FEITIAN

Rockey7.NET Production Tool Help

V2.0

Feitian Technologies Co., Ltd.

Website: www.FTsafe.com

Revision History:

Date	Revision	Description
Sep. 2009	V1.0	1st release of the document
Sep. 2010	V2.0	2nd release of the document

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 hardware 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.

3. Warranty – Feitian warrants that the hardware 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.

Contents

Chapter 1. Overview 1

1.1 Introduction to Rockey7.NET 1

1.1.1 ROCKEY7.NET Features 3

1.1.2 Supported Environments 4

1.2 Introduction to ROCKEY7.NET Production Tool 4

1.3 Production Process 4

Chapter 2. Using ROCKEY7.NET Initialization Tool 6

2.1 Generating New Rockey7.Net License File 6

2.2 Editing Existing Rockey7.Net License File 9

2.3 Initializing Rockey7.Net User Dongle 10

Chapter 3. Using ROCKEY7.NET Production Tool 13

3.1 Producing User Dongle 13

3.2 Setting Production Options 15

Chapter 1. Overview

1.1 Introduction to Rockey7.NET

ROCKEY7.NET is the ideal solution to protect .NET application against piracy. The 32bits high performance smart card chip based hardware with built-in .NET virtual machine allows .NET applet being executed within the device. Software developers can download .NET applet to the device and make it work with outer .NET application. The secure communication technique is applied to protect the data transferred between the device and host machine. Each ROCKEY.NET dongle can run multiple .NET applets simultaneously so as to protect multiple .NET application at the same time. The design of ROCKEY7.NET follows the up-to-date .NET technology. Its stable and effective performance provides excellent protection to Microsoft .NET applications.

With the advent of .NET technology, its simple, powerful and effective features are widely accepted by the users in various fields. The .NET program will be compiled into an IL (Intermediate Language) scripts and being rendered in different .NET runtime environment. The same IL scripts can be used in different platforms, which can significantly reduce the development and maintenance cost. However, the simple structure of IL language makes it too easy to be decompiled into source code. Thus the .NET program is easy to be cracked. How to protect .NET IL scripts is the most important topic of .NET application protection. Normally, the following techniques can be applied to achieve the goal:

- Anti-debugging

  The protected .NET application has the feature that it cannot be debugged or audited by .NET debugger tools.

- JIT runtime decode

  The instructions of .NET application will be decoded before it is rendered by JIT in runtime. The.NET application in memory is not complete.

- Function body encryption

  The body of function will be encrypted with cryptographic algorithm. The .NET application is stored in cipher form.

- Keyword confusion

  The keywords in the .NET application like function name, variable name will be replaced with irrecognizable code. So even if the application is decompiled, it is not easy for cracker to understand the meaning of source code.

- Communication encryption

  For the applications involving communication with hardware, the transfer datagram is encrypted and

protected against interception.

All the above methods can protect the .NET application. But they have a common feature. The protected application is completely stored in the computer, regardless it is stored in cipher or plain format. So it is easy for experienced cracker to obtain the complete original .NET application by using decryption tools. However, ROCKEY7.NET is designed to terminate the risk coming in this situation.

ROCKEY7.NET software protection solution provides a unique “function migration” method to protect the .NET application. It can download a part of the functions in original application to its built-in .NET virtual machine and execute the code in its .NET runtime environment. The internal .NET applet can cooperate with outer application to achieve the same functionality as the unprotected application. The ROCKEY7.NET built-in .NET virtual machine runtime environment makes this innovative design achievable. Worldwide, only a few manufactures can develop .NET virtual machine runtime environment, Feitian is one of them. Feitian .NET virtual machine runtime environment has been certified by Microsoft. Based on 32bits high performance smart card chip, Feitian .NET smart card provides a stable, effective platform for .NET applets being executed onboard. It is the perfect environment for .NET smart card applications. ROCKEY7.NET solution is based on Feitian .NET smart card. The Enveloper can intelligently analysis the target .NET application, migrate the core part of the application to the Feitian .NET card automatically. The protected application is not complete. A core part of the program is protected by the Feitian .NET smart card. The migration concept can guarantee that the outer application can never be fully reversely decom3piled.

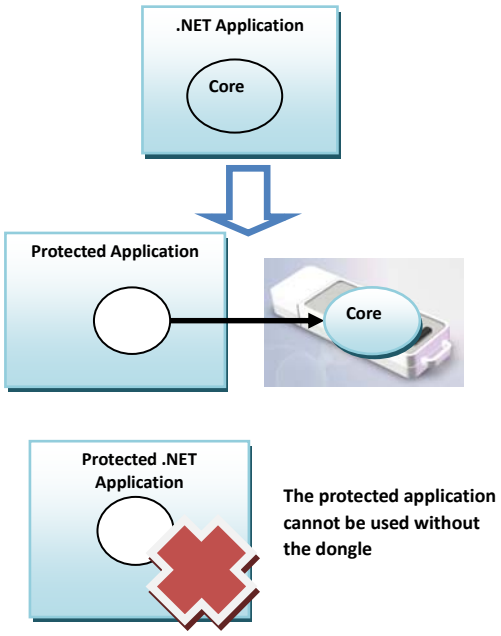

Figure 1.1 ROCKEY7.NET Software Protection Principle

ROCKEY7.NET provides many utilities to help developer maintain the entire software life cycle. The Enveloper, initialization tool, production tool, remote update tool can help developer protect and control the software remotely. Normally, there are several roles in software maintain cycle: administrator, developer, production staff,

sales staff and support staff. For each role, there are according tool to help developer perform the operation. This design can effectually separate the responsibility and task for each different role.

