

西安邮电大学

毕业设计（论文）

题目：_____ 基于 STM32 微控制器的

_____ 电子时钟的设计 _____

学院：_____ 自动化学院 _____

专业：_____ 自动化 _____

班级：_____ 自动 1403 _____

学生姓名：_____ 王泽鹏 _____

学号：_____ 06141094 _____

导师姓名：_____ 姚霁 _____ 职称：_____ 讲师 _____

起止时间：2017 年 12 月 5 日至 2018 年 6 月 10 日

毕业设计（论文）声明书

本人所提交的毕业论文《基于STM32微控制器的电子时钟的设计》是本人在指导教师指导下独立研究、写作的成果，论文中所引用他人的文献、数据、图件、资料均已明确标注；对本论文的研究做出重要贡献的个人和集体，均已在文中以明确方式注明并表示感谢。

本人完全理解《西安邮电大学本科毕业设计（论文）管理办法》的各项规定并自愿遵守。

本人深知本声明书的法律责任，违规后果由本人承担。

论文作者签名：

日期： 年 月 日

西安邮电大学本科毕业设计(论文)选题审批表

申报人	姚霁	职称	讲师	学院	自动化学院		
题目名称	基于 STM32 微控制器的电子时钟的设计						
题目来源	科研				教学	√	其它
题目类型	硬件设计	√	软件设计		论文		艺术作品
题目性质	应用研究		√		理论研究		
题目简述	利用 ST 的相关开发板, 设计实现基于 STM32 微控制器的电子时钟设计。						
对学生知识与能力要求	1. 熟练掌握嵌入式系统开发流程及方法; 2. 熟练掌握 C 语言; 3. 具有硬件电路调试基础; 4. 有较强的自学能力和分析能力。						
具体任务以及预期目标	1. 了解 Cortex-M4 处理器核的内部结构; 2. 掌握 STM32F4 的内部 RTC 模块工作原理; 3. 掌握 LCD 显示屏的驱动原理; 4. 学会使用开发工具对进行嵌入式开发设计; 5. 设计的电子时钟要在 LCD 上显示年月日小时分秒等信息, 并可进行时间的设置和闹铃等功能。						
时间进度	2017. 12. 5~2018. 1. 15 了解课题的背景知识, 具体细节, 明确所涉及的内容, 确立毕业设计的题目, 撰写开题报告; 2018. 1. 16~2018. 3. 12 学习有关本课题设计的相关知识, 包括 STM32 硬件电路及相关开发软件的学习; 2018. 3. 13~2018. 4. 1 规划设计方案, 硬件选型及购买, 搭建硬件电路; 2018. 4. 1~2018. 5. 10 编写电子时钟的硬件代码、仿真, 硬件调试; 2018. 5. 10~2018. 5. 30 系统联调, 完成实物设计, 同时撰写毕业设计论文; 2018. 6. 1~2018. 6. 10 修改、打印毕业设计论文, 准备 PPT , 完成毕设答辩。						
系(教研室)主任 签字	年 月 日			主管院长 签字	年 月 日		

西安邮电大学本科毕业设计(论文)开题报告

学号	06141094	姓名	王泽鹏	导师	姚霁
题目	基于 STM32 微控制器的电子时钟的设计				
<p>选题目的（为什么选该课题）</p> <p>（1）电子时钟应具有读取方便、显示直观、功能多样、电路简洁等诸多优点，符合电子仪器仪表的发展趋势，具有广阔的市场前景。</p> <p>（2）课题偏向实践，在实践中能学习到更多技巧和方法，可以很好地锻炼自己。</p> <p>（3）课题与本专业匹配程度高，本人将对该单片机数字电子时钟采用 stm32 自带的 RTC，用 1LCD12864 能够准确显示时间。</p>					
<p>前期基础（已学课程、掌握的工具，资料积累、软硬件条件等）</p> <p>（1）Verilog 编程与调试</p> <p>（2）Modelsim 仿真软件的应用</p> <p>（3）嵌入式控制系统 B、数字控制系统课程设计、数字电路与逻辑设计 B</p> <p>（4）参考资料《单片微机测控系统设计大全》</p> <p>（5）实验设计基本原则，分析、排查程序和电路故障的基本方法</p>					
<p>要解决的问题（做什么）</p> <p>用 STM32 设计一个数字电子时钟，采用 LCD12864 来显示并修改时间或闹钟。整个系统用 STM32 单片机作为中央控制器，由单片机执行采集内部 rtc 值，时钟信号通过单片机 I/O 口传给 LCD12864，单片机模块控制驱动模块驱动显示模块，通过显示模块来实现信号的输出，LCD12864 的显示及相关的控制功能。</p>					
<p>工作思路和方案（怎么做）</p> <p>工作思路：本设计拟实现的基本功能为计时显示功能，能够体现微控制器电子时钟的可扩展优越性，加入时间调整程序，使用按钮，调整年月日及时间功能。</p> <p>（1）第 1-2 周：选择合适的逻辑控制器-stm32 是一个低功耗，高性能 32 位单片机，片内含 4kBytesISP 的可反复擦写 1000 次的 Flash 只读程序存储器；</p> <p>（2）第 3-12 周：数码管动态显示。动态显示，即各位数码管轮流点亮，对于显示器各位数码管，每隔一段延时时间循环点亮一次。利用人的视觉暂留功能可以看到整个显示，但必须保证扫描速度足够快；硬件电路设计：本电子时钟设计所需电源电压为直流电压大小为 5v 的电压源；</p> <p>（3）第 13-14 周：对成品进行调试并达到最佳效果，完成论文的编写；</p>					
指导教师意见					
签字 年 月 日					

西安邮电大学毕业设计(论文)成绩评定表

学生姓名	王泽鹏	性别	男	学号	06141094	专业 班级	自动 1403
课题名称	基于 STM32 微控制器的电子时钟的设计						
指导 教师 意见	<div> <div>评分（百分制）：</div> <div>指导教师(签字)：_____</div> <div>年 月 日</div> </div>						
评阅 教师 意见	<div> <div>评分（百分制）：</div> <div>评阅教师(签字)：_____</div> <div>年 月 日</div> </div>						
验收 小组 意见	<div> <div>评分（百分制）：</div> <div>验收教师(签字)：_____</div> <div>年 月 日</div> </div>						
答辩 小组 意见	<div> <div>评分（百分制）：</div> <div>答辩小组组长(签字)：_____</div> <div>年 月 日</div> </div>						
评分比例	指导教师评分（20%） 评阅教师评分（30%） 验收小组评分（30%） 答辩小组评分（20%）						
学生总评 成绩	百分制成绩				等级制成绩		
答辩委员 会意见	<div> <div>毕业论文(设计)最终成绩(等级)：</div> <div>学院答辩委员会主任(签字、学院盖章)：_____</div> <div>年 月 日</div> </div>						

目 录

摘 要.....	1
ABSTRACT.....	2
第一章 引言.....	3
1.1 背景介绍.....	3
1.2 课题内容.....	3
第二章 方案设计.....	4
2.1 总体方案设计.....	4
2.2 关键问题.....	5
第三章 硬件设计.....	6
3.1 STM32F407ZGT6 芯片.....	8
3.2 启动电路模块.....	10
3.3 复位电路.....	10
3.4 TFT-LCD 触摸屏模块.....	11
3.5 蜂鸣器电路.....	12
第四章 软件设计.....	13
4.1 主程序模块.....	14
4.2 固件库的使用.....	15
4.3 时钟程序模块.....	16
4.4 TFT-LCD 显示程序.....	18
第五章 系统测试.....	19
5.1 安装 ST-LINK 驱动.....	19
5.2 MDK.....	19
5.3 TFT-LCD 测试.....	19
5.4 系统总体测试与调试.....	22
第六章 总结与展望.....	24
致 谢.....	25
参考文献.....	26

