的其他功能模块,提出一种基于 STM32+ucGUI 构架的手持北斗显示终端系统,用于实现北斗(BD2)信号的接收和解析,并实现简单导航界面显示。本文重点研究系统的总体结构、北斗定位数据接收解析和电子罗盘界面设计等内容。

## 2 系统总体结构与思路

系统在硬件上以 STM32 嵌入式微处理器为核心,扩展了 LCD、GPS 模块、存储器、温湿度传感器和气压传感器,实现了对经纬度、高度、速度和温湿度信息的实时采集,根据用户需求,可对特定的应用领域进行精确化管理和科学预测。如图 1 所示。

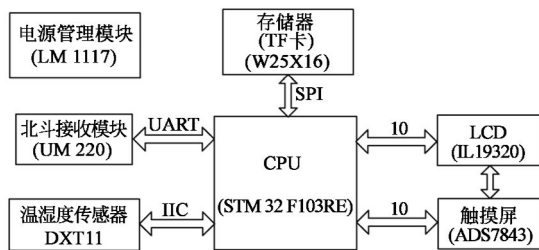


图 1 系统组成结构

Fig. 1 Structure of system

系统处理器采用 STM32F103BE,使用高性能的 ARM Cor-texTM-M3 32 位的 RISC 内核,工作最高频率可达 72 MHz,内置高速存储器(512K 的 Flash 和 64K 的 SRAM),不需要使用专门的外部 Flash 或 ROM 固化代码,具有丰富 IO 端口,以及 ADC、I2C、SPI、TIMER、UART 等模块<sup>[7]</sup>,通过科学的复用可以满足应用需要。

系统存储器包括串行 Flash 和大容量 TF 卡,均通过 SPI 总线与 CPU 进行数据传输。串行 Flash 使用 W25X16,用于存储系统的中英文字库、触摸屏校验参数等。TF 卡用于存储系统图标、应用图片和文本等大容量数据;LCD 模块是

由一块真彩液晶屏(240×320)和 LCD 控制器集成,LCD 控制器位 ILI9320,通过 16 位的 8080 总线与 CPU 进行数据交换,同时提供 4 线电阻采样端与触摸屏的 AD 芯片 ADS7843 相连;CPU 使用 4 个 IO 端口,模拟 SPI 总线与触摸屏 ADS7843 进行通信;CPU 通过 IIC 总线,与温湿度传感器 STH1X 进行通信。

在软件上,系统采用“驱动层+GUI”的形式,即在底层硬件驱动的基础上,直接移植 ucGUI 图形用户接口。系统对实时性要求不高,且任务量相对较少,完全可以使用 CPU 中的定时器和中断实现简单的任务调动;同时为了保证 GPS 信息的可视化显示,应该使用图形用户接口(GUI),而 ucGUI 又是最合适本系统的,可为任何使用 LCD 图形显示的应用,提供高效的独立于处理器及 LCD 控制器的图形用户接口,可独立于操作系统,直接使用,支持丰富的图形库、字体和窗口管理器<sup>[8]</sup>。

本文重点阐述 BDII 定位信息的接收、解析和显示等内容的设计。

## 3 北斗定位模块控制与数据解析

### 3.1 UM220 模块及其初始化

UM220 是北斗联星公司针对车辆监控、气象探测和电信电力授时等应用推出 BD2/GPS 双系统模块。UM220 是市场上尺寸最小的 BD2/GPS 模块,集成度高,功耗低,非常适合北斗系统大规模应用的需求。这里只研究其中北斗定位信息的提取和显示。产品具有尺寸小、重量轻、功耗低等优点,同时支持 BD2 和 GPS 的但系统独立定位和多系统联合定位,支持 UART, SPI, 1PPS 和 I2C 等多种接口。

UM220 模块正常工作前,要通过指令对其进行初始化。主要配置如下信息;

