

ePass1000ND Developer's Guide

FEITIAN

Copyright © 1999-2005, Feitian Technologies Co., Ltd.

All rights reserved.

<http://www.FTsafe.com>

Feitian Technologies Co., Ltd. has made all attempts to make the information in this document complete and accurate. Feitian Technologies is not responsible for any kind of direct or indirect loss or damage from inaccuracies or omissions.

This document will not necessarily reflect all of the updates to the ePass1000ND hardware or software.

Revision History

Date	Version	Memo
Mar. 2005	1.0	1 st Release
Aug 2005	1.1	2 nd version

Feitian Technologies Co., Ltd

Software Developer's Agreement

All Products of Feitian Technologies Co., Ltd. (Feitian) including, but not limited to, evaluation copies, diskettes, CD-ROMs, hardware and documentation, and all future orders, are subject to the terms of this Agreement. If you do not agree with the terms herein, please return the evaluation package to us, postage and insurance prepaid, within seven days of their receipt, and we will reimburse you the cost of the Product, less freight and reasonable handling charges.

1. **Allowable Use** – You may merge and link the Software with other programs for the sole purpose of protecting those programs in accordance with the usage described in the Developer's Guide. You may make archival copies of the Software.
2. **Prohibited Use** – The Software or ePass1000ND hardware token or any other part of the Product may not be copied, reengineered, disassembled, decompiled, revised, enhanced or otherwise modified, except as specifically allowed in item 1. You may not reverse engineer the Software or any part of the product or attempt to discover the Software's source code. You may not use the magnetic or optical media included with the Product for the purposes of transferring or storing data that was not either an original part of the Product, or a Feitian provided enhancement or upgrade to the Product. You may not place the software on a server and make it publicly available.
3. **Warranty** – Feitian warrants that the ePass1000ND tokens and Software storage media are substantially free from significant defects of workmanship or materials for a time period of twelve (12) months from the date of delivery of the Product to you.
4. **Breach of Warranty** – In the event of breach of this warranty, Feitian's sole obligation is to replace or repair, at the discretion of Feitian, any Product free of charge. Any replaced Product becomes the property of Feitian.

Warranty claims must be made in writing to Feitian during the warranty period and within fourteen (14) days after the observation of the defect. All warranty claims must be accompanied by evidence of the defect that is deemed satisfactory by Feitian. Any Products that you return to Feitian, or a Feitian authorized distributor, must be sent with freight and insurance prepaid.

EXCEPT AS STATED ABOVE, THERE IS NO OTHER WARRANTY OR REPRESENTATION OF THE PRODUCT, EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

5. **Limitation of Feitian's Liability** – Feitian's entire liability to you or any other party for any cause whatsoever, whether in contract or in tort, including

negligence, shall not exceed the price you paid for the unit of the Product that caused the damages or are the subject of, or indirectly related to the cause of action. In no event shall Feitian be liable for any damages caused by your failure to meet your obligations, nor for any loss of data, profit or savings, or any other consequential and incidental damages, even if Feitian has been advised of the possibility of damages, or for any claim by you based on any third-party claim.

6. **Termination** – This Agreement shall terminate if you fail to comply with the terms herein. Items 2, 3, 4 and 5 shall survive any termination of this Agreement.

Contact Information

World Wide Web:

www.FTsafe.com

Feitian Technologies Co., Ltd.

Tel: +86-10-62304466

Fax: +86-10-82304477

Email: World.Sales@FTsafe.com

Addr: Bldg 7A, 5th Floor, 40 Xueyuan Road, Haidian District, Beijing
10083, China

Please Email any comments, suggestions or questions regarding this document to us
at World.Sales@FTsafe.com

EC Attestation of Conformity



ePass1000ND is in conformity with the protection requirements of CE Directives 89/336/EEC Amending Directive 92/31/EEC. ePass1000ND satisfies the limits and verifying methods: EN55022/CISPR 22 Class B, EN55024: 1998.

FCC Standard

This device is in conformance with Part 15 of the FCC Rules and Regulation for Information Technology Equipment.

Operation of this product is subject to the following two conditions: (1) this device may not cause harmful interference, and (2) this device must accept any interference received, including interference that may cause undesired operation.

USB



This equipment is USB based.

Table of Contents

Chapter 1 Introduction	1
1.1 Why Use ePass1000ND	2
1.2 Information for ePass1000ND Developers	3
1.3 ePass1000ND Architecture	6
Chapter 2 Installing the ePass1000ND Software	10
2.1 Platforms Supported by ePass1000ND	11
2.2 Installing ePass1000ND SDK	11
2.3 Uninstalling ePass1000ND SDK	17
Chapter 3 The ePass1000ND Console Editor	23
3.1 Using the ePass1000ND Console Editor	24
3.2 Opening ePass1000ND	26
3.3 Turn on/off the LED	26
3.4 Formatting ePass1000ND	27
3.5 Get and Change Access Control Settings	27
3.6 Managing the ePass1000ND File System	28
3.7 Management of the SO and User PIN	30
3.8 Closing ePass1000ND	30
Chapter 4 Distributing ePass1000ND Application	31
Appendix I: Technological Specifications for ePass1000ND	32

Chapter 1

Introduction

Security is an essential requirement of business over Internet today. Clients certainly realize this need for safety in Ecommerce as in any other business field. And ePass1000ND is the key to protect and provide E-business with reliability, convenient operation and many more.

This reference guide is intended to assist you in integrating ePass1000ND security into your applications or customer solutions.

This chapter covers the following topics:

- ✓ Product Overview
- ✓ Why Use ePass1000ND
- ✓ Information for ePass1000ND Developers
- ✓ ePass1000ND Architecture

Product Overview

ePass1000ND is a fully portable, driverless and low cost device that connects to the Universal Serial Bus (USB) port of a Personal computer and normally used under Windows 98 SE or above platform (ePass1000ND may also be used in MAC and Linux operating systems. See Appendix I.) ePass1000ND does not require an additional power supply or any card-reader and is only about the size of your thumb.

ePass1000ND is the ideal solution to user authentication and access control operation. Applications can benefit from ePass1000ND's built-in MD5-HMAC algorithm unit, which provides a powerful challenge/response mechanism for authentication services. The challenge/response authentication model is more secure than the traditional user-name & password model because in challenge/response the "shared secret" information is never exposed during the authentication process.

ePass1000ND is also a perfect solution for portable storage of sensitive information. Digital certificates, private keys, passwords, credit card numbers and other security credentials may be safely and conveniently stored (on) ePass1000ND and taken with you.

1.1 Why Use ePass1000ND

The benefits that ePass1000ND can provide to your organization or application center around protecting and securing network communications.

- ✓ Passwords alone are not strong security. Users may share passwords, or write them down near their PCs. Passwords can be intercepted by network sniffing devices and there are free software programs on the Internet to help hackers crack user passwords. ePass1000ND is a tremendous solution for two factors security measures. Here two factors authentication system means user needs to possess the ePass1000ND device beside knowing its PIN or password.
- ✓ Credit card numbers, bank account numbers, digital certificates and other security credentials can be damaged or deleted by computer viruses. Those can also be stolen by hackers or otherwise compromised if they are left in the insecure Personal Computer environment. The ePass1000ND on-board MD5 Hash algorithm guarantees that information stored in it is far more secure than it would be in the PC environment.
- ✓ ePass1000ND need not a driver and is plug & play device. It's a low cost and portable hardware that you can use to store and carry your security credentials from home to office PC or anywhere safe and conveniently.

