

目录

摘要.....	3
一、实验任务.....	4
1.1 基本要求.....	4
1.2 扩展部分.....	4
二、设计目标.....	5
三、关于我们.....	5
3.1 团队成员.....	5
3.2 READSOME 名字的由来.....	5
四、TFT 模块的设计与实现.....	6
4.1 功能简介.....	6
4.2 主要参数.....	6
4.3 界面设计流程图.....	11
4.4 遇到的问题.....	15
五、MP3 模块的设计与实现.....	17
5.1 硬件设计与选型.....	17
5.2 软件设计.....	22
六、时钟模块的设计与实现.....	26
6.1 硬件模块.....	26
6.2 软件设计.....	26
6.3 调试中遇到的问题.....	27
七 最终成果展示.....	28
八、硬件课设心得与体会.....	29
九、参考文献.....	32
附录（一） 核心板电路图.....	33
附录（二） 整机电路图.....	33
附录（三） READSOME 使用说明书.....	34
附录（四） 所用器件清单.....	34

摘要

随着嵌入式技术的飞速革新,小型消费类电子产品在人们生活中起到了越来越重要的作用。其中,电子书阅读器是现代图书信息的载体,也是传统阅读方式的创新和未来发展的必然趋势。TI 在各种领域提供了相应的解决方案和器件。

此项目在运用 MSP430 及德州仪器先进的模拟半导体及通信器件的基础上自主设计深入研究制造出低成本低功耗电子书阅读系统,它能让您在移动中随时阅读您 SD 卡中的图书,也可以摇身一变成为电子相框或音频播放器。

关键词:msp430 , 电子书 ,sd卡 , fat , TFT显示

ABSTRACT

With the rapid innovation of embedded technology, small consumer electronics products has played an increasingly important role in people's lives. Among them, the e-book reader is the information carrier of modern books, but also the innovation of the traditional way of reading and the inevitable trend of future development. TI's solutions and devices in various fields. Independent design this project using the MSP430 and TI advanced analog semiconductors and communication devices based on in-depth study to create a low-cost low-power e-book reader system, which allows you to move at any time to read books on your SD card, can also be transformed into a digital photo frame or audio player.

Key Words:msp430 ,e-books , SD-card , fat, TFT display

一、实验任务

设计并实现多功能电子书阅读器，可从 SD 卡上读取文本文档并提供翻页等流畅的阅读体验，在此基础上进行图片显示、音乐播放等扩展功能。

1.1 基本要求

预期实现的基本功能如下：

- A.创建 SD 卡相应文件目录；
- B.能够选择并读取 SD 卡内 txt 格式文件并在 TFT 显示屏上正确显示汉字内容；
- C.用户能上下滚动文字内容，选择自动/手动翻页模式，返回文件目录；
- D.能够选择并读取 SD 卡内 bin 格式文件的图片并在 TFT 显示屏上合理显示；

1.2 扩展部分

- 1、提供 wav 格式音乐播放功能，并有音量增减/上下一曲的控制功能，可以快进快退，循环播放；
- 2、实现 TFT 的触摸屏操作代替传统按键操作；
- 3、显示阅读时间，提醒用户注意休息，保护视力；
- 4、完美的单词的处理，即一个单词不会因为换行而被拆开；
- 5、动态显示页码，当然，这是一般电子书应有的功能；
- 6、开机即进入系统时间设置，并在屏幕左下方显示；

1.3 产品特色

- 1. 全屏触摸，无需任何按键。
- 2. 友好的人机界面，无需开发者告诉你怎样使用，通过屏幕界面很容易理解。
- 3. 完美的单词换行处理，采用该行不能显示则采用换行处理的方式。
- 4. 价格便宜，开发成本的降低，是我们能赢得市场的关键。
- 5. 完美的环保封装，美白的外壳，还具有缠绕耳机线的地方。
- 6. 采用电池供电，完全脱离开发环境。

二、设计目标

基于 TI 的超低功耗 16 bit RISC 混合信号处理器的 MSP430™ 微控制器平台为各种低功耗和便携式应用提供了良好的解决方案。本项目主要研究如何利用 msp430 为控制核心,设计 SD 卡模块实现单片机从外围通用存储的信息读取,设计 TFT 彩色液晶显示模块合理显示文本图片等内容,另外提供基本按键实现用户对电子书基本操作,例如从 SD 卡上选择文件,读取文件,阅读时上下移动,暂停返回等基本功能。

除此之外,READSOME 电子书阅读器也将添加包括图片显示、MP3 音乐播放器和时钟计时等扩展部分功能,细化到诸如控制音量大小、返回上一菜单、显示播放时间、显示已使用时间并温馨提示等功能。

三、关于我们

3.1 团队成员

我们 READSOME 小组由三人组成,分别是:

陈明哲 通信工程 0909 (中英 0902) 班

万 晨 通信工程 0909 (中英 0902) 班

明 星 通信工程 0907 (中英 0901) 班

小组分工如下:

陈明哲: TFT, SD 卡, FAT32

万 晨: MP3 模块, 时钟

明 星: 电池, 外围包装, 答辩资料

3.2 READSOME 名字的由来

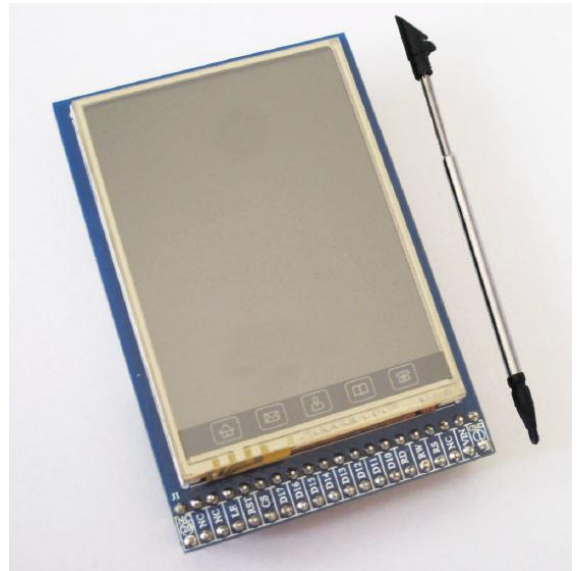
我们的灵感来自于 Draw Something (我画你猜)。这是近期一款非常火热的游戏,当然国内的火热完全由于国外的影响。它的火爆不可小觑:游戏刚刚发售 6 周,下载量就已经达到 3 千 5 百万,并且仍旧以每天 25 万下载量的程度继续增长。并且根据苹果 App Store 数据统计,Draw Something 是 84 个国家的“单词游戏”(Word Game)排行第一名。而

我们利用 MSP430 制作的电子书阅读器轻便且易携带，同时还兼具电子相册和 MP3 播放器的多重功能，必定也可以吸引很多的眼球。故取名为 ReadSome，以电子书功能为主，带来视觉和听觉的多重享受。

四、TFT 模块的设计与实现

4.1 功能简介

Thin Film Transistor (薄膜场效应晶体管)，是指液晶显示器上的每一液晶象素点都是由集成在其后的薄膜晶体管来驱动。从而可以做到高速度高亮度高对比度显示屏幕信息。目前在手机上 TFT 使用最为广泛，中高端彩屏手机中普遍采用的屏幕，分 65536 色及 26 万色，1600 万色三种，其显示效果非常出色。随着技术的进步，TFT 不仅应用在手机上，许多智能仪表，工控人机界面也都在使用 TFT 取代之前的黑白屏。



4.2 主要参数

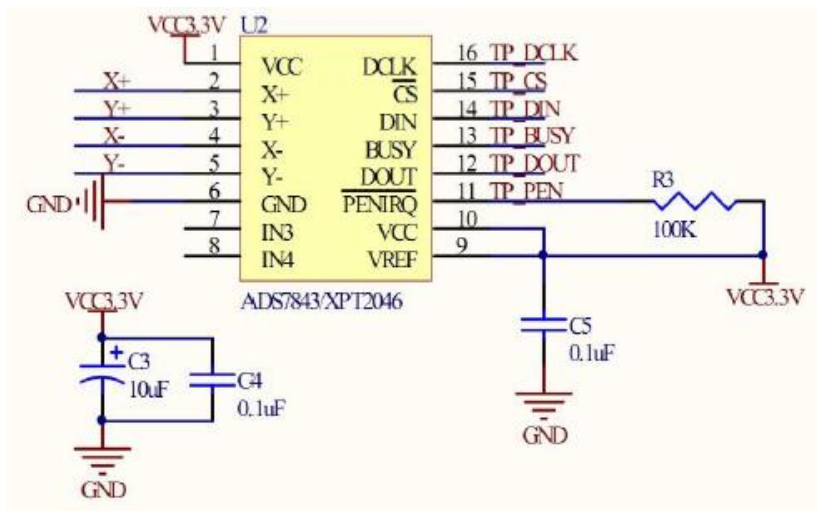
NBCTFT2.6_V2.2 主要有如下参数特点：

类目	参数
模块尺寸 (长x宽)	76mmx51mm
像素	320x240
颜色	26万色
驱动IC	ILI9320
触摸类型	电阻式
背光类型	LED

4.2.1 硬件介绍

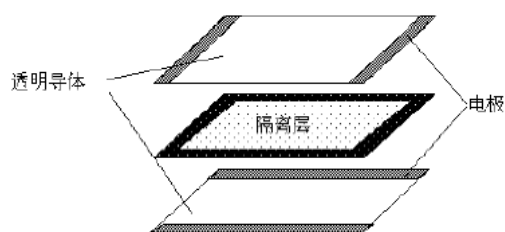
2.6 寸 TFT 模块电路共分为 FPC 接口电路、触摸控制电路、IO 扩展电路、电源电路、TFT 控制接口电路、SD 接口电路等 6 部分。

