

# Wince sqlite3 移植

Date: 2013/04/20

公开资料

类别	内容
关键词	WINCE、sqlite
摘要	描述如何在 wince6.0 上移植 sqlite3.

修订历史:

版本	日期	原因
V1.00	2013-04-20	创建文档

# 目录

1 概述.....	3
2 SQLite 移植.....	4
2.1 下载 SQLite .....	4
2.2 创建 SQLite Dll 工程 .....	4
2.3 SQLite 编译 .....	6
3 SQLite 使用.....	8
4 免责声明 .....	11

# 1 概述

SQLite 是一款轻量的遵守 ACID 的关系数据库管理系统，它包含在一个相对较小的 C 库中。它的资源占用非常低，在嵌入式设备中可能只需几百 K 的内存就够了。它能够支持 Windows/Linux/Unix 等主流的操作系统。

本文档描述如何在 wince 下移植和使用 sqlite。

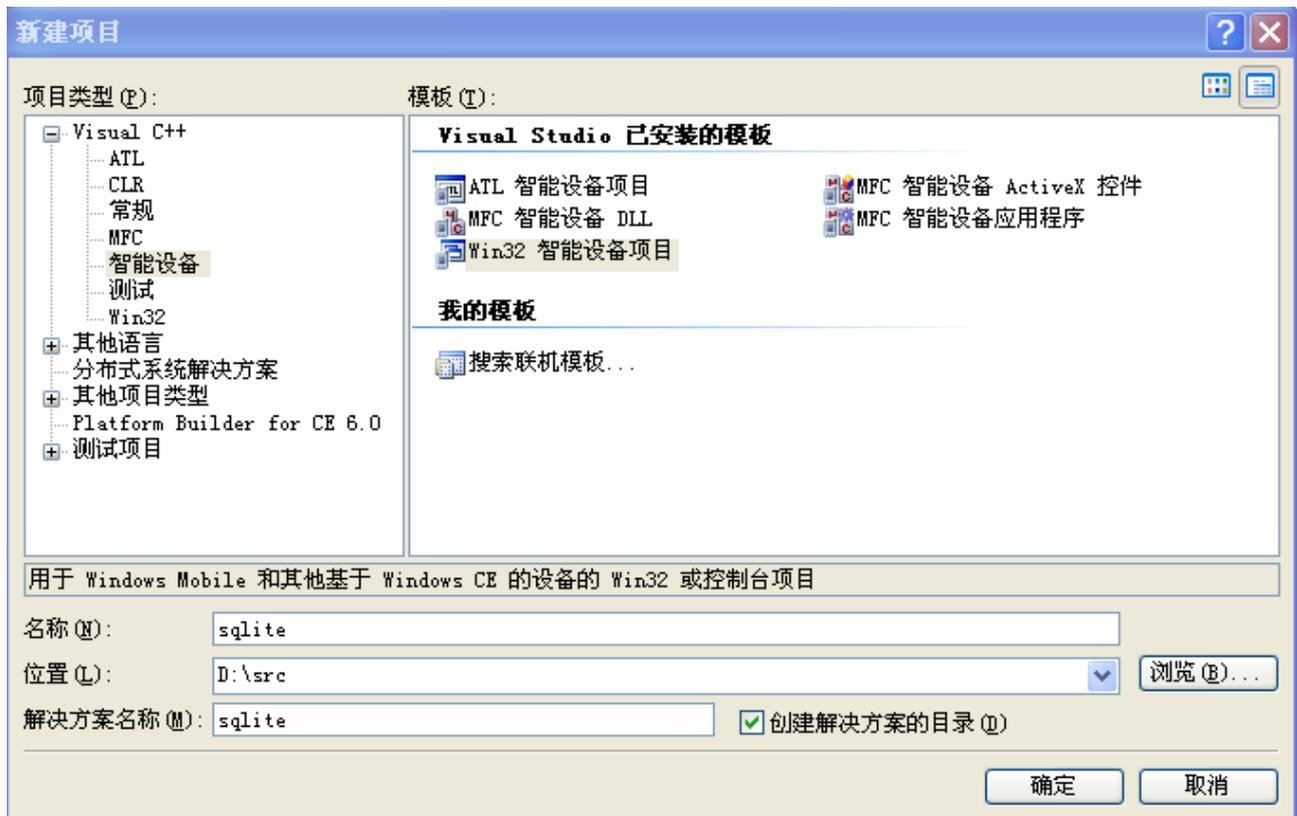
## 2 SQLite 移植

### 2.1 下载 SQLite

从 SQLite 官方网站 <http://www.sqlite.org/download.html> 下载最新的源代码，编写本文档时的版本为 3.7.16。一共包含如下几个文件 `sqlite-amalgamation-3071602.zip`，`sqlite-dll-win32-x86-3071602.zip`，`sqlite-src-3071602.zip`，然后解压。

