ARM Cortex™-M0

32-BIT MICROCONTROLLER

NuMicro™ Family
MCU Technology Guide

Nuvoton is providing this document only for reference purposes of NuMicro microcontroller based system design. Nuvoton assumes no responsibility for errors or omissions.

All data and specifications are subject to change without notice.

For additional information or questions, please contact: Nuvoton Technology Corporation.

www.nuvoton.com

4 MAIN SYSTEM

4.4 Security System

4.4.1 Security Lock

4.4.1.1 Security Lock Register

Nuvoton provides a function for the NuMicro™-M0 series to lock the chip securely by means of the user configuration register, Config0[1], LOCK bit. As shown in *Figure 4.1*, if the LOCK bit is set as 0, user can only get the chip's data in Config0 and Config1 through *Nuvoton's* NuMicro ICP programming tool, NuGang programmer, or a third party programming tool, and the other data in flash will be shown as 0xFFFF_FFFF. This protection mechanism can prevent the original source code from being stolen. In addition, *Nuvoton* provides other applications to enhance and strengthen the protection of user's source code. Please refer to section *4.4.2* for details.

Config0 (Address = 0x0030_0000)

31	30	29	28	27	26	25	24
Reserved			CKF	Reserved	CFOSC		
23	22	21	20	19	18	17	16
CBODEN	CBOV1	CBOV0	CBORST	Reserved			
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
CBS	Reserved					LOCK	DFEN

[1]	LOCK	**Security Lock** 0 = Flash data is locked 1 = Flash data is not locked When flash data is locked, only device ID, Config0 and Config1 can be read by writer and ICP through serial debug interface. Others data is locked as 0xFFFFFFFF. ISP can read data anywhere regardless of LOCK bit value.

Fig 4.1 LOCK Bit in Config0 Register

4.4.1.2 Locking a Chip via ICP or ISP Tool

Besides the third party writer, user can also use *Nuvoton's* NuMicro ICP or ISP Programming Tool to lock the source code during the process of chip programming. Moreover, Nuvoton provides an interface for user to get the contents of flash data from the *On-board Flash* window wherein the flash data is changed to 0xFFFF_FFFF after the ICP tool finishes "Security Lock" process. Through such a convenient tool, source code protection can be greatly improved. If someone wants to read the flash data of the locked chip, the ICP tool will pