

读写 SD 卡操作实验

实验目的

1. 输入字节数组或者字符串保存在 SD 卡中
2. 从 SD 卡读取数组或者字符串并显示在串口屏上

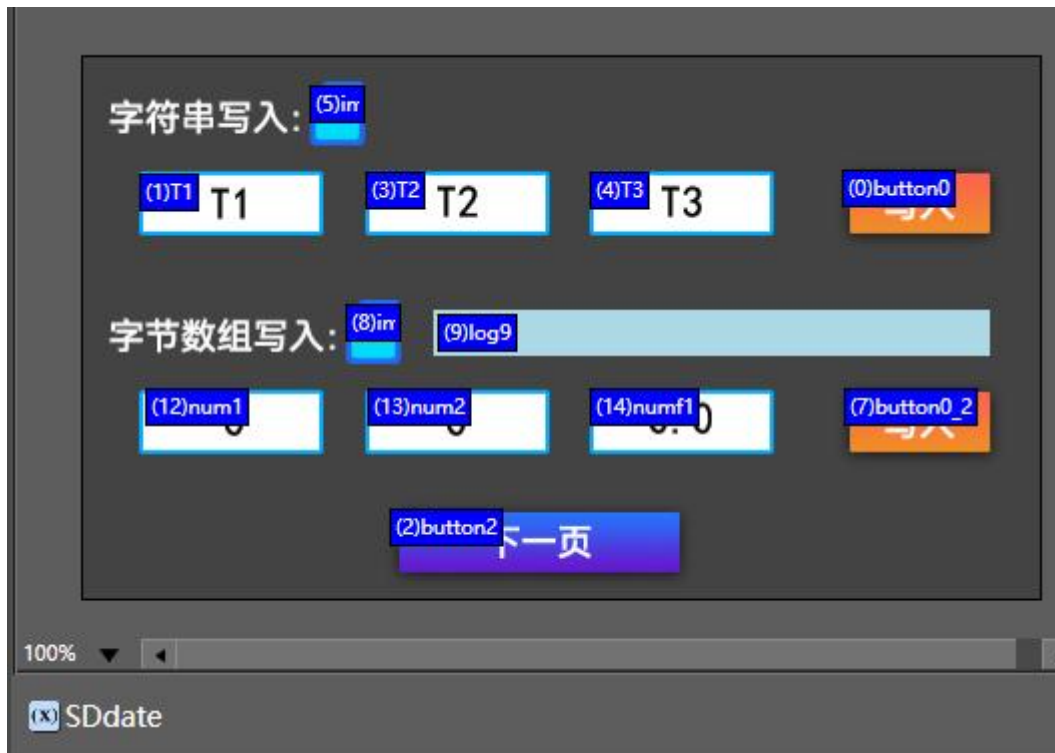
实验工具

1. 任意型号三易串口屏，本工程所用型号为（SANY-WQ-S43-01-C）
2. SD 卡，FAT32 格式
3. 上位机软件 VP，编辑界面和工程

页面布局

实验工程分为两个界面，写 SD 卡和读 SD 卡。用翻页函数 `showpage()` 关联

Filewrite





控件介绍

1. Filewrite（写入页）

控件名	属性		功能
Image 序号 5 和 8	可见	图片	通过切换不同的图片内容标，记写入是否成功
T1-3 序号 1, 3, 4	可见	文本	调用键盘输入并显示要写入的字符串
Button0	可见	按钮	执行事件脚本实现写入 SD 卡
Log9	可见	日志	显示要写入的字节数组
Num1, num2	可见	整数	输入要写入的整数
Numf1	可见	浮点数	输入要写入的浮点数
button0_2	可见	按钮	执行写入操作
Button2	可见	按钮	翻页到下一页
SDdate	不可见	变量	变量用于缓存拼接之后的字符串，使脚本简洁

2. Fileread（读取页）

控件名	属性		功能
Textbox4	可见	文本框	显示读取的字符串
Log6	可见	日志	显示读取的字节数组
Button0	可见	按钮	执行事件脚本实现读取 SD 卡字符串
Button1	可见	按钮	执行事件脚本实现读取 SD 卡字节数组
Button2	可见	按钮	翻页到上一页

逻辑实现

写入 (fwrite)

1. 字符串输入

生成键盘控件之后文本控件可以直接在属性栏选择调用，此处无脚本
文本脚本如下：

按下事件：T1.bgImg = user.image.page1_P; //切换背景图片实现点击效果

弹起事件：T1.bgImg = user.image.page1; //切换背景图片实现点击效果

T1-T3，逻辑和脚本一致，pix 工程可查看

2. 字符串拼接与写入

Button0 按钮 按下事件 脚本如下

```
/****** 格式 1：数据存储在 txt 格式文件内  
******/
```

//将三个文本字符串拼接起来数据，数据之间用空格隔开一下

```
SDdate.txt = T1.txt + " " + T2.txt + " " + T3.txt;
```