4.2.1.1 触摸控制电路



典型触摸屏的结构一般由三部分组成：两层透明的阻性导体层、两层导体之间的隔离层、电极。触摸屏工作时，上下导体层相当于电阻网络。当某一层电极加上电压时，会在该网络上形成电压梯度。如有外力使得上下两层在某一点接触，则在电极未加电压的另一层可以测得接触点处的电压，从而知道接触点处的坐标。比如，在顶层的电极(X+,X-)上加上驱动电压，则在顶层导体层上形成电压梯度，当有外力使得上下两层在某一点接触，在底层就可以测得接触点处的电压，再根据该电压与

电极(X+)之间的距离关系，知道该处的 X 坐标。然后，将驱动电压切换到底层电极 (Y+,Y-) 上，并在顶层测量接触点处的电压，从而知道 Y 坐标。计算触点的 X, Y 坐标具体分为如下两步： 1. 通过触摸屏控制芯片 ADS7843/XPT2046 分别采集触点在 X 轴和 Y 轴上产生的电压数字量，如下面公式中的 X 值和 Y 值。



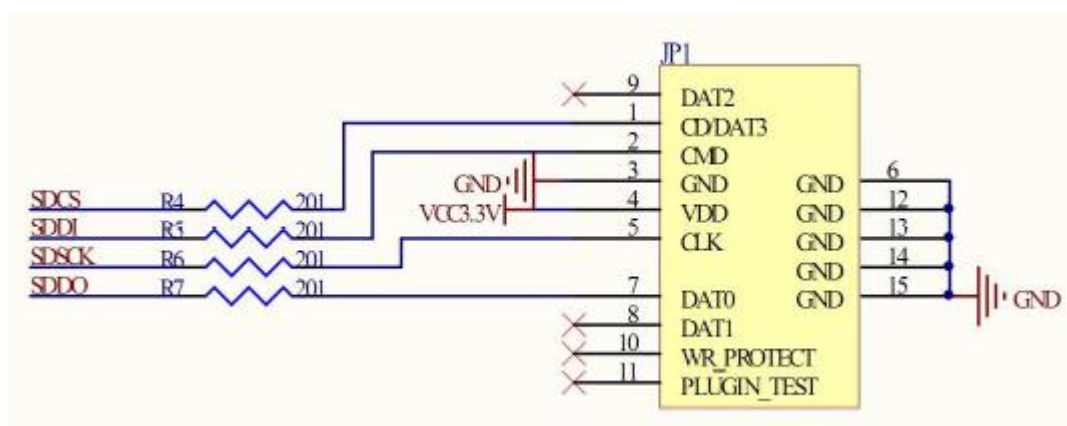
触摸屏结构

$$X_{LCD} = \frac{X - X_{min}}{X_{max} - X_{min}} \times W$$

$$Y_{LCD} = \frac{Y - Y_{min}}{Y_{max} - Y_{min}} \times H$$

4.2.1.2 SD 接口电路

SD 卡控制电路如下图所示，接口为模拟 SPI 接口，读写实现较为简单。



4.2.1.3 TFT 控制接口电路

图中

第 1 脚到第 20 脚是为兼容 12864 接口而设计：第 1 脚为电源地；

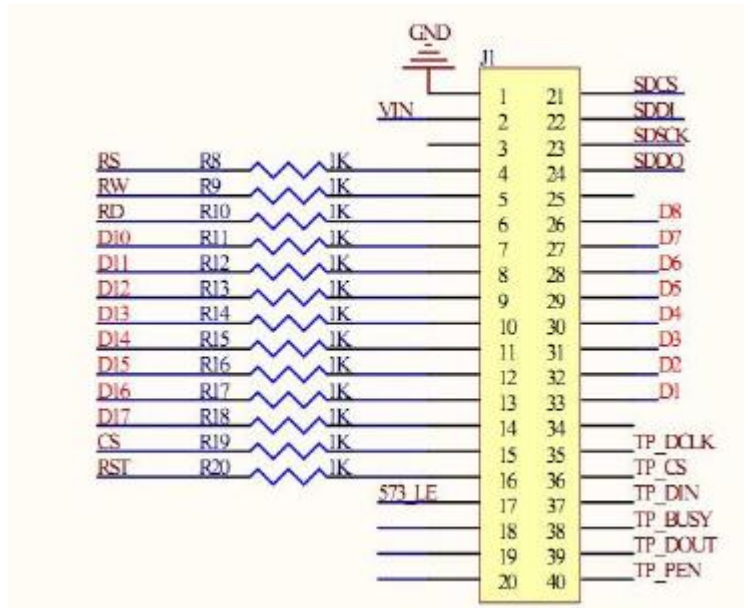
第 2 脚为电源输入；

第 4 脚到第 16 脚与控制器间增加了一个限流电阻，作用是在使用 5V 控制器驱动 TFT 时，保护 TFT 以免被过流损坏；第 17 脚为 IO 扩展控制位；

第 21 脚到第 24 脚为 SD 卡控制接口；

第 26 脚到第 33 脚为 TFT 数据口的低 8 位。

第 35 脚到第 40 脚为触摸控制接口；



4.2.2 软件介绍

4.2.2.1 TFT 初始化

TFT 初始化分为

1. Start Initial Sequence
2. Power On sequence
3. Adjust the Gamma Curve
4. Set GRAM area
5. Partial Display Control
6. Panel Control

详细参数设置参照附录的原代码

4.2.2.2、触摸控制

```

unsigned char Getpix() //取采样值，此处使用软件滤波，2次取样的值相差太大的视为噪声
{
    unsigned char flag=0;
    struct struct1 pix1;
    struct struct1 pix2;
    P1DIR2=0; //设置笔中断位为输入
    if ((P1IN&0x04)==0x00)
    {
        pix1=AD7843();
        pix2=AD7843();
        if(pix_filter(pix1,pix2)==1) //得到当前TP的采样值，此处使用
        {
            if((coordinate.x>Xmin)&&(coordinate.y>Ymin))
            {
                lx=240.0*(coordinate.x-Xmin)/(Xmax-Xmin); //坐标
                ly=320.0*(coordinate.y-Ymin)/(Ymax-Ymin); //Xmi.
                flag=1;
            }
        }
    }
    return flag;
}

```

4.2.2.3、SD 卡初始化

SD 卡初始化条件：

1. 频率在 400KHZ 以下
2. 电压在 3.3V
3. 上拉电阻为 1K 欧姆
4. SD 卡为一代卡

初始化代码：

```

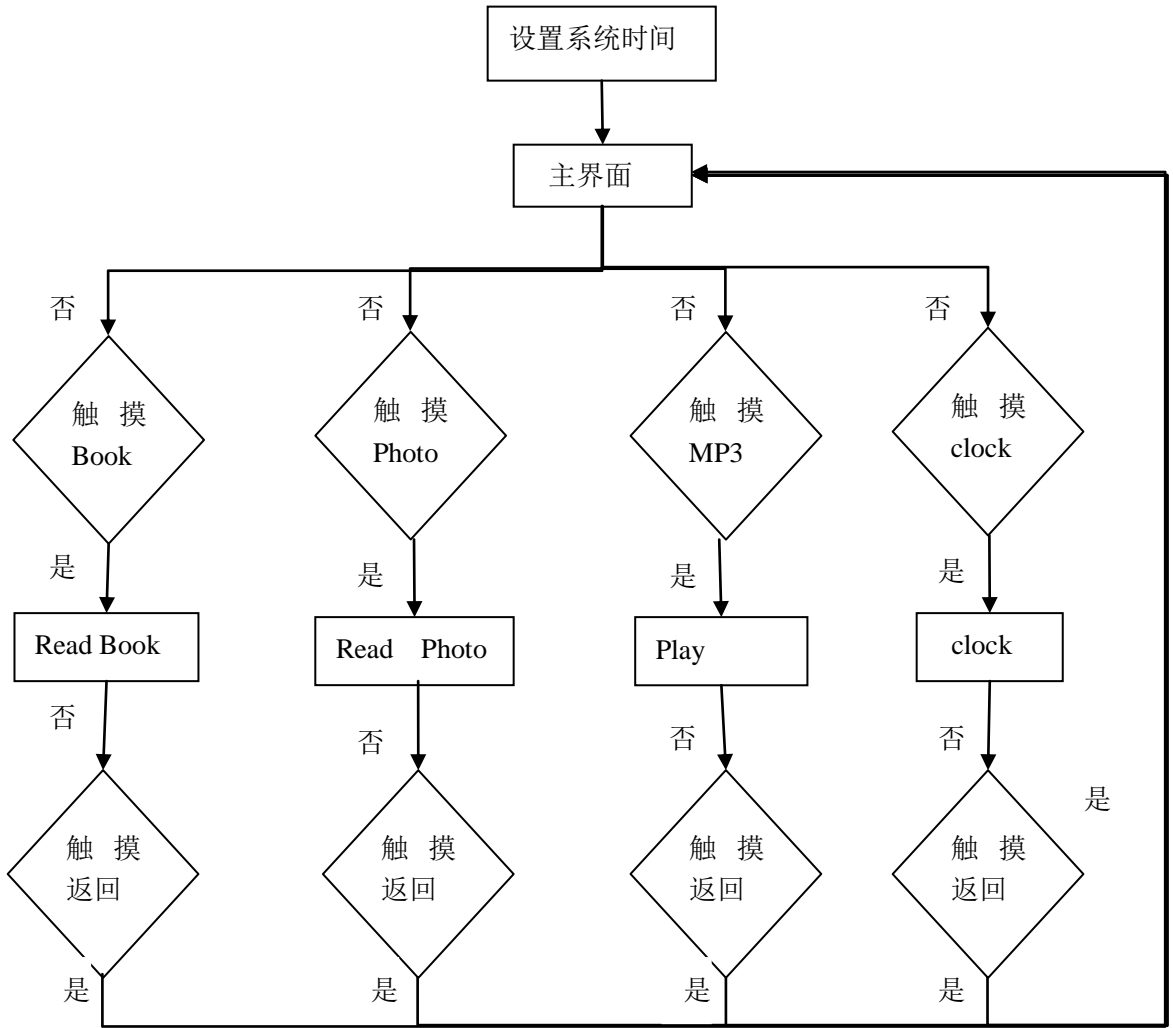
unsigned char SD_Init() //初始化, 使用CMD1 (1号命令)
{
    unsigned char time,temp;
    unsigned char pcmd[] = {0x41,0x00,0x00,0x00,0x00,0xff};
    SD_CSL;
    time=0;
    do
    {
        temp=SD_Write_Cmd(pcmd);
        time++;
        if(time==TRY_TIME)
        {
            return(INIT_CMD1_ERROR); //CMD1写入失败
        }
    } while(temp!=0);
    //is_init=1; //初始化完毕, 将is_init设置为0, 为了提高以后
    SD_CSH; //关装SD卡的片选
    SD_spi_write(0xff); //按照SD卡的操作时序在这里补8个时钟
    BCSCTL2 &=~DIVM_2;
    return(0); //返回0, 说明初始化操作成功
}

