

基于数字兆欧表自动切换档位的研究与实现

戎真真¹, 虎恩典², 刘勇³, 鲁研青¹, 杨喆³

(1. 北方民族大学 计算机科学与工程学院, 银川 750021; 2. 宁夏大学 机械工程学院, 银川 750021;
3. 北方民族大学 电气信息工程学院, 银川 750021)

摘要: 随着电力、电气设备的日益复杂化, 对于其安全性能也有了更高的要求, 数字兆欧表作为测量绝缘阻值的重要工具, 已在工业领域中广泛应用; 为了满足测量精度以及兆欧表的可靠性, 通过基于 S3C2440 的嵌入式平台, 提出了一种基于数字兆欧表的自动切换档位的实现方案, 以达到更加精确的测量; 此方案通过判断采集到的电压值选择相应的采样电阻来实现, 并结合硬件平台和软件设计, 分别介绍了驱动程序和应用程序的编写与测试, 并很好地应用到了实际电路中; 经测试, 能够达到预期的效果, 测量精度高, 测量误差控制在 $\pm 2\%$ 。

关键词: S3C2440; 自动切换; 驱动程序; 应用程序; 测量精度

Research and Implementation of Automatic Switching Gear of the Digital Megger

Rong Zhenzhen¹, Hu Endian², Liu Yong³, Lu Yanqing¹, Yang Zhe³

(1. School of Computer Science and Engineering, North University of Ethnics, Yinchuan 750021, China;
2. School of Mechanical Engineering, Ningxia University, Yinchuan 750021, China;
3. School of Electrical and Information Engineering, North University of Ethnics, Yinchuan 750021, China)

Abstract: With the growing complexity of electric power and electrical equipment, a higher requirement has been brought up for the safety performance. As an important tool of measuring insulation resistance, Digital megger has been widely used in industrial field. In order to satisfy the measurement precision and the reliability of the megger and achieve a more accurate measurement, a realizable scheme has been put forward based on digital megger switch gear that is based on the S3C2440 embedded platform. This plan will choose a corresponding sampling resistance by judging collected voltage value, and combined with the hardware platform and software design, introducing the preparation and testing of the drivers and applications respectively. what's more, a good application has been used in practical circuits. After testing, it can achieve the desired results which makes a higher measurement precision and Controls the measurement error about $\pm 2\%$.

Keywords: S3C2440; automatical switch; drivers; application; measurement accuracy

0 引言

绝缘电阻是我国计量法规定的电气安全检测项目中的强检项目。本课题来源于宁夏回族自治区教育厅的科技攻关项目, 围绕绝缘电阻的测量展开的。兆欧表是测量绝缘电阻的专用仪表, 适用于电力、邮电、铁路、通信、矿山等部门测量各种装置的接地电阻以及低电阻的导体电阻值, 所以应用非常广泛。传统的兆欧表主要是手摇指针式兆欧表。它的主要不足之处是测量时必须用手摇动发电机并保证有 120 r/min 的速率才能维持正常的输出电压, 而且同一台摇表的电压等级少, 量程范围小, 表针指示读数, 刻度为非线性, 测量误差和读数误差都较大。而研制一种准确、稳定、方便、适用范围广的数字兆欧表具有了非凡的意义。而测量精度的实现无疑成为了设计中至关重要的环节, 本数字兆欧表共分为 4 个电压等级 (500 V, 1 000 V, 2 500 V, 5 000 V), 并且每一个电压档位又分别对

应 6 个采样电阻, 通过不同被测电阻产生的采样电压来自动地切换采样电阻, 实现更加精确的测量, 测量误差控制在 $\pm 2\%$ 。

1 软硬件平台

1.1 硬件平台

S3C2440 处理器是基于 ARM920T 内核的 32 位 RISC 嵌入式芯片。该 ARM 内核的 CPU 主频最高可达 533 MHz, 此处使用 499 MHz, 它除了集成 3 个串口、SD 卡控制器、USB Host 控制器、LCD 控制器、Nand Flash 控制器以及实时时钟外, 还增添了工业控制总线 (CAN)、Camera 控制器 (数码相机接口)、PCMCIA 接口 (可接无线网卡或调制解调器及其他外设)。另外, 用 1 个 96 针总线插槽引出 CPU 的局部总线, 可外接其他总线设备并与多方通信。本开发板提供了一个 34Pin 2.0 mm 间距的 GPIO 接口, 并采用 GPG 口来控制档位的切换, GPF 口来实现采样电阻的切换, 具体的接口连接如图 1 所示。