1.2 Information for ePass1000ND Developers

Developers who want to use the low level API interface of ePass1000ND to add access control and entity authentications to their programs, and manage memory inside ePass1000ND to store sensitive information.

ePass1000ND has a hierarchical file system similar to those found on PCs, except that the ePass1000ND file management system is more specific regarding classifications of permission and security.

Each ePass1000ND unit has a unique hardware serial number. The serial number may serve as a unique identifier for ePass1000ND enabled applications or administrative functions.

ePass1000ND uses the MD5 Hash algorithm to protect passwords and other security information. The hash algorithm keeps the user passwords or other security information safe and unreadable directly from ePass1000ND device. Moreover the algorithm verifies user validity by calculating password saved in hardware and random character string inputted. This procedure is explained in more detail below.

In Diagram 1.1 ePass1000ND is shown connected to a simplified network.

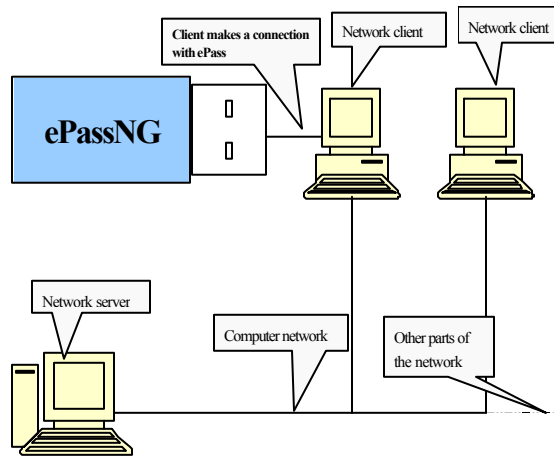


Diagram 1.1

The client machine will first make a request to the server for identification services. The server will then extract the password (or “Key”) that corresponds to the user name. (See Diagram 1.2)

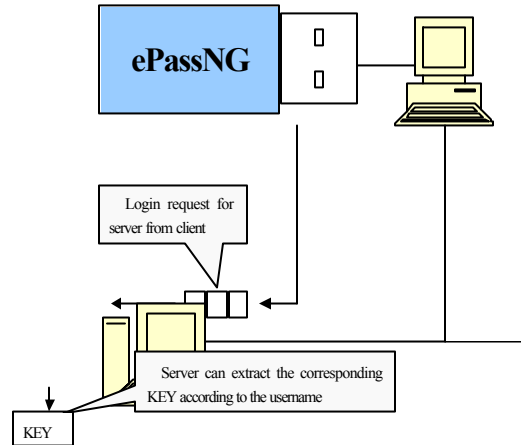


Diagram 1.2

When the server receives the login request from the client it replies with a random character string X sent back to the client machine. The client machine forwards the random character string to ePass1000ND. The server also extracts the password corresponding to the user name from its database. The server performs a calculation involving the random character string X that it sent to the client and the user's password. The result is referred to as Rh. (See Diagram 1.3)

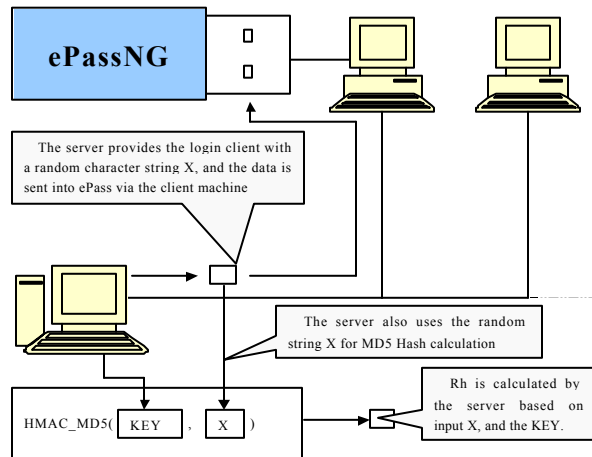


Diagram 1.3

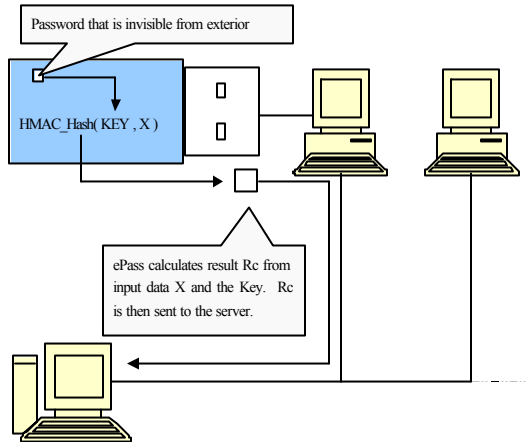


Diagram 1.4

ePass1000ND uses the random character string from the server and the password in its own file system to calculate a result, R_c . The calculated result R_c may then be returned to the server. (See Diagram 1.4)

The server compares the two results, R_h (server) and R_c (client). The user is authenticated to the network if the two results match. The logon attempt is denied if the two results don't match. (See Diagram 1.5)

The results (R_h and R_c) change according to the value of the randomly generated character string and the password may not be induced from the calculated results. With ePass1000ND in place then, the password is kept safe and secure inside the ePass1000ND container, even during the authentication process.

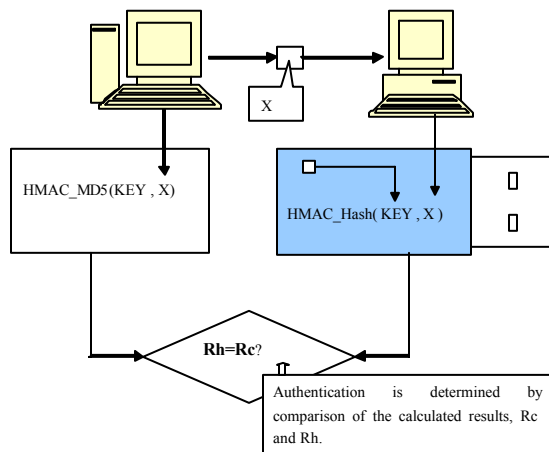


Diagram 1.5

1.3 ePass1000ND Architecture

This section describes the basic structure and concepts pertaining to ePass1000ND.

1.3.1 Security States

The ePass1000ND hardware supports a three level security structure: Security Officer (SO), User and Guest.

Security Officer (SO) State

SO is the most privileged security state. SO requires a Personal Identification Number (PIN). This state allows changes to sensitive parameter settings and token initialization. If the ePass1000ND SO PIN is lost or forgotten it cannot be retrieved. In that case the device must be returned to Feitian where it will be reset to factory defaults.

User State

The User state also requires entry of a PIN. ePass1000ND may be configured to allow a user to reset or change the User PIN. Personal information stored in ePass1000ND is normally accessed in the User state. There is a hardware counter in ePass1000ND to track user logon failure. The counter decrements each time the user fails at an attempt to log onto ePass1000ND. The user is locked out of ePass1000ND if the counter decrements to zero. The SO PIN would then be needed to reset the hardware counter.

Guest State

The Guest state is the default state for access to ePass1000ND. Guest state allows read-only access to public information only.