### 2.2 创建 SQLite DLL 工程

A、运行 VS2005，点击文件 (F) -> 新建 (N) -> 项目 (P)，项目类型为 Visual C++ -> 智能设备 -> Win32 智能设备项目，项目名称为 `sqlite`。然后点击确定。



B、在 Win32 智能设备项目向导中的平台设置根据实际情况选择，此处我们选择 EAC-0945。然后点击下一步。



C、在应用程序设置中，应用程序类型选择 DLL(D)，附加选项选择空项目(E)，然后点击完成。

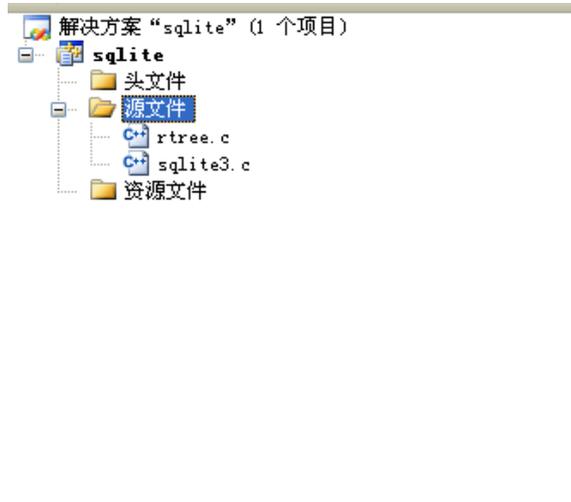


D、复制 sqlite-amalgamation-3071602 目录下的 sqlite3.c, sqlite3.h, sqlite3ext.h 到工程源文件目录。

复制 sqlite-src-3071602\ext\rtree 目录下的 rtree.c, rtree.h, sqlite3rtree.h 到工程源文件目录。

复制 sqlite-dll-win32-x86-3071602 目录下的 sqlite3.def 到工程源文件目录。

然后在**解决方案**窗口, 右键点击 Sqlite 工程下的**源文件**, 选择**添加->添加现有项**, 加入工程目录下的 rtree.c 和 sqlite3.c。添加完毕后, 工程窗口类似如下:



## 2.3 SQLite 编译

A、修改 sqlite3.c, 将

```
#if !defined(HAVE_LOCALTIME_R) && !defined(HAVE_LOCALTIME_S) && \
    defined(_MSC_VER) && defined(_CRT_INSECURE_DEPRECATED)
```

```
#define HAVE_LOCALTIME_S 1
```

```
#endif
```

改为

```
#if !defined(HAVE_LOCALTIME_R) && !defined(HAVE_LOCALTIME_S) && \
    defined(_MSC_VER) && defined(_CRT_INSECURE_DEPRECATED) && !defined(SQLITE_OS_WINCE)
```

```
#define HAVE_LOCALTIME_S 1
```

```
#endif
```

并在 `static int osLocaltime(time_t *t, struct tm *pTm)` 增加如下语句

```
#ifdef SQLITE_OS_WINCE
```

```
struct tm *__cdecl localtime(const time_t *t);
```

```
#endif
```

这是因为 wince 并没有实现 `localtime_s` 和 `localtime_t`, 必须使用使用 `sqlite3.c` 中给 wince 实现的

`struct tm *__cdecl localtime(const time_t *t)` 函数。如果不做以上修改, 编译的时候会出现

`unresolved external symbol localtime_s referenced in function osLocaltime`。

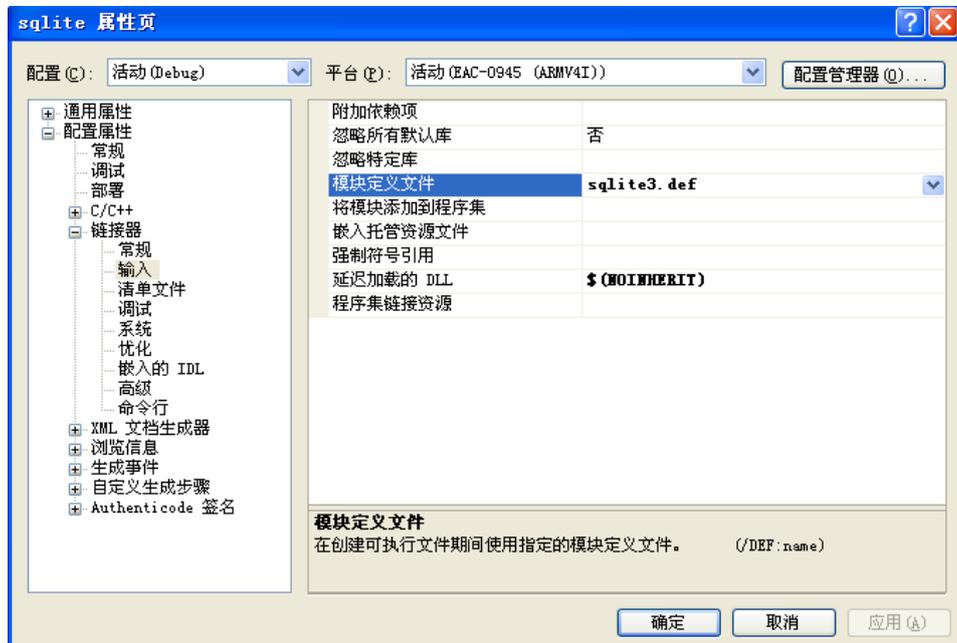
B、将工程属性中的**配置属性->C/C++->高级->编译为**设置为**编译为 C 代码(/TC)**, 在**预处理器->预处理器定义**中增加 `SQLITE_ENABLE_COLUMN_METADATA`。如果不增加这个宏, 则编译的时候会出现如下信息:

```
sqlite3.def: error LNK2001: 无法解析的外部符号 sqlite3_column_database_name
```

