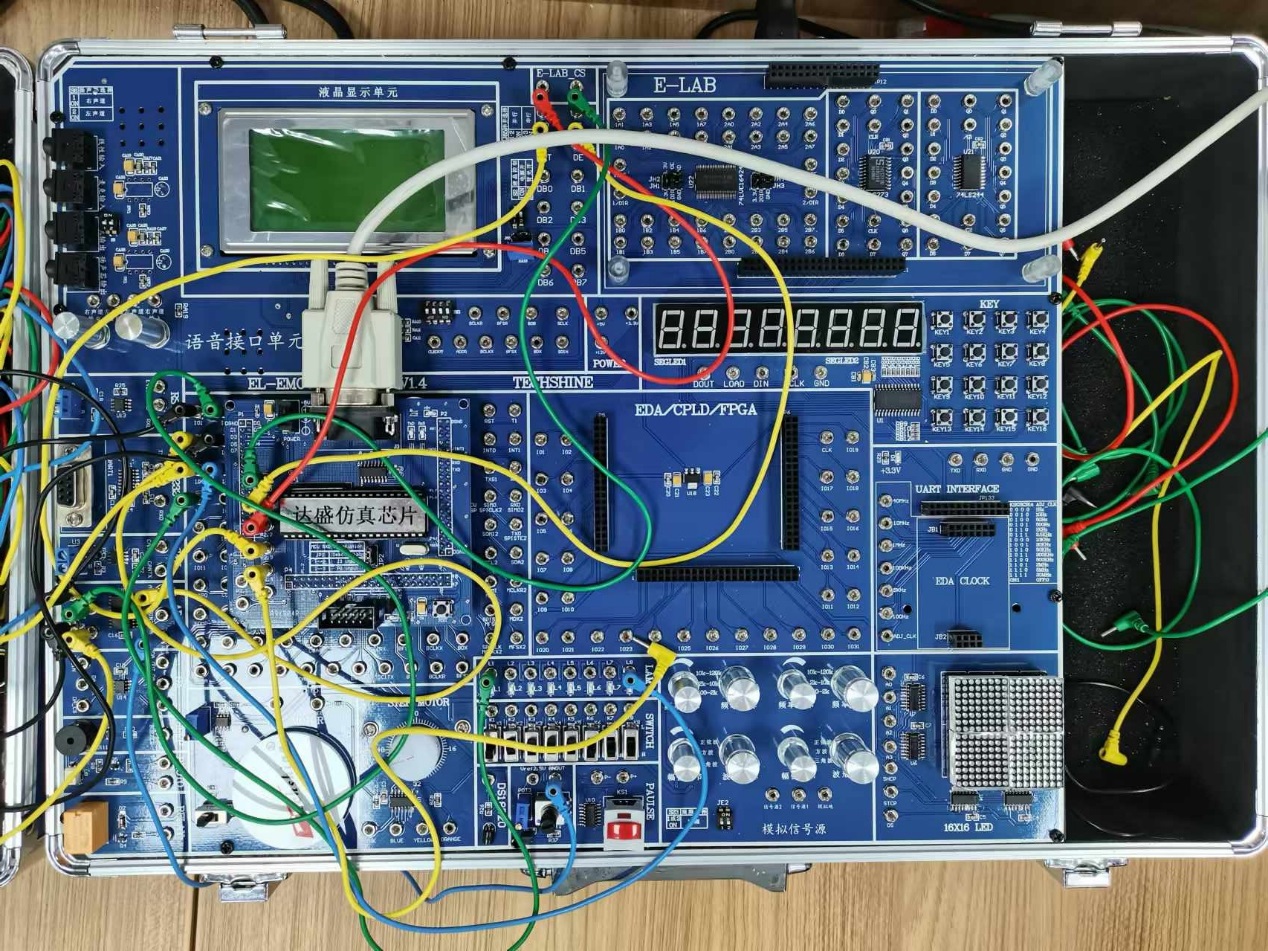
**实验报告六 液晶显示屏和AD转换**

|  |  |  |  |
| --- | --- | --- | --- |
| 姓名 | 学号 | 组内分工 | 工作量占百分比 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