1.3.2 Device Attributes

Serial Number

Each ePass1000ND unit has a 64-bit globally unique serial number. The serial number is burned into the unit at the factory and may be used by applications for quick reference to a specific unit.

LED

Each ePass1000ND is equipped with a Light Emitting Diode (LED) that can be controlled by applications. And its status can indicate if the driver is successfully installed.

1.3.3 Cryptographic Services

Random Number Generator

ePass1000ND can generate random numbers in hardware. Random numbers may be used when creating authentication digest code as well as seed for other cryptographic functions.

MD5 algorithm

The MD5 algorithm is an industry standard hashing algorithm that takes a message of arbitrary length as input and produces a 128-bit message digest as output. The output digest is believed non-reversible, meaning that no one can figure out the input data from the output MD5 digest.

MD5 HMAC

Although much more reliable than simple checksum methods, MD5 does not provide a data integrity check because anyone can alter the input data and generate a corresponding output digest. Obviously, the hashed value needs to be protected. That is the target of the Hashed Message Authentication Code (HMAC). HMAC can be used with the MD5 hash algorithm and a secret key to authenticate a message or collection of data. ePass1000ND supports this industry standard method to provide a secure way for end users or applications to be authenticated without exposing their secret keys.

TEA

TEA is an excellent encryption algorithm. This algorithm is more simple than DES (Data Encryption Standard). TEA has high anti analyze ability and it's faster than DES. The encryption key can be 128 bits for 64 bits data. TEA is so secure.

1.3.4 File System

ePass1000ND has a built-in file system which can be fully managed from the API library. The file system is a flexible method for storing, protecting and retrieving data in ePass1000ND.

The ePass1000ND file system has the following attributes:

- ✓ 2 levels of directories.
- ✓ File sizes are pre-allocated upon creation. The size of a file cannot be changed after it has been created. To change the size of a file, you must first delete the file and then recreate it.

- ✓ Free space is calculated on the entire token.
- ✓ Files are named with a digital ID instead of a character string.
- ✓ The ePass1000ND file system supports both 32-bit and 16-bit directories and file IDs.
- ✓ The scope of directory IDs is local to the current directory.
- ✓ The scope of file IDs is local to the current directory.
- ✓ ePass1000ND supports named directories, not longer than 32 bytes. Names are global to the entire file system and are case-sensitive. Duplicates are not permitted.
- ✓ ePass1000ND supports 16-byte Globally Unique Identifiers (GUIDs) for directories. GUIDs are global to the file system. Duplicates are not permitted.

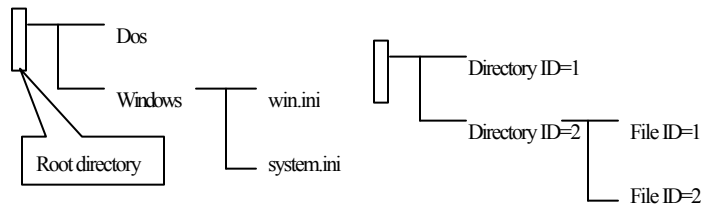


Diagram 1.6

Notes:

The Master File (MF) is the root directory and uses the ID of zero.

The MF may contain files and directories.

Directories are defined by 32-bit ID: 1... 0xFFFFFFFF (ID of 0 is reserved for MF).

File IDs are defined by 32-bit ID: 0...0xFFFFFFFF

Applications should use 16-bit file IDs (0...0xEFFF), and 16-bit Directory IDs (1...0xEFFF) to ensure future compatibility with a new device. Applications should avoid using 32-bit IDs.

Type of Files

The ePass1000ND file system uses two types of files:

Type	Descriptions
DATA	Any variable length binary data
KEY	Data used for cryptographic operations

File Access Settings

Each file on the ePass1000ND has three access types. Each file has its own access setting.

Access Types	File Types	
	Data	KEY
Read	Control	Prohibited
Write	Control	Control
Crypt	Meaningless	Control

Note: Read and Write access are the functions that control ePass1000ND data transfers. Crypt access is meaningless for DATA file type. Cryptographic operations on KEY files are performed inside of ePass1000ND.

File Access Rights

Attributes	Description
ALWAYS	Access is always permitted. Security state is ignored.
NEVER	Never grant access. Security state is ignored.
PIN	Access is granted in User State or SO State.
SO PIN	Access is granted only in SO State.

Note: Multiple applications accessing the same ePass1000ND hardware should coordinate their use of directory IDs and file IDs to avoid collision.

1.3.5 Multi ePass1000ND Token Applications

ePass1000ND supports multi-token applications; multiple ePass1000ND tokens may work together on the same computer.

Chapter 2

Installing the ePass1000ND Software

The ePass1000ND software must be properly installed before it's being used with the computer. This chapter will provide instructions for installing the ePass1000ND software in Windows environment.

- ✓ Platforms Supported by ePass1000ND
- ✓ Installation of the ePass1000ND SDK

2.1 Platforms Supported by ePass1000ND

The following platforms are currently supported by ePass1000ND:

- ✓ Windows 98SE
- ✓ Windows ME
- ✓ Windows 2000
- ✓ Windows XP
- ✓ Linux
- ✓ MAC

Note: Please first logon with Administrator privileges before installing the ePass1000ND software.

2.2 Installing ePass1000ND SDK

The ePass1000ND SDK software will install the following components (on) in your computer:

- ✓ ePass1000ND Console Editor
- ✓ ePass1000ND SDK Documents
- ✓ ePass1000ND Header Files and Libraries
- ✓ ePass1000ND Sample Codes

Note: Please uninstall any earlier version of the ePass1000ND SDK before installing the new version.

The installation of ePass1000ND SDK begins by inserting the installation CD into the CD-ROM drive and selecting the setup.exe file

The following window should appear, please choose a setup language here, see Figure 2.1:

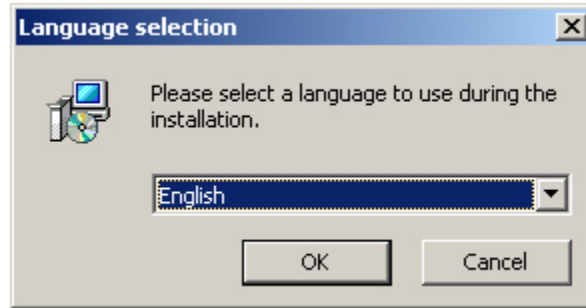


Figure 2.1

Here we choose “English” for demonstration. Click “OK” to continue, then the installation welcome window will appear.

