多量程直流电压表

陈逢源　冯诚　项渊博

May 27, 2012

Outline

1. 实现方案
2. 仪表放大器
3. Atmega128A
4. LCD12864

电路和芯片

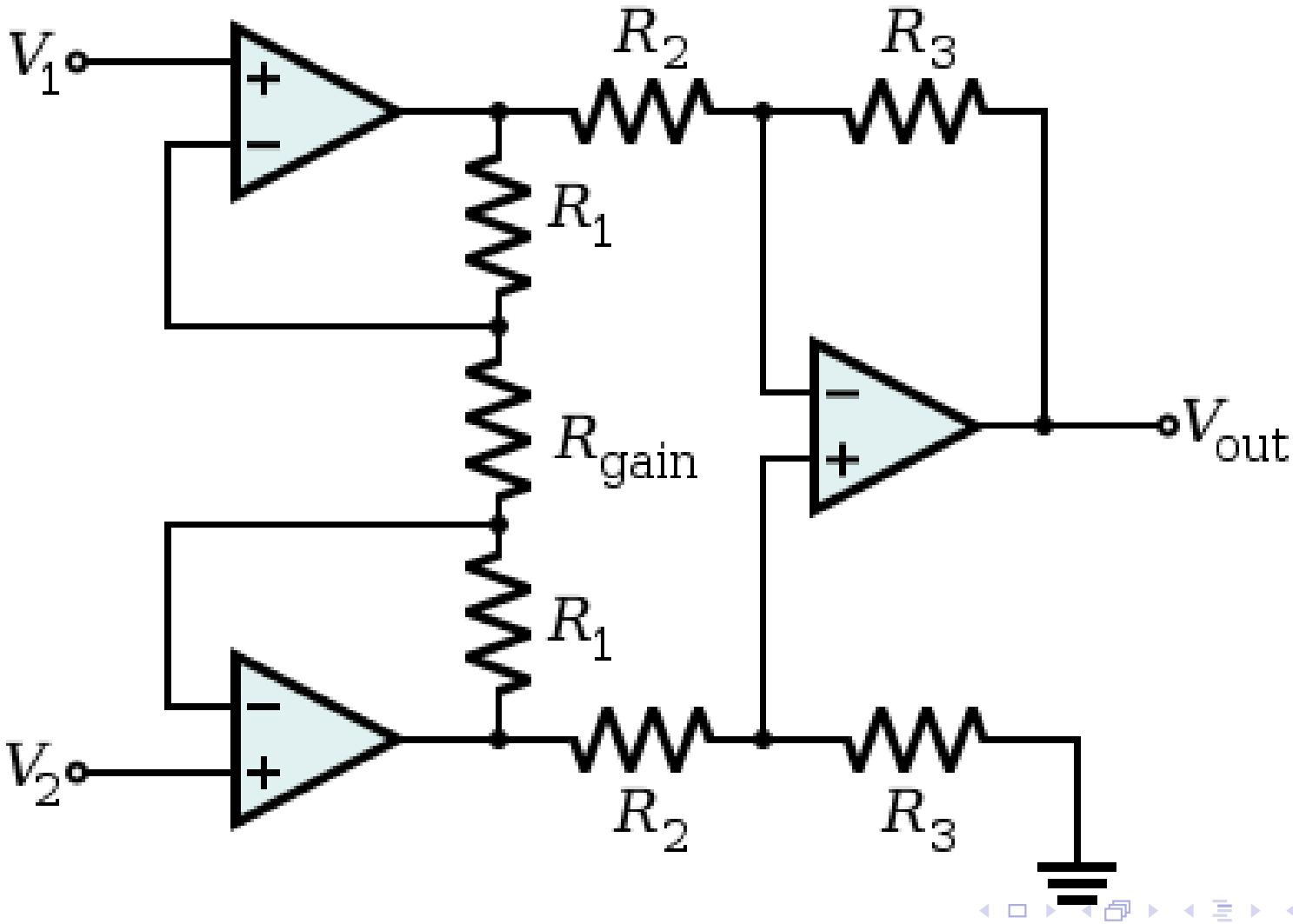
- 仪表放大器
- Atmega128A
- lcd12864
- CD4051 模拟开关