(1)设置串口指令,对模块输出数据的端口和协议进行配置,指令如下 \$CFGPR1,1,0,115200,3,2。这里使用串口 1,数据长度为 8 bit,波特率为 115 200,模块输入的数据格式为和芯星通软件接口协议,即对模块进行配置,输出的数据格式为 NMEA 协议。

(2)设置卫星系统配置,接收机收到该指令后会自动复位,设置的开启卫星频点在复位后生效。

\$CFGSYS,h10。这里为了验证北斗系统的定位效果,仅启动北斗定位系统。

(3)设置输出 NMEA 配置,\$CFGNMEA,h30.输出的 NMEA 版本为在 version3 基础上扩展的 BD2 相关语句。

(4)设置消息输出频度。通过该指令可设置模块 NMEA 消息类中各条指令的输出频率,也可屏蔽不需要指令。\$CFGMSG,0,1,0,\$CFGMSG,0,2,0,\$CFGMSG,0,3,0,\$CFGMSG,0,5,0,关闭 GLL,GSA,GSV,VTG 等 4 条指令,

指令输出频率保留默认设置,即 1 Hz。

### 3.2 BD2 数据的接收

UM220 每条 NMEA 指令都是一串以“\$”开始,以“\*”结束的字符串,不同指令的数据长度也不同,正确的存储 BDGGA 和 BDRMC 指令是正确解析所需的经纬度、高度、速度和角度信息的前提。在硬件电路上,UM220 模块与 STM32 的 UART2 相连,每隔 2 s 两条指令发送一次,因此将使用 STM32 的 UART2 中断函数来处理接收的数据更为方便。需要设置的变量如表 1 所示。

表 1 BD2 数据获取函数所需变量

Tab.1 Variables of getting BD2 data function

| 序号 | 变量名称        | 变量类型   | 变量含义                                                                                                             |
|----|-------------|--------|------------------------------------------------------------------------------------------------------------------|
| 1  | HEADER_BUF  | * char | 指令类型(报头字符串)缓存                                                                                                    |
| 2  | BDGGA_BUF   | * char | BDGGA 指令的数据缓存                                                                                                    |
| 3  | BDRMC_BUF   | * char | BDRMC 指令的数据缓存                                                                                                    |
| 4  | BD_RX_LEN   | U8     | 各缓存当前的长度标识                                                                                                       |
| 5  | BD_RX_EN    | U8     | 接收 BD 数据使能,与 BD 解析程序通信的信号量,防止缓存中的数据在没有读完的情况下又被更新。1:可以更新缓存;0:不可以更新缓存。                                             |
| 6  | BD_CMD_NUM  | U8     | 已接收的 BD 指令数量。由于只接收 BDGGA 和 BDRMC 两条指令,0~2。在 BD 获取程序中为 2 时,令 BD_RX_EN=0;在 BD 解析程序中,两条指令解析完成后,复位(0),同时 BD_RX_EN=1; |
| 7  | BD_BUF_FLAG | U8     | 更新 BD 数据缓存使能。1:可以;0:不可以。                                                                                         |
| 8  | BD_RX_STA   | U8     | 0 表示正在接收指令包头,如“BDGGA 或 BDRMC”;1 表示正在接收 BDGGA 的数据;2 表示正在接收 BDRMC 的数据。                                             |

BD 数据获取函数分成 3 个部分,即 BD 指令类型的识别,按照类型存储两种指令数据和 BD 指令接收结束。

(1)BD 指令类型识别,在 BD\_RX\_EN==1 时,如果收到指令开始符号“\$”,则表明一条完整的指令开始,设置 BD\_BUF\_FLAG 为 1,GPS\_RX\_STA 为 0,BD\_RX\_LEN 为 0,即允许更新 BD 数据缓存,开始接收指令报头。

(2)按照类型存储两种指令,紧随“\$”开始符后的是指令报头字符串,即“BDGGA”或“BDRMC”,若 BD\_RX\_STA 为 0,即正在接收报头字符串时,则将接收到的指令字符串存储在