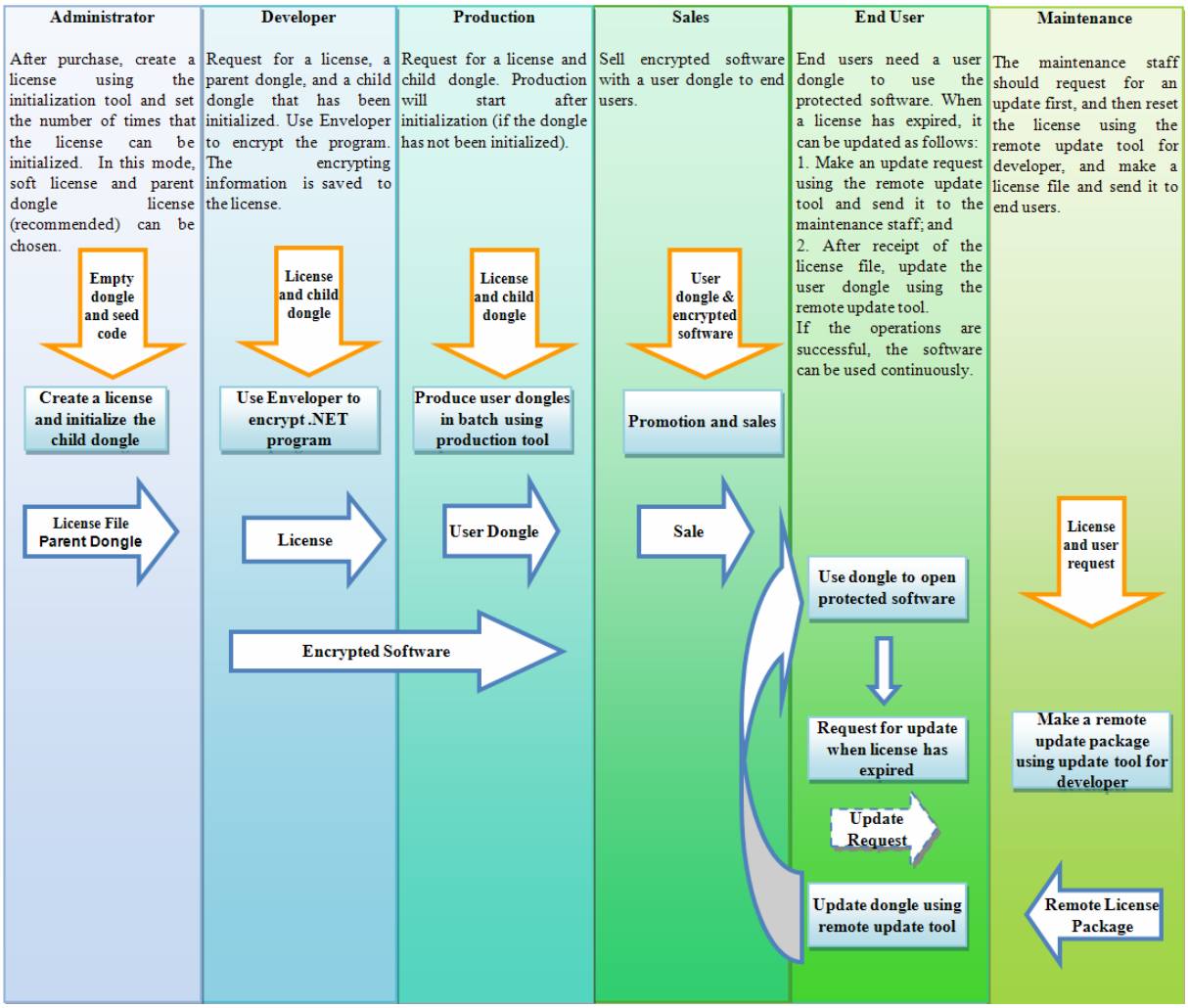

Figure 1.2 ROCKEY7.NET Software Maintenance Flow

1.1.1 ROCKEY7.NET Features

- 32-bit high performance smart card
- Globally unique hardware serial number
- User-defined 4-byte product ID and 8-byte management password
- No user installation of driver needed: CCID driver provided with operating system available or the driver can be installed automatically online
- Communicated packets are encrypted, preventing listening simulation software
- Counting or timing mode for user selection
- Easy and secure remote upgrade

- .NET virtual machine system with Feitian's own IPR
- Smart enveloper, no development work for encrypted product, smart and automatic analysis of .NET code functions of application, automatic migration of resulting algorithms and functions into card to run
- A complete set of easy-to-use tools for controllable license, and role and process-based management of encrypted products

1.1.2 Supported Environments

- Framework: Framework 2.0, Framework 3.0, Framework 3.5, Framework 4.0
- Supported development languages: C#, VB.NET, Delphi.NET, ASP.NET
- Supported .NET application programs: UI, console, service, control, and component programs
- Supported operating systems: Windows 2000, Windows 2003(32/64-bit), Windows XP(32/64-bit), Windows Vista(32/64-bit), Windows 2008 (32/64-bit), and Windows 7 (32/64-bit)

1.2 Introduction to ROCKEY7.NET Production Tool

ROCKEY7.NET Producing Tool includes an initialization tool and a producing tool. The initialization tool is used to initialize the ROCKEY7.NET hardware to allow it to work with the Enveloper Tool. It can initialize the PID and SO PIN of the ROCKEY7.NET device. If the hardware has not been initialized, the initialization tool will set the PIN and SO PIN according to the seed entered by the user. If it has been initialized, or used in other projects, the user must enter a correct PID and SO PIN before the initialization tool initializes the hardware using current password of the hardware, while clearing all data on the hardware. All the information on the hardware will be lost after initialization, and the hardware cannot work with the original program.

The ROCKEY7.NET producing tool is used to produce the dongle distributed to end users in batches. A project file generated by the Enveloper Tool is required for the producing tool to import it. In addition, new timer or counter software license can be designed as required. Then, the producing tool writes the functions to be migrated to the dongle. The produced ROCKEY7.NET dongle can work with a corresponding .NET program.

1.3 Production Process

The production process is as follows:

1. Initialize the dongle with the initialization tool, setting the PID and SO PIN using a seed.
2. Protect the program with the Enveloper Tool to generate a project file.
3. Open the project file with the producing tool, set a software license, and produce the dongle in

batches.

The whole production process is clear and specific. The producing tool runs efficiently and stably.

Chapter 2. Using ROCKEY7.NET Initialization Tool

2.1 Generating New Rockey7.Net License File

Start ROCKEY7.NET initialization tool. You need to create a license file or open an existing license file. If this is the first time you use ROCKEY7.NET, you must create a license file. The license file will be required when you use the production tool, the enveloper, or the remote update tool.

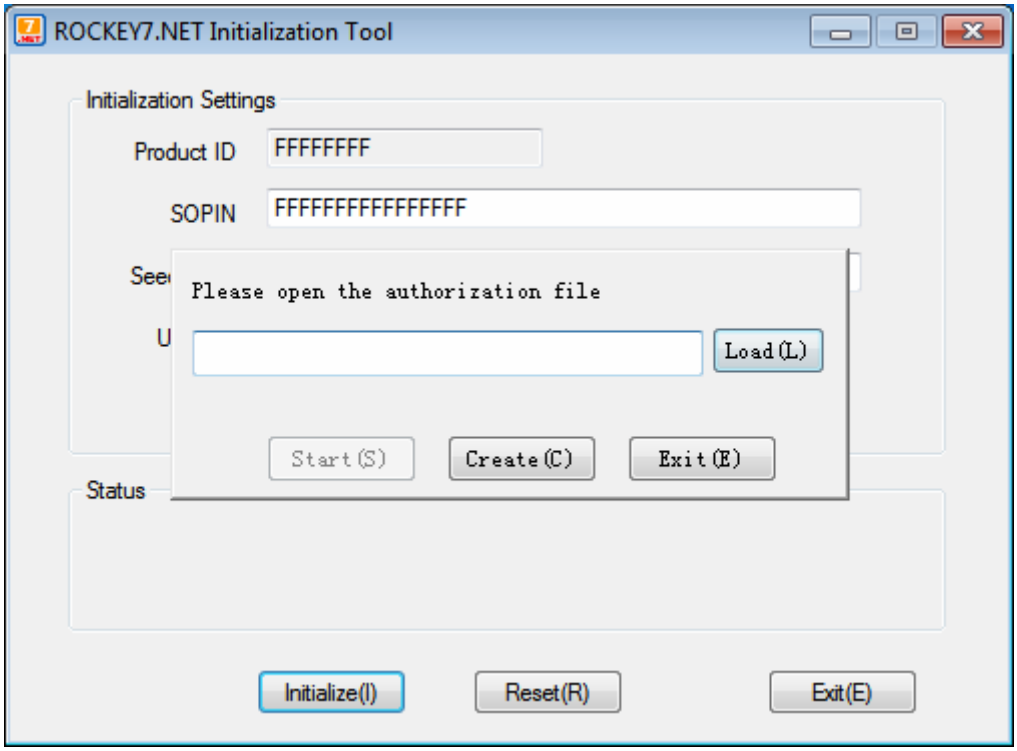

Figure 2.1 Initialization Tool Logon

Attach an empty ROCKEY7.NET dongle and click Create License button. Because the dongle has not been initialized, its product ID and SO PIN are all the default (“FF”s). At this point, you can enter a seed code to create a new license.

The seed code is used to generate a product ID and a SO PIN. Different seed codes produce different product IDs and SO PINs. The product ID and SO PIN cannot be configured directly. Thus, it is almost impossible to duplicate a dongle with the same product ID and SO PIN as another.