仪表放大器特性

非常低的直流偏移，低漂移，低雜訊，非常高的 [開路增益]，非常大的共模抑制比（CMRR）和高輸入阻抗.
电路的增益为:

$$\frac{V_{out}}{V_2 - V_1} = (1 + \frac{2R_1}{R_{gain}})\frac{R_3}{R_2}$$

标准仪表放大器电路图



软件开发环境

emacs(24.0)	编辑器
avr-gcc	编译器
avr-libc	C library for Atmel AVR microcontrollers
avrdude	AVR Downloader
make	Standard tool to compile source trees

ATmega128A 芯片性能概述

High-performance, Low-power Atmel® AVR® 8-bit Microcontroller

- Up to 16MHz Throughput at 16MIPS
- 8 channel,10-bit ADC
- 53 Programmable I/O lines
- 4Kbytes Internal SRAM
- SPI Interface for In-System Programming

ATmega128A AD 性能

- 10-bit Resolution
- 0.5 LSB Integral Non-linearity
- 13 μs - 260 μs Conversion Time
- Sleep Mode Noise Canceler
- 8 Multiplexed Single Ended Input Channels
- Interrupt on ADC Conversion Complete

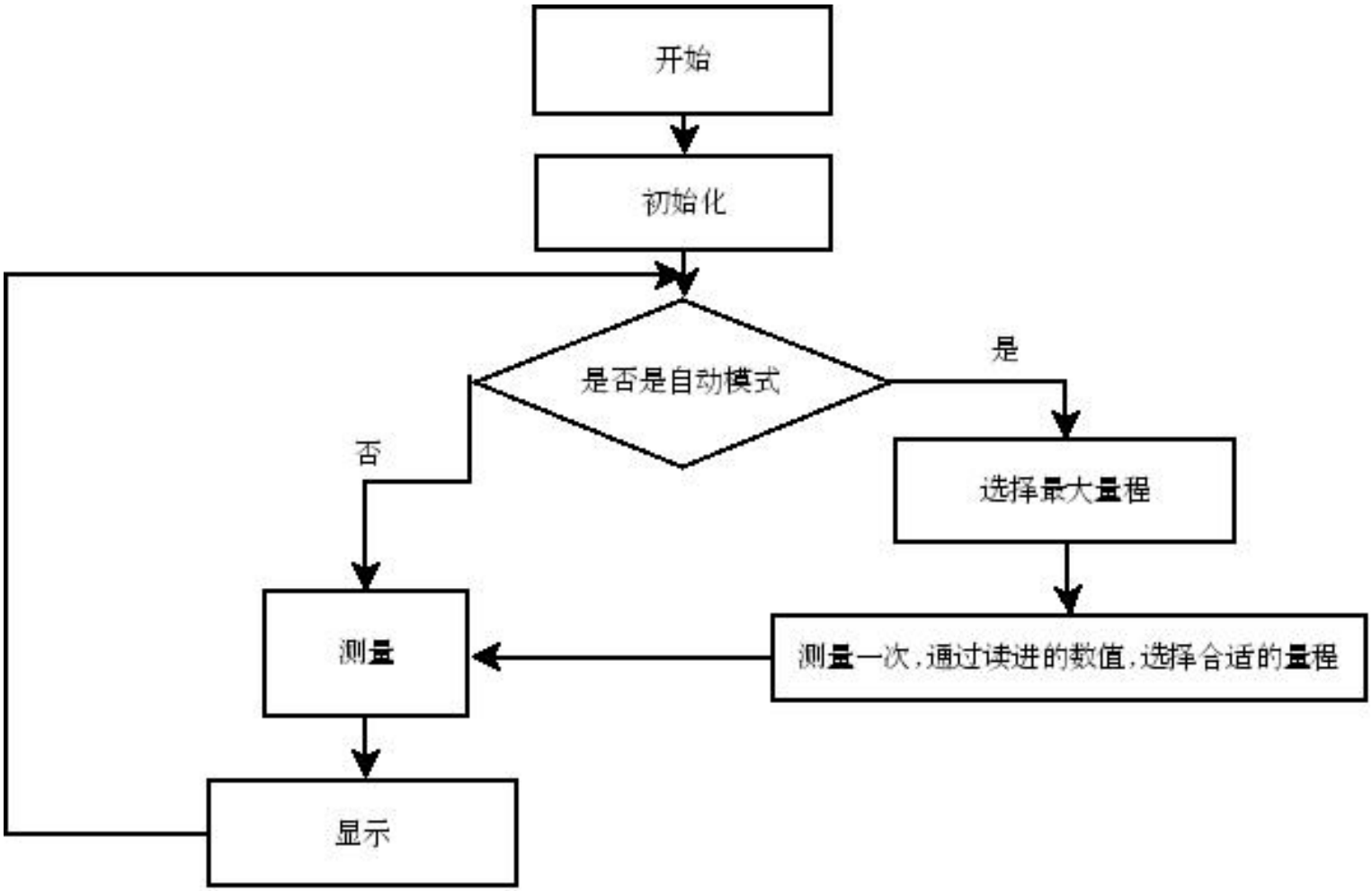
AD 转换

- 转换公式 $ADC = \frac{V_{IN} \cdot 1024}{V_{REF}}$
- AD 寄存器
- 程序中 ADCSRA 的设置
  - $((1 \ll ADEN)|(1 \ll ADIE)|(1 \ll ADPS2)|(1 \ll ADPS1)|(1 \ll ADPS0))$
  - 打开 ADC
  - AD 转换完成中断打开
  - AD 时钟预分频为 128
- AD 程序