HEADER\_BUF[] 中,由于设置模块仅输出“BDGGA”和“BDRMC”两种指令,因此根据第 3 位判断当前输出指令的类型,若 HEADER\_BUF[2]为‘G’,则表明当前为 BDGGA 指令,设置 BD\_RX\_STA 为 1,若 HEADER\_BUF[2]为‘R’,则表明当前为 BDRMC 指令,设置 BD\_RX\_STA 为 2。在此过程中,执行 BD\_RX\_LEN++,当 BD\_RX\_LEN==5 时,则根据 BD\_RX\_STA 的值,将指令的其余字符串存储到相应的缓存中,即 BD\_GGA\_BUF[]或 BD\_RMC\_BUF[]。

(3)BD 指令接收结束,当串口接收到‘\*’后,则表明一条指令的结束,另 BD\_CMD\_NUM

加 1, 当 BD\_CMD\_NUM 为 2 时, 表明分别接收到 1 条 BDGGA 和 1 条 BDRMC 指令, 此时令 BD\_RX\_EN 为 0, 停止接收模块发送的指令, 保护指令缓存, 等待对指令的解析, 并将计数器 BD\_RX\_LEN 清零。

这里需要重点强调是, BD\_RX\_EN 和 BD\_CMD\_NUM 两个变量实际上完成了 BD 数据接收程序和 BD 数据解析程序之间的信号量通信功能, 保证了接收缓存存在同一时刻只能被一个任务占用, 避免了缓存读写冲突的发生。

### 3.3 BD 数据的解析

为了便于应用程序调用各类信息参数, 系统必须对正确接收到的两条指令进行解析, 即提取出用户需要的时间、经(纬)度、高度、速度和方向等信息, 故需要定义如下变量。每条 NMEA 中包含的不同信息之间以“,”相间隔, 这里以对 BDGGA 为例。\$BD2GGA, 063952.000, 4002.2299, N, 11618.0968, E, 1, 4, 2.788, 37.254, M, 0, M, \*71 是一条典型的 GGA 指令, 各段依次含义为时间(063952.000)、纬度(4002.2299)、南纬或北纬指示(N)、经度(11618.0968)、东经或西经指示(E)、定位状态标识(1)、参与定位的卫星数(4)、水平精度因子(2.788 3)、椭球高(37.254)、椭球高单位(M)、海

平面分离度(0)、海平面分离度单位(M)、差分校正时延(空)、参考 ID(空)、结束符合和校验和(71)。

当 BD\_CMD\_NUM 为 2, 停止对 BD2 指令的接收, 依次对进行解析。通过变量 seg 的值确定各信息段, 再根据各段信息的长度即可提取出所需的信息, 函数流程如图 2 所示。

表 2 BD2 数据解析函数所需变量

Tab. 2 Variables of resolution function

| 序号 | 变量名称      | 变量类型   | 变量含义            |
|----|-----------|--------|-----------------|
| 1  | time      | * char | 存储时间            |
| 2  | longitude | * char | 经度数值            |
| 3  | longti_an | char   | 经度方向, E 或 W     |
| 4  | latitude  | * char | 纬度数值            |
| 5  | lati_an   | char   | 纬度方向, N 或 S     |
| 6  | lock      | char   | 定位标识            |
| 7  | height    | * char | 高度              |
| 8  | speed     | * char | 行进速度            |
| 9  | angle     | * char | 行进方向            |
| 10 | seg       | U8     | 对 BD2 语句中的“,”计数 |
| 11 | buf_p     | U8     | 对指令缓存中的字节计数     |
| 12 | inf_p     | U8     | 对各段信息中的字节计数     |

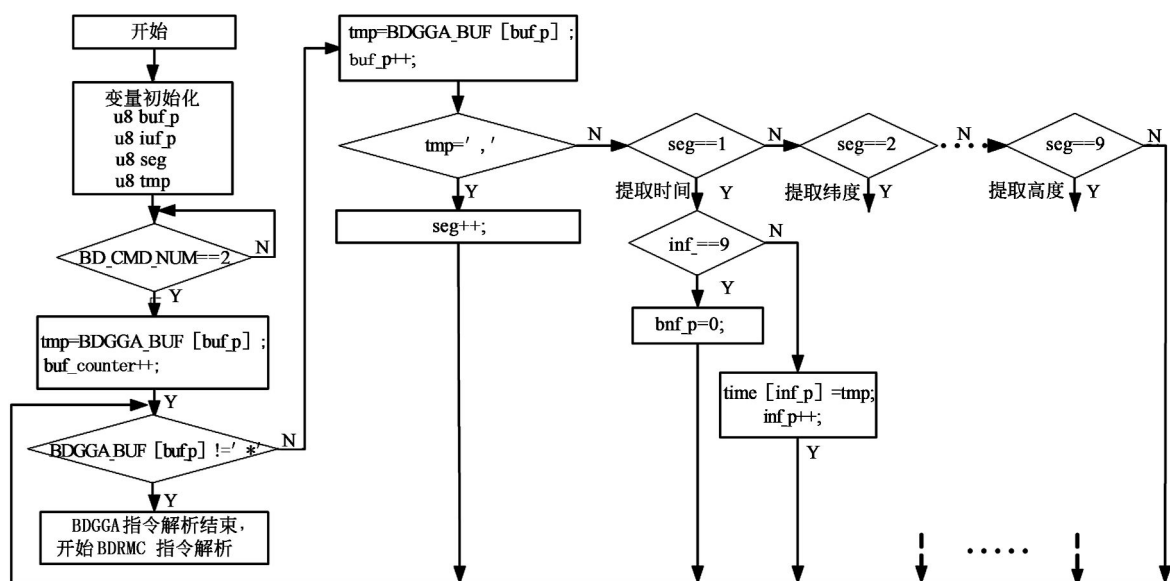


图 2 BD2 数据各段信息提取函数流程图

Fig. 2 Flow of the information extraction of BD2 data

## 4 显示界面设计

### 4.1 ucGUI 的移植

ucGUI作为一种精简、高效的嵌入式图形支持系统,在使用过程中主要完成两个工作,一是要将其移植到自己的硬件平台上,二是要根据需求编写图形应用程序。

由于ucGUI可以支持任意型号的LCD控制器和CPU,即可独立运行也可与ucOS操作系统配合使用,因此关于ucGUI移植相关的参考文献很多<sup>[8-10]</sup>,但是在具体移植步骤也有所不同,主要体现在对配置文件的修改和底层LCD驱动控制程序的编写。

本系统在移植ucGUI时,结合LCD控制器的型号,在编写驱动函数和修改配置文件时主要参考了文献<sup>[10]</sup>,但是由于是独立运行ucGUI,没有系统支持,因此需要在移植进行修改。具体的修改内容和移植过程,前期已撰写并发表于网上。

### 4.2 自动设备对象(Auto device object)

根据要求,系统的导航定位信息应以电子罗盘的形式直观形象地提供给用户。电子罗盘是在2-D图形的基础上,对几何图形进行静态(表盘)或动态(指针旋转)的显示。由于定位信息的实时更新,界面中的部分几何图形也需要实时刷新,而ucGUI中的自动设备对象(Auto device object)就是针对需要实时更新或改变显示内容的需求场合,尤其是像指针这样屏幕中的小部分实时更新的场合,它基于内存设备(memory devices),可以自动区分由固定对象组成的静止区域和由移动变化对象组成的动态更新区域,不需要更新整个界面,因此节约CPU计算时间,有效地避免屏幕抖动<sup>[8]</sup>。使用对象的主要过程及关键代码如下:

(1)定义绘制路径信息的结构体变量。

```
typedef struct {
    GUI_AUTODEV_INFO AutoDevInfo; //
    自动设备对象信息
    GUI_POINT aPoints[5]; //组成指针的5
    个顶点
    float Angle; //指针旋转角度
} PARAM;
编写绘制函数
```

```
static void Draw(void * p) {
    PARAM * pParam = (PARAM *)p;
    if (pParam->AutoDevInfo.DrawFixed) {
        /* 绘制固定对象组成的静态区域 */
        GUI_ClearRect(0,40,240,280); //清空表
        盘区域
        GUI_SetColor(GUI_BLUE); //设置前景色
        (蓝色)
        GUI_AA_FillCircle(Center_X, Center_Y,
        Radius_EX); //绘制外实心圆
        GUI_SetColor(GUI_WHITE); //设置前景
        色(白色)
        GUI_AA_FillCircle(Center_X, Center_Y,
        Radius_IN); //绘制内实心圆
    }
    /* 绘制移动指针 */
    GUI_SetColor(GUI_BLUE);
    GUI_AA_FillPolygon(pParam->aPoints, cou-
    ntof(_aNeedle), MAG * 120, MAG * 160);
    if (pParam->AutoDevInfo.DrawFixed) {
        /* 以字符串格式显示方向信息 */
        GUI_SetTextMode(GUI_TM_TRANS);
        GUI_SetColor(GUI_RED);
        GUI_SetFont(&GUI_Font8x16);
        GUI_GotoXY(120,140);
        GUI_DispString(angle); }
    调用自动设备对象
    void _DemoCompass(void) {
        ... ..
        while (1) {
            GUI _ MEMDEV _ CreateAuto
            (&AutoDev); //创建自动对象设备
            GUI _ RotatePolygon ( Param. aPoints, _
            aNeedle, countof(_aNeedle), Param. Angle); //
            旋转指针
            GUI _ MEMDEV _ DrawAuto ( &AutoDev,
            &Param. AutoDevInfo, &_Draw, &Param); //
            绘制自动设备对象
            ... ..
            if(GUI_GetKey==BACK){ //如果按下返
            回键
                GUI _ MEMDEV _ DeleteAuto (&AutoDev); //
                删除自动对象设备
```

```

GUI_Clear();
return
}
}

```

## 5 图片显示及界面效果

系统对 BMP 或 JPG 格式的图片显示,主要针对以下两种应用:一是复杂图形界面的显示,如



图 3 系统最终效果图

Fig. 3 Final effect of sysetm

主界面的背景及按键 LOGO;二是简易电子地图的预览及缩放功能,对大范围高分辨率的电子地图,

根据需要,按照地图上的像素位置与实际地理坐标位置的对应关系,对图片上的指定区域进行剪裁显示,以及缩小和放大显示。

上述功能的实现,需要调用 ucGUI 的位图显示功能,这里需要使用如下 API 函数,由于篇幅限制,具体调用方法这里不再叙述。

系统的主界面、电子罗盘界面以及电子地图界面最终效果如图 3 所示。

## 6 结 论

本文研究的系统,硬件上以 STM32 嵌入式处理器和北斗 II 模块 UM220 为核心,软件设计上以 ucGUI 为核心实现了电子罗盘界面,实现了精确定位及信息的可视化显示。分别在静止,步行(1~2 m/s)和车载(60~100 m/s)等三种环境下对系统性能进行测试,结果表明系统冷启动时的定位时间小于 2 min,热启动时的定位时间小于 1 min,经度纬度不受运动速度影响,误差小于 3 m,而由于 GPS 定位原理的局限性,实际运动速度越快,定位结果越精确。同时,与其他同类定位终端相比,该系统的最大优点是成本低,功耗小,可单独应用于定位领域,也可以此系统为平台,进行二次开发和扩展,应用于更多领域。