摘 要

单片机问世以来，以此为基础的应用技术飞速发展，但是随着需求的不断变化，51系列单片机已经不能满足社会的发展。与此同时产生的Cortex-M4主要应用于要求高性能、低成本、低功耗的嵌入式应用，以Cortex-M4为主控芯片的各种产品及衍生物更是在生活中的各个方面都有所应用，其已成为广大开发者进行嵌入式应用开发的首选之一。

本次毕业设计采用便是基于Cortex-M4为内核的STM32F4开发板，主要应用了其内部的RTC时钟模块，同时用TFT-LCD液晶显示屏作为显示模块，以STM32F4开发板为硬件平台，采用Keil uVision5开发环境进行设计，最终实现24小时制的时钟显示、日期显示和可设定任意时间的闹钟等功能，以上功能的显示和设置均通过TFT-LCD液晶显示屏完成。

结合当下市场的需要，通过液晶显示的电子时钟已经越来越流行，特别适合在家庭居室、办公室、大厅、会议室、车站和广场等地方使用，它具有显示清晰直观、走时准确、可以进行夜视等功能，并且还可通过丰富的接口扩展出若干子功能。

关键词：STM32F4；电子时钟；TFT-LCD；RTC；

ABSTRACT

Since the advent of SCM, the application technology based on it has developed rapidly. However, with the continuous changes in demand, 51 series single-chip microcomputer can no longer meet the development of society. At the same time, the Cortex-M4 is mainly used in embedded applications that require high performance, low cost, and low power consumption. The Cortex-M4 is the main product for the control of various products and derivatives in all aspects of life. With its application, it has become one of the first choices for developers to develop embedded applications.

This graduation design uses the STM32F4 development board based on Cortex-M4 as the core. It mainly uses its internal RTC clock module, uses the TFT-LCD liquid crystal display as the display module, and uses the STM32F4 development board as the hardware platform, adopting Keil. uVision5 development environment for the design, the ultimate 24-hour clock display, date display and alarm clock can be set to any time and other functions, the above functions are displayed and set through the TFT-LCD LCD screen.

Combining the needs of the current market, electronic clocks through liquid crystal display have become more and more popular, especially suitable for use in homes, offices, halls, conference rooms, stations and plazas, etc. It has a clear and intuitive display, accurate travel time, can carry out night Equivalent functions, and several sub-functions can also be extended through a rich interface.

Key words: STM32F4; Electronic clock; TFT-LCD; RTC;

第一章 引言

1.1 背景介绍

随着时代的快速更迭，科技的快速发展，从最原始的观太阳和时钟摆钟到目前的电子时钟，许多人为之不断的研究。目前，人们的生活节奏日新月异，每个人都对生活质量有着不同的需求。简单计时功能的时钟已经不能满足当下每个人的需求。希望在确保准确时间的基础上增加一些附加功能，例如日历等。

应用最广泛的MCS-51单片机虽然最容易上手学习，由于速度不足或缺乏存储空间，其性能在许多情况下都会受到影响。因此，由于MCS-51自身结构的局限性，与当前的新技术存在着明显的脱节。显然MCS-51单片不能满足本次毕业设计的需要，故采用STM32单片机进行开发。

基于STM32微控制器的电子时钟设计，因为其具有非常好的开放性和可操作性，可以迅速掌握其硬件以及已有固件库的使用方法。集时间，日期，星期等功能于一体的电子时钟，该系统具有直观的功能并且易于操作，市场前景广阔。该系统利用STM32F4开发板内部的RTC实时时钟模块，通过相应的软件和算法的设计实现计时的功能，并在TFT-LCD液晶显示屏上显示年，月，日，时，分，秒，周等。该系统具有电路简单、下一年自动校正、SPB界面友好、系统可靠性良好等特点。

综上所述，电子时钟应具有硬件电路实现简单，界面友好，系统可靠性高等诸多优点。可以预见，其符合当下电子仪器的发展潮流，拥有广阔的市场前景。

1.2 课题内容

整个系统选用STM32F407微处理器作为微处理器，电子时钟功能是通过开发板上的TFT-LCD液晶显示屏和蜂鸣器完成显示和铃响的。

实现功能：

（1）日历和时间可以独立调整，可以通过触摸实现小时，分，秒，年，月，日的调整，并可自由设置开机后的初始值，同时掉电再上电后走时保持准确。

（2）具备闰年的自动调整功能。

（3）具有闹钟计时器性能：可通过TFT-LCD液晶显示屏设定闹铃计时功能，在SPB界面中可自定义设置。到达闹钟时间时，蜂鸣器运行并保持5s。

第二章 方案设计

2.1 总体方案设计

本次毕业设计总体设计框图见图2.1。整体电路的三个组成部分为：单片机最小系统、蜂鸣器电路和LCD显示屏电路。其中单片机最小系统由启动模块、晶振模块、电源模块、RTC时钟模块、MCU模块和复位模块构成。在本次毕业设计中，闹钟的铃响由蜂鸣器实现。当到达预先设定的闹钟时间，蜂鸣器按照一定的频率运行，并在5秒钟后自动停止。

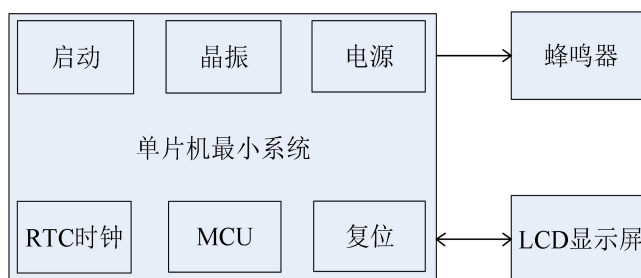


图 2.1 总体设计框图

(1) 单片机最小系统

在单片机最小系统中RTC时钟模块是芯片内部的一个时间运行模块，用于对时间运行进行计算，通过编程和算法实现年，月，日，时，分，秒的计算，实现了秒钟满60清0进1、分钟满60清0进1、小时满24清0进1、天数满28/29/30/31清0进1、月数满12清0进1、年份满10清0进1。

(2) LCD显示屏电路

触摸屏显示模块采用TFT-LCD液晶显示屏，采用TFT-LCD模块驱动芯片中的ILI 9320芯片显示触摸屏，TFT-LCD不同于TN-LCD和STN-LCD的简单矩阵。每个TFT可在LCD的每个像素上使用，此举可以减少非选择性状态下的干扰。扫描线数量的多少与LCD的静态特性非线性关系，从而大大提高了图像质量。TFT-LCD液晶显示屏的最重要的组成为电阻薄膜屏幕，其与监视器表面紧密匹配。手指碰触屏幕引起电阻变化，在X坐标和Y坐标上产生准确的信号。结果被传递到触摸屏控制器。控制器检测触头并计算其在平面内的位置，然后根据所获得的位置模拟鼠标模式操作。

(3) 蜂鸣器电路

蜂鸣器电路由定时器进行控制，时间运行到已设置的闹钟时间时触发闹钟功能，蜂鸣器按已设定好的计时时间响铃，从而实现闹钟功能。

2.2 关键问题

(1) 硬件设计中，整个电子时钟系统的核心处理器采用STM32，通过STM32内的RTC进行时间的控制和日期的读取。最后由单片机驱动TFT-LCD液晶显示器，并在屏幕上显示结果。