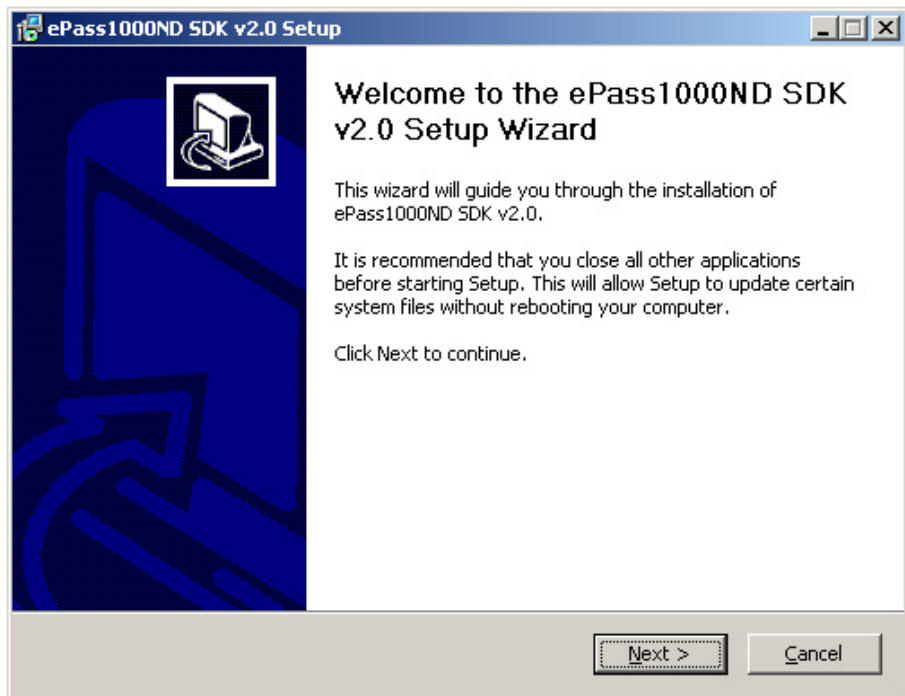


Figure 2.2

Click the “Next” button to continue. The license window will appear.

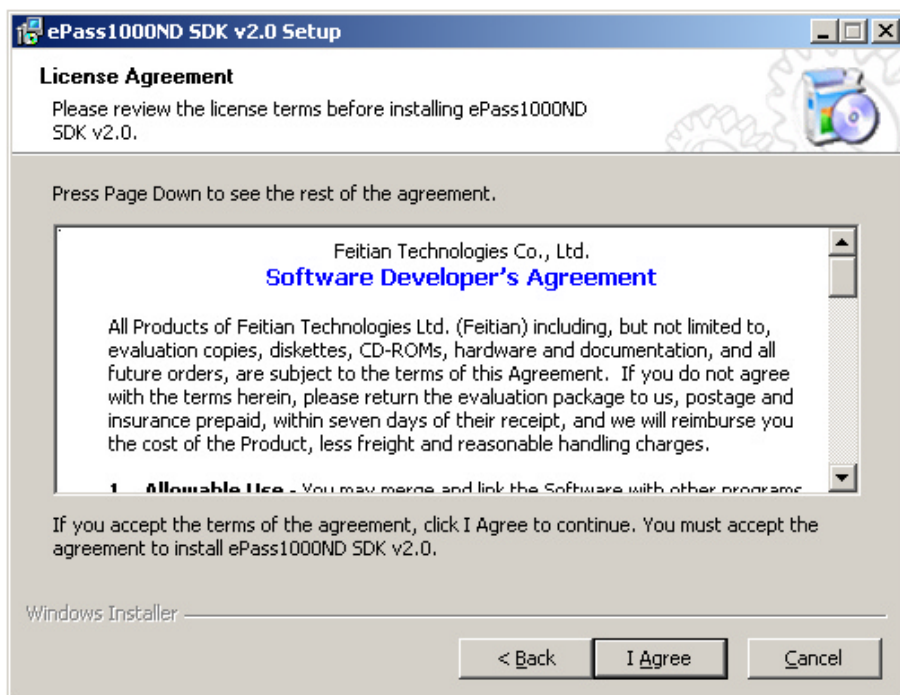


Figure 2.3

Carefully read the License Agreement. Click “ I Agree” if you agree with the terms therein.

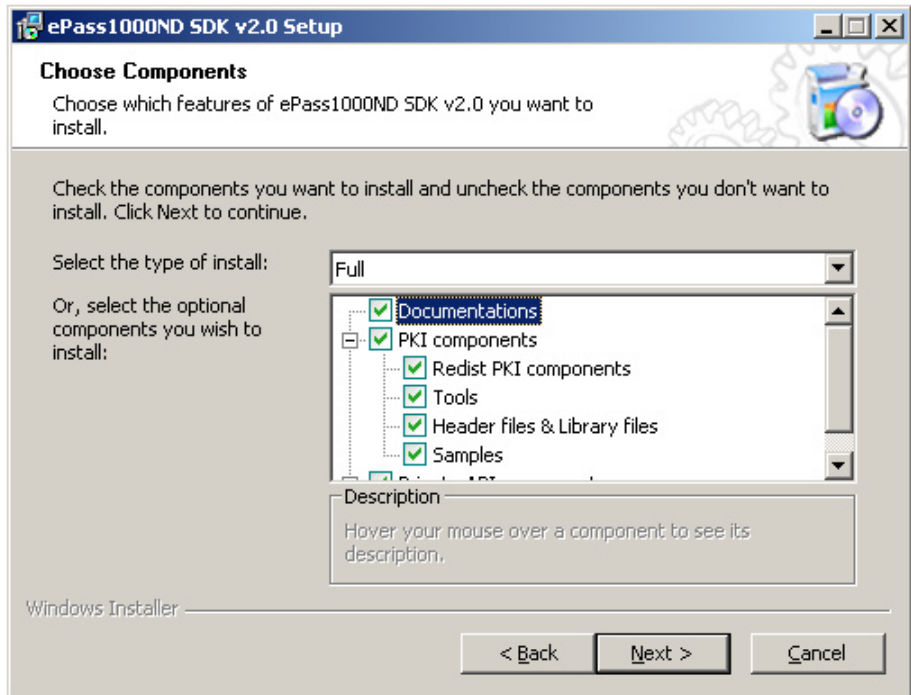


Figure 2-4

Installation program lists the type of install and optional components. Select a installation type from the drop down list. You may select components to install and choose “Next”.

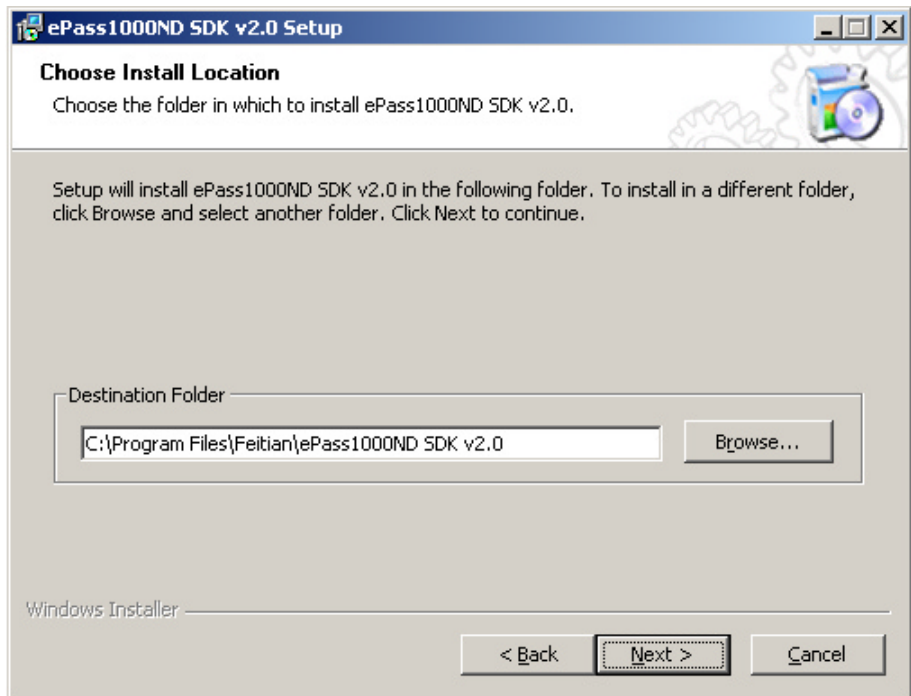


Figure 2.5

You may click “ Browse...” to change the installation path. Click “ Next” to continue.