## 参 考 文 献:

- [1] 吴红兰,黄圣国. 嵌入式 GPS 数据采集与分析系统设计[J]. 实验室研究与探索, 2011, 30(1): 54-57.  
WU H L, Huang S G. Design of embedded GPS data receiving and analysis system [J]. *Research and Exploration in Laboratory*, 2011, 30(1): 54-57. (in Chinese)
- [2] 魏强,王晓浩,周兆英,等. GPS 的微型系统及其制导[J]. 光学精密工程, 2009, 17(7): 1660-1664.  
Wei Q, Wang X H, Zhou Z Y, et al. Guidance and implement of micro system based on GPS [J]. *Optics and Precision Engineering*, 2009, 17(7): 1660-1664 (in Chinese)
- [3] 张洋溢,王忠. 基于 ARM 的 GPS/GPRS 多功能手持终端的设计与实现[J]. 计算机测量与控制, 2011, 19(12): 3050-3053.  
Zhang Y Y, Wang Z. Design and implementation of an embedded GPS/GPRS multi-function handheld terminal based on Linux [J]. *Computer Measurement & Control*, 2011, 19(12): 3050-3053. (in Chinese)
- [4] 沈鹏,徐景硕,张建忠. 北斗导航系统及其在无人机上的应用[J]. 航空电子技术, 2007, 38(2): 5-9.  
Shen P, Xu J S, Zhang J Z. BeiDou navigation positioning system and its application to unmanned aerial vehicle [J]. *Electronics for Aeronautics and Astronautics*, 2007, 38(2): 5-9. (in Chinese)
- [5] 钱成越,薛原. 基于北斗导航系统的移动导航定位终端设计[J]. 现代计算机, 2010, 6: 165-169.  
Qian C Y, Xue Y. Design of a mobile navigation terminal based on Beidou (COMPASS) navigation satellite systems [J]. *Modern Computer*, 2010, 6: 165-169. (in Chinese)

- [6] 张晓安, 逯亮清, 杨彪. 应用延长基线法实现北斗双星的快速定向[J]. 光学精密工程, 2010, 18(11): 2461-2466.  
Zhang X A, Lu L Q, Yang B. Fast orientation of Beidou bi-satellite by extending baseline [J]. *Optics and Precision Engineering*, 2010, 18(11): 2461-2466. (in Chinese)
- [7] STMicroelectronic. STM32 Reference Manual [EB/OL]. <http://www.st.com/web/catalog/mmc/FM141/SC1169/SS1031/LN1565/PF164485>.
- [8] uc-GUI Manual v4. [www.micrium.com](http://www.micrium.com) [EB/OL]. <http://micrium.com/rtos/ucgui/overview>.
- [9] 北京翔宇星通. UM220 接口协议, 2011. [EB/OL]. <http://download.csdn.net/download/snowdrinker/5508139>.  
BeiJing xiangyuxingtong. UM220 interface protocol, 2011 [EB/OL]. <http://download.csdn.net/download/snowdrinker/5508139>. (in Chinese)
- [10] Code\_HXH. 基于STM32的ucGUI移植和优化[EB/OL]. <http://www.haogongju.net/art/73074>. 2011/7/27.  
Code\_HXH. The transplant and optimization of ucGUI based on STM32 [EB/OL]. <http://www.haogongju.net/art/73074>. 2011/7/27. (in Chinese)
- [11] 张洋, 刘军, 严汉字. 原子教你玩STM32[M]. 北京: 北京航空航天大学出版社, 2013.  
Zhang Y, Liu J, Yan H Y. *YUAN Zi teach you how to play STM32* [M]. Beijing: BEIHANG University Press, 2013. (in Chinese)

**作者简介:** 罗卫兵(1969—), 男, 湖南益阳人, 博士, 教授, 主要从事信息与通信工程领域的研究。E-mail: luoweibing@21cn.com

胡健生(1984—), 男, 辽宁兴城人, 博士研究生, 讲师, 主要从事嵌入式系统及军事通信的研究。E-mail: hujian-sheng121@163.com