```

4.3 界面设计流程图

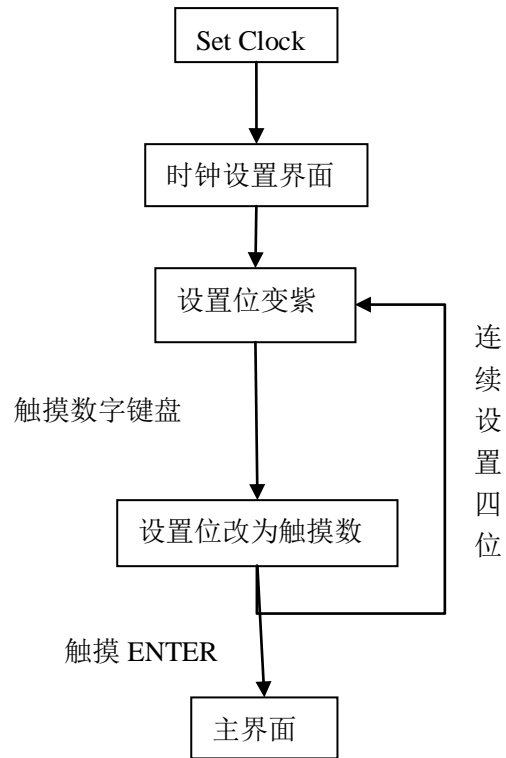
软件分为四个功能: Read Book ,Read photo, Play MP3 ,Show clock

其流程图如下图所示:

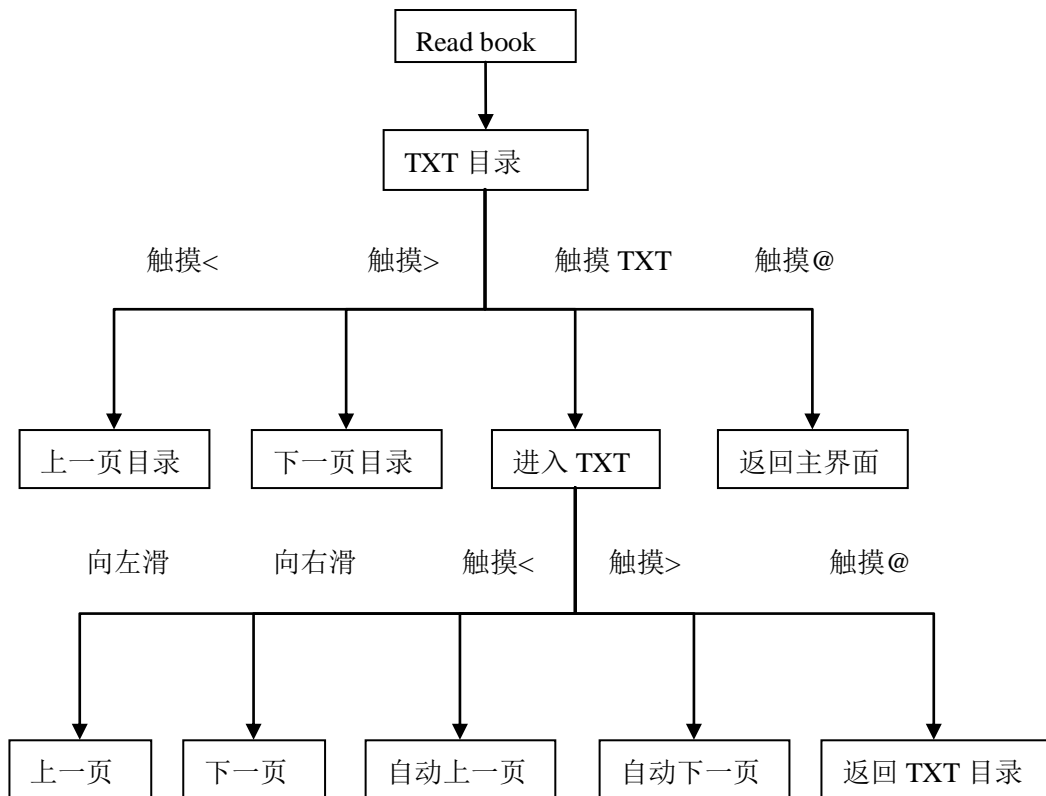


可以看到，整个功能模块均由界面和触摸来完成，现在我们分别讨论各模块的实现原理。

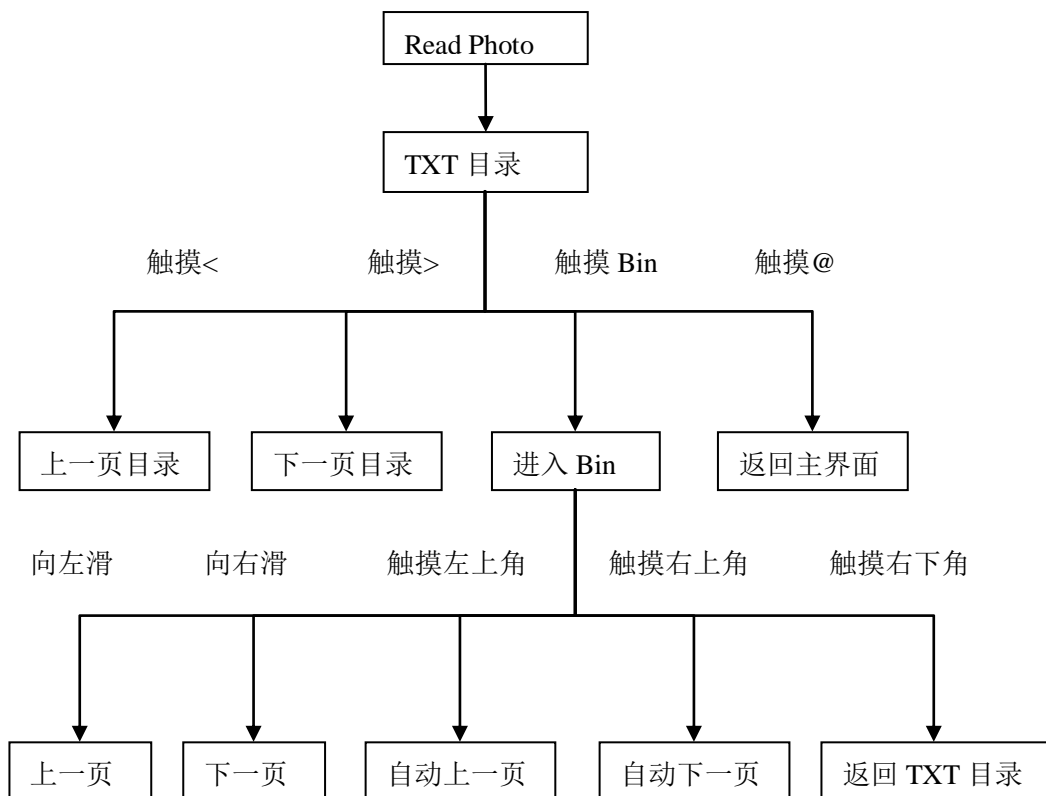
4.3.1、Set clock



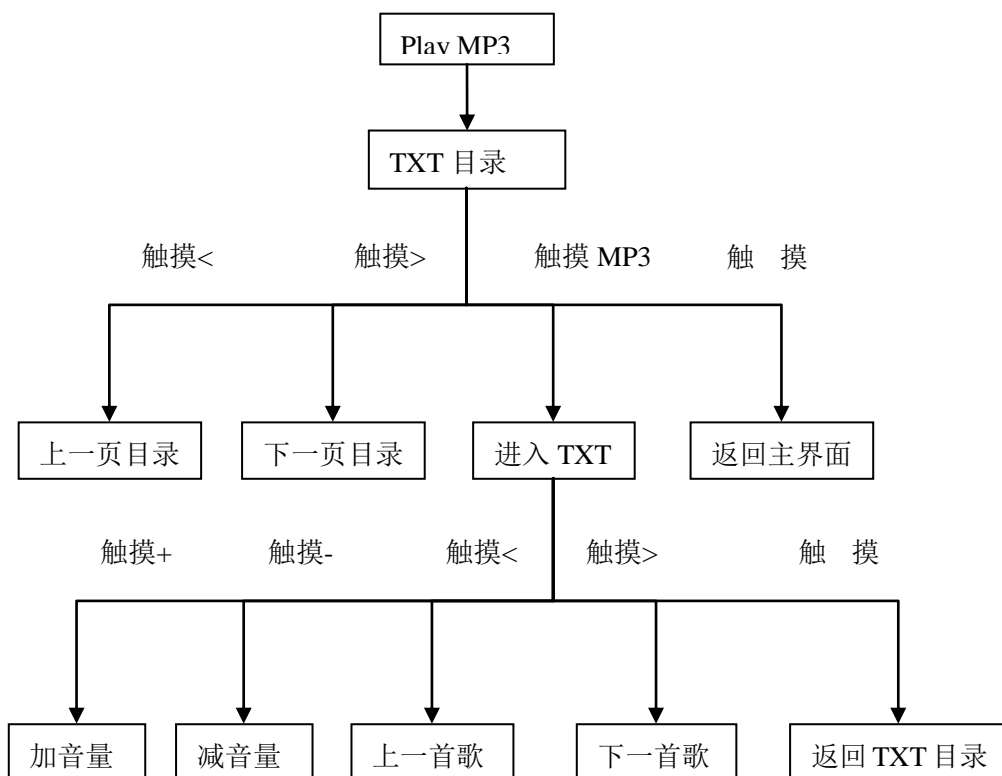
4.3.2、Read Book



4.3.3、Read Photo



4.3.4、Play MP3



4.4 遇到的问题

4.4.1、SD 卡初始化

网上代码很多，但真正要把它用起来是非常难的，对于时钟的限制是初始化的关键，而在代码中采用延时是非常不可取的，因为无法精确到时钟在 400HZ 以下，最后还是采取改变时钟频率的方法，将频率降下来，初始化后将频率升高到 8MHZ，实施快速的数据传输。

另外，对于 SD 卡的版本也是不一样的，8G 以上的卡是第二版的卡，而目前网上的代码基本上都是第一版本的，因此，必须要用 2G 或 2G 以下的卡才能正常驱动。

4.4.2、TFT 文字显示

问题：翻页怎么来控制每次向前或向后翻页后，其字符都是一模一样的？

解决方案：在添加了英文末尾字符的处理后，不存在一个确定的字符数显示在屏幕上，这样就会导致翻页无法向前或向后来回都一模一样，若采用标记位，当连续翻页后，标记则会无法继续进行，最后想到的办法就是固定每页的显示字符数，且每页的字符数正好利于翻页，为此最后决定采用 256 个字节一页，即扇

区的一半，这样哪怕再多次的翻页也很容易的解决记录位置的问题，从而成功解决了显示字符的问题。

4.4.3、FAT 文件系统

问题：如何从 51 直接移植现有的 FAT132 系统？

解决方案：选好可行的代码是关键，我们组采用的是振南 FAT 系统，它是唯一一个有示例，有完整功能说明的代码，根据它给的例子不难调出最后的结果。但是，51 和 MSP430 在变量方面稍微有些不同，导致给我们的移植带来了不少困难，在把变量修改后基本可以进行使用。

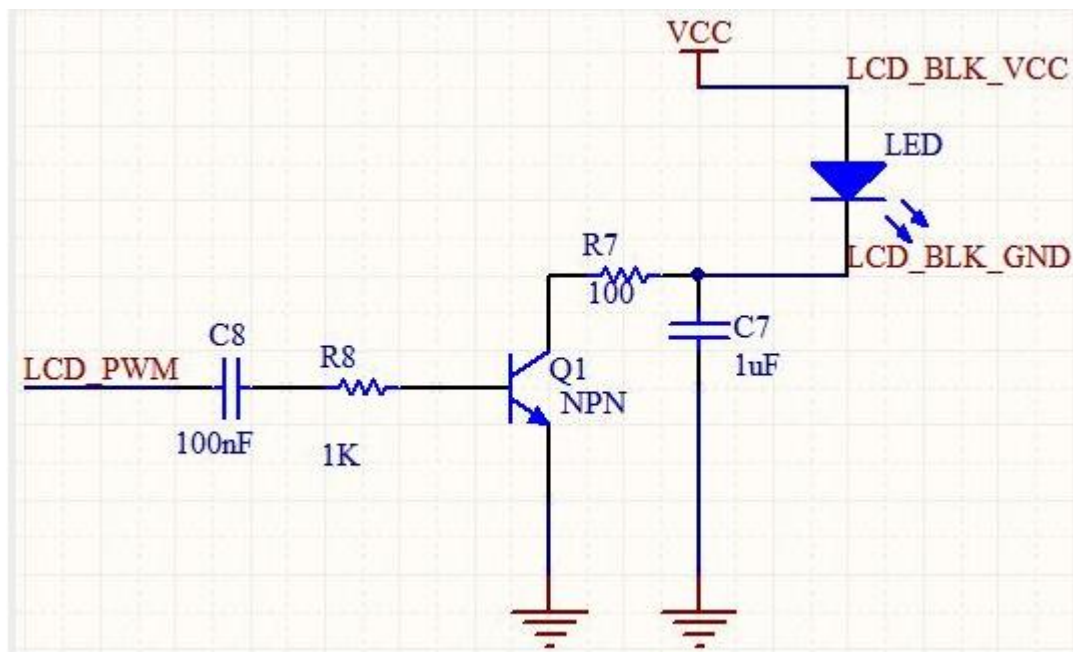
因此，在以后设计时，代码的变量尽量统一化，方便以后在不同平台上进行移植，也有利于大工程的开发。

4.4.4、背光亮度调节

问题：如何利用 PWM 放大进行背光调节？

解决方案 1：利用非非门放大电路，虽然增加了驱动能力，但是其电路并没有增大，导致此方法完全不可行。

解决方案 2：利用三极管进行放大，电路图如下：



此方法可以进行背光的亮度调节，但是此方法难以用于 TFT 调节。

原因如下：

1. PWM 调节方式是对电压的大小进行调节，而 TFT 不仅是亮度的显示，还有

触控控制，触控控制模块需要的却是稳定的电压，对于集成好的 TFT 而言，电压的改变即意味着触控电路是否能正常工作，若要保证触控电路进行正常工作，则无法使 TFT 有亮度的明显差异。

2. 我们购买的 TFT 不仅包含有触控电路，还包含了 SD 卡，而 SD 卡的初始化更是依赖于 3.3V 的电压，对于电压的改变，将使 SD 卡无法正常的驱动，而无法正常驱动即意味着整个电子阅读的瘫痪，因此这也是以 PWM 调节亮度的不可实现的一大重要原因。

4.4.5、RAM 不够怎么办？

现象：对于一个搭载了 FAT32 系统，音频播放，电子阅读，电子相册，电子时钟的综合功能，2k 的 RAM 是完全不够用的，而 RAM 的不够将直接抑制了程序的开发，因此我们必须要跟换 RAM.

解决方案：

1. 更换 RAM（430 无外设 IO，此方法在 430 不可用）
2. 外加 EEPROM

此方法即外加一个读写存储器，但在有限的时间内研究其读写是非常划不来的，而且影响了外设的美观。

3. 更换主芯片（我们采用的方法）

F149 有 60K 的 ROM 以及 2K 的 RAM，48 个引脚，而 1611 有 48KROM 和 10KRAM，引脚却是与 F149 一模一样，整个换芯片的过程就是将主芯片拆下，再将 1611 焊上，即可完全进行代码调试，连外设都不需要移动，这是最快也是最省事的方法。

4. 采用 SD 卡读写，此方法没进行测试，其速度应该会大大减慢。

五、MP3 模块的设计与实现

5.1 硬件设计与选型

为了能够播放 MP3 歌曲，我们决定采用硬件来直接解码 MP3 歌曲，msp430 主控芯片用来进行对硬件的控制和文件数据流的传送。目前，单片机使用的 MP3 解码方案有很多，其中 vs1003 体积小，功能全，而且价格上都具有很大的优势，使用起来也较为简单，于是

我们选择使用它作为解码芯片。经过寻找，我们购入一块带有 vs1003 芯片的解码模块。

5.1.1、VS1003B 芯片介绍

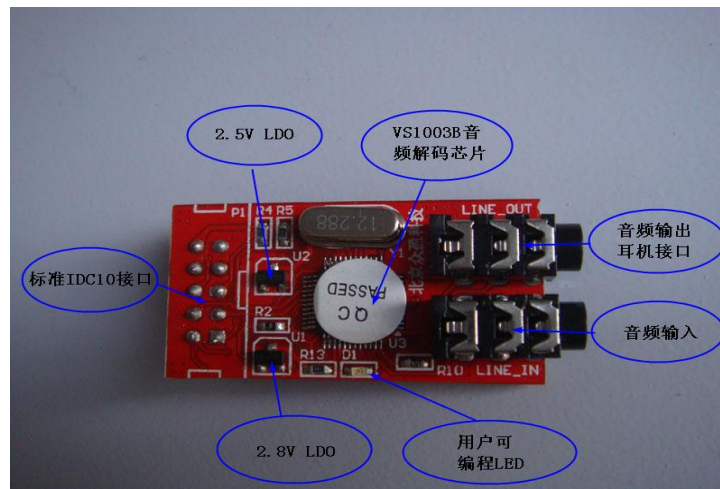
VS1003 语音解码芯片由芬兰公司 VLSI 生产，是目前解码效果甚佳的 DSP 核的解码器，LQFP48 小型封装，支持解码音乐格式包括 MP3、WMA、WAV、MIDI、P-MIDI，编码格式 IMA ADPCM（单声道），音质属于中高端。

5.1.2、VS1003 音频模块特性

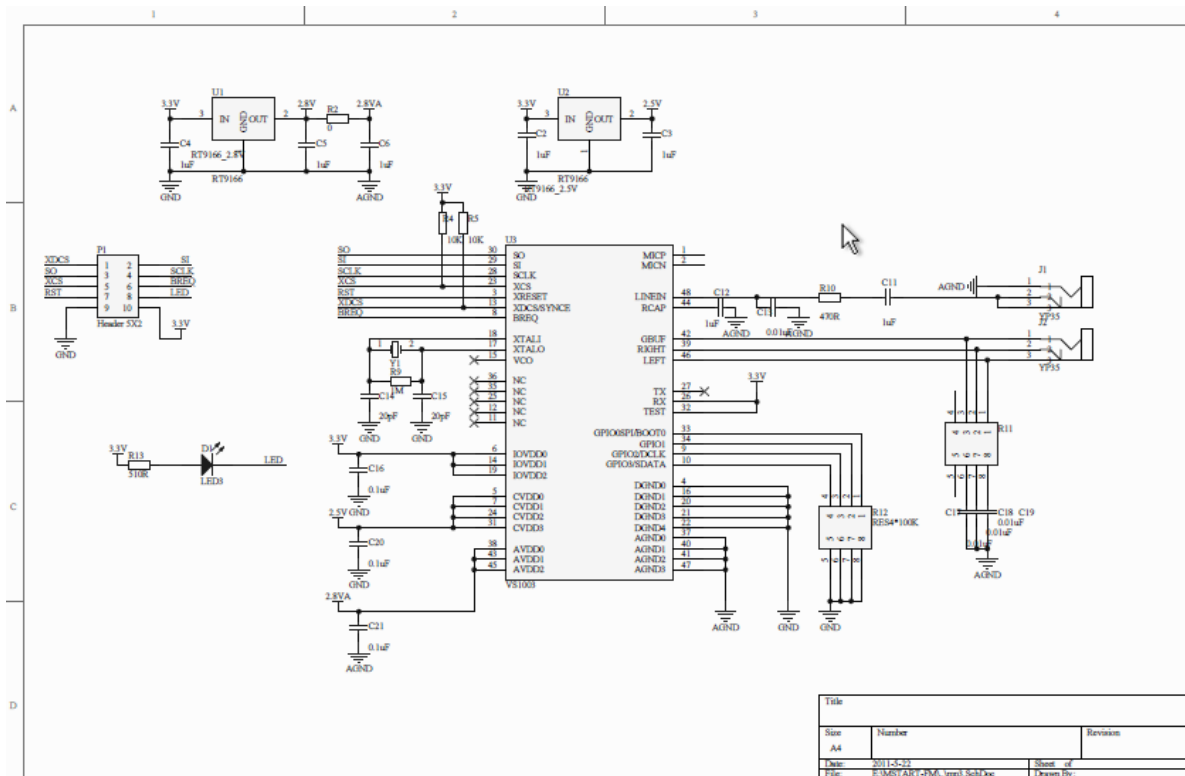
VS1003 音频解码格式：MP3，MP3+V，WMA，WAV，MIDI，SP-MIDI；录音编码格式：IMA ADPCM（单声道），线入（Line input）输入方式；支持 MP3 和 WAV 流；低功耗；具有内部锁相环时钟倍频器；高质量的立体声数模转换器（DAC）；16 位可调片内模数转换器（ADC）；高质量的立体声耳塞驱动（30 欧）；单独的模拟、数字和 IO 供电电源；串行的数据和控制接口（SPI）；

- SPI 接口，控制信号线均引出
- 1 个耳机和音响输出接口
- 1 个 line_in 输入接口
- 板内 LDO 提供 2.8V 和 2.5V 电源
- 模块电源供电电压: 3.3V

5.1.3、购买的模块实物展示图



5.1.4、购买的模块电路图



5.1.5、 控制方式

主控芯片 msp430 与 vs1003 芯片模块采用 spi 总线方式连接。

5.1.5.1 Msp430 SPI 总线

SPI (Serial Peripheral Interface) 串行外围模块接口是 Motorola 首先在其 MC68HCXX 系列处理器上定义的，它是一种同步的高速串行通信协议。它可以使单片机与外围模块之间进行数据交换，比 EEPROM,Flash,实时时钟，AD 转换，数字信号处理器和数字信号解码器之间交换数据，同时应用比较多的是用来作为按键等外设的扩展。SPI 总线系统是一种同步串行外设接口；是一种高速的，全双工，同步的通信总线，并且在芯片的管脚上只占用四根线，节约了芯片的管脚，同时为 PCB 的布局上节省空间，提供方便，正是出于这种简单易用的特性，现在越来越多的芯片集成了这种通信协议。SPI 有主从两种工作方式，可以工作在 3 线或者 4 线模式下。

MSP430 单片机特点：

- (1) SPI 模式支持 3 线和 4 线模式；
- (2) 支持主机与从机模式；

- (3) 接受和发送有各自独立的发送移位寄存器和缓冲器；
- (4) 接受和发送都有独立的中断能力；
- (5) 移位时钟的极性和相位可编程；
- (6) 字符长度可以是 7 位或者 8 位。

当 Msp430 USART 模块控制器 UxCTL 的位 SYNC 置位时，USART 模块工作于同步模式，对于 149 即工作于 SPI 模式，若是 169，USART0 可以支持 I2C，可以通过另一控制位 I2C 控制，I2C 位 0 则工作于 SPI。在 SPI 模式下，允许单片机以确定的速率发送和接收 7 位或 8 位数据。

同步通信与异步通信类似；同步通信和异步通信寄存器资源一致，具体寄存器的不同位之间的功能存在差异；具体寄存器内容参见 TI 提供的用户指南。

USART 模块的 SPI 操作可以是 3 线和 4 线，其信号如下：

SIMO：从进主出，主机模式下，数据输出；从机模式下，数据输入。

SOMI：从出主进，主机模式下，数据输入；从机模式下，数据输出。

UCLK：USART SPI 模式时钟，信号有主机输出，从机输入。CLK 时钟只能由主机提供。

STE：从机模式发送接收允许控制脚，用于 4 线模式，控制多主从系统中多个从机，避免发生冲突。

SPI 的寄存器

Register	Short Form	Register Type	Address	Initial State
USART control register	U0CTL	Read/write	070h	001h with PUC
Transmit control register	U0TCTL	Read/write	071h	001h with PUC
Receive control register	U0RCTL	Read/write	072h	000h with PUC
Modulation control register	U0MCTL	Read/write	073h	Unchanged
Baud rate control register 0	U0BR0	Read/write	074h	Unchanged
Baud rate control register 1	U0BR1	Read/write	075h	Unchanged
Receive buffer register	U0RXBUF	Read	076h	Unchanged
Transmit buffer register	U0TXBUF	Read/write	077h	Unchanged
SFR module enable register 1†	ME1	Read/write	004h	000h with PUC
SFR interrupt enable register 1†	IE1	Read/write	000h	000h with PUC
SFR interrupt flag register 1†	IFG1	Read/write	002h	082h with PUC

† Does not apply to MSP430x12xx devices. Refer to the register definitions for registers and bit positions for these devices.

Table 14-2. USART1 Control and Status Registers

Register	Short Form	Register Type	Address	Initial State
USART control register	U1CTL	Read/write	078h	001h with PUC
Transmit control register	U1TCTL	Read/write	079h	001h with PUC
Receive control register	U1RCTL	Read/write	07Ah	000h with PUC
Modulation control register	U1MCTL	Read/write	07Bh	Unchanged
Baud rate control register 0	U1BR0	Read/write	07Ch	Unchanged
Baud rate control register 1	U1BR1	Read/write	07Dh	Unchanged
Receive buffer register	U1RXBUF	Read	07Eh	Unchanged
Transmit buffer register	U1TXBUF	Read/write	07Fh	Unchanged
SFR module enable register 2	ME2	Read/write	005h	000h with PUC
SFR interrupt enable register 2	IE2	Read/write	001h	000h with PUC
SFR interrupt flag register 2	IFG2	Read/write	003h	020h with PUC

SPI 的初始化及其复位(和 UART 公用一套寄存器)

在初始化或者重新配置 USART 的 SPI 时，必须按照以下顺序进行：

- (1) UxCTL 寄存器的第 0 位 SWRST 置位；
- (2) 在 SWRST 置位的条件下，初始化所有的 SPI 寄存器，包括 UxCTL 寄存器；
- (3) 通过置位模块使能寄存器 MEx 的 URXEx 和 UTXEx 位使能 SPI 的接受和发送使能模块；
- (4) 通过软件复位 UxCTL 寄存器的第 0 位 SWRST；
- (5) 通过中断使能寄存器 IEx 的 URXIEx 和 UTXIEx 来使能发送和接受中断。

5.1.5.2 vs1003 SPI 模式

SPI 总线，被应用于 VS1003 的串行数据接口 SDI 和串行控制接口 SCI。
SDI管脚SCI管脚描述

XDCS /XCS 低电平有效片选输入，高电平强制使串行接口进入standby模式，结束当前操作。高电平也强制使串行输出SO 变成高阻态。如果SM_SDISHARE 为 1，不使用XDCS，但是此信号在XCS中产生。

SCK 串行时钟输入。串行时钟也使用内部的寄存器接口主时钟。SCK 可以被门控或是连

续的。对任一情况，在XCS变为低电平后，SCK 上的第一个上升沿标志着第一位数

据被写入。

SI 串行输入，如果片选有效，SI就在SCK 的上升沿处采样。

SO 串行输出，在读操作时，数据在 SCK 的下降沿处从此脚移出，在写操作时为高阻态。

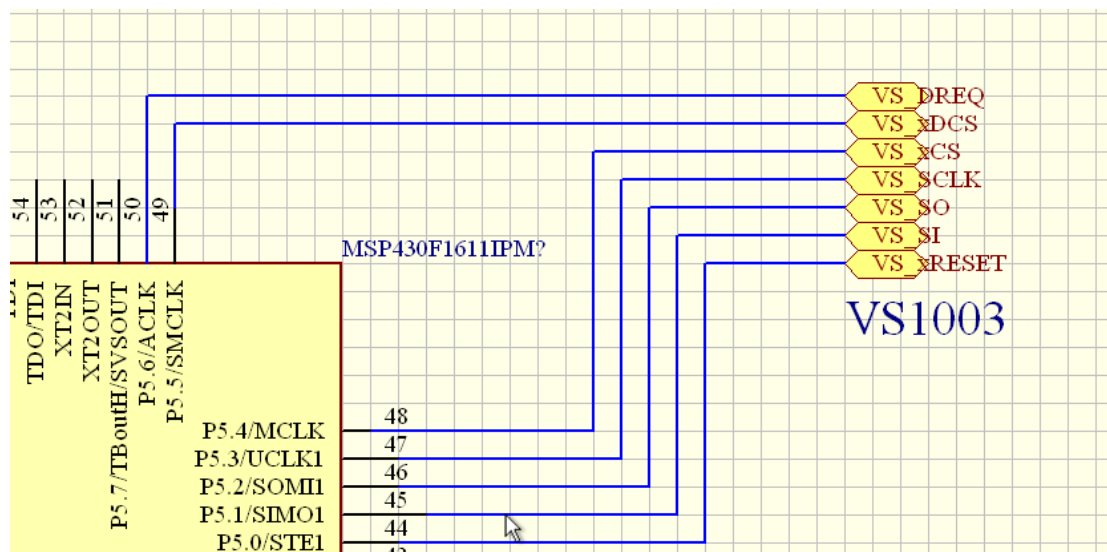
5.2 软件设计

5.2.1、Vs1003 驱动程序

驱动程序需要完成对 msp430 io 管脚的接口设置，vs1003 芯片的初始化操作，向芯片写数据，写控制寄存器。

使用 msp430f1611 的 UART1 模块，接口配置如下：

P5.4=xCS, P5.5=xDCS, P5.3=SCLK, P5.2=SO, P5.1=SI, P5.6=DREQ , P5.0=xRESET



5.2.2、写控制字—vs1003 的正弦测试模式

VS1003 的 SPI 总线用来传送 MP3 数据和控制命令。当要传送 MP3 数据时 xDCS 须置为低电平，而 xCS 置 1。此时 SPI 总线称作 SDI（串行数据接口）。VS1003 拥有以下几种测试模式：存储器测试、SCI 总线测试和正弦测试。所有的这些测试都有相同的步骤：硬件复位，置位模式寄存器 SPI_MODE 的 bit5: SM_TESTS，发

送测试命令到 SDI 总线上。测试命令总共包含 8 个字节的数据，前 4 个字节为命令代码，后四个字节为 0。正弦测试步骤如下：

1) 进入 VS1003 的测试模式：SPI_MODE = 0x0820

2) 选择写控制字模式，向 VS1003 发送正弦测试命令：0x53 0xEF 0x6E 0x30
0x00 0x00 0x00 0x00。

3) 延时 500ms 后，退出正弦测试，发送命令：0x45 0x78 0x69 0x74 0x00 0x00
0x00 0x00

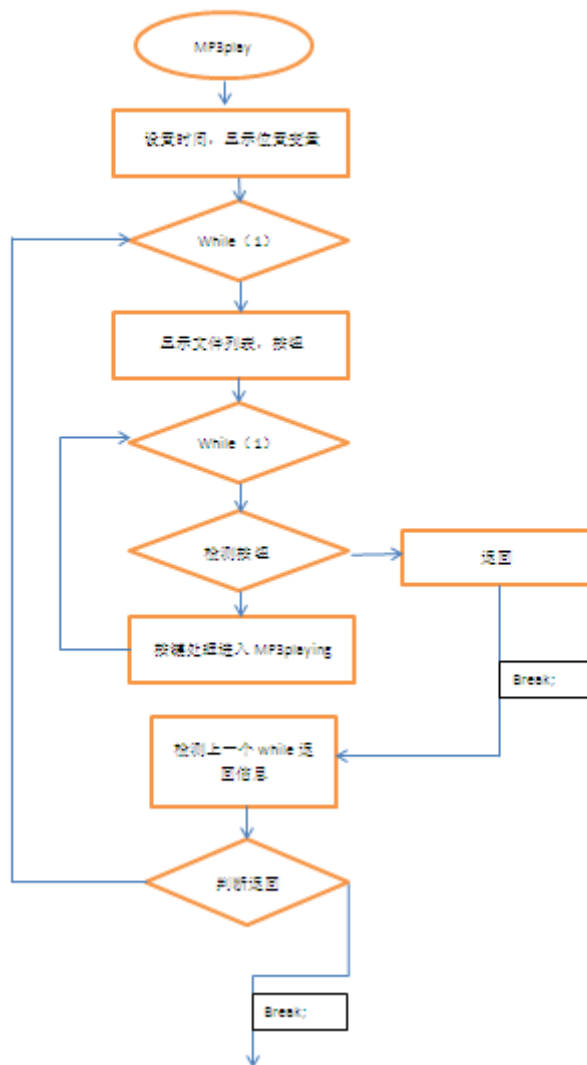
5.2.3、写数据—基本的 MP3 播放

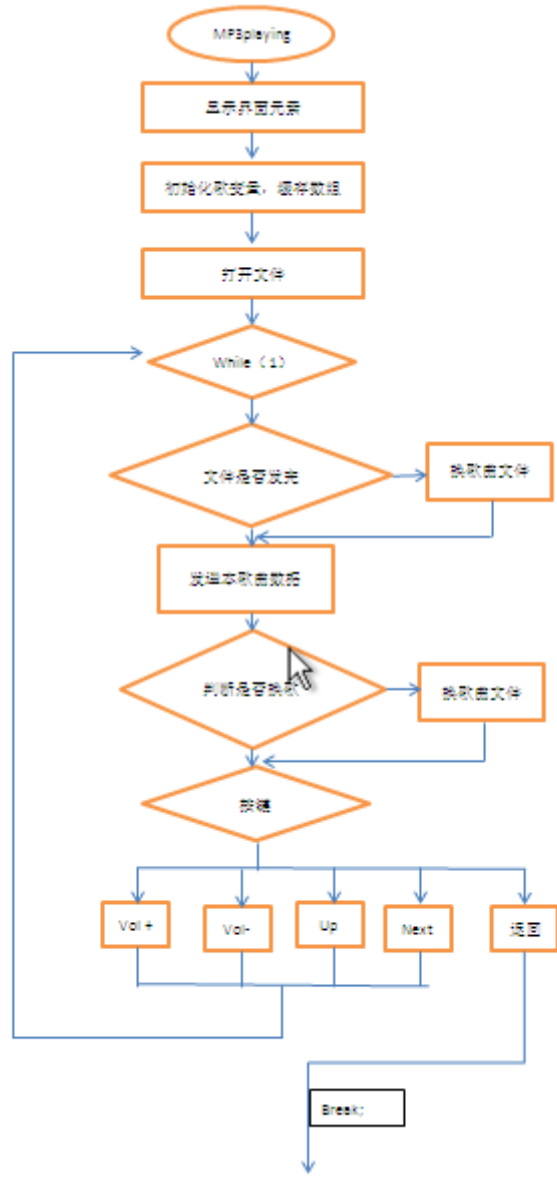
整个数据流是单片机内设置一个缓冲数组，大小为 1024 字节，每次从 sd 卡制定扇区位置开始读取 1024 字节，再向 vs1003 发送数据，经试验，此缓冲数组大小与音乐播放流畅程度有关系。若此数组设置得较小，音乐播放中会出现卡顿杂音现象，影响音乐的正常流畅播放，1024 字节能够保证音乐的流畅性。

5.2.4、文件系统的引入与进阶 MP3 播放文件的控制

文件播放控制原理是，设置当前播放标志和歌曲标志，当有按键按下下一曲换歌时，歌曲标志加 1 后两者值不同，每次发送一段数据就判断这两个值，不同就立刻换下首歌地址，并且将两者值该为相等。上一曲原理相同。文件系统引入后，就不用事先计算好物理扇区地址，只用按照文件名称打开相应 MP3 文件，在换歌后或者播放完后关闭此文件并打开下一文件。

软件流程图如下：





5.2.5、音量控制

音量控制实际上是向 vs1003 音量控制寄存器写值。方法是播放过程中要控制音量时关闭 spi 数据写，打开 spi 控制写，向音量控制寄存器写对应数值。值范围为 0~255，每增 1 衰减 0.5db，可以左右声道分开控制，此项目中两声道值相同。