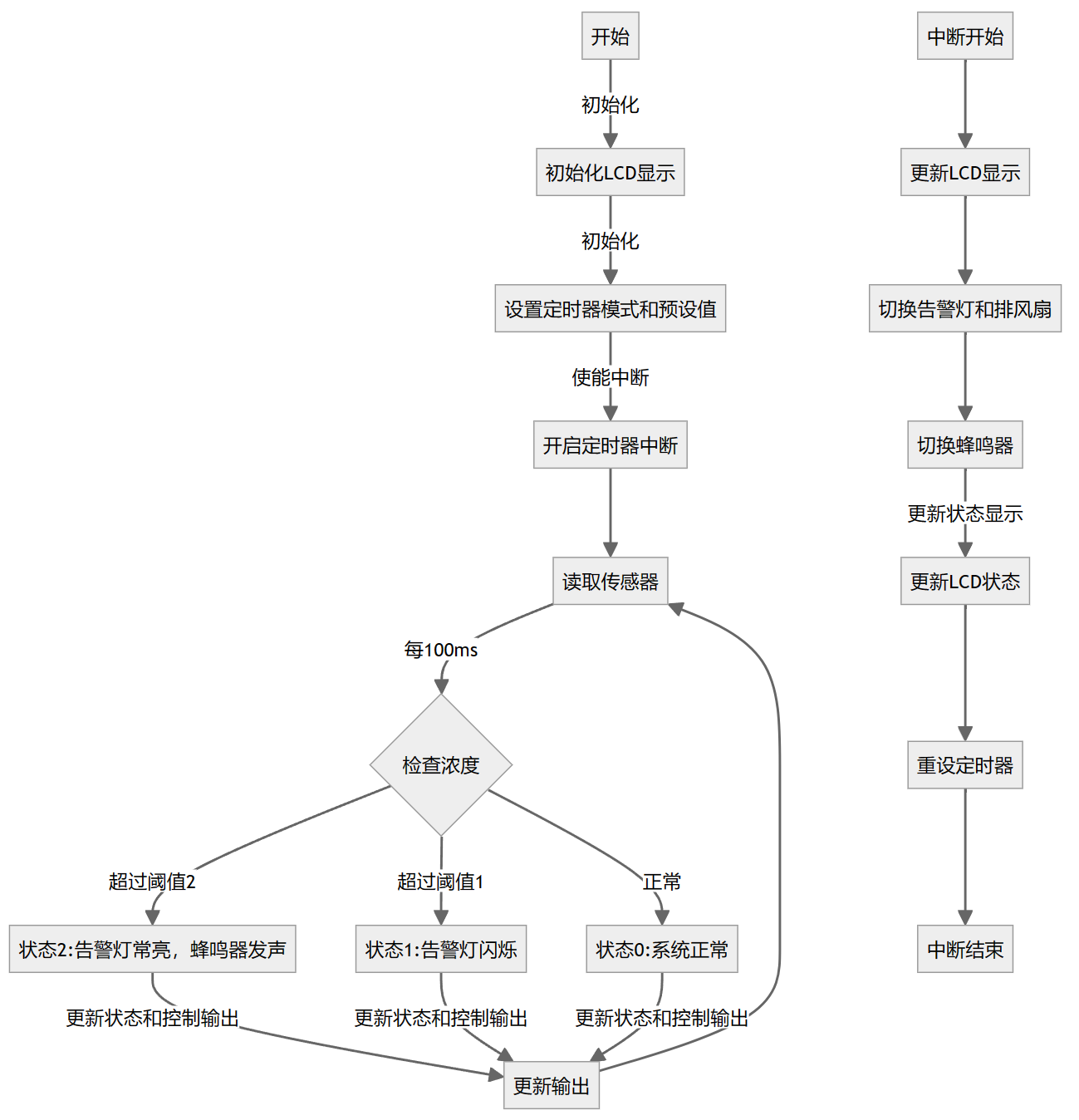
1. 电路图

LCD\_CS连接P2.0，LCD\_SCLK连接P2.1，LCD\_SID连接P2.2，LCD\_RST连接P2.3。P0.5连接告警灯，P0.6连接排风扇，P2.4连接蜂鸣器。DS18B20的DQ连接P0.0。



1. 程序分析

1、流程图的形式给出程序设计的思路，



2、代码及必要注释。

#include <reg51.h>

#include <intrins.h>

#include <stdio.h>

#include "st7920.h"

#include "tlc2543.h"

#include "delay.h"

#define THR\_1 1500

#define THR\_2 3500

unsigned int value;

unsigned char lamp;

unsigned char beep;

sbit IO6 = P0 ^ 5; // 告警灯

sbit IO7 = P0 ^ 6; // 排风扇

sbit IO8 = P2 ^ 4; // 蜂鸣器

unsigned state = 0; // 系统状态：0正常，1超过阈值1，2超过阈值2

#define TIMER\_PRESET\_VALUE 50000

// 定时器中断服务函数

void timer\_ISR(void) interrupt 1

{

    char printText[16];

    static unsigned char count = 0;

    count += 1;