此数字兆欧表的整体流程结构为通过倍压生成模块将 12 V 直流电源生成系统所需的直流负高压, 然后通过模拟开关 4662 加至采样电阻两端, 采样电阻的模拟输入信号经过 10 位 CMOS 的 ADC 模数转换器转换为数字信号, 并能根据采集到的不同数值自动切换采样电阻, 提高测量精度^[1]。然后 S3C2440 微处理器将数字信号以及相应的采样电阻通过计算转

收稿日期: 2014-01-09; 修回日期: 2014-03-04。

基金项目: 宁夏回族自治区教育厅重点项目。

作者简介: 戎真真 (1990-), 女, 河南汝洲市人, 硕士研究生, 主要从事计算机控制和嵌入式技术方向的研究。

虎恩典 (1956-), 男, 河南密县人, 教授, 硕士生导师, 主要从事检测、机电控制及计算机控制方向的研究。

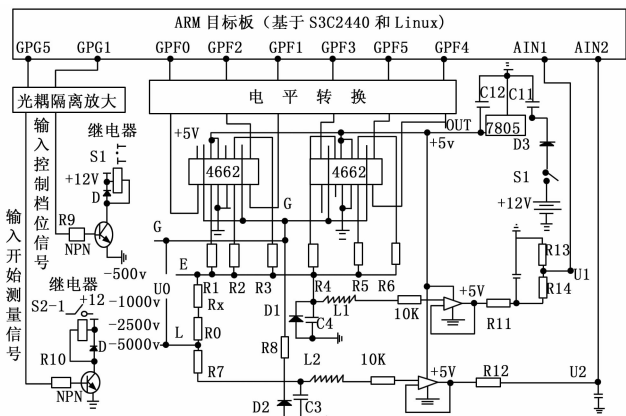


图 1 gpio 接口电路图

化为绝缘阻值，最后在 ARM 目标板 LCD 上显示出来。其中 ARM 目标板通过 GPG 端口和继电器相连，GPG1 是控制是否开始测量，并产生负高压，GPG5 通过继电器控制开关电源控制器芯片 TL594，达到控制数字兆欧表测量档位的功能，整体电路原理如图 2 所示。

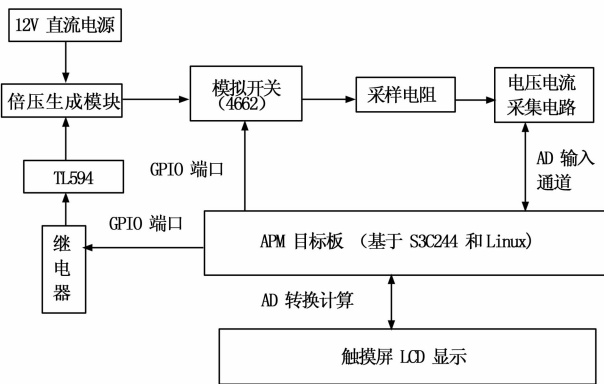


图 2 数字兆欧表原理框图

1.2 软件平台

Linux 操作系统是一种性能优良、源码公开且被广泛应用的免费操作系统，由于其体积小、可裁减、运行速度快、良好的网络性能等优点，可以作为嵌入式操作系统。实时可靠性是嵌入式应用较为普遍的要求，本设计采用的是 Linux 2.6 内核，尽管 Linux 2.6 并不是一个真正的实时操作系统，但其改进的特性能够满足响应需求。Linux 2.6 已经在内核主体中加入了提高中断性能和调度响应时间的改进，在嵌入式领域，Linux 2.6 除了提高其实时性能，系统的移植更加方便，同时添加了新的体系结构和处理器类型——包括对没有硬件控制内存管理方案的 MMU-less 系统的支持，可以支持大容量内存模型、微控制器，同时还改善了 I/O 子系统，增添更多的多媒体应用功能^[2]。