5.2.6、界面设计和功能整合

界面上，进入 MP3 模块后能够显示 sd 卡中歌曲目录界面，点击对应歌曲后进入播放界面。播放界面有音量增减，上一曲下一曲，返回按钮。上方加入正在

播放的歌曲名和正在播放的时间（见时间模块）。左下方还加入系统时间显示（见时间模块）。

六、时钟模块的设计与实现

6.1 硬件模块

硬件使用 msp430 中的 TIMER_A 定时器。用 8MHz 主频时钟，可以利用定时器计数生成 0.5 秒中断，然后利用中断服务程序完成对时间和时间显示刷新的计数。

相关寄存器如下：对应要设置时钟源，时钟分频，计数模式

Table 11-3. Timer_A Registers

Register	Short Form	Register Type	Address	Initial State
Timer_A control	TACTL	Read/write	0160h	Reset with POR
Timer_A counter	TAR	Read/write	0170h	Reset with POR
Timer_A capture/compare control 0	TACCTL0	Read/write	0162h	Reset with POR
Timer_A capture/compare 0	TACCR0	Read/write	0172h	Reset with POR
Timer_A capture/compare control 1	TACCTL1	Read/write	0164h	Reset with POR
Timer_A capture/compare 1	TACCR1	Read/write	0174h	Reset with POR
Timer_A capture/compare control 2	TACCTL2	Read/write	0166h	Reset with POR
Timer_A capture/compare 2	TACCR2	Read/write	0176h	Reset with POR
Timer_A interrupt vector	TAIV	Read only	012Eh	Reset with POR

6.2 软件设计

因为采用了 led 灯闪烁，所以采用每 0.5 秒中断，每次进入中断 led 状态翻转。每偶数次进入中断则秒计数加一。并且在中断中完成对时间显示的刷新，设置刷新控制 showmode 来对不同界面的时钟进行刷新。

播放时钟控制采用另外一套计时变量，在进入歌曲或者切换到另一首歌曲是从 0 开始计秒，时间显示也在中断中刷新。

基本设置：

CCTL0 = CCIE; 打开定时器中断

CCR0 = 2047; 0.5 秒计时变量

TACTL = TASSEL_1 + ID_3 + MC_1; 选择 1 时钟源，选择 8 分频，计数方式是增至 CCR0。

时间计数程序：

时分秒的计数 timeradd () ；

播放时间计数 ptimeradd () ；

时分秒都是两个字符型变量，便于 lcd 中相关函数输出。

中断服务程序：

根据 showmode 值来刷新时间显示。刷新方式是先要在要改变位置用空格清空，在刷新成要改变的量，为了避免时分秒同时刷新时闪烁现象，决定只在要改变的位刷新，刷新控制 change 值即改变位数。

6.3 调试中遇到的问题

6.3.1、调试方式

因为我们缺少例如 JTAG 调试的工具，为了检测和调试程序，我们最开始决定用串口输出信息。在串口程序上，我们搜集也修改了很多代码，但始终的不到想要的稳定输出程序信息的效果。串口经常不稳定，无法输出或者输出不完整。好在开发板上提供了四个 led 灯，在程序不同地方加入 led 控制，就可以通过 led 的不同组合初步调试程序状态，在 lcd 屏幕模块加入后，也可以通过屏幕输出来调试。

6.3.2、IAR 文件管理

IAR 文件管理方式有些特殊，将工程复制后存在其他路径并不能成为真正的副本，在修改当前文件时，由复制所得的其他路径下文件也同时被修改。所以要注意副本的备份，可以压缩成压缩文档存副本。

6.3.3、Sd 卡容量，代数

Sd 卡分为一代和二代，市面上大容量卡（4G 及以上）都是二代卡，而我们的单片机是不支持二代卡的，所有操作和相关驱动是基于一代 sd 存储卡而编写。

6.3.4、按键中断\查询

刚开始想用物理按键查询方式控制，但是发现按键中断会受很多因素影响，例如复用的引脚需要特殊处理，要用延时控制按键灵敏度，并且中断程序不能满足我们对少量按键在不同界面不同功能的要求。所以程序全部采用查询方式检测

触摸屏上虚拟按键。

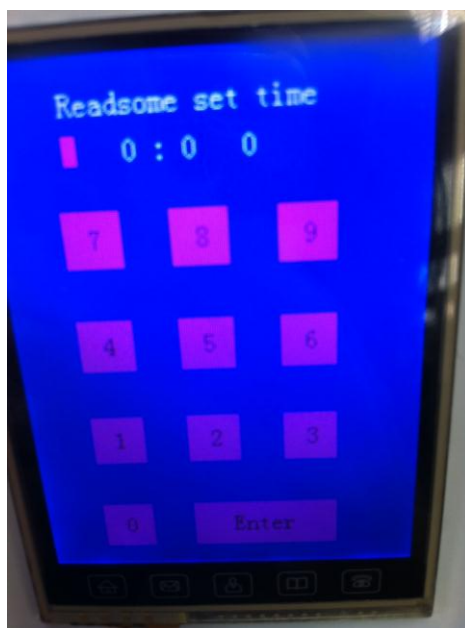
6.3.5、文件系统

没有文件系统时，能够播放歌曲，前提是知道歌曲起始扇区，这样是十分原始的，而且不能判断歌曲何时结尾。在移植了文件系统后，我们能够直接按文件名称来进行操作，切换歌曲，显示列表也依赖文件系统的支持。

七 最终成果展示



主界面图



设置时间界面

保持供电情况下，拨动开关，会出现如上右图所示的设置系统时钟的图样。输入当前的时间，点触“Enter”方块即可进入系统主界面，如上图所示。不同颜色的四个区域即是我们所实现的四大主要功能，包括英文电子书阅读、电子相册以及 MP3 播放器和时钟显示功能。



封装完成效果图

如果您对其他具体操作感兴趣的话，详情可参见[附录（三）READSOME 使用说明书](#)

八、硬件课设心得与体会

硬件课设个人总结

——陈明哲

本次硬件课设从第八周开始一直到期末考试后，经历接近两个多月的调试与开发，与软件课设相比，硬件的不确知性要大得多，而代码无法运行的理由有什么也很离谱，一样的代码，在他那就可以运行，而放到我的上面就不行了，很简单的东西却要折腾很久，串口通信就是一个典型的例子，为了调试 SD 卡，我们必须先把串口通信弄好，而串口通信看似就是那么几条语句，却整整搞了我们一个星期，几乎把它的初始化都完完全全弄懂了，最终才弄清楚原因，调试运行成功。

我们开始都是抱着一种在网上找代码，然后烧进去看能不能用，而不注重原理，随着课设的深入，发现与其盲调，不如先对其进行原理的分析，否则调到后面根本不知道为什么就是不出来，而别人却告诉你这是可以的，这时候就需要对其进原理性的分析，与其开始浪费时间不如先弄懂原理，这样效率会更高。

其次，对于代码的获得方法有很多种，除了从淘宝买器件得到的代码外，我

们自己觉得比较好的就是从淘宝的商家评论里面找买家，然后通过好心的买家来获得代码，而我们有的代码就是通过这种方式得到的，也许不是很好用，但对整个课设还是起到了一定的帮助作用。

另外，对于 FAT 系统，我们还专门加了振南的 QQ，请他辅助我们完成 51FAT32 系统到 MSP430 的移植，这也是我们实质性的飞跃。

最后，一个团队分工真的很重要，在我们比赛中，有的组甚至就一个人专门在做，而其他人在旁边打酱油，不仅整个课设的进度放慢，更重要的是失去了团队合作的意义，当今社会，一个人的模式已经无法适应开发的需要，因此，培养团队能力是非常关键和必要的。

硬件课设个人总结

——万 晨

项目任务的分解与整合

在入手本题目后，我们小组共同商讨完成了整个系统的初步分析，并且拟定了初步的方案及器件选型，然后，我们分工开始入手各自的模块，更深入地查找资料联系商家。确定方案后，我们小组开始了各个模块地独立编写和调试。在独立调试地过程中，我们仍然商讨这一些共同接口及整个系统最终地操作模式。在完成各自模块后，需要进行系统联调，此时需要统一代码的编写风格和原理。例如在界面显示这块，音乐时钟部分的退出，菜单选择功能的代码与电子书模块的代码风格进行了统一。整合中还会出现头文件不统一，变量设置冲突等等问题，这需要小组成员及时沟通，共同制定出整合方案。

科学的工作方式

进度安排采用工作周报和后期的每日站立会议模式。工作周报刚开始是对我们的很大的压力，因为我们的进度需要及时上报，给我们一种很紧张的感觉。但是，工作周报对整个项目的积极促进是很明显的。因为如果想要提高就必须做好每天的工作记录，并且将工作的结果分为成功的和失败的两种，失败的那部分必须做分析，搞清楚究竟是自己的哪些方式有问题，导致了失败的结果。如果自己分析不清楚还可以请教身边的同学老师。对于成功的部分，要看是不是每一次采用的方式相同，如果同样的事情，每一次采用的方式都相同，说明方法是正确的，基本上就可以固化下来，这就是我们通常所说的经验。而且我们及时的反馈当前

状态和问题，还可以对整个进度有着较为清晰的计划。我们在工作周报制度中慢慢改掉拖延的毛病，有思路和想法时立即实施，一步一步迭代开发，在工作周报的制度中也让我们记录了开发过程，有了明确的时间线索。随着这条线索，我们可以及时地整理当周的问题，积累代码，完成相关的文档记录。