In addition, the license file can be used to control the number of the child dongles that can be initialized. The number can be specified in Unit Left field. The license file can be opened by the initialization tool. You can make changes to it. For better protection effects, you can save the number limit to the parent dongle. Thus, users need to open the initialization tool and attach the parent dongle if they want to make changes to the licensed number.

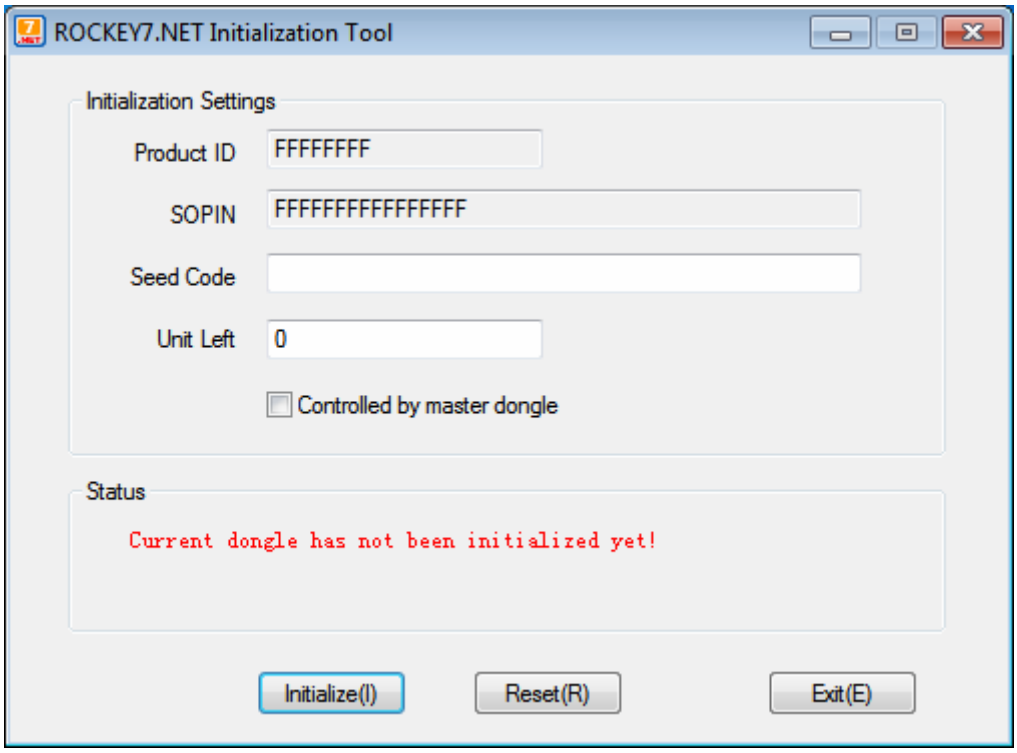

Figure 2.2 Initialization Tool Main Interface

Users can enter a seed code and specify the number of user dongles that are allowed to produce. It is recommended to store the information into the parent dongle. Click Initialize button.

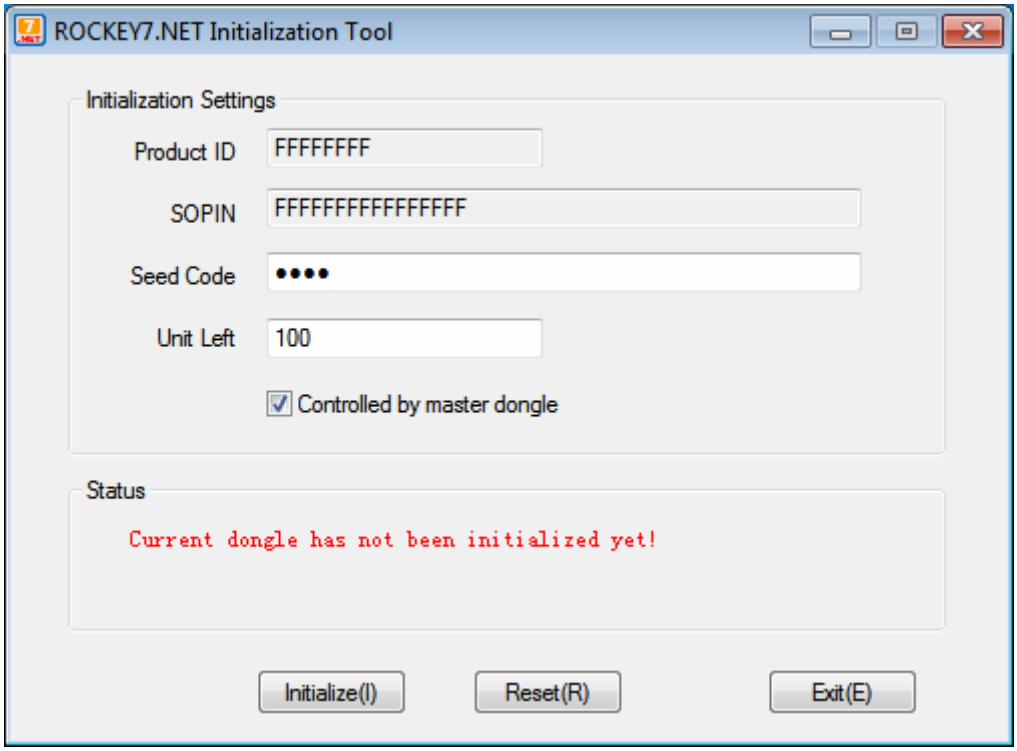

Figure 2.3 Entering Seed Code

After initialization, choose a directory to save the license file.

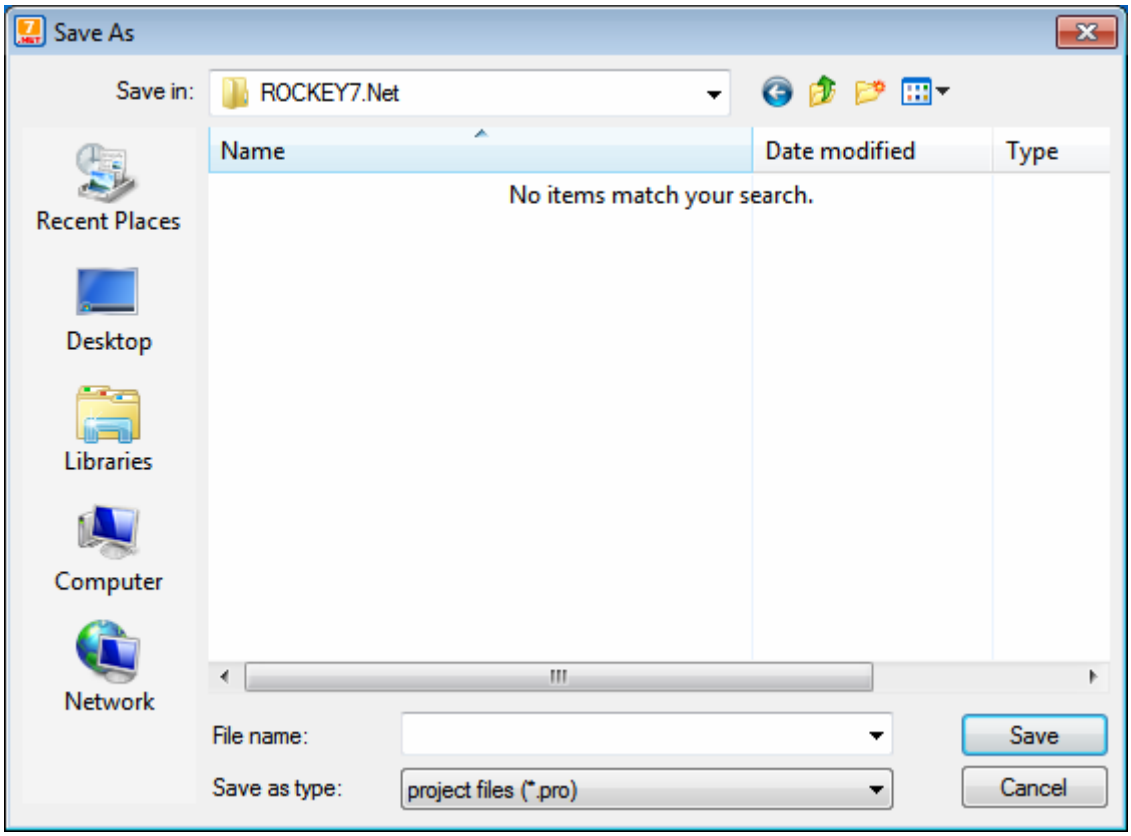

Figure 2.4 Saving License File

After the license file is saved, the new product ID and SO PIN will be displayed in the initialization tool. Take good card of the license file. It will be needed later.