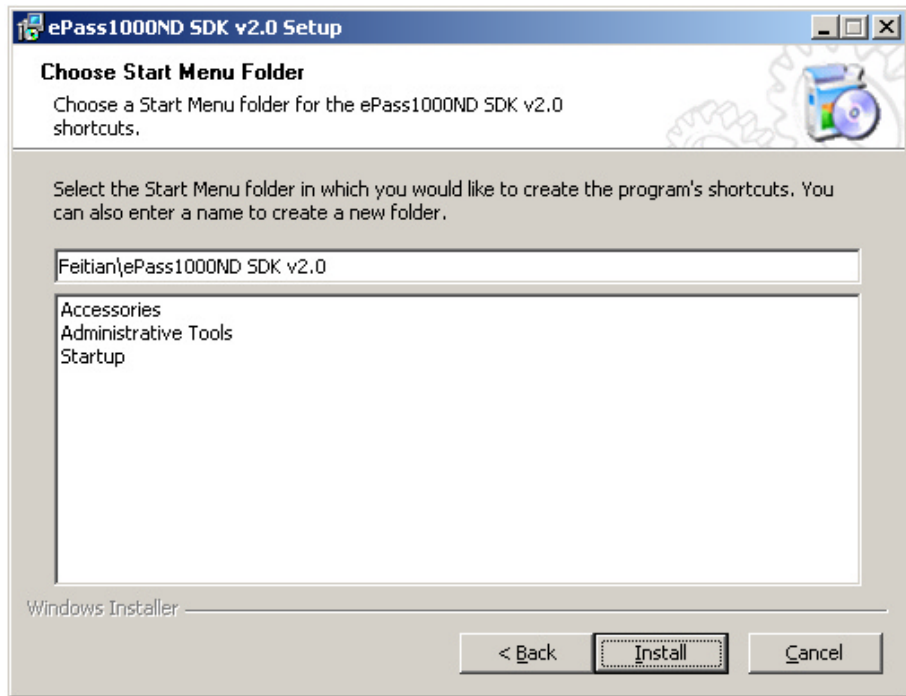


Figure 2.6

Click “ Install” and the ePass1000ND SDK v2.0 will be installed on your computer.

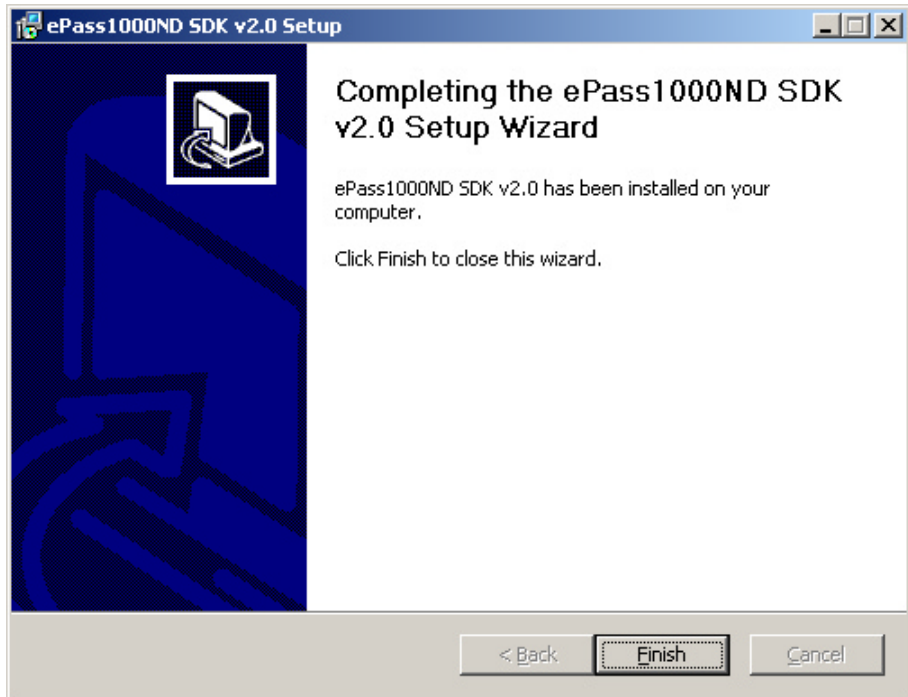


Figure 2.7

Click “ Finish” to complete the ePass1000ND SDK v2.0 installation.

2.3 Uninstalling ePass1000ND SDK

You may uninstall ePass1000ND SDK v2.0 in the same way that you uninstall other applications, from Control Panel and Add/Remove Programs. See Figure 2.8. Choose “ePass1000ND SDK v2.0 (Remove only)” in “ Add/ Remove Programs” to remove it.

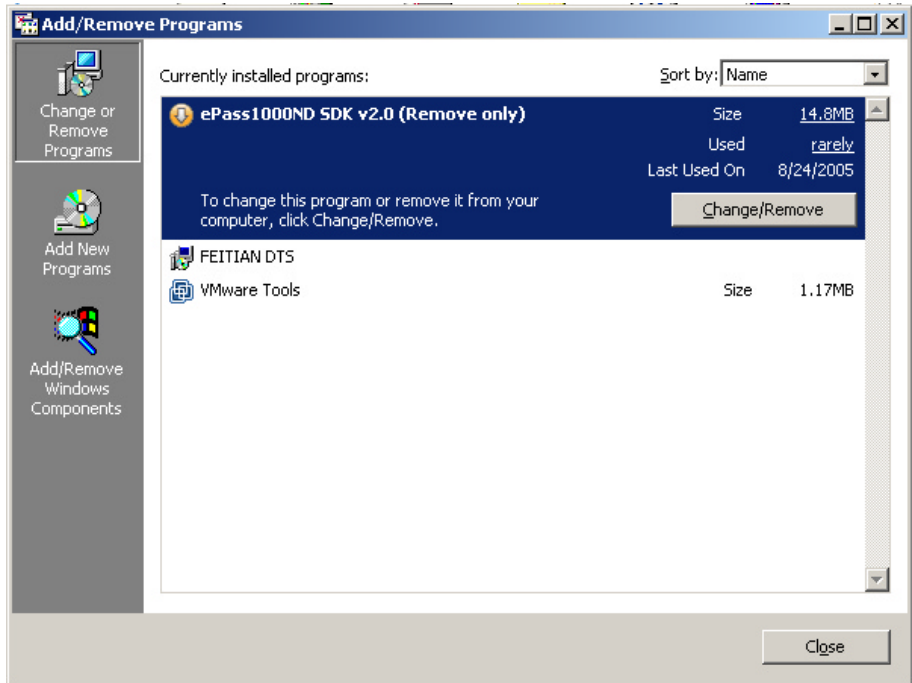


Figure 2.8

Then click “ Change/Remove” to continue. Following figure will appear.