(2) 在软件设计中，以下研究集中于：

如何编写TFT-LCD的一系列驱动功能，包括LCD读取功能、写入功能、初始化功能和清屏功能。

要用C语言在LCD上绘图，必须先编写点函数。这是绘图的基本要求，因为需要在TFT-LCD液晶显示器上显示模拟时钟，所以必须编写圆和线的绘制曲线函数。通过函数绘制表盘，秒针、分钟和小时等，以便可以在TFT-LCD液晶显示器上模拟电子时钟。

STM32内部RTC负责时间以及日期信息的读取^[1]，同时具有自动计时、微调等功能，通过STM32已有的固件库可以实现读取一个字节的函数，写入一个字节函数，指定读取数据函数的位置，指定写入数据函数的位置等。LCD的信息传输是通过重新读取时间函数来实现的。

第三章 硬件设计

在本次毕业设计中，STM32F4 开发板及板载资源如图 3.1 所示。STM32F407 的内部资源在开发板上均得到充分利用，资源十分丰富。STM32F407 的所有内部资源都可以在开发板上进行实验验证，扩展十分丰富。整个开发板上的接口和功能模块非常简明。显影板的尺寸为 121mm×160mm。开发板的设计侧重考虑了人性化的设计。

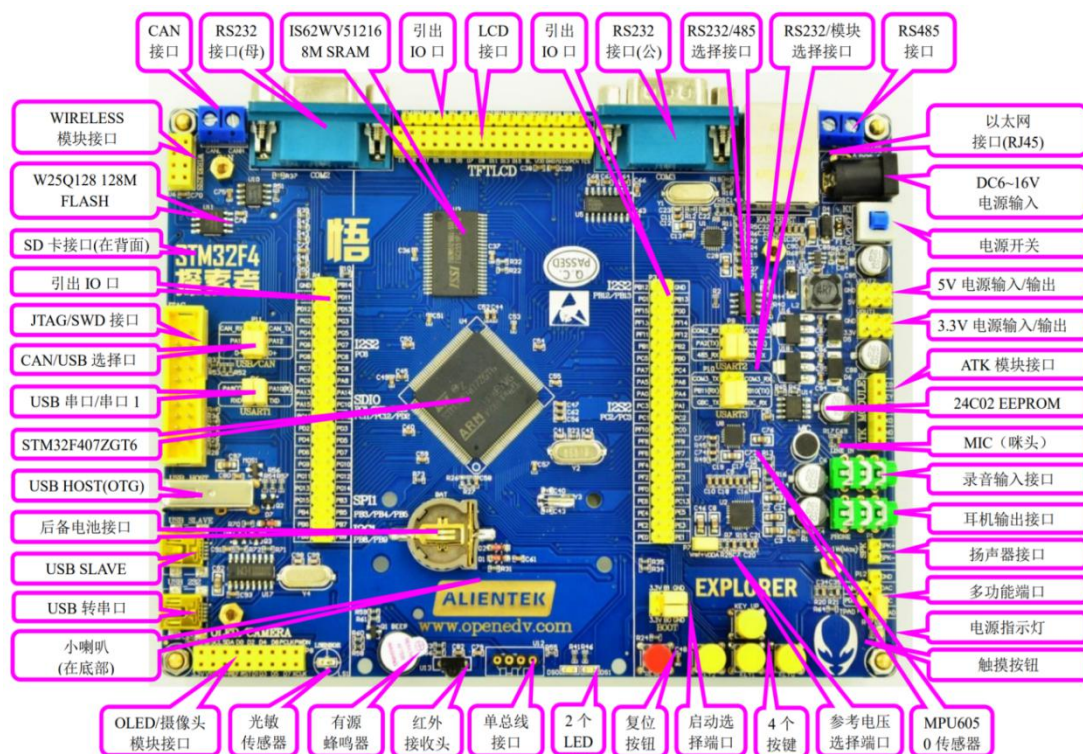


图 3.1 STM32F4 开发板

开发板 STM32F4 板载资源：

- ◆ CPU：STM32F407ZGT
- ◆ 1 个电源指示灯（蓝色）
- ◆ 2 个指示灯（DS0：红色，DS1：绿色）
- ◆ 1 个配备红外遥控器的红外接收头
- ◆ 1 个容量 256 字节的 EEPROM 芯片
- ◆ 1 个高性能音频编解码芯片，WM8978
- ◆ 的 2 路 RS232 串口（一公一母）接口
- ◆ 1 路传感器的单总线接口
- ◆ 1 个 ATK 模块接口
- ◆ 1 个支持电容触摸屏标准的 4.3 寸 LCD 接口

- ◆ 1 个 OLED 模块接口
- ◆ 1 个可用于程序下载和代码调试的 USB 串口
- ◆ 1 个 USB SLAVE 接口，用于 USB 从机通信
- ◆ 1 个 USB HOST (OTG) 接口，用于 USB 主机通信
- ◆ 1 个有源蜂鸣器
- ◆ 1 个录音头（MIC/咪头）
- ◆ 1 组多功能端口
- ◆ 1 组 5V 电源供应/接入口
- ◆ 1 组 3.3V 电源供应/接入口
- ◆ 1 个参考电压设置接口
- ◆ 1 个直流电源输入接口
- ◆ 1 个启动模式选择配置接口
- ◆ 1 个 RTC 后备电池座，并带电池
- ◆ 1 个复位按钮，可用于复位 MCU 和 LCD
- ◆ 4 个功能按钮，其中 KEY_UP 兼具唤醒功能
- ◆ 1 个电容触摸按键
- ◆ 1 个电源开关，控制整个板的电源

开发板STM32F4特性：

（1）丰富的接口：该板提供十种标准接口，便于各种外设的实验和开发。

（2）灵活的设计：板上的许多资源可以灵活配置以满足不同的条件。引出了除晶体振荡器占用的IO端口以外的所有IO端口，极大地方便了每个人的扩展和使用。此外，板上单击下载功能可以避免频繁设置B0和B1。基于STM32的开发只通过一根USB线即可实现。

（3）充足的资源：主芯片STM32F407ZGT6采用1M字节闪存以满足大容量存储器和大容量数据存储的需要，可扩展1M字节SRAM和16M字节闪存。为了满足各种应用的需要，高性能音频编解码器芯片、六轴传感器、快速以太网卡、光学传感器和各种接口芯片均已安装在板上。

（4）人性化的设计：每个接口都有一个丝印标签，并且框架使其易于使用。一些常用的外围设备通过丝网标记以方便搜索。界面位置设计方便，可充分、合理利用资源。

3.1 STM32F407ZGT6 芯片

STM32在运行速度、存储空间、内部功能模块的集成，基于串口的外设扩展等方面都符合本次毕业设计的需求，适合采用高级语言编程，并且在开发技术和仿真调试方面比MCS-51核心微控制器要更加超前。它包含50多个模型，用户可以轻松地选择满足他们需要的产品。

STM32F407ZGT6芯片具有如下特点：

（1）复用IO口重映射功能。

由于一些复用功能可以重新映射，所以在设计STM32开发板的PCB时更加容易。

（2）中断的输入可以使用全部引脚。

所有IO端口均可用作中断输入，这比许多ARM更有优势。当想要使用中断时，可以轻松使用该IO端口，而无需接收特定数量的脚。原理图的设计十分简单，同时PCB的设计也十分轻松。