一个完整的嵌入式程序首要的条件就是开发环境的搭建，这里包括 QtE 开发环境和 Linux 环境，应用程序和操作系统通过编译生成可执行文件，然后下载到开发板来驱动硬件的动作。本设计中采用的是 fedora14 嵌入式平台，QtE-4.6.3 的 QT 开发源代码，在 QtCreator 应用平台上设计 Qt 人机交互界面，然后通过编译生成可以在开发板 mini2440 上运行的可执

行文件，最后运行，驱动硬件，自动切换档位，实现最终被测电阻的精确测量。嵌入式程序设计流程如图 3 所示。

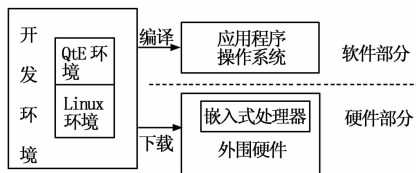


图 3 嵌入式程序设计流程图

软件部分和硬件部分的交互，需要驱动程序来完成。Linux 的设备驱动主要分为 3 类：字符驱动、块设备驱动及网络设备驱动等。而本设计中兆欧表的自动切换采样电阻属于字符设备驱动。设备驱动程序软件封装了如何控制这些设备的技术细节，并通过特定的接口导出一个规范的操作集合；内核使用规范的设备接口（字符设备接口和块设备接口）通过文件系统接口把设备操作导出到用户空间程序中^[3]。设备驱动程序与外界接口的示意如图 4 所示。

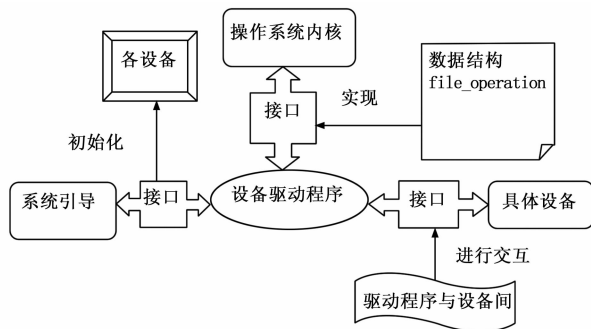


图 4 设备驱动程序与外界接口

2 档位自动切换部分驱动程序设计

兆欧表主要用来检查电气设备、家用电器或电气线路对地及相间的绝缘电阻，以保证这些设备、电器和线路工作在正常状态，避免发生触电伤亡及设备损坏等事故。因此对于其测量精度的要求较高。而本设计依据不同被测阻值会产生不同的采样电压，从而自动地切换采样电阻，实现更加精确的测量。档位驱动程序主要由初始化函数、设备的打开和释放函数、读写函数、控制函数、设备的中断和轮询处理、文件结构及设备卸载函数等组成^[4]。

2.1 驱动初始化

档位驱动程序除了完成档位设备主、次设备号的申请和档位设备的注册外，还应包括硬件的初始化。下面是根据 S3C2440 处理器的特性编写的设备初始化程序：

```
static int __init dev_init(void)
{
    int ret;
    int i;
    for (i = 0; i < 8; i++) {
        s3c2410_gpio_cfgpin(dangwei_table[i], dangwei_cfg_table[i]); //设置 dangwei 对应的端口寄存器的状态为输出(OUTPUT)
        s3c2410_gpio_setpin(dangwei_table[i], 0); //设置 dangwei 对应的端口寄存器为低电平输出
    }
}
```

```
ret = misc_register(&misc); //注册设备
printk (DEVICE_NAME"/tinitialized/n");//打印初始化信息
return ret;
}
```