Figure 2.9

Click “ Next” to continue. See Figure 2.10.

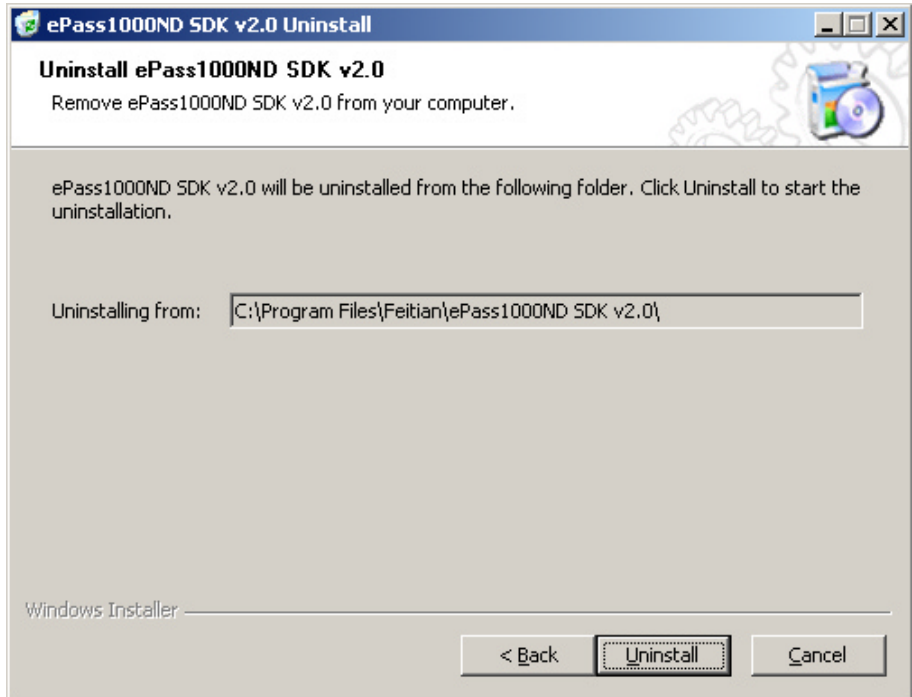


Figure 2.10

Click “Uninstall” to start un-installation process.

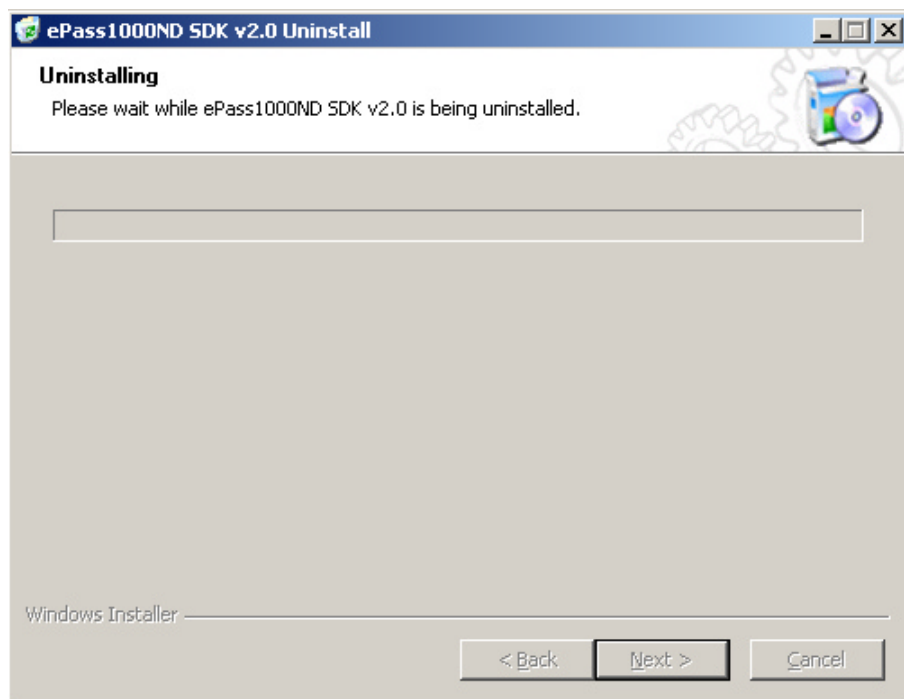


Figure 2.11

In figure 2.11 un-installation progress is displayed.

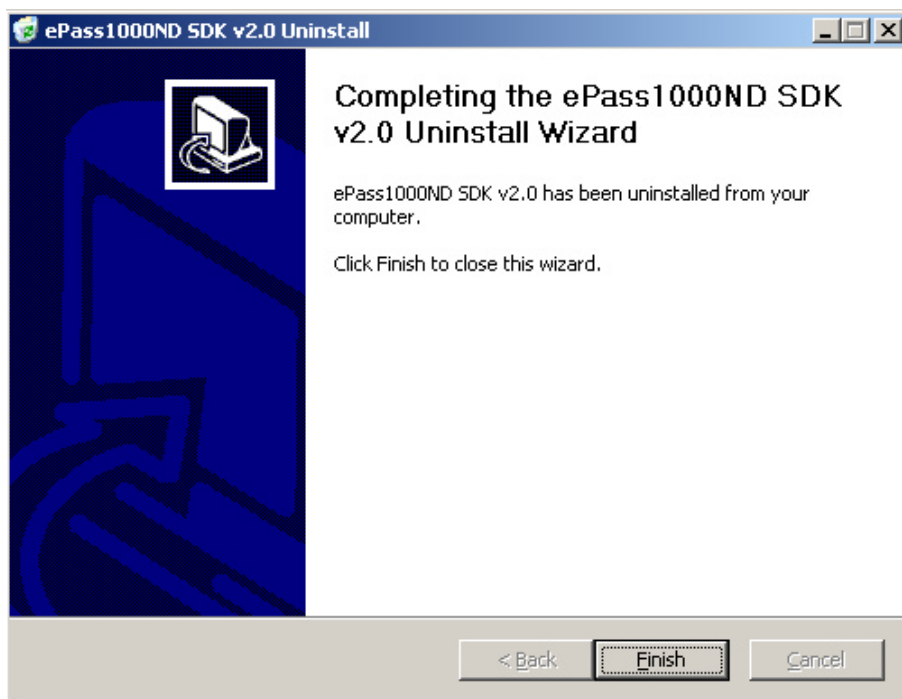


Figure 2.12

Click "Finish" to complete un-installing ePass1000ND SDK v2.0.

Chapter 3

The ePass1000ND Console Editor

In this chapter we will cover some of the common ePass1000ND functions and demonstrate the ePass1000ND Console Editor. The ePass1000ND Console Editor is programmed with ePass1000ND' C language API and may be used to manage many of the ePass1000ND functions without extra programming.

To know more about ePass1000ND API, please refer to “ePass1000ND Developer’s Reference” . This document describes ePass1000ND API in detail.

Note: PKI developers may want to skip this chapter unless they want to improve their over-all familiarity with the product.