（3）串口下载程序。

许多ARM芯片具有串行下载代码的功能，并且STM32保持了这种优秀的设计，大大降低了开发成本（没有JLink、ULink等，不需要专用的下载器）。

STM32F407ZGT6的释义表如表3.1所示：

表 3.1 STM32F407ZGT6 释义表

STM32	STM32 代表 ST 品牌 Cortex-Mx 系列内核（ARM）的 32 位 MCU
F	F4xx 系列为 1.8~3.6V
407	ARM Cortex-M4 内核，不加密模块
Z	Z 代表 144 管脚
G	G 代表 1M 字节 Flash
T	T 这一项代表封装，T 代表 LQFP 封装
6	工业级的工作温度范围：-40℃-85℃

STM32F407ZGT6的简单引脚图如图3.2所示：

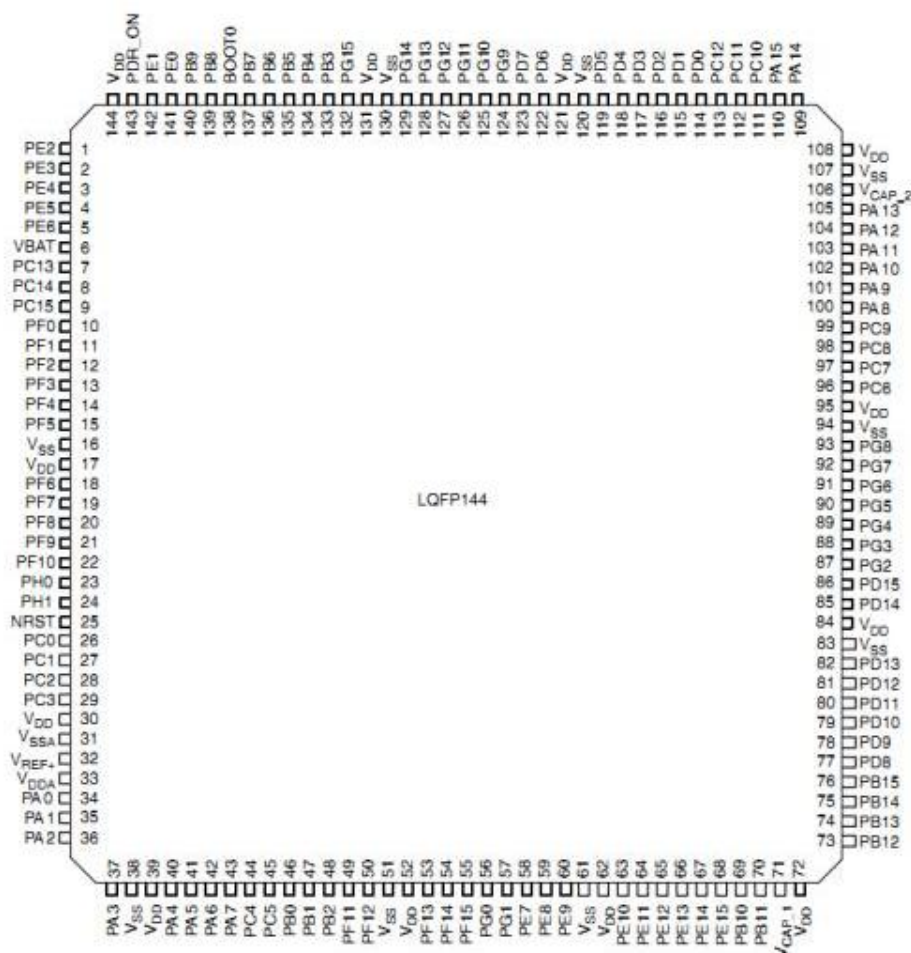


图 3.2 STM32F407ZGT6 的简单引脚图

STM32F407ZGT6 系列使用 ARM 的 Cortex-M4 32 位 RISC 内核。在正常工作期间，频率为 168MHz，其功能非常强大。选择 STM32F407ZGT6 作为此次设计的微处理器，主要因为其外围接口多种多样，在实现其它功能的时候更加方便，并且有足够大的闪存和内存。

ARM CORTEX-M4 处理器是最新的嵌入式应用专用的开发处理器。为了满足控制和信号的高效、易驱动的要求，增强了浮点运算，增加了 DSP 和并行计算的功能。其信号处理的高效和结合了 CORTEX-M 微控制器系列低功耗、低成本和易驾驶的优点。它可广泛应用于电机控制、车辆工程、电源管理、嵌入式音频和工业自动化需求市场，并提供了一种新的敏捷解决方案。

3.2 启动电路模块

可通过 STM32F407ZGT6 芯片的 BOOT [1: 0] 引脚，来控制三种不同的引导模式。启动模式如表 3.2 所示：

启动模式选择引脚		启动模式	说明
BOOT1	BOOT0		
X	0	主闪存存储器	主闪存存储器被选为启动区域
0	1	系统存储器	系统存储器被选为启动区域
1	1	内置SRAM	内置SRAM被选为启动区域

表 3.2 启动模式分配图

在系统复位后，SYSCLK 的第 4 个上升沿，BOOT 引脚的值将被锁存。用户可以通过设置 BOOT1 和 BOOT0 引脚的状态，来选择在复位后的启动模式。

在从待机模式退出时，BOOT 引脚的值将被重新锁存，因此，在待机模式下 BOOT 引脚应保持为需要的启动配置。在启动延迟之后，CPU 从地址 0x0000 0000 获取堆栈顶的地址，并从启动存储器的 0x0000 0004 指示的地址开始执行代码。

因为固定的存储器映像，代码区始终从地址 0x0000 0000 开始（通过 ICode 和 DCode 总线访问），而数据区（SRAM）始终从地址 0x2000 0000 开始（通过系统总线访问）。Cortex-M3 的 CPU 始终从 ICode 总线获取复位向量，即启动仅适合于从代码区开始（典型地从 Flash 启动）。STM32F103VCT6 微控制器实现了一个特殊的机制，系统可以不仅仅从 Flash 存储器或系统存储器启动，还可以从内置 SRAM 启动。

3.3 复位电路

如图 3.3 所示，使用板上的板载复位按钮重置 STM32。还具有重置 LCD 显示器的功能。由于 LCD 模块的复位端连接到 STM32 的复位端，因此按此键可将 STM32 与液晶一起复位。当 STM32 复位到低电位，系统电路复位到低电位，R 和 C 构成通电复位电路。同时，开发板将 TFT-LCD 的复位引脚连接到复位。复位按钮不仅可用于复位 MCU，还可用于重置 LCD。

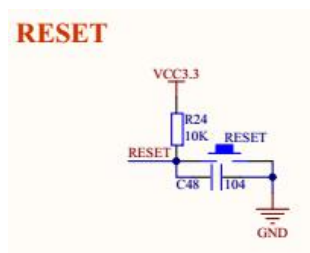


图 3.3 复位电路原理图

3.5 蜂鸣器电路

开发板 STM32F4 包含一个有源蜂鸣器见图 3.5:

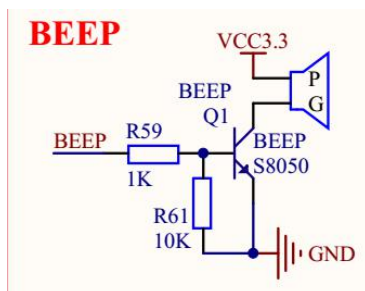


图 3.5 蜂鸣器原理图

有源蜂鸣器是带有振荡电路的蜂鸣器。当蜂鸣器连接到电源时，蜂鸣器瞬间短暂自动响起。在无源蜂鸣器的情况下，需要将多个频率（2 至 5KHz）的驱动信号施加到声音。这种设计的选择使用一个有源蜂鸣器，以便任何人都可以使用它。