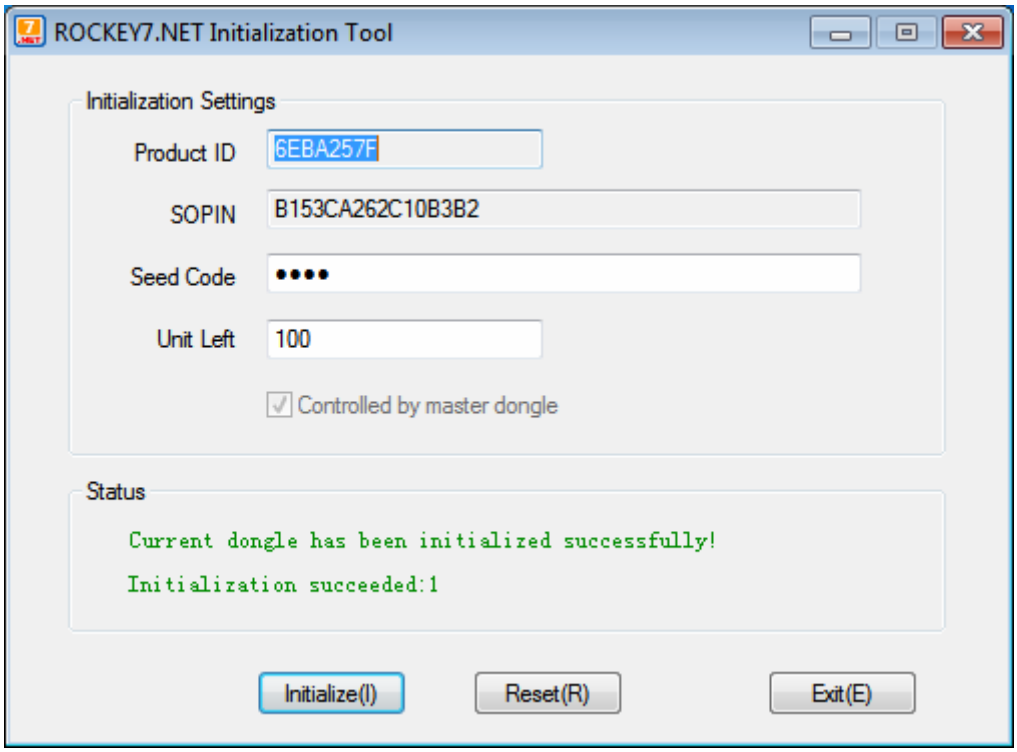

Figure 2.5 Completing Initialization

2.2 Editing Existing Rockey7.Net License File

After a user dongle is produced using the production tool, the number of the child dongles that can be initialized will be decreased by 1 until it becomes 0. Then, the license must be updated to produce again. You can use the initialization tool to edit an existing license file.

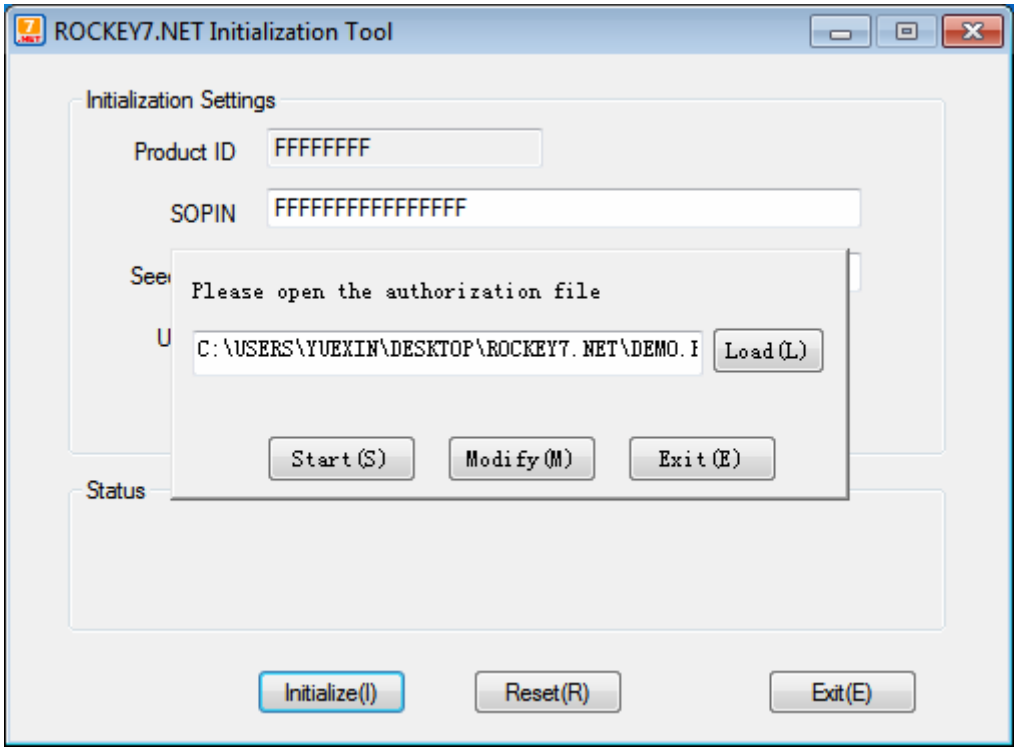

Figure 2.6 Initializing Used Hardware

Start the initialization tool. Click Import button. Choose a license file. If you have specified to save the information into the parent dongle, you need also to attach the parent dongle to your computer. Click Modify Authorization button.