    // 每秒钟

    if (count % 20 == 0)

    {

        sprintf(printText, "  测量值 %4d  ", value);

        LCD\_write\_string(3 << 8, printText);

        if (lamp == 1)

        {

            lamp = 2;

            IO6 = 0;

            IO7 = 1;

        }

        else if (lamp == 2)

        {

            IO6 = 1;

            IO7 = 1;

            lamp = 1;

        }

        else if (lamp == 3)

        {

            lamp = 3;

            IO6 = 1;

            IO7 = 1;

        }

        else

        {

            lamp = 0;

            IO6 = 0;

            IO7 = 0;

        }

        if (beep >= 1 && beep <= 3)

        {

            IO8 = 0;

            beep += 1;

        }

        else if (beep >= 4)

        {

            IO8 = 1;

            beep += 1;

            if (beep > 6)

                beep = 1;

        }

        else

        {

            beep = 0;

            IO8 = 1;

        }

        if (state == 0)

        {

            LCD\_write\_string(2 << 8, "      正常      ");

        }

        else if (state == 1)

        {

            LCD\_write\_string(2 << 8, "      阈值1     ");

        }

        else if (state == 2)

        {

            LCD\_write\_string(2 << 8, "      阈值2     ");

        }

        else

        {

            LCD\_write\_string(2 << 8, "      错误      ");

        }

    }

    TH0 = (65536 - TIMER\_PRESET\_VALUE) / 256; // 设置定时器的高字节

    TL0 = (65536 - TIMER\_PRESET\_VALUE) % 256; // 设置定时器的低字节

}

int main()

{

    unsigned long measured;

    // 设置定时器的工作模式和预设值

    TMOD = 0x01;                              // 定时器0工作在模式1

    TH0 = (65536 - TIMER\_PRESET\_VALUE) / 256; // 设置定时器的高字节

    TL0 = (65536 - TIMER\_PRESET\_VALUE) % 256; // 设置定时器的低字节

    // 开启定时器并使能定时器中断

    TR0 = 1;

    ET0 = 1;

    EA = 1;

    LCD\_init();

    LCD\_write\_string(0 << 8, "  有害气体检测  ");

    LCD\_write\_string(1 << 8, "  Gas Detector  ");

    LCD\_write\_string(2 << 8, "      正常      ");

    LCD\_write\_string(3 << 8, "  测量值     0  ");

    while (1)

    {

        delay\_ms(100);

        value = read2543(0);

        if (value > THR\_2)

        {

            if (state != 2)

            {

                state = 2;

                lamp = 3;

                beep = 1;

            }

        }

        else if (value > THR\_1)

        {

            if (state != 1)

            {

                state = 1;

                lamp = 1;

                beep = 0;

            }

        }

        else

        {

            state = 0;

            lamp = 0;

            beep = 0;

        }

    }

    return 0;

}

1. 思考题（见PPT）

1. 什么是开环控制？

开环控制系统是一种控制系统，其中控制动作不依赖于输出。这意味着输出并不反馈给输入用以校正。在开环控制系统中，输入直接转化为输出，只基于预先设定的条件和规则。其特点是简单，代价低，但缺乏灵活性，因为它不根据输出调整输入。一旦系统启动，就会持续执行相同的操作，不考虑外界条件是否变化。典型的例子包括洗衣机的基本周期、交通信号灯等。

2. 能否在汉字模式下，使得液晶显示屏显示字符串“J吉L林U大+学”，如有此类需要，应当如何完成？

在英文字符后加入一个空格，使得汉字对齐。

3. 在真实场景，可能会发生输入值在阈值附近左右摇摆的情况，会导致输出设备不断的重启，严重时造成设备损坏。如何编程避免此种情况出现？

* **滞后阈值（Hysteresis）**：设置两个阈值，一个用于开启动作，另一个用于关闭动作。
* **时间延迟**：一旦达到阈值，等待一段固定时间（后再检查值是否仍然超过阈值。
* **滤波**：对于模拟信号，可以通过硬件（如低通滤波器）或软件方法（运行平均、加权平均等）来平滑输入读数从而减轻瞬间的波动影响。
* **逻辑限制**：添加额外的逻辑来判断设备是否在很短时间内重复开关，如果是，则在某个时间段内阻止它再次切换状态，或者仅在确认真的有必要时才使设备切换状态。

1. 问题分析

1.告警灯、排气扇和蜂鸣器的状态转换较为复杂，容易出错。