在图 3.5 中，Q1 用于扩展流量，而 R61 是一个下拉电阻，用于防止 MCU 复位时偶尔发出蜂鸣声。BEEP 信号直接连接到 MCU 的 PB8。PB8 可通过 PWM 作为 PWM 输出和驱动蜂鸣器。

第四章 软件设计

本设计中使用的软件 Keil uVision5 是与 Keil 软件开发的 C 语言开发语言兼容的软件开发工具。与传统装配相比，C 语言在结构、可维护性、结构和功能等方面具有明显的优势。Keil 主体包含实时操作系统的核心（这也是嵌入式开发工程师选择该软件的主要原因），C 编译器，库管理器，宏汇编器，连接器和调试器，固件。与以前的版本相比，Keil uVision5 为开发应用程序提供了一个高效，紧凑的平台。其他更多的 ARM 芯片可以同时支持。

软件首先需要配置 STM32 的 RTC。

第一步是启用电源和备份存储器的时钟，然后更改区域以开始正常操作。

第二步是取消备份存储器的写保护，以方便在备份存储器中读写数据。同时，配置的时钟被识别以避免每次重置后重置时钟。

第三步是使外部低速振荡器，并选择 RTC 时钟源，以提供基本时钟信号用于电子时钟的操作。

第四步是启用 RTC 并配置它，包括分频、RTC 允许的配置位、更新配置和设置中断。这些设置为 RTC 根据设计的需要工作提供了基础。

第五步主要是描述中断服务功能。对于中断，有必要确定当前时间值的实时更新。

编程过程中有几个重要的函数。第一个是 TimeTySt 函数。这个函数的主要功能是设置软件的启动时间。这也对应于实时时钟的校准。Time_set 函数有六个参数：年，月，日，小时，分钟和秒。它们用于显示用户希望设置的年，月，日，小时，分钟和秒钟。返回值为 0 个显示设置失败。1 显示设置成功。程序的算法如下：如果年设置为大于 1970，则年将从 1970 减去，并乘以对应于变量存储年的秒数。如果设定的月份大于 1 月份，则该月份的数值将从该月份扣除。第二个变量存储在另一个月，然后进行类比，然后添加每个参数的相应变量，并获得 1970 年 1 月 1 日 0:00:00 秒的秒数。最后，秒数被写入 RTCYCNT 寄存器作为初始计数值。

另一个主要特性是 TimeIGET 函数。该函数的主要功能是读取计数器的值并将其转换为当前时间，如 time_set 函数的反函数。首先将 RT CYCNT 寄存器的值存储在变量中。该变量用于执行与时间缩放相关的计算。同时，需要定义结构日历来存储转换后的每个变量的值。规则也是一个简单的数学问题。可以使用商和剩余计算来获取与第二个变量相对应的年，月和日变量。

中断服务功能的处理也很简单。只需确定标志位并确定生成的中断是否需要第二个中断。如果是则调用 TimeIGET 函数并刷新当前时间。主要关注的是每个模块的初始化，包括 STM32 系统、RTC 模块和 TFT-LCD 显示模块。由于没有外部

键盘扩展，下载程序后需要进行时间校准，手动更改 TimeTySt 函数的参数值。由于中断时间被刷新，TFT-LCD 也能实时显示当前时间，实现电子时钟的效果。

4.1 主程序模块

主要功能有初始化功能、中断功能、计时功能和主要功能，如图 4.1 所示。它还包含显示屏和触摸的相应设置。在 main 函数中，初始化各种设置，然后设置变量的初始值，如年、月和日。最后，设置相应的 LCD 屏幕。数据被处理并被调用以使其出现在触摸屏上。字体颜色、字体大小或背景颜色设置。

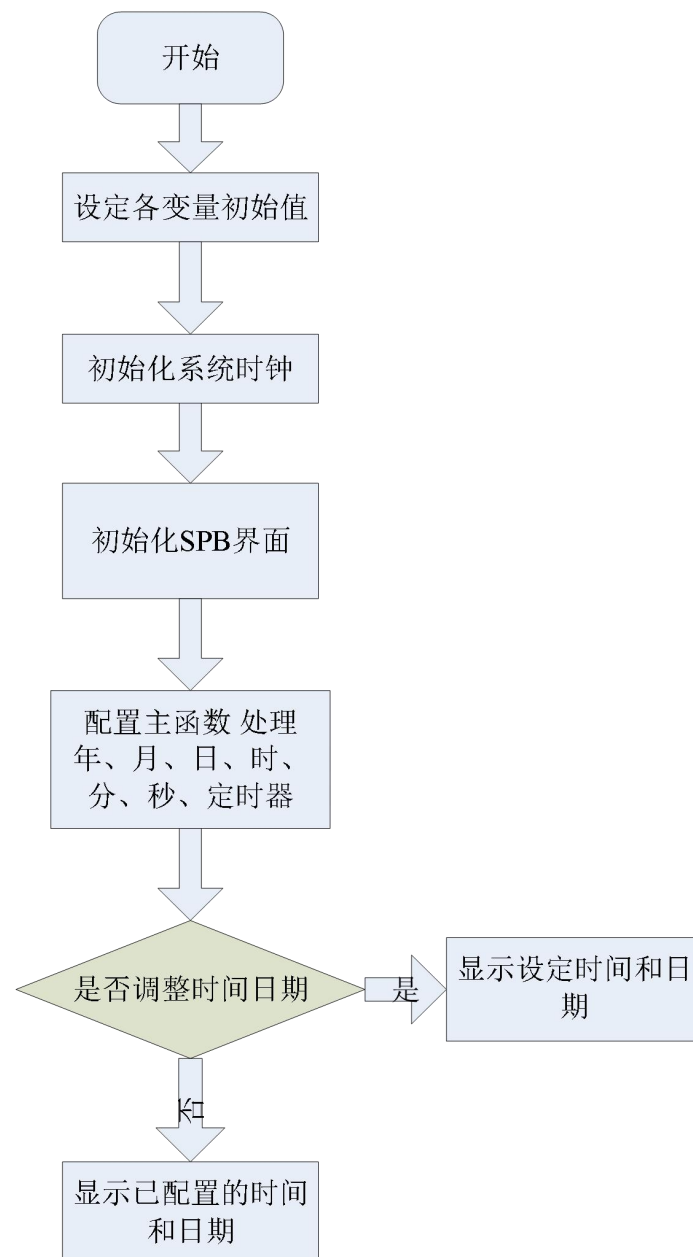


图 4.1 主程序流程图

```

//主程序
void main_task(void *pdata)
{
    u8 selx;
    ul6 tcnt=0;
    spb_init();          //初始化 SPB 界面
    spb_load_mui();      //加载 SPB 界面
    slcd_frame_show(0); //显示
    while(1)
    {
        selx=spb_move_chk();
        system_task_return=0;//请退出标志
        switch(selx)//发生了双击事件
        {
            case 1:calendar_play();break; //时钟
            case 2:sysset_play();break;   //系统设置
        }

        if(selx!=0XFF)spb_load_mui();//显示主界面
        delay_ms(1000/OS_TICKS_PER_SEC);//延时一个时钟节拍
        tcnt++;
        if(tcnt==500) //500 个节拍为一秒钟
        {
            tcnt=0;
            spb_stabar_msg_show(0);
        }
    }
}