可以看到, 这里涉及到两个函数分别是 `s3c2410_gpio_cfgpin`, `s3c2410_gpio_setpin`, 这两个函数均在 `/linux-2.6.32.2/arch/arm/plat-s3c24xx/gpio.c` 文件中定义。它们分别对 8 个 GPIO 口进行配置, 从函数名来看, `cfgpin` 对引脚寄存器状态进行配置, 而 `setpin` 是对寄存器数据值进行配置^[5]。其中的注册设备是把它注册成了混杂设备, 主、次设备号如下:

```
.minor = 253,
.name = DEVICE_NAME,
.fops = &dev_fops,
```

2.2 配置 gpio 端口

档位以及切换采样电阻时对应的 GPIO 端口列表:

```
static unsigned long dangwei_table [] = {
S3C2410_GPF(0),
S3C2410_GPF(1),
S3C2410_GPF(2),
S3C2410_GPF(3),
S3C2410_GPF(4),
S3C2410_GPF(5),
S3C2410_GPG(1),
S3C2410_GPG(5),
}
```

这里 `S3C2410_GPF` 和 `S3C2410_GPG` 宏定义在 `mach/gpio-nrs.h` 中, 其中 6 个 GPF 端口是用来控制采样电阻的, 两个 GPG 端口是用来控制量程档位的。

档位对应的端口将要输出的状态列表如下, 此处分配的 8 个 gpio 端口全部置为输出的状态:

```
static unsigned int dangwei_cfg_table [] = {
S3C2410_GPIO_OUTPUT,
S3C2410_GPIO_OUTPUT,
S3C2410_GPIO_OUTPUT,
S3C2410_GPIO_OUTPUT,
S3C2410_GPIO_OUTPUT,
S3C2410_GPIO_OUTPUT,
S3C2410_GPIO_OUTPUT,
S3C2410_GPIO_OUTPUT,
```

2.3 ioctl 函数

`ioctl` 是设备驱动程序中对设备的 I/O 通道进行管理的函数。所谓对 I/O 通道进行管理, 就是对设备的一些特性进行控制, 例如串口的传输波特率、马达的转速等等。

```
static int s3c2440_dangwei_ioctl(
struct inode * inode,
struct file * file,
unsigned int cmd,
unsigned long arg)
{
switch(cmd) {
case 0:
case 1:
if (arg > 7) {
return -EINVAL;
```

```
}
s3c2410_gpio_setpin(dangwei_table[arg], ! cmd);
return 0;
default:
return -EINVAL;
}
}
```

此段程序分别给 6 个采样电阻对应的 gpio 引脚的数据值进行设置。

2.4 文件结构函数

```
static struct file_operations dev_fops = {
.owner = THIS_MODULE,
.ioctl = s3c2440_dangwei_ioctl,
};
```

2.5 退出注销设备

```
static void __exit dev_exit(void)
{
misc_deregister(&misc);
}
```

3 配置各文件参数

3.1 修改 KConfig 文件

在 KConfig 文件中添加:

```
config DANGWEI_MINI2440
tristate "DangWei Support for Mini2440 GPIO dangwei"
depends on MACH_MINI2440
default y if MACH_MINI2440
help
```

This option enables support for dangwei connected to GPIO lines

on Mini2440 boards.

设置一些内核配置选项, 在配置内核时添加一个 `DangWei Support for Mini2440 GPIO dangwei` 的选项, 让配置者可以选择是否把 `dangwei` 的驱动编译进内核, 如果选了这个选项, 编译时 (`CONFIG_DANGWEI_MINI2440`) 就等于 `y`。这一选项支持 `dangwei` 与 `gpio` 接口的连接。

3.2 修改 makefile 文件

在 makefile 文件中添加以下一行代码:

```
obj-(CONFIG_DANGWEI_MINI2440) += mini2440_dangwei.o
```

也就是说当 `CONFIG_DANGWEI_MINI2440` 是 `y` 时, 就把驱动 `mini2440_dangwei` 编译进内核

4 编译内核并下载到开发板

在 Linux 终端进入内核目录 `Linux2.6.32.2` 下打开配置文件, 在命令行输入 `make menuconfig`, 在弹出的配置窗口中添加档位设备选项。然后保存, 退出。