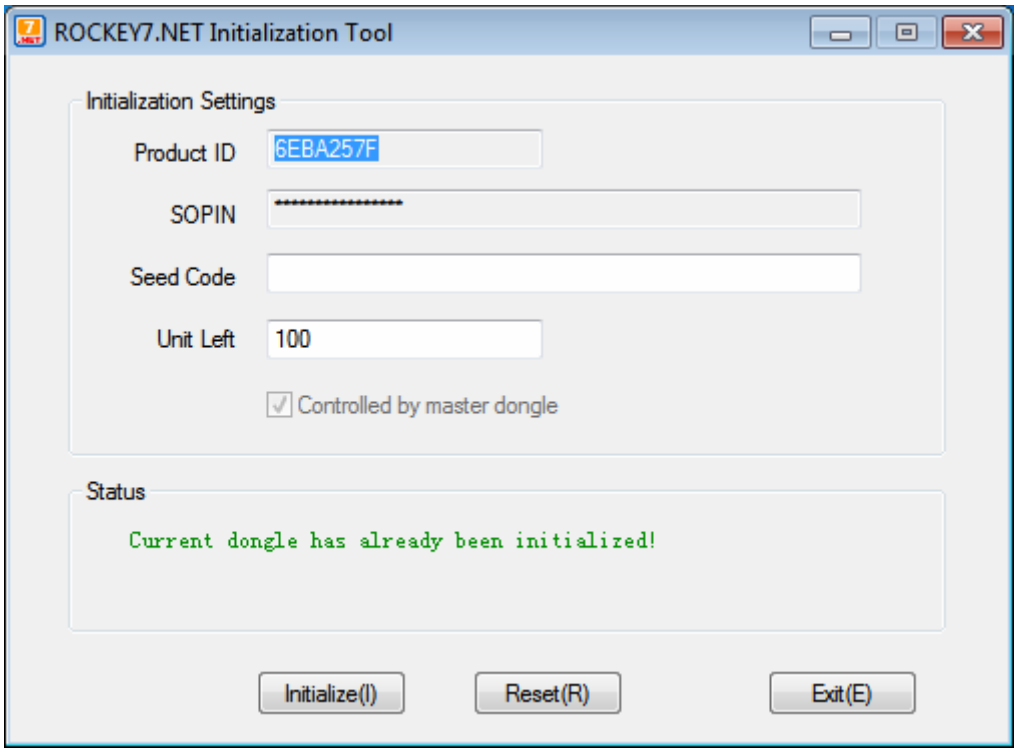

Figure 2.7 Editing License File

Enter a seed code to generate a new license file. If you enter a seed code that is not compliant with the product ID, a new product ID and SO PIN will be generated and the license file will be updated, which cannot be used in the original project again, although the initialization will proceed anyway.

2.3 Initializing Rockey7.Net User Dongle

The ROCKEY7.NET user dongle is used by end user to open the .NET application program protected by the enveloper. In addition, it is also needed when using the enveloper. Make sure that you have initialized at least one user dongle before protecting the .NET application program with the enveloper.

When initializing the ROCKEY7.NET user dongle, you need to attach a parent dongle and a child dongle.

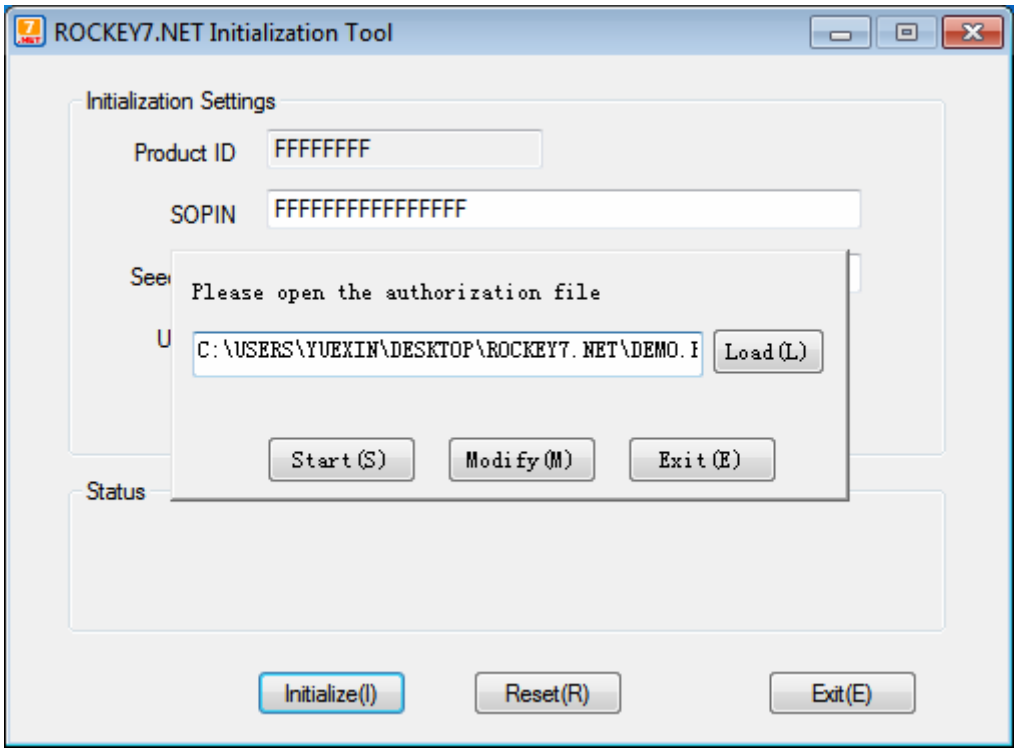

Figure 2.8 Initialization Tool Logon

Choose a license file. Click Start button to start initialization. The child dongle will be detected automatically by the initialization tool and its status will be shown.

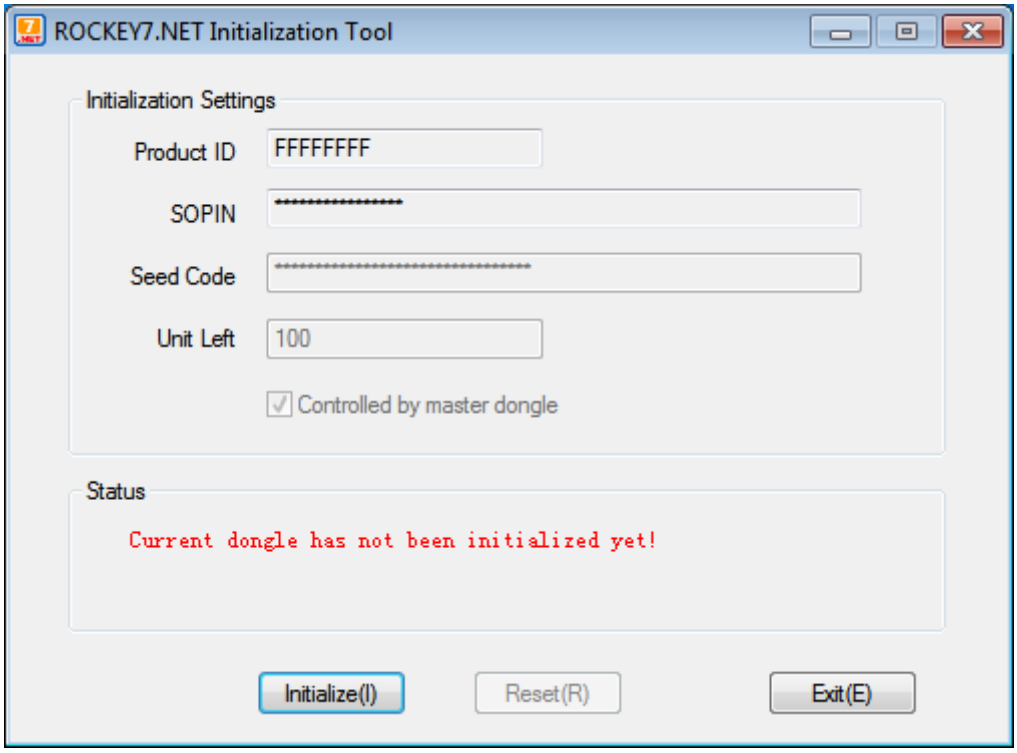

Figure 2.9 Initializing Child Dongle

Click Initialize button to initialize the child dongle. If the operation is successful, a success prompt will be shown and the number of child dongles that can be initialized will be decreased by 1.