```

4.2 固件库的使用

STM32 标准外设库的旧版本被称为固件或固件包中包含的程序、数据结构和包括微控制器在内的所有外围设备的性能特征。该库还包括每个外围设备驱动程序的描述和应用程序示例，并为开发人员提供了访问底层硬件的中间 API。通过使用固件库，开发人员不再需要了解基本的硬件细节，开发人员可以轻松应用每

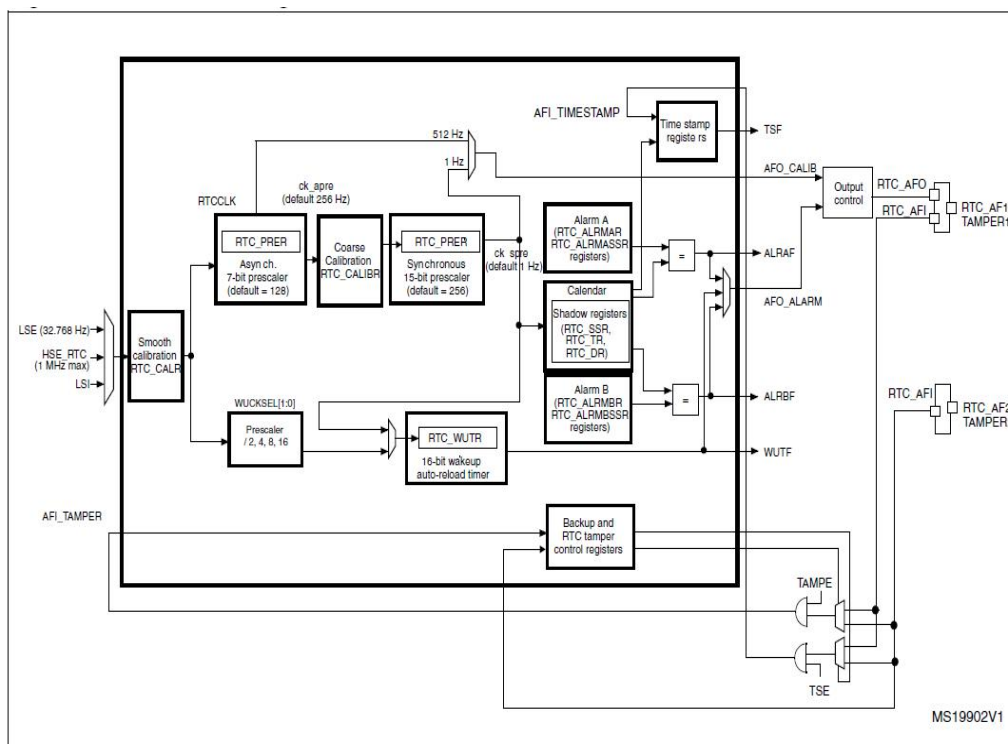
个外部应用程序。故使用固件库可以大大减少用户的研发周期并降低开发成本。每个外围驱动器由覆盖外围设备的所有功能的一系列功能组成。每个设备的开发都是由通用的 API（应用程序编程接口）驱动的，它可以规范驱动程序的结构、功能和参数名称。简而言之，使用标准外围库的最大优点是开发者可以灵活地使用每个外围设备而不知道硬件的细节。标准外围库涵盖了 GPIO 计时器、CAN、I2C、SPI、UART 和 ADC 的所有标准外围设备。相应的 C 源代码只使用最基本的 C 编程知识。所有代码都经过严格测试，易于理解和易于使用，包括完整的文档。这对于两种开发和应用都是非常方便的。

4.3 时钟程序模块

STM32F4 具有 RTC 实时时钟见图 4.2。使用的代码可以计算年、月、日、分钟、秒和周，并计算和输出函数。当系统正常运行时，内部 RTC 实时时钟使用 3. 系统的。当系统关闭时，电池供电的保证时间信息不会丢失。

RTC 主要由两部分组成。第一部分是 APB1 接口连接到 APB1 总线。APB1 总线时钟驱动，主要用于 APB 1 总线的读写。第二部分是一组可编程计数器。它由预分频模块和可编程计数模块组成。前者用于产生 1 秒的时基，后者用于初始化当前时钟并执行时钟计数。

通过设置 RTC_CRH 的最低有效位，可以暂停系统所需的秒数。在 RTCHCRL 中，位 5 至 0 是有效位，RTC 关闭、配置标记、寄存器同步标记、溢出符号、警告标志和第二符号。其中一些值得注意。例如，位 5 是 RTC 操作位。该位只能由硬件操作。该软件是只读的。在执行 RTC 操作之后，有必要确定操作是否完成。



1. On STM32F40x and STM32F41x devices, the RTC_AF1 and RTC_AF2 alternate functions are connected to PC13 and PI8, respectively.

图 4.2 RTC 结构图

判断日期是否正确：年月日时分秒小于零则不正确，月日时分秒都不能超过最大值，正确之后在判定是否是闰年，若为闰年，则判断二月是否满足要求。

闰年的判定：闰年应该是 4 整除（例如闰年 2004，闰年 2001），但不是所有的除以 4 的年份都是闰年。在一个可以被除以 100 的年份，一个可以被除以 400 的年份是闰年（例如，2000 是闰年）。一年可以分为 100，不能被 400 整除（例如，18001900 和 2100）不是闰年。

闹钟定时：可以很据自己的需求调节时间并且设置闹钟，可自行更改蜂鸣器响铃模式，闹钟时间到则蜂鸣器响。

下附闰年算法的部分代码：

```
/*闰年算法*/
if ( (year_number % 4 == 0) && (year_number % 100 != 0) || (year_number % 400 == 0) )
{
if ( (month_number == 1) || (month_number == 3) || (month_number == 5)
|| (month_number == 7) || (month_number == 8) || (month_number == 10)
|| (month_number == 12) )
{
```

```
if( month_number == 12)
{
if( day_number > 31 )
{
month_number = 1;    //月份加一
day_number = 1;      //天数为一
year_number++;
}
}
```

4.4 TFT-LCD 显示程序

在图 4.3 中可以很容易地显示任何 LCD 的操作过程。硬复位和初始化序列只能执行一次。绘图过程如下。在液晶屏上设置坐标、写克命令、写彩色数据、显示对应点和书写颜色。读取点的过程是读取坐标数据，读取 GRAM 命令，读取颜色数据以获得对应于点的颜色数据。

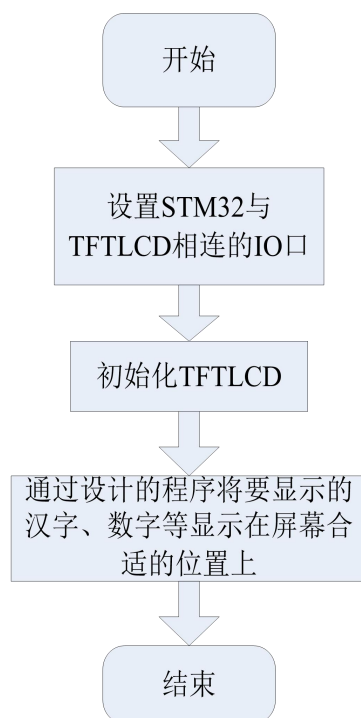


图 4.3 TFT-LCD 显示流程图

第五章 系统测试

5.1 安装 ST-LINK 驱动

ST-LINK 是 STMicroelectronics STM 8 和 STM 32 系列芯片的仿真器。它是一款最适合学习和使用 RAM 的开发工具，并且与 Keil uVision5 编译器环境完全兼容。易于操作，易于学习。