//根据函数返回值来控制 SD 卡图标的显示和隐藏（返回 0：写入成功 返回-1：写入失败）

```
if(fileWriteLine("StringREC.txt",SDdate.txt) == 0)//存入 StringREC.txt 文件中
```

```
{  
    image38.visible=1;  
}else  
{  
    image38.visible=0;  
}
```

```
/****** 格式 2：数据存储在 csv 格式文件内  
******/
```

```
/*
```

//将三个文本字符串拼接起来数据，存储为 CSV 数据，数据之间必须用英文逗号(,)隔开

```
SDdate.txt = T1.txt + "," + T2.txt + "," + T3.txt;
```

//根据函数返回值来控制 SD 卡图标的显示和隐藏（返回 0：写入成功 返回-1：写入失败）

```
if(fileWriteLine("StringREC.csv",SDdate.txt) == 0)//StringREC.csv 文件中
```

```
{
```

```

        image38.visible=1;
    }else
    {
        image38.visible=0;
    }
    */

```

/*
注：

上面两种方法选择使用一种即可；

fileWriteLine("StringREC.txt",SDdate.txt)函数，参数为：存储数据的目标文件 要存储的文本数据

StringREC.txt 文本文件可不用手动新建，函数写入时，StringREC.txt 文件

StringREC.txt 文件也可放到文件夹内，例如放到 SD_WR 文件夹内，此时参数应写完整路径："SD_WR/StringREC.txt"

前提是文件夹 SD_WR 需要提前手动新建好，并且函数中路径输入不能出错

```

    */
    弹起事件脚本：image38.visible=0;//隐藏图片

```

3. 字节数组输入

输入整数和浮点数

生成键盘之后整数控件可以直接调用，此处无需脚本

整数控件脚本如下

按下事件:num1.bgImg = user.image.page1_P;//切换背景图片实现点击效果

弹起事件:num1.bgImg = user.image.page1;//切换背景图片实现点击效果

Num1 num2 逻辑脚本一致，numf1 输入浮点数脚本逻辑一致

4. 写入字节数组

button0_2 按钮执行写入操作脚本如下

//按下写入按钮，将输入的数字转换拼接为数组形式，并写入 SD 卡内

byte arr[8];//定义一个数组，长度：2 字节整数 + 2 字节整数 + 4 字节浮点数

```

//byte 0-1
arr[0] = (byte)((num1.val >> 8) & 0xff);//数据拼接
arr[1] = (byte)((num1.val >> 0) & 0xff);

//byte 2-3
arr[2] = (byte)((num2.val >> 8) & 0xff);//数据拼接
arr[3] = (byte)((num2.val >> 0) & 0xff);

//byte 4-7
floatToBytes(numf1.valf,arr,4,1);//浮点数转字节数组。
//4 个参数:需要转换的浮点数 缓存转换结果的数组 起始位置 大小端

//根据函数返回值来控制 SD 卡图标显示和隐藏（返回 0：写入成功 返回-1：写入失败）
if(fileWriteBytes("BytesREC.bin",0,arr,0,8) == 0)//写入到 BytesREC.bin 文件中
{
    image39.visible=1;
}else
{
    image39.visible=0;
}

//利用 log 控件显示数组 arr 的数据,注意显示 hex 数据时，要勾选控件的 hexMd 属性
log9.clear();//清空
log9.addBytes(arr,0,8);//显示

/*
注：
    fileWriteLine("DateREC.txt",SDdate.txt)函数，参数为：存储数据的目标文本，要存储的文本数据

    DateREC.txt 文本文件可不用手动新建，函数写入时，会自动创建 DateREC.txt 文件

    DateREC.txt 文件也可放到文件夹内，例如放到 SD_WR 文件夹内，此时参数应写完整
    路径："SD_WR/DateREC.txt"
    前提是文件夹 SD_WR 需要提前手动新建好，并且函数中路径输入不能出错

*/
弹起事件脚本如下
image39.visible=0;//隐藏图片

```

5. 翻页至下一页

Button2 按钮

脚本 如下

弹起事件: showPage(1);//显示第一页

读取 (fileread)

