**USBKEY开发实验报告**

# 实验工具和语言

VC6.0(开发测试界面和硬件连接的动态链接库)

ZsoftDev(做硬件开发)

XP 虚拟机一台（编译硬件代码）

windows 7 笔记本一台（开发界面和动态链接库，连接开发板）

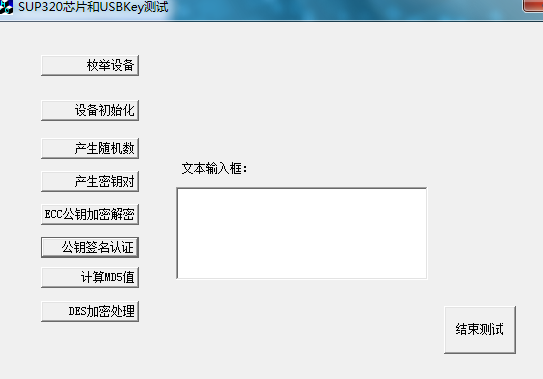
# 实验步骤

1. 首先了解测试界面--> HDL –> COS之间消息传递机制和流程
2. 在现有框架上调试和实现实验一的基本功能

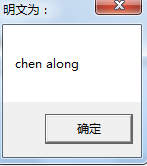
* 实现ECC加密算法（该算法已经在demo中实现，但是在里面做了一些改动）
* 实现MD5 算法(该算法在demo中没有实现，我们实现该算法，并且调试通过)
* 实现私钥签名认证（该算法已经在demo中实现，但是运行时存在一些问题，通过调试解决，现在能够正常运行）
* 实现DES加密算法（该算法在本地调试已经通过，但是由于该算法存在的全局变量过大，导致内存溢出，所以从开发板中移出来，在界面中实现）。

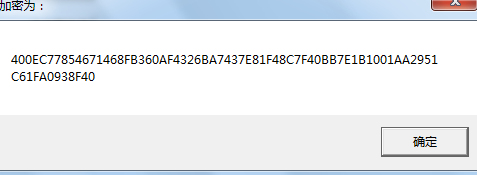
# 实验详细步骤

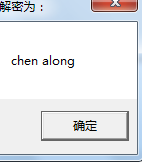
**程序主界面如下所示：**



1. 完善ECC加密算法，原始ECC加密算法中加密类容是写死的，通过增加一个文本框类进行加密类容的输入，实验结果如下图所示：







1. 在开发板上实现MD5算法

* 在DLL工程中，在文件Ukey\_Only.h 中添加一行代码如下

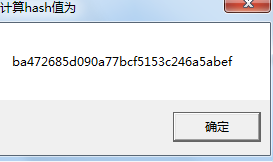
#define MD5\_FUNC = 0x81,表示命令类型

* 在Cactisrc(ATOM)中的mydef.h文件中添加如下代码

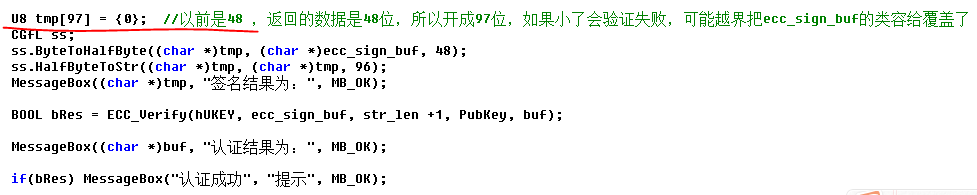
#define MD5\_FUNC 0x81

* 在COS中实现MD5算法（在\Demo U+Key\Cactisrc(ATOM)\app\udisk目录下增加md5.h 和md5.c文件）。
* 实验结果如下所示：





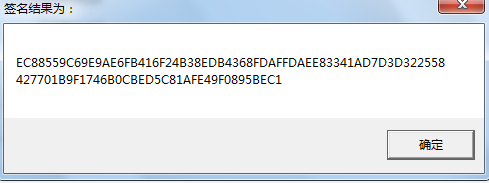
1. 完善私钥签名认证函数

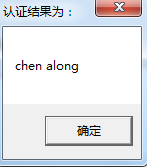


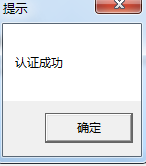
如上图所示：tmp[97],以前是48，通过程序验证，如果太短，可能导致接收不了密文，导致密文丢失，再进行解密失败，最后会弹出认证失败的窗口。通过分析加大tmp数组的长度，该问题得到解决。

实验如下图所示：









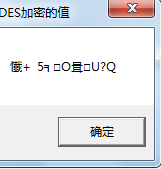
1. 实现DES算法

在COS中实现DES算法（在\Demo U+Key\Cactisrc(ATOM)\app\udisk目录下增加des.h 和des.c文件）。

实验结果如下所示：









# 总结

1. 在做该实验我们收获很多，了解了一些硬件编程的知识
2. 掌握了一些调试程序的方法
3. 了解了硬件和电脑消息传递的一些机制

# 注意事项

* 软件ZsoftDev版本太旧，在windows 7上必须以兼容XP模式进行安装
* 在windows 7 上用ZsoftDev软件进行编译可能会出现问题，我们一直没有解决，最后在xp虚拟机上进行的编译成功
* 在开发板上进行开发时，要注意内存使用情况，根据说明书指示，只有2K的内存，如果在COS中添加代码，特别有比较大的全局变量时，如果超出2K，在硬件连接电脑时就会提示连接错误。
* 开发时是串口线进行调试，可以事半功倍