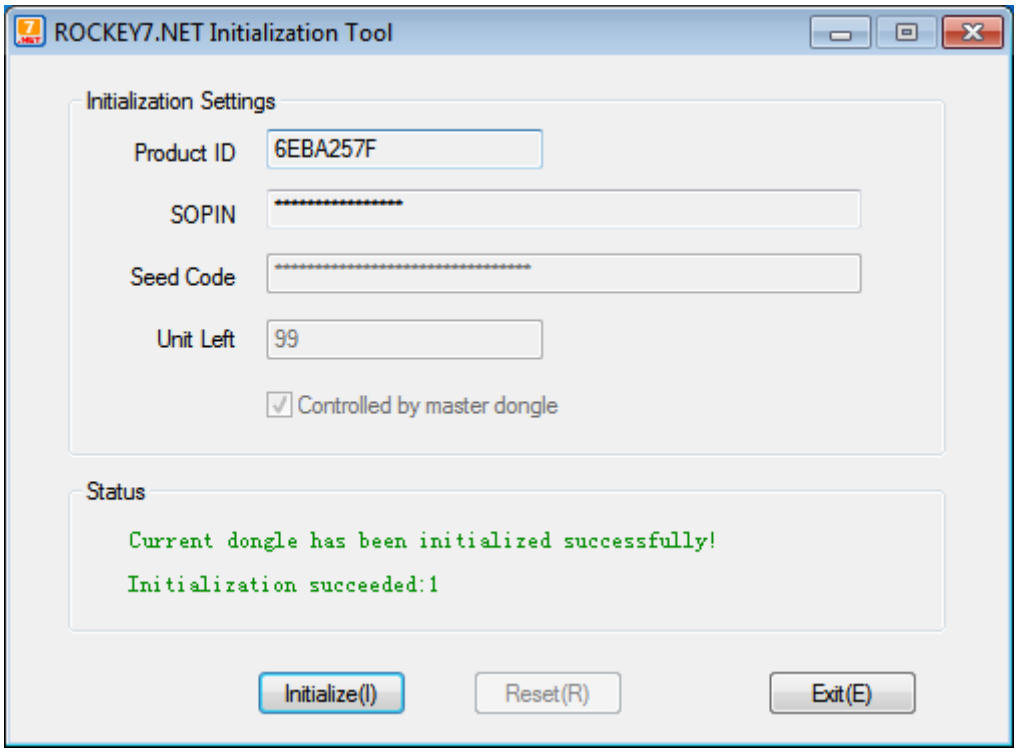

Figure 2.10 Completing Initialization of Child Dongle

Chapter 3. Using ROCKEY7.NET Production Tool

The production tool is used to produce a batch of dongles under a license file. Make sure the .NET application program has been protected successfully and the license file has been updated through the enveloper before using the production tool.

3.1 Producing User Dongle

Start the production tool. If you choose to store the information into a parent dongle during initialization, you need attach a parent dongle and a child dongle to your computer. You will be asked to open ROCKEY7.NET license file.

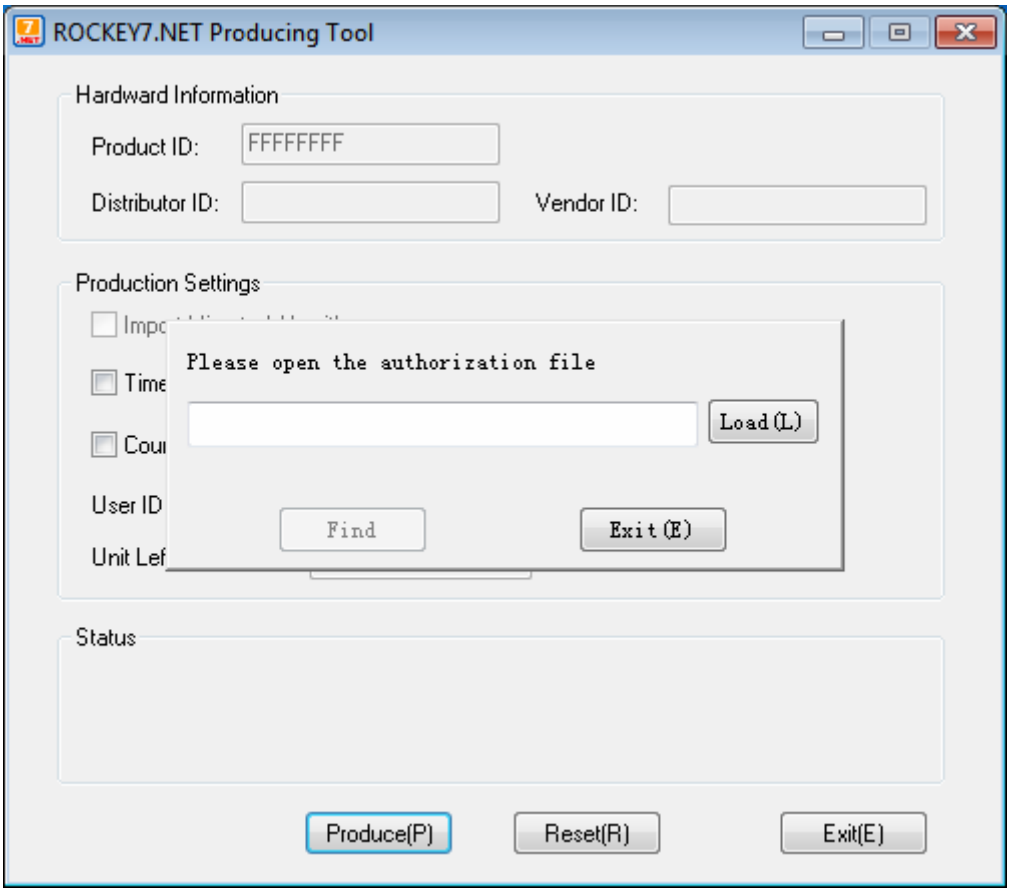

Figure 3.1 Production Tool Logon

Choose the license file and click Find button. After finding the valid parent dongle and child dongle, you will be directed to the main interface.

In Basic Information area is the information in the dongle, including product ID, distributor ID, and customer ID.

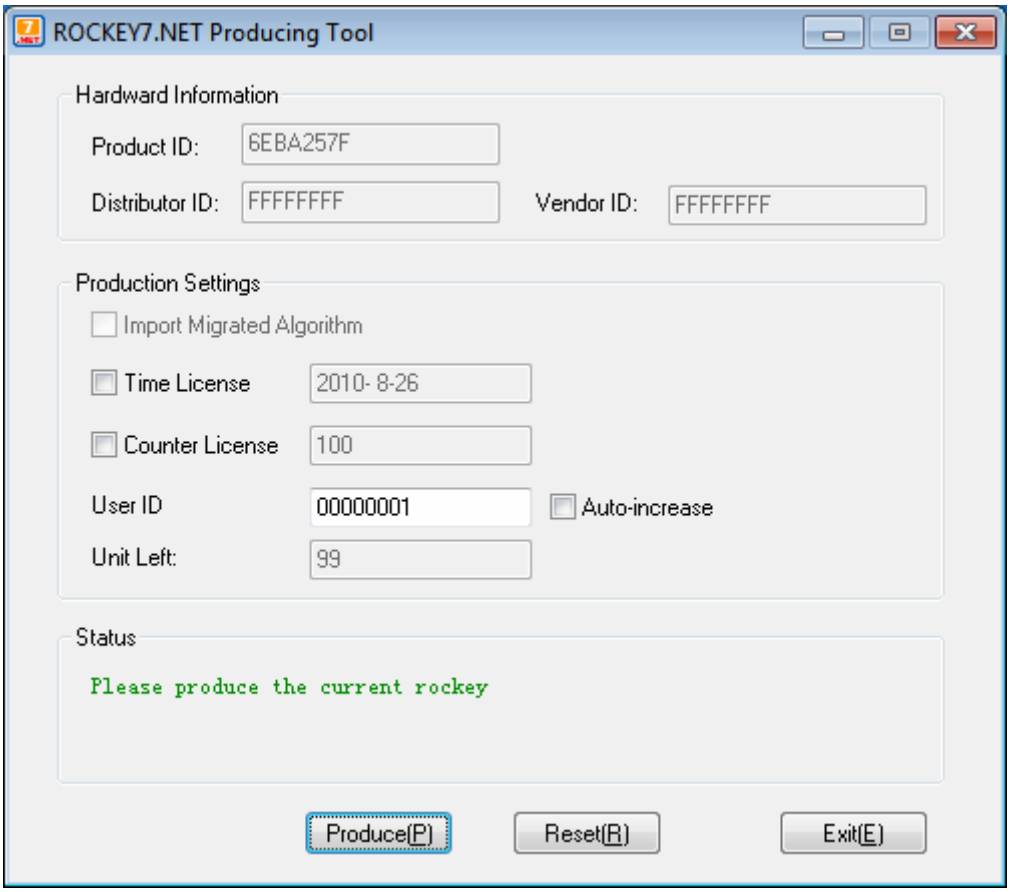

Figure 3.2 Production Tool Main Interface

When starting the production tool, you can also attach a parent dongle and an empty dongle. After importing a license file, you will be directed to the main interface.