输入: `make zImage` 开始编译内核。

在内核编译完成之后通过 `minitools` 工具, 下载内核到开发板。

5 应用测试程序的设计

数字兆欧表在测量阻值时, 考虑到精度的问题, 需要适时的切换不同的电路, 以达到更加精确的测量。本设计的兆欧表电路中总共设有 6 个档位, 量程档位的切换主要是通过模拟开

关的通断来实现的，并通过开发板的 GPIO (GPFO 至 GPF5) 接口来输出控制信号。

5.1 设置采样电阻对应的端口状态

```
void MainWindow::setRm(int i)
{
    int fd = ::open("/dev/dangwei", 0);
    if (fd < 0) { return; }
    ioctl(fd, 1, 0);
    ioctl(fd, 1, 1);
    ioctl(fd, 1, 2);
    ioctl(fd, 1, 3);
    ioctl(fd, 1, 4);
    ioctl(fd, 1, 5);
    ioctl(fd, 0, i);
    ::close(fd);
}
```

fd 就是用户程序打开设备时使用 open 函数返回的文件标识符，ioctl 函数是文件结构中的一个属性分量，这里我们的驱动程序提供了对 ioctl 的支持，所以就可以直接在用户程序中使用 ioctl 函数控制设备的 I/O 通道。ioctl 函数的返回值是一个整数类型的值，如果命令执行成功，ioctl 返回零，如果出现错误，ioctl 函数应该返回一个负值^[6]。这个负值会作为 errno 值反馈给调用此 ioctl 的用户空间程序。

本设计中先对端口进行初始化，然后再对 i 对应的 gpio 口置位，也就是说这里在选择好了采样电阻时，对其相对应的 gpio 口置位，从而实现硬件电路的连接^[7]。

5.2 档位自动切换子程序

```
int MainWindow::selectRm(int j)
{int ret;
 setRm(j);
 update_adc();
 if(U1>0.8 && U1<3.9){ret=j;}
 else if(U1>3.9){j++;if(j>5)
 return -1;ret=selectRm(j);}
 else if(U1<0.8){j--;if(j<0)
 return -2;ret=selectRm(j);}
 return ret;
}
```

测量时，首先设置某一支路处于接通的状态，然后通过判断从数据采集和转换程序中采集到的电流信号支路采样点电压值大小，来控制档位的切换，保证采样点的电压值在 0.8~3.9 V 之间（读入 A/D 的值在 0.4~1.95 V 之间）。如果采样点电压值高于 3.9 V（读入 A/D 的值高于 1.95 V），就要从当前的支路往高档位支路切换；如果低于 0.8 V（读入 A/D 的值低于 0.4 V），就要从当前的支路往低量程支路切换；如果在第六档时，采样点电压值仍高于 3.9 V，则视为超出测量范围；如果量程档位在 0.8~3.9 V 之间，则置位变量 ROK，否则，一直循环判断档位，直至寻找到合适档位，并提取出此时相应的电阻值。变量 ROK 为全局变量，用于标志量程档位是否合适。其中 update_adc() 函数是读取从开发板采集到的电压值。档位自动切换程序流程如图 5 所示。

其中 update_adc() 在读取电压值时，考虑 mini2440 中所采集到的最大电压值是 3.3 V，所以为了避免击穿电路，这里采取了分压的方式，取一半读入 AIN1 中，最后在应用程序