sqlite3.def : error LNK2001: 无法解析的外部符号 sqlite3\_column\_database\_name16  
 sqlite3.def : error LNK2001: 无法解析的外部符号 sqlite3\_column\_origin\_name  
 sqlite3.def : error LNK2001: 无法解析的外部符号 sqlite3\_column\_origin\_name16  
 sqlite3.def : error LNK2001: 无法解析的外部符号 sqlite3\_column\_table\_name  
 sqlite3.def : error LNK2001: 无法解析的外部符号 sqlite3\_column\_table\_name16  
 sqlite3.def : error LNK2001: 无法解析的外部符号 sqlite3\_table\_column\_metadata



在链接器->输入->模块定义文件中输入 sqlite3.def。



C、编译工程，即可生成 sqlite.dll 和 sqlite.lib

## 3 SQLite 使用

A、点击文件 (F) -> 新建 (N) -> 项目 (P) ,项目类型为 Visual C++ -> 智能设备 -> Win32 智能设备项目, 项目名称为 `sqlite3test`。然后点击确定。

B、在 Win32 智能设备向导中平台设置为 EAC-0945, 应用程序类型设置为控制台应用程序(O), 然后点击完成。



C、复制 `sqlite3.h`, `sqlite3.lib`, `sqlite3.dll` 到工程目录下, 并将 `sqlite3test.cpp` 中的文件内容替换如下:

```
// sqlite3test.cpp : 定义控制台应用程序的入口点。  
//
```

```
#include "stdafx.h"  
#include <windows.h>  
#include <commctrl.h>  
#include "sqlite3.h"
```

```
int _tmain(int argc, _TCHAR* argv[])  
{
```

```
sqlite3 *db = 0;
int rc;
char *sql;

rc = sqlite3_open("\\nandflash\\test.db", &db);
if(rc)
{
    printf("Can't open database:%s\n", sqlite3_errmsg(db));
    sqlite3_close(db);
    exit(1);
}

// Create a table
sql = "CREATE TABLE SensorData( ID INTEGER PRIMARY KEY,SensorID INTEGER,SiteNum INTEGER,Time
VARCHAR(12),SensorParameter REAL);";
sqlite3_exec( db , sql , 0 , 0 , 0);

//insert data record
sql = "INSERT INTO SensorData VALUES( NULL , 1 , 1 , '200605011206', 18.9 );";
sqlite3_exec( db , sql , 0 , 0 , 0);

sql = "INSERT INTO SensorData VALUES(NULL, 2, 1, '200605011207', 19.0);";
sqlite3_exec( db , sql , 0 , 0 , 0);

do{
    // inquiry data
    int i,j;
    int row=0, col=0;
    char **result;
    sql = "SELECT * FROM SensorData ";
    sqlite3_get_table( db , sql , &result , &row , &col , 0 );
    printf("row:%d column=%d \n" , row , col);

    for(i=0;i!=(row+1);i++)
    {
        for(j=0;j<col;j++)
            printf("%s\t",result[i*col+j]);
        printf("\r\n");
    }
    sqlite3_free_table(result);
}while(0);
printf("Press Enter key to exit. \r\n");
```

```
c = getchar();  
sqlite3_close(db);  
return 0;
```

```
}
```

D、在项目属性的配置属性->链接器->输入->附加依赖项中增加sqlite3.lib，然后编译工程。

E、将编译所得的sqlite3test.exe和sqlite.dll复制到设备的\\NandFlash目录，运行sqlite3test.exe即可。

每运行一次，表的记录都会多2项。

运行后，按Enter键即退出程序。

## 4 免责声明

本手册所陈述的产品文本及相关软件版权均属深圳盈鹏飞科技有限公司所有，其产权受国家法律保护，未经本公司授权，其它公司、单位、代理商及个人不得非法使用。

您若需要我公司产品及相关信息，请及时与我们联系，我们将热情接待。

深圳盈鹏飞科技有限公司将会不断地完善本手册的相关技术内容，请客户适时从公司网站下载最新版本的数据手册，不再另行通知。