本设计就需要用到 ST-LINK 下载和调试程序，在此之前应该先安装 ST-LINK 的驱动。下载 ST-LINK 驱动后，直接双击驱动文件 dpinst_amd64，然后根据提示进行安装即可。安装后，有必要检查安装是否成功。把 ST-LINK 插到电脑上 USB 接口，然后在我的电脑\管理\设备管理器\通用串行总线控制器中看到一个 ST-Link driver，表明驱动安装成功。

5.2 MDK

MKK 是指 Keil FDK，也称为 DK-ARM，UVIEW 5 等。Keil FDK 软件为 CORTEX-R4、CORTEX-M、ARM7、ARM 9 处理器设备提供了一个完整的开发环境。Keil DK 是很容易学习和非常强大的。这是一个很好的嵌入式开发工具。

5.3 TFT-LCD 测试

开机显示姓名和学号，在蜂鸣器响一声后正常显示主界面，如图 5.1 所示。

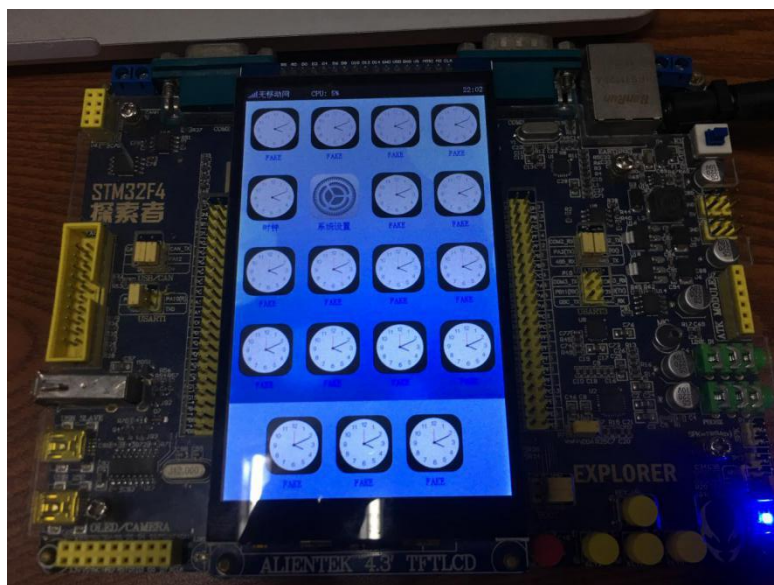


图 5.1 主界面图

下面为部分代码：

```

m_spb_dev spbdev;
//背景图路径选择
u8*const spb_bkpic_path_tbl[3][2]=
{
    {
        "1:/SYSTEM/SPB/BACKPIC/back_480610. jpg",
    },
};
//界面图表的路径表
u8*const spb_icos_path_tbl[3][2][8]=
{
    {
        {
            "1:/SYSTEM/SPB/ICOS/clock_110. bmp",
            "1:/SYSTEM/SPB/ICOS/clock_110. bmp",
            "1:/SYSTEM/SPB/ICOS/clock_110. bmp",
            "1:/SYSTEM/SPB/ICOS/clock_110. bmp",
            "1:/SYSTEM/SPB/ICOS/clock_110. bmp",
            "1:/SYSTEM/SPB/ICOS/set_110. bmp",
            "1:/SYSTEM/SPB/ICOS/clock_110. bmp",
            "1:/SYSTEM/SPB/ICOS/clock_110. bmp",
        },
        {
            "1:/SYSTEM/SPB/ICOS/clock_110. bmp",
            "1:/SYSTEM/SPB/ICOS/clock_110. bmp",
            "1:/SYSTEM/SPB/ICOS/clock_110. bmp",
            "1:/SYSTEM/SPB/ICOS/clock_110. bmp",
            "1:/SYSTEM/SPB/ICOS/clock_110. bmp",
            "1:/SYSTEM/SPB/ICOS/clock_110. bmp",
            "1:/SYSTEM/SPB/ICOS/clock_110. bmp",
            "1:/SYSTEM/SPB/ICOS/clock_110. bmp",
        },
    },
};

```

```
},
};
//三个主图标的路径表
u8*const spb_micos_path_tbl[3][3]=
{
{
    "1:/SYSTEM/SPB/ICOS/clock_110.bmp",
    "1:/SYSTEM/SPB/ICOS/clock_110.bmp",
    "1:/SYSTEM/SPB/ICOS/clock_110.bmp",
},
};
//界面图标的名字
u8*const icos_name_tbl[GUI_LANGUAGE_NUM][2][8]=
{
{
    {
        "FAKE", "FAKE", "FAKE", "FAKE",
        "时钟", "系统设置", "FAKE", "FAKE",
    },
},
};

//主图标对应的名字
u8*const micos_name_tbl[GUI_LANGUAGE_NUM][3]=
{
{
    "FAKE", "FAKE", "FAKE",
},
};
//初始化 SPB 各个参数
//返回值：0 操作成功
//其它，错误代码
```

5.4 系统总体测试与调试

此部分主要是把各模块的代码函数一起在工程里运行，把程序通过 ST-LINK 下载到开发板上然后再运行。最后的调试阶段由于需要频繁修改，故每完成一小块的功能后，就须立马调试，确保无误后再写下一段代码。在 Keil uVision5 软件内可以添加断点开始调试，另有单步调试、跳过全部函数调试，以上的功能都给工程的调试带来了极大的便利性。另外，还可以使用联机调试的功能，此功能需对整个工程全部编译后才可使用故效率略低。