课程设计的意义

在课程设计中，我们对 msp430 单片机也有了更深入的认识，从亲自动手的过程中，我们能够学到比嵌入式课堂上更多更实用的知识，能够更深入地理解单片机，也对我们的动手能力和学习能力有很大的提高。另外，搜集资料也是重要的能力之一。在互联网上如何寻找前人开发的经验和实例，最快的吸收并化为己用对开发进度是很有帮助的。在课设过程中我们也学习到了严谨科学的项目开发方法和思路。这无疑对我们今后的发展是有好处的。

兴趣

兴趣是最好的老师，在课设中培养兴趣，才是最有价值的。

硬件课设个人总结

——明 星

算下来硬件课程设计进行到现在已经差不多4个月了，从一开始的选题，提交TI项目申报书到第一次在南一楼集中开会，老师就告诉我们这将是漫长且需要花费足够精力和时间的课程，最初我们对于TI和TI公司的MSP430系列的了解是微乎其微的，直到通过一些官方的视频和文字资料以及从学校申请领到的launchpad的上手我们逐渐对这些知识有了初步的了解，然后慢慢地通过实际的践行以及对不可预知问题的最终寻求方案并解决的过程中我们逐渐对MSP430单片机有了更深入的了解，而这些是远远超过我们在嵌入式理论基础课上那些单纯的原理性教授形式所赋予我们的对于新鲜事物的认识。

在这个过程中要感谢许许多多帮助我们的人。首先是我们的指导老师——曾喻江老师，给我们提供很多技术上或者实际问题上的帮助和指引，能在百忙之中抽出时间给我们开视频会议传达精神；其次是我们这个小组的成员，毕竟我们的大多数工作都是在考试周之前就基本结束了，而不是像其他有些组到最后才突击，我认为这是与我们良好的规划和友好的团队合作分不开的，我们每周都会提交工作周报，对我们本周的任务予以总结并提出下周计划，在最后两周更是每天

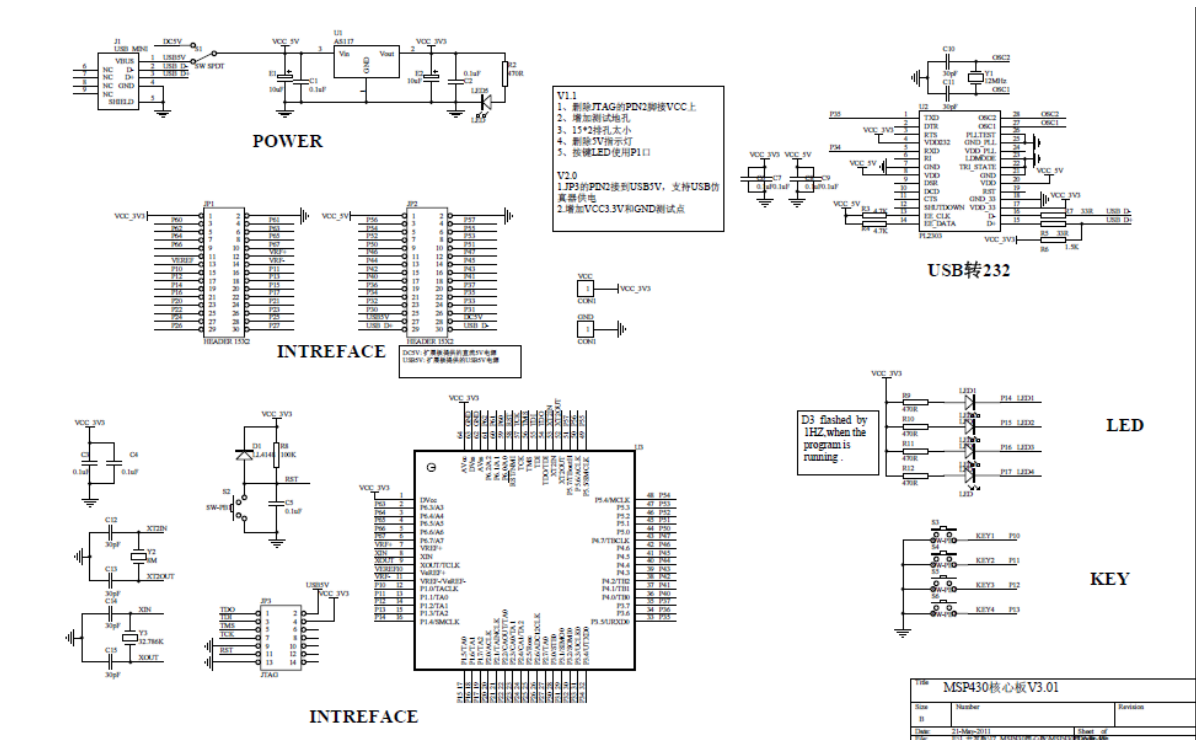
都要提交站立会议纪要，这种略显得让人有压力的报告方式现在看起来是我们能够高效率、高质量完成课程设计各种琐碎的事情的重要基础；然后还有很多淘宝上的陌生人，不但在我们购买的元器件中附带丰富的学习资料和良好示例，并且在之后使用中遇到不懂的问题请教时仍然能够给我们细心且耐心的讲解和提供可能的解决方案给我们参考选择。

我们选择参加TI杯，是希望通过自己的付出为我们前三年的大学生活画上圆满句号。与我个人而言，这次硬件课设也让我对我们通信\电信专业有了更加立体的领悟，在大二大三的时候我经常会觉得有些科目学起来似乎感觉没什么实际用处，而这次在此过程中才发现并不是它们没有用，而是我们在慢慢的积累和融会贯通之后将会有更深的理解并也能运用到我们的实际操作中，即小原理完成大功效。我相信这次宝贵的经历，无论其结果如何，都会是我们大学生活的重要一笔，也将为我们将来的生活学习和工作提供诸多参考价值。当然在此过程中，我们也有很多不足，比如在有些问题的考虑上还不够周全等等，但是我们组队员从没想过放弃，我们终将感谢硬件课程设计陪我们度过漫长岁月。

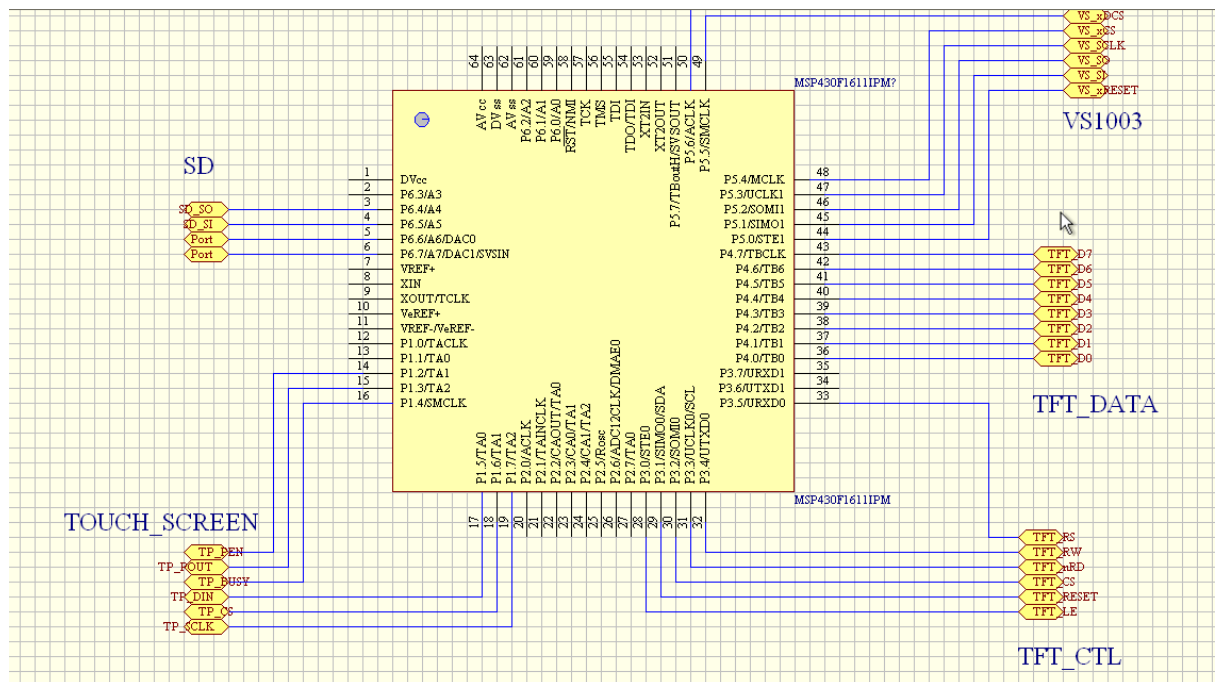
九、参考文献

- 1.MSP430x13xx_MSP430x14xx Mixed Signal Microcontroller (Rev. F).pdf
- 2.用户手册 MSP430x1xx Family User's Guide .pdf
- 3.MSP430 系列超低功耗 16 位单片机原理与应用(华中科技大学).pdf
- 4.使用 MSP430 系列入门.pdf
- 5.MSP430(tm) Hardware Tools User's Guide.pdf

附录（一） 核心板电路图



附录（二） 整机电路图



附录（三） READSOME 使用说明书 （附在后面）

附录（四） 所用器件清单

器件	单价	数量	总价
MSP430F1611	95	1	¥95
TFT	50	1	¥50
VS1003	50	1	¥50

总金额为：195 元