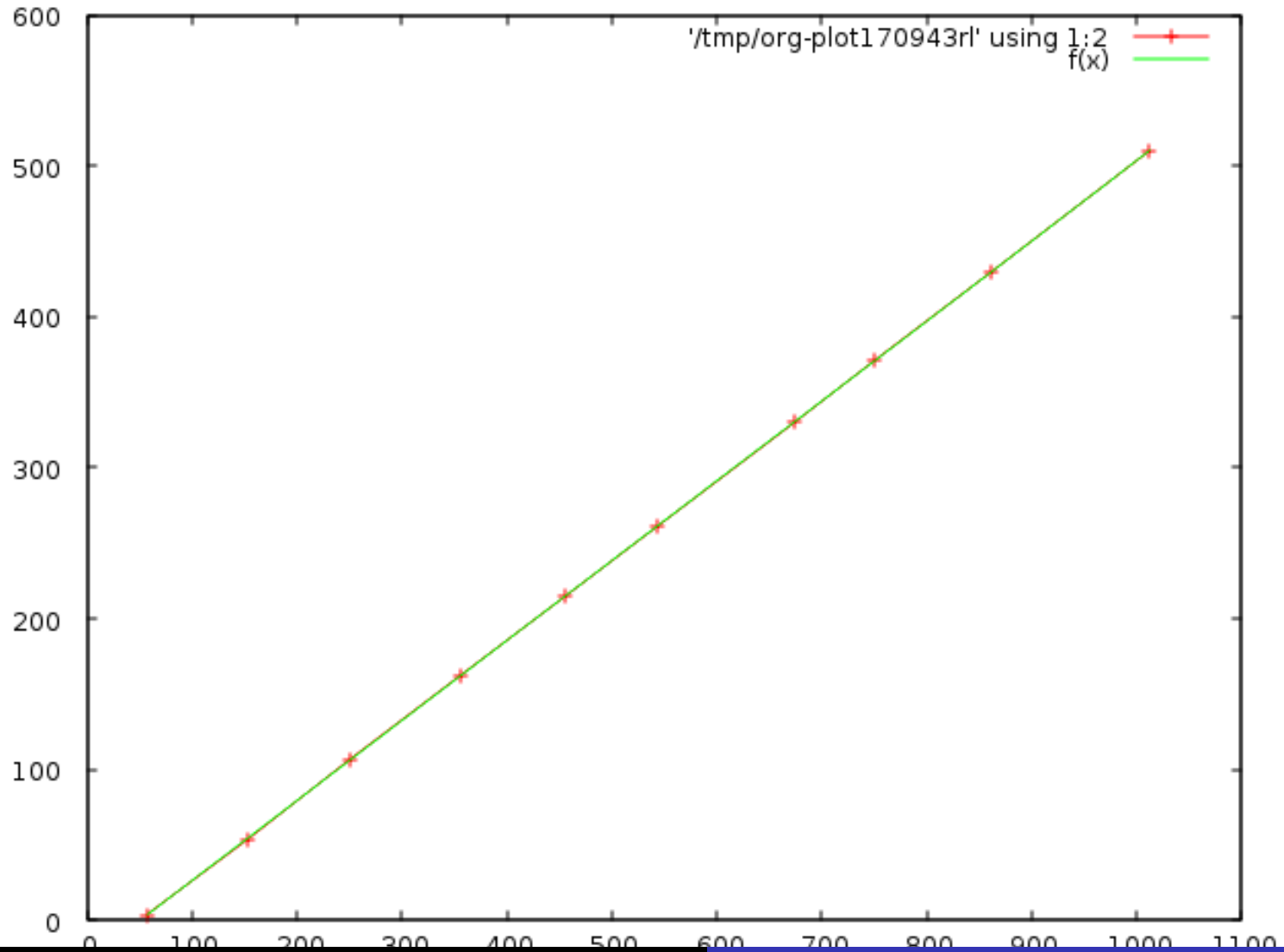
量程切换

- 中断寄存器
- 中断程序
- 具体代码
- 自动切换量程

流程图



0~500mA


'/tmp/org-plot170943rl' using 1:2
f(x)
600
500
400
300
200
100
0
0
100
200
300
400
500
600
700
800
900
1000
1100

性能

- 性能 128x64 液晶点阵,8x4 中文显示
- 并行 8 位数据通行
- 独立 LED 背光电源
- 标准 ASCII 字符库和简体中文字库

单片机和 LCD12864 通信

```
void lcd12864_init(void);
void lcd12864_clear(void);
void lcd12864_move_cur(uint8_t x,uint8_t y);
void lcd12864_write_cmd(uint8_t command);
void lcd12864_write_data(uint8_t wrdata);
```

显示数字和字符串

```
void lcd12864_dis_num(int32_t num);
void lcd12864_dis_str(char * str);
```

显示程序

```
void dis(int32_t average, uint8_t mode)
{
        int32_t reality;
        lcd12864_clear();
        lcd12864_move_cur(0, 0);
        lcd12864_dis_str("多量程直流电压表");
        lcd12864_move_cur(0, 1);
        lcd12864_dis_str("挡位:");
        lcd12864_move_cur(3, 1);
        lcd12864_dis_str(mode_str[mode]);
        lcd12864_move_cur(0, 2);
        lcd12864_dis_str("电压:");
        reality = convert(average, mode);
        if (below(reality, mode)) {
                if (mode == 2 || mode == 5) {
                        sprintf(str_temp, "%ld.%03ld", reality / 1000,
                                reality % 1000);
                        lcd12864_move_cur(6, 2);
                        lcd12864_dis_str("伏");
                        lcd12864_move_cur(3, 2);
                        lcd12864_dis_str(str_temp);
                } else {
                        lcd12864_move_cur(3, 2);
                        lcd12864_dis_num(reality);
                        lcd12864_move_cur(6, 2);
                        lcd12864_dis_str("毫伏");
                }
```