This chapter discusses:

- ✓ Using the ePass1000ND Console Editor
- ✓ Opening ePass1000ND
- ✓ Turn on/off the LED
- ✓ Formatting ePass1000ND
- ✓ Get and Change Access Control Settings
- ✓ Managing the ePass1000ND File System
- ✓ Management of the SO and User PIN
- ✓ Using the encryption function
- ✓ Closing ePass1000ND

3.1 Using the ePass1000ND Console Editor

The ePass1000ND Console Editor was programmed with ePass1000ND's C language API set. It may be used to invoke all of the ePass1000ND functions. Each command issued in the editor corresponds to one or more C language API functions. You may use it to explore all of the ePass1000ND hardware functions and programming interfaces.

The most common ePass1000ND programming interfaces are integrated to the Console Editor. You only need to use the Console Editor for most of the task unless your applications need low-level control of ePass1000ND (i.e. storage management, user authentication or cryptographic operations).

The following table lists the functions that correspond to each operation in the Console Editor:

ePass1000ND Operations	ePass1000ND C Language API Functions
Open the ePass1000ND	epas_OpenDevice: Open an ePass1000ND that is connected to the computer. You may open the specified ePass1000ND by system enumerating order, hardware serial number, etc.
Turn on/off the LED	epas_SetProperty: The LED can be used to indicate the operating state of ePass1000ND.
Format ePass1000ND	epas_DeleteDir: Delete all directories and files in ePass1000ND. epas_SetProperty: Set or change the name of the token.
Get and Change Access Control settings	epas_GetProperty: Retrieve access control settings of ePass1000ND. epas_SetProperty: Change access control settings of ePass1000ND.
Manage the ePass1000ND File System	epas_CreateDir: Create directories. epas_DeleteDir: Delete directories. epas_ChangeDir: Change current directory. epas_CreateFile: Create files. epas_DeleteFile: Delete files.

	epas_OpenFile : Open file for further operation. epas_Read : Read data from files. epas_Write : Write data to files. epas_CloseFile : Close a file.
Manage SO and User PIN	epas_ChangeCode : Change User or SO PIN. epas_Verify : Verify User or Security Officer PIN.
Use Cryptographic functions of ePass1000ND	epas_GenRandom : Get a random number from the random number generator inside ePass1000ND. epas_HashToken : Perform an MD5 calculation. epas_MD5_HMAC : Perform an MD5 HMAC calculation.
Close ePass1000ND	epas_CloseDevice : Close ePass1000ND.

To invoke ePass1000ND Console Editor, you may open the Windows Start menu >>Programs>>Feitian Technologies>>ePass1000ND SDK v2.0>>Use private API, and then select Console Editor. The user interface for the Console editor looks like the following figure.

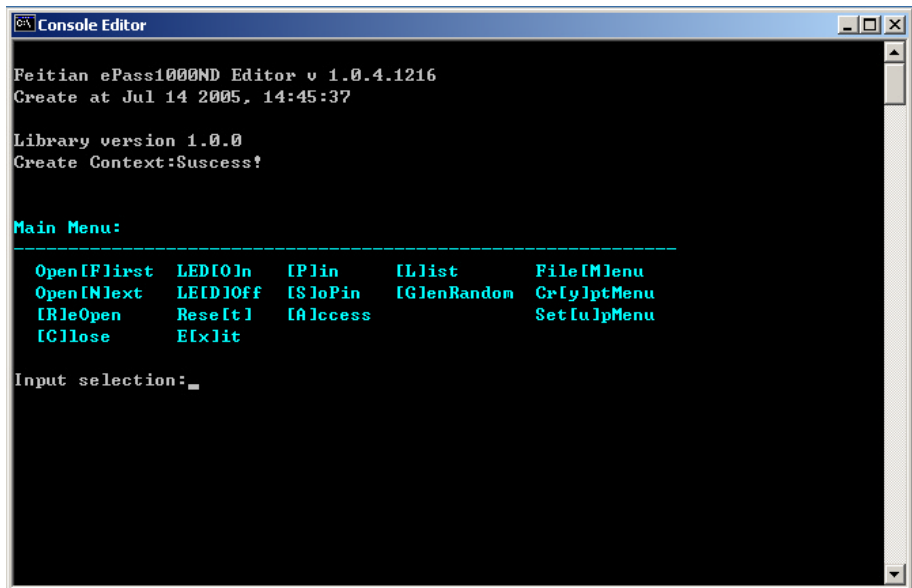


Figure 3.1

3.2 Opening ePass1000ND

ePass1000ND must first be Opened before you can access its functions. ePass1000ND is a USB device, and in theory there could be as many as 32 ePass1000ND devices attached to a computer. The C language API supports several ways to find and open ePass1000ND devices. For detail information regarding the ePass1000ND Open function, `epas_OpenDevice()`, please refer to the ePass1000ND Developer's Reference.

Open the first ePass1000ND device: After inserting ePass1000ND into the USB port, type 'F' and press the 'Enter' key in the Console Editor. The ePass1000ND Console Editor will attempt to open the first ePass1000ND connected to the system. The error message, "Device not found" will be displayed if no ePass1000ND is found. The Editor will display the screen seen below upon successful completion of the operation, see Figure 3.2.

```

Console Editor
[Ctrl]lose      E[x]it

Input selection:f
Open device:Success?

=>> Firmware Version: 1.06
=>> Product Code: 10
=>> Capabilities: 3
=>> Total memory size: 8192 bytes
=>> Free memory space: 7336 bytes
=>> Max directory levels: 2
=>> File system type: 1
=>> Friendly token name: ePass1000ND
=>> Hardware serial number: 0x42F81BC15722B44C

Main Menu:
-----
Open [F]irst  LED [O]n  [P]in  [L]list  File [M]enu
Open [N]ext  LED [O]ff  [S]loPin  [G]enRandom  Cr [Y]ptMenu
[R]leOpen    Rese [t]  [A]ccess  Set [u]lpMenu
[Ctrl]lose   E [x]it

Input selection:_

```

Figure 3.2

3.3 Turn on/off the LED

The ePass1000ND LED is useful to indicate the state of ePass1000ND. The developer may use the LED to indicate when data is being sent to or retrieved from ePass1000ND. This could help prevent the unintended loss of data. ePass1000ND

should not be removed from the USB port during a data transfer.

To turn on the LED, input 'O' in 'Input selection'. To turn off the LED, input 'D'.

If the ePass1000ND LED is not working properly, it may indicate bad contact, or that the driver was not successfully installed.

3.4 Formatting ePass1000ND

The formatting operation will empty ePass1000ND storage space and initialize it. Feitian recommends that ePass1000ND be initialized before distribution. All information inside ePass1000ND will be deleted after formatting and cannot be recovered again. Please perform this operation carefully.

The SO PIN must be verified before the format operation (**can be**) is invoked. This requirement ensures that only Security Officer can format the key.

- ✓ **To verify the SO PIN:** Input 'S' and press 'Enter', then input the SO PIN and press 'Enter' again. The message "Verify SO-PIN successfully" will appear if the SO PIN, entered was correct.
- ✓ **To format ePass1000ND:** Input 'U' and press 'Enter', then input 'D' to format from the "Setup Menu".

You may assign a friendly name to ePass1000ND, just like you would label a logical drive.

- ✓ **To set a token name:** access the "Setup Menu", input 'T' and then enter the token name (Maximum of 31 characters).

3.5 Get and Change Access Control Settings

Access control settings are critical for working with ePass1000ND.

To retrieve current access control settings from ePass1000ND: Go to the "Main Menu" of ePass1000ND Console Editor and input 'A'. See the example of an ePass1000ND access control settings display in Figure 3.3.