1. 读取字符串

Button0 按钮写入脚本, 实现读取字符串并显示在 textbox4 文本框控件

脚本如下

```
int value;
int i;
/*
//读取文件的行数,函数参数为文件路径
value = fileReadLineCount("StringREC.txt");

textBox4.txt = "";//读出显示前, 先清空显示区
for(i=0;i<value;i++)
{
    //利用函数读取每行所有的字符, 每行后面添加一个换行, 给文本框控件显示
    textBox4.txt += fileReadLine("StringREC.txt",i,20)+"\r\n";
    //参数为: 文本路径, 第几行, 读取的字符数
}
*/

//读取文件的行数,函数参数为文件路径
value = fileReadLineCount("StringREC.txt");

textBox4.txt = "";//读出显示前, 先清空显示区
for(i=0;i<value;i++)
{
    //利用函数读取每行所有的字符, 每行后面添加一个换行, 给文本框控件显示
    textBox4.txt += fileReadLine("StringREC.csv",i,20)+"\n";
    //参数为: 文本路径, 第几行, 读取的字符数
}
```

2. 读取字节数组

Button1 按钮写入脚本，实现读取 SD 卡 Bin 文件字节数组并打印在 log6 日志控件显示
脚本如下

```
byte arr[8];
```

```
//读取文件的字节数据
```

```
fileReadBytes("BytesREC.bin",0,arr,0,8);
```

```
//参数为：文件路径 起始位置 用于缓存所读数据的数组 数组的缓存起始位置 字节数
```

```
//利用 log 控件显示数组 arr 的数据,注意显示 hex 数据时，要勾选控件的 hexMd 属性
```

```
log6.clear();//清空
```

```
log6.addBytes(arr,0,8);//显示
```

3. 翻页至上一页

Button2 按钮

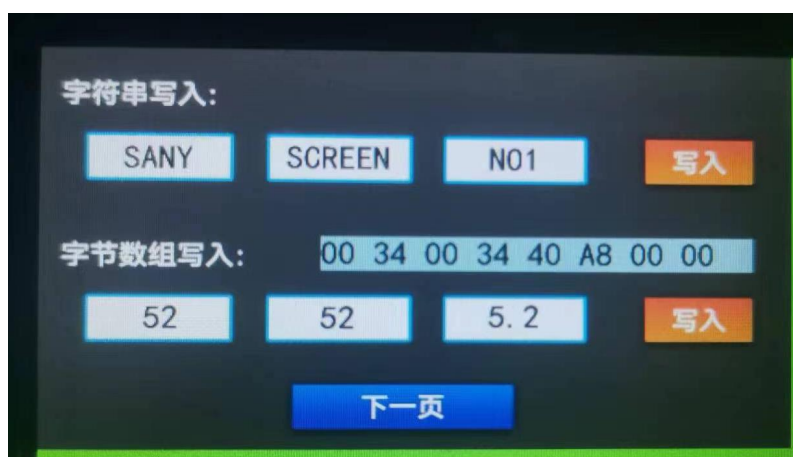
脚本 如下

弹起事件：showPage(0);//显示第 0 页

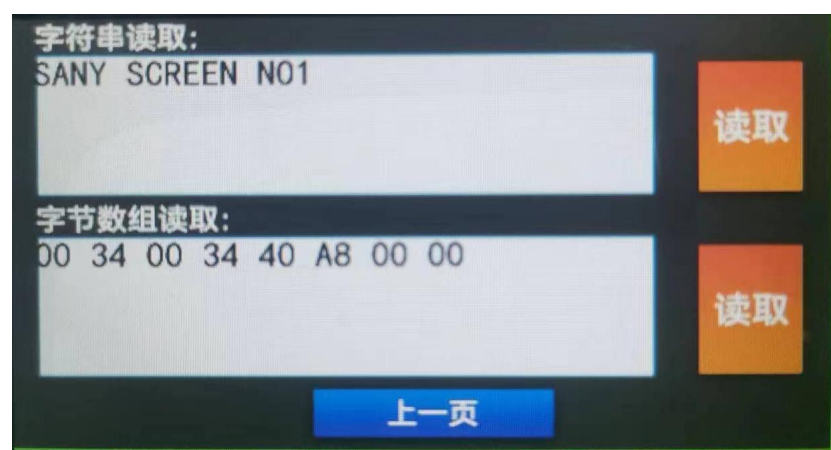
实验结果

1. 字符串写入分别输入 SANY SCREEN NO1 点击写入 sd 卡图标闪烁一次写入成功
2. 整数浮点数分别输入 52 52 5.2 点击写入 sd 卡图标闪烁一次下写入成功切页
3. 点击读取，字符串显示为 SANY SCREEN NO1
4. 字节数组点击读取 显示为 00 34 00 34 40 a8 00 00

写入界面



读取界面



SD 卡内容
SD 卡界面

窗 > U 盘 (G:)					▼	🔄
名称	修改日期	类型	大小			
BytesREC.bin	2018/8/4 22:48	BIN 文件	1 KB			
StringREC	2018/8/4 22:48	文本文档	1 KB			

StringREC 文件