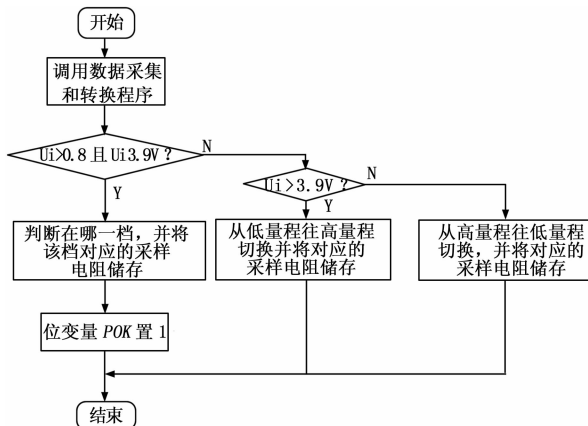


图 5 档位自动切换程序流程图

中放大至实际的电压值。

5.3 读取储存的采样电阻值

```
void MainWindow::readRm()
{ int num = -1;
 int R[6] = {R1, R2, R3, R4, R5, R6};
 num = selectRm(2);
 Rm = R[num];
 printf("%d", Rm);
}
```

R [6] 数组存储的是 6 个采样电阻的阻值，当判断确定位于支路电压范围内的采样电压，此时对应的电阻值就是采样电阻 Rm，同时储存此采样电阻到应用层，用于最终被测电阻的计算。

6 运行测试

本 QT 人机交互界面是在 Linux 环境下，采用 QtE-4.6.3 版本的源代码，在 QtCreator 应用平台下设计的数字兆欧表的程序。并将此程序在 QT 中编译运行，生成可执行文件 Digital_Megger，然后下载到开发板。

修改脚本文件 qt4:

```
cd/usr/local/Trolltech/QtEmbedded-4.6.3-arm/demos/embedded/fluidlauncher
cd /bin
./fluidlauncher - qws
./Digital_Megger -qws
运行程序 qt4[8]。
```

与被测电路连接之后，10 M 的被测电阻显示的运行结果如图 6 所示。

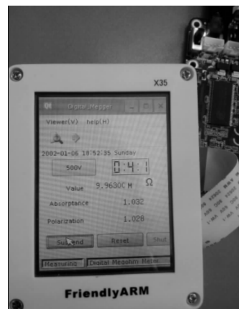


图 6 10 M 电阻运行结果图

系数的关系曲线，其中横风系数通过关系式 (2) 求得，从图中可以看出横风系数近似为常数 0.9，由此便知 $C=0.9$ ；图 7 (c) 为风速与估算值的线性回归曲线，其中估算值是指在空气密度为 1.1614 kg/m^3 、横风系数为 0.9 的条件下针对不同压差计算出的风速值。风速与估算值的线性回归方程为 $y=0.995x+0.0475$ ，回归系数为 0.9998，从图中可以看出二者近似相等，线性相关，由此可以验证关系式 (2) 在工程设计中具备较高的可信度。

3.2 传感器标定方法

由于差压传感器 ASDX-D4 输出是经过偏置的数字信号，其差压零点值易受外界环境影响发生零点漂移，必须进行传感器标定。标定方法如下：将横风传感器探头置于无风密闭环境中，通过串口以异步通信方式发送标定命令，下位机认定当前压差为零点，并将该压差值保存在 MSP430 的 Flash 中，避免掉电丢失。

4 结论

数字式横风传感器研制成功后，经过长期测试工作稳定，实现了预期的设计效果。该传感器主要应用于军事火控领域、还可应用于航天航空、汽车工业等各种需要检测横风量的领域。

作者创新观点：基于风洞实验提出一种检测横风风速及风向的设计方案，通过数学建模的方法简化研究对象，根据相关理论推导得到风速测量的关系式，并对其可信度进行实验验证；

利用 Matlab 对 IIR 数字滤波算法进行仿真研究，证明滤波器设计的可行性和合理性，编写滤波算法并在微处理器中实现该算法；采用差压传感器测量气流压差，温度传感器测量环境温度，通过查表法测得空气密度，实时监测气流变化输出横风风速；实现全双工异步通信功能，可通过向串口发送命令的方式对传感器进行零点标定，消除零点漂移。