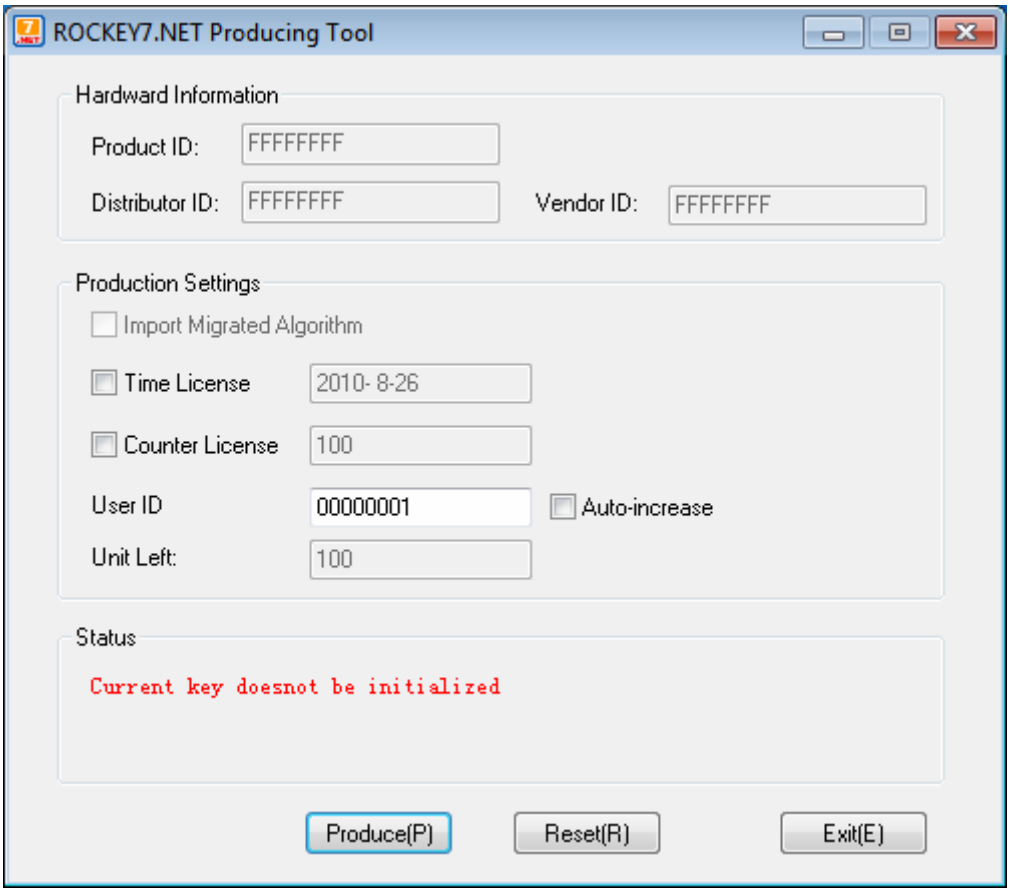

Figure 3.3 When Child Dongle Not Initialized

Click Ok to complete initialization and production of child dongle.

3.2 Setting Production Options

The production options include the algorithm function, the timer, the counter, and the user ID.

- Algorithm Function

It is a function migrated to the dongle by the Enveloper Tool. It must be chosen to download when producing the dongle for the first time. Otherwise, the produced dongle cannot work with the protected program.

If the dongle has been initialized, the producing of the device requires only the setting of the timer, the counter and user ID. The algorithm function download is not necessary.

- Timer License

This kind of license limits the use of the protected software to a time scope.

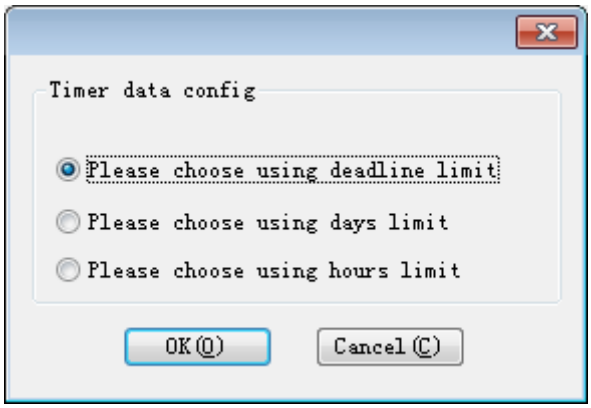

Figure 3.4 Timing Options

If you select the deadline option, you can specify an expiration date. Once the date is reached, the software cannot be opened normally. In this case, a remote update is needed. Or, reproduce the dongle with the producing tool and set a new timer license to resume use of the software.

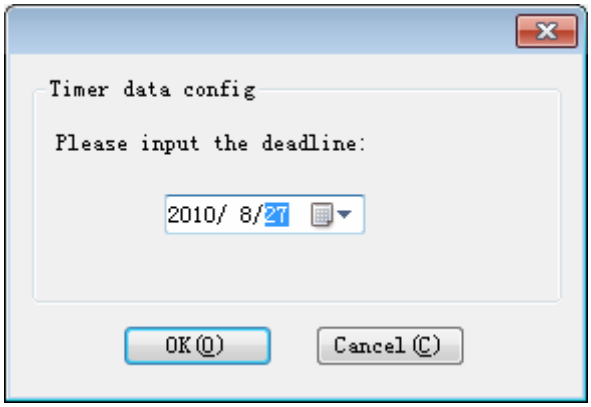

Figure 3.5 Selecting Expiration Date

If you select the days limit, the protected software does not work the specified days after it is used for the first time. Users must remotely update the dongle. Or, reproducing of the dongle with the producing tool and setting a new timer license are needed to continue to use the software.

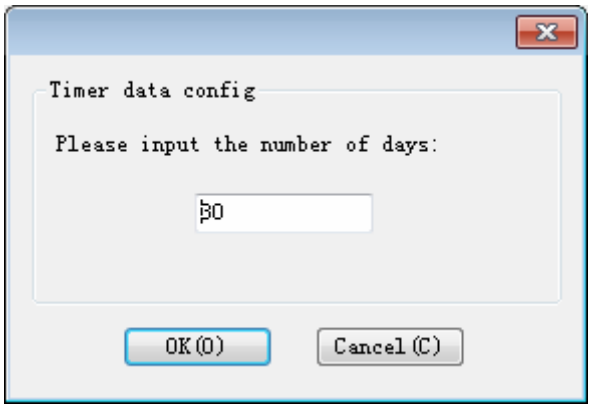

Figure 3.6 Entering Days of Use

If you select the hours limit, the protected software will not work after the accumulated time of use reaches the specified value. The amount of time of use of software by users can be controlled exactly in this way.