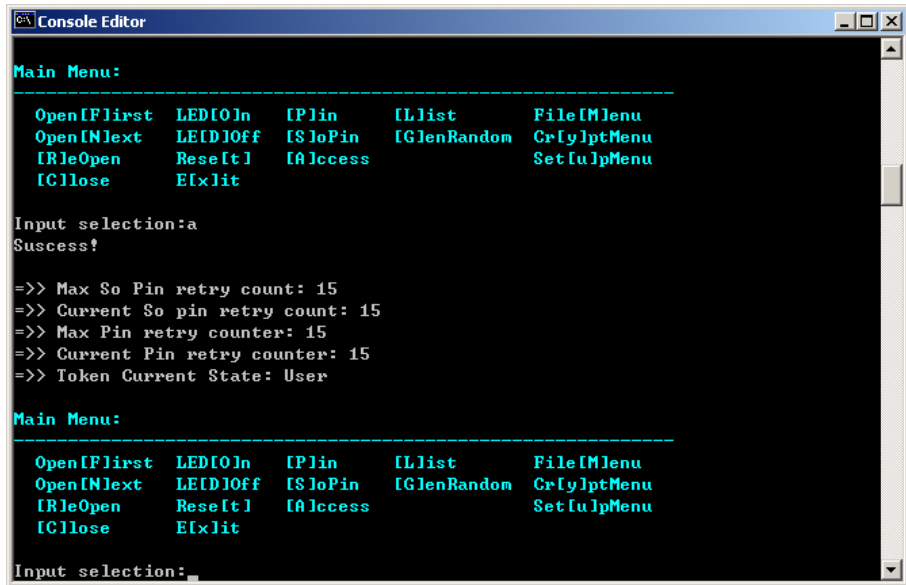


Figure 3.3

To set the access control settings of ePass1000ND: from the “Setup Menu” of ePass1000ND Console Editor, input ‘A’. Then set the value of the access control settings.

3.6 Managing the ePass1000ND File System

As we discussed in previous chapter, ePass1000ND has a two-level directory file system. You may create, delete, read and write files, and change access control settings to the files. To operate on directories or files, first access the “File Menu” of the ePass1000ND Editor.

In the file system of ePass1000ND a directory can be identified by the long ID (32 bits), short ID (16 bits), name (ASCII string) or by the GUID (128 bits). Your application may choose one or several of these four ways to identify a directory.

To create a directory with an ID: Go to “File Menu” of ePass1000ND Editor and input ‘D’. Then input the ID of the directory.

To create a directory with a name string or GUID: Go to “File Menu” of ePass1000ND Editor and input ‘A’. Then input the name string or/and GUID.

To view the settings of all files and directories: Input ‘L’. See Figure3.4 below:

```

E:\epass_proj\epassND_M8\source\output\Debug\epassNDEdit.exe

C[h]Dir    [L]ist    Create[D]ir    Create[F]ile    Create[A]pp
De[D]ir    De[le]teFile
[O]pen     [R]ead     [W]rite        [C]lose
Cr[yp]tMenu    Set[ul]Menu    E[x]it

Input selection:1

DirID      FileID     Type      Read      Write     Delete    Crypt      Size
-----
0000       FFFF      DATA     ANYONE    ANYONE    ANYONE    ANYONE     512
0000       FFFE      DATA     ANYONE    ANYONE    ANYONE    ANYONE     32
0000       F000      DIR       NONE      NONE      NONE      NONE       0
                                Name: "ASP_DEMO"
F000       0001      MD5       NONE      ANYONE    ANYONE    USER      16
F000       0002      MD5       NONE      ANYONE    ANYONE    USER      16
0000       0000      FREE     ANYONE    ANYONE    ANYONE    ANYONE     7544

File Menu:
-----
C[h]Dir    [L]ist    Create[D]ir    Create[F]ile    Create[A]pp
De[D]ir    De[le]teFile
[O]pen     [R]ead     [W]rite        [C]lose
Cr[yp]tMenu    Set[ul]Menu    E[x]it

Input selection:

```

Figure 3.4

To change the current directory. Go to “File Menu” of ePass1000ND Editor and input ‘H’. Input the ID of the directory.

Only one file may be opened or operated on at a time. The currently opened file will close automatically if a new file is opened. There are three file access settings that may be applied to the binary or crypt file types: read access privilege, write access privilege and cryptography access privilege. (See the File Access Settings table in section 1.4.)

When a file is created its size must be specified. The size cannot be changed after the file is created. The file type must also be specified and cannot be changed later either.

To create a file: First enter the directory path where you want to store the file. Then from the “File Menu” of the ePass1000ND Editor and input ‘F’. Then input the file ID, file size, file type and file access settings. See Figure 3.6. Feitian recommends to use a short ID when creating a file. Some file IDs are reserved by Feitian, please refer to ePass1000ND Developer’s Reference for detail.

To delete a file: Go to “File Menu” of the ePass1000ND Editor and input ‘E’. Then input the file ID.

3.7 Management of the SO and User PIN

The application should determine if the user will be required to enter an SO or User PIN. If security is not a concern or if the application has some other means of protecting the data, files may be set with an attribute that allows it to be accessed without PIN.

To change the SO PIN: Go to “Setup Menu” of ePass1000ND Editor, input ‘S’ and press ‘Enter’. Then input the current SO PIN and a new SO PIN.

The factory initialized (default) SO PIN is “rockey”. Before distribution of ePass1000ND you should change the SO PIN.

To change the User PIN: Go to “Setup Menu” of ePass1000ND Editor, input ‘P’ and press ‘Enter’. Then input the current User PIN and a new User PIN. The factory initialized User PIN is “12345678”. Before distribution of ePass1000ND you should change the User PIN.

3.8 Closing ePass1000ND

ePass1000ND is designed to be accessed by only one application at a time. If your application opens ePass1000ND, it cannot be accessed by any other application until your application closes ePass1000ND.

To close ePass1000ND: Go to “Main Menu” of the ePass1000ND Console Editor, input ‘C’ and press ‘Enter’.

Chapter 4

Distributing ePass1000ND Application

ePass1000ND library files must be properly installed before other application programs or drivers access it.

To program with the ePass1000ND API, the following files are required:

Files	SDK Paths
FT_ND_API.h	/Include
FT_ND_API.lib	/Lib
FT_ND_API.dll	/Lib (Must be available at run time)
FT_ND_SC.dll	/lib(Must register the DLL for control running)
FT_ND_MOD.dll	/lib(Must register the DLL for control running)

Appendix I:

Technological Specifications for ePass1000ND

Supported Operating Systems	Windows98SE/ME/2000/XP; Mac OS 10.3 ; Linux 2.6 or above
Memory Size (by Model)	1.4k
On-Board Security Algorithms	MD5, TEA
Chip Security Level	Secured and Encrypted Data Storage
Power Dissipation	< 250 mW
Operating Temperature	0 C to 70 C (32 F to 156 F)
Storage Temperature	-40 C to 85 C (-40 F to 185 F)
Humidity Rating	0 to 100% without condensation
Connector Type	USB type A (Universal Serial Bus)
Casing	Hard Molded Plastic, Tamper Evident
Memory Data Retention	At least 10 years
Memory Cell Rewrites	At least 100,000 times