电子时钟显示模块测试：电子时钟的算法无误，如图 5.2 所示。

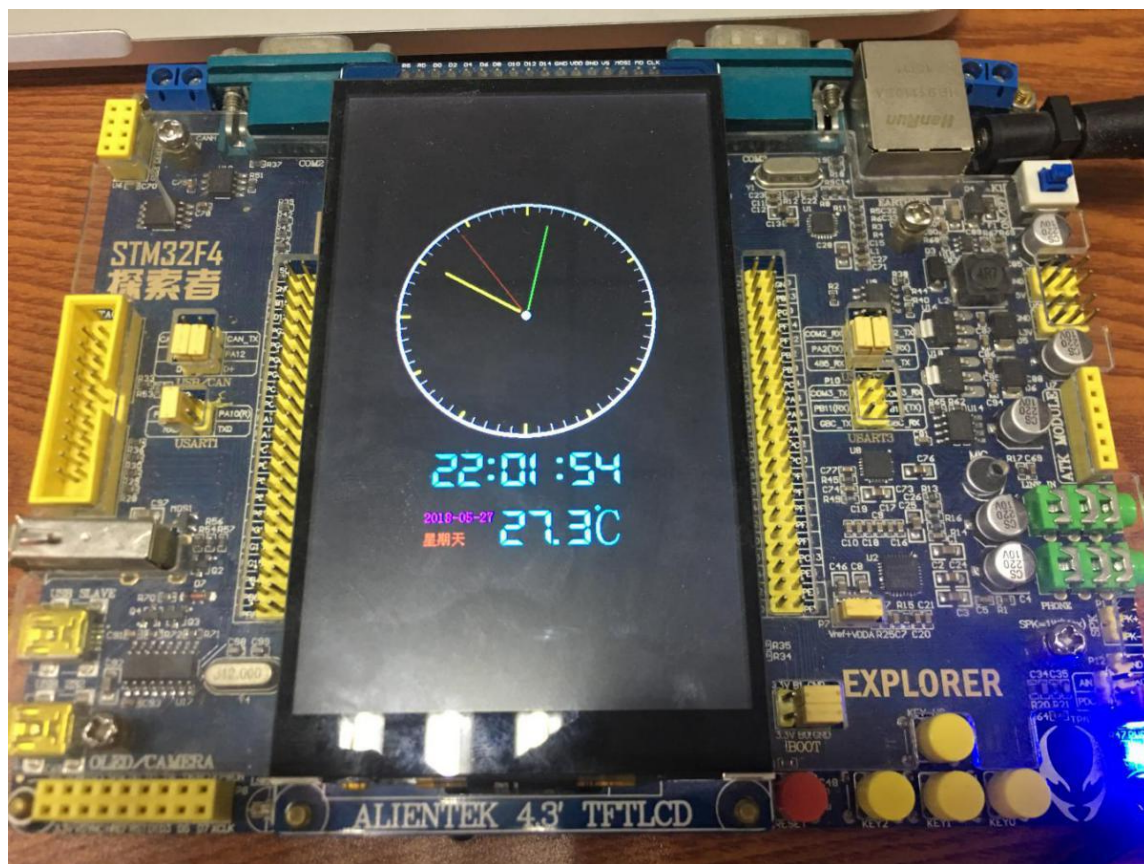


图 5.2 时钟显示图

闹钟的设置模块测试：电子时钟可以通过触摸调整数据，如图 5.3 所示。



图 5.3 闹钟设置界面图

定时报警模块测试：可发出声音，持续 5 秒钟，如图 5-4 所示。

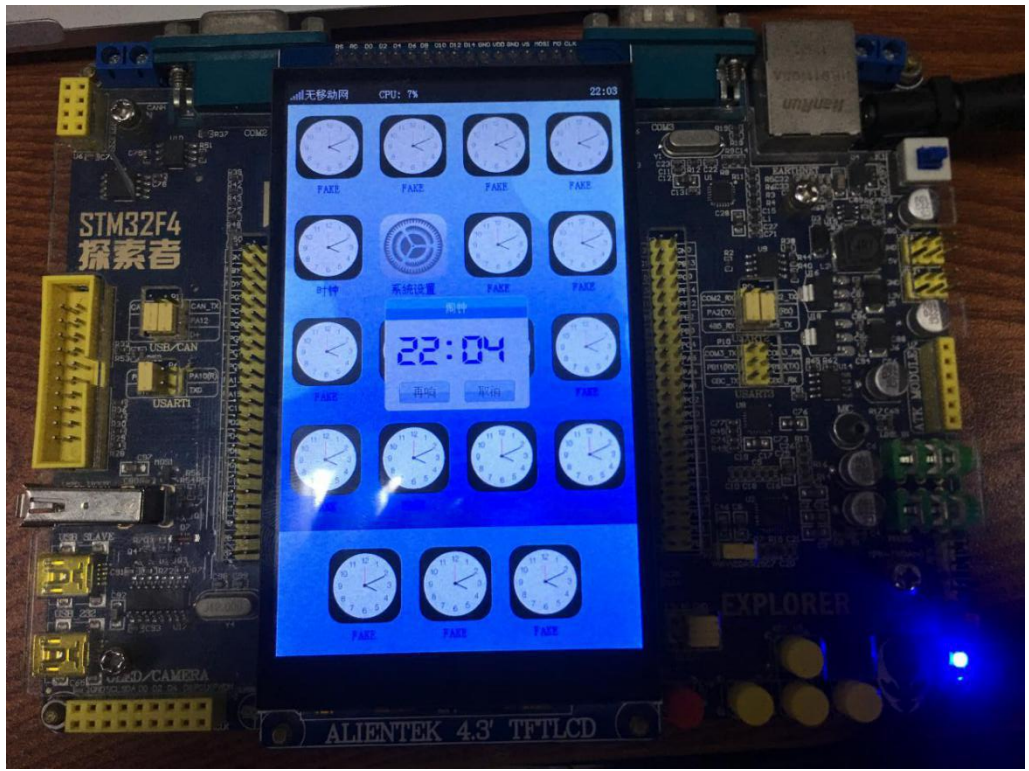


图 5-4 闹钟提醒图

第六章 总结与展望

在 STM32 F407 上实现的电子时钟使用户能够控制时间和警报变化。为了使使用者能够通过直观的方式进行控制，本设计使用了 TFT-LCD 显示屏，此举符合嵌入式设备和时钟的发展要求。作为本科毕业设计，此篇文章设计的作品以功能性作为首当其冲的考虑，此基础上尽可能的降低成本。本论文主要介绍了电子时钟的设计和实现方法，并且此基础上实现了目前广泛应用的可视化控制操作。叙述了 STM32F407 的硬件结构，RTC 时钟功能，闹钟中使用的数据结构和算法。事实上，RTC 只是一个计时器，所有的信息在关机后都会丢失，所以需要找到存储这些信息的区域，故将使用备份寄存器。即使电源关闭后，也可以使用纽扣电池供电，以便随时保存。

在完成此次毕业设计的过程中，本人对 ARM 体系有了更层次的了解，通过驱动 TFT-LCD 显示屏的方法，充分发挥 STM32F407 的强大性能。本论文完全实现了电子时钟的各项功能，因此也对嵌入式系统也更加体会深刻。基于 STM32F4 开发板良好的可扩展性，在已完成的电子时钟的基础上仍可以添加其它功能，例如：温度显示、自选闹钟音乐等。本次毕业设计给本人提供了不可多得的实践经验，我相信对我将来的生活、工作都会有很大的帮助。

致 谢

本次论文是在毕业设计指导老师姚霁导师的悉心指导下完成的，从最初的选题到论文的完成姚导师都付出了很大的心血与精力，姚老师推荐的一些相关的资料在本次毕业设计中有很大帮助，并且解答了在编写论文时遇到的很多疑惑，姚老师的知识很渊博，在遇到苦难或者有疑惑的地方都能及时的指导本人，衷心的感谢我的导师。

在本次毕业论文的完成过程中，一直得到了自动化学院领导和老师们的热情帮助和指导。从论文选题、课题研究到学位论文完成的这些过程中，都夹杂着每一个人的辛勤汗水。在此谨向他们表示我最诚挚的谢意和感激之情。

在本次论文研究的过程中，得到了很多同学的帮助和关心。遇到不懂的地方就会及时向他们请教，再重新梳理自己的思路。对论文的完成起到了很大的作用，对于他们的支持和帮助表示诚挚的谢意！最后，我向给予我帮助和支持的父母、老师、同学和朋友再次表达我衷心的感谢与感激之情！

参考文献

- [1] Labrosse Jean J., UC/OS 一 II--源码公开的实时嵌入操作系统, (邵贝贝译), 中国电力出版社, 2001 年
- [2] 冉明等, 基 UC/OS 一 II 的嵌入式系统的设计, 冉明等, 微机发展, 2004. 8
- [3] 谭浩强, C 程序设计(第二版), 清华大学出版社, 1999. 12
- [4] uC/OS 环境下的 C 语言编程, 北京单片机开放网, www.bimcu.com
- [5] 杜春雷. ARM 体系结构与编程. 北京:清华大学出版社, 2003. 7. 14
- [6] ARM 公司《ARM Architeture ReferenceM annual》 2000
- [7] STM32F407ZGT6 数据手册
- [8] 滕振芳, 张昆. 基于单片机的电子万年历的设计[J]. 价值工程, 2010(6):63.
- [9] FatFs 文件系统中文资料.pdf
- [10] STM32F10xxx_Library_Manual_ChineseV2, ARM 公司
- [11] UM0427 Oct.2007 Rev 2, ARM 公司, 2007