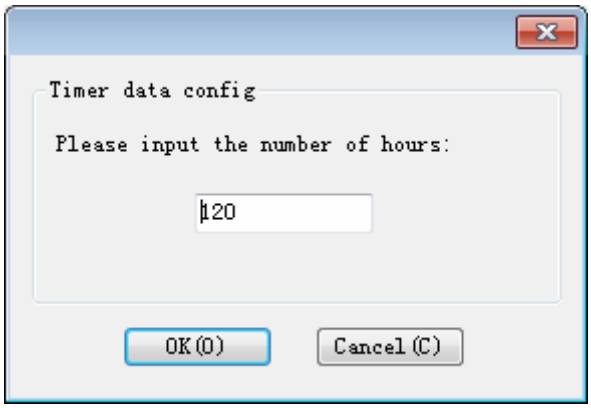

Figure 3.7 Entering Hours of Use

- Counter License

This kind of license limits the number of the times of the use of the software. When the specified number is reached, the protected program cannot be opened. Users must remotely update the dongle. Or, reproducing of the dongle with the producing tool and setting a new counter license are required to continue to use the software.

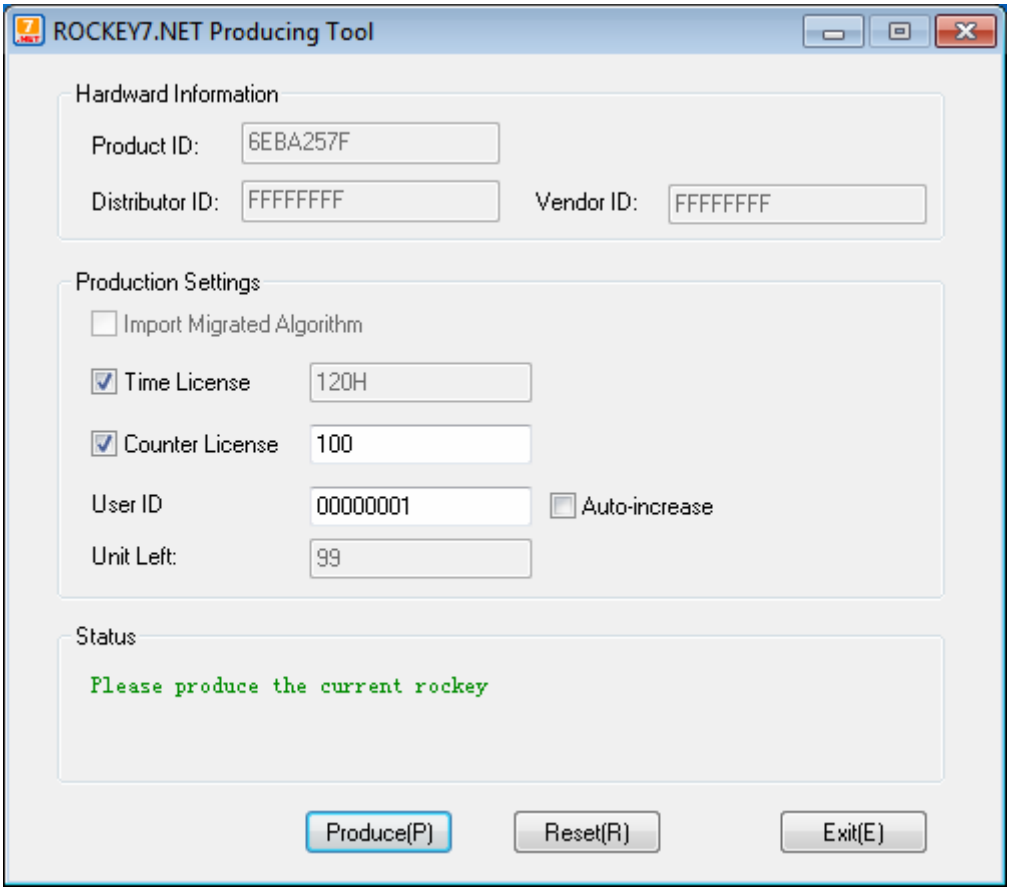

Figure 3.8 Specifying Counter License

- User ID

The user ID is used to distinguish different users by the developer. After setting a user ID, the developer can selectively perform a specific operation for a user, such as remote update. We recommend developers assign a user ID to each dongle when producing it, which is helpful for future maintenance.

When selecting the incremental user ID, the user ID increases by 1 each time a dongle is produced. Thus, each produced dongle has a unique user ID.

After specifying the producing options, click OK to produce the dongle in batches. The producing tool will keep the statistics of successes and failures.

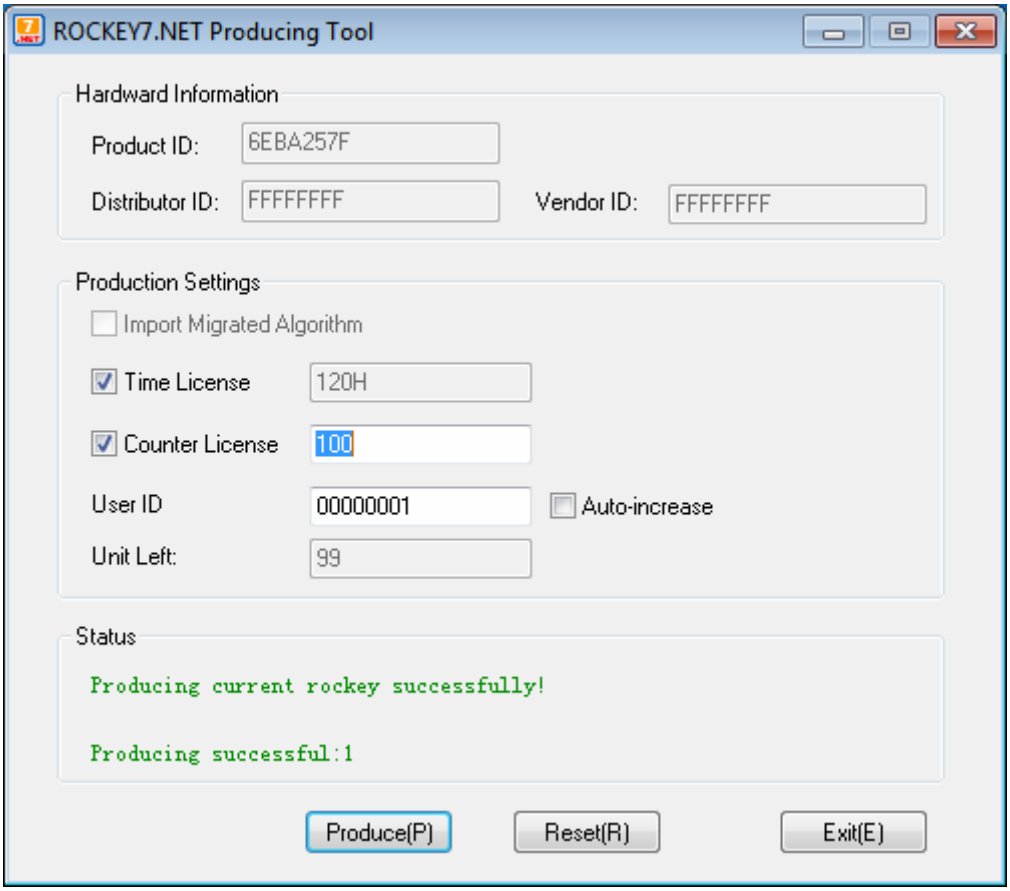

Figure 3.9 Producing Dongle

After the producing is complete, users can choose to save the producing configuration data to a file. In this way, users do not need to reset the producing options next time they want to produce the dongle.