参考文献：

- [1] Lu R H, Zhao C S, Yang Z W. Research on computational method for extend target posture angle [J]. Chinese Journal of Scientific Instrument, 2006, 27: 1211-1212.
- [2] Chandrakssan A, Brodersen R. Low-power COMS digital design [J]. IEEE Journal of Solid-State Circuits, 1992, 27: 473-483.
- [3] 吴成富, 冯 喆, 等. 一种改价的基于红外传感器的无人机姿态测量方法 [J]. 计算机测量与控制, 2013, 21 (4): 883-885.
- [4] Xiao J X, Chen G J, Xie Y L. Test and design-for-testability of IIR filter [J]. Journal of Computer-Aided Design & Computer Graphics, 2007, 19: 203-209.
- [5] Mukherjee N, Rajski J, Tyszer J. Testing schemes for IIR filter structure [J]. IEEE Transactions on Computers, 2001, 50: 674-688.
- [6] Dabrowski A, Sozanski K. Control circuit for active power-harmonic-compensation filter in power systems realized With DSP TMS320C50 [A]. 3rd European DSP Education & Research Conference [C]. Paris, France, 2000; 1-7.

(上接第 2700 页)

经测试，每更换一个电阻，均能实现可靠精确的测量，测量误差为 $\pm 2\%$ ，在工业测试领域具有很好的借鉴作用。这里采集了 20 个电阻的测量值，500 V 电压档位 1~9 M 之间电阻所测得的电阻值分布如图 7 所示，同一档位下 10~100 M 之间电阻所测得的电阻值分布如图 8 所示。

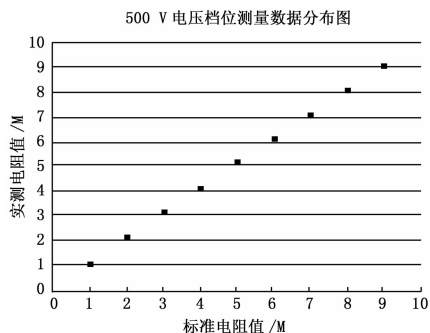


图 7 1 M 间隔的被测电阻的数据分布图

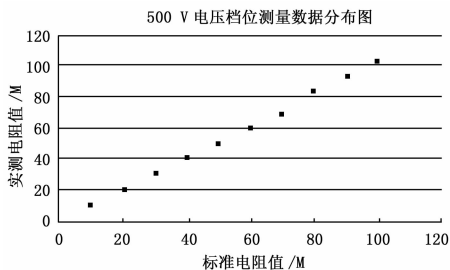


图 8 10 M 间隔的被测电阻的数据分布图

7 结束语

本文介绍了基于 S3C2440 的数字兆欧表自动切换采样电阻的驱动程序和应用程序的实现方案。经运行验证，该驱动程序加入内核，并烧写到开发板上后，运行应用程序能够可靠的工作，且性能稳定，测量精度高。这一设计方案同时对在其他嵌入式设备上扩展 gpio 口也有一定的参考价值 and 意义，应用程序也对其他的设备有一定的借鉴作用。

参考文献：

- [1] 虎恩典, 张广忠. 单片机控制的智能型电子兆欧表 [J]. 测控技术, 2004, 23 (12): 71-73.
- [2] 李 俊. 嵌入式 Linux 设备驱动开发详解 [M]. 北京: 人民邮电出版社, 2008.
- [3] James W G. 测试驱动的嵌入式 C 语言开发 [M]. 尹 哲, 译. 北京: 机械工业出版社, 2012.
- [4] 冯国进. 嵌入式 Linux 驱动程序设计从入门到精通 [M]. 北京: 清华大学出版社, 2008; 151-168.
- [5] 王润琪, 欧阳益锋, 等. 基于 QT 的触摸屏驱动在 LINUX 下的应用设计 [J]. 计算机测量与控制, 2005, 13 (12): 1429-1430.
- [6] 赵振亮, 徐立鸿, 邓梧鹏, 等. 基于 Linux 系统中嵌入式 GUI 的研究与分析 [J]. 微型电脑应用, 2006, 22 (11): 42-44, 50.
- [7] 闫锋欣, 曾泉人, 张志强, 等. C++ GUI Qt4 编程 (2 版) [M]. 北京: 电子工业出版社, 2008.
- [8] Rubini A. Linux 设备驱动程序 [M]. 北京: 中国电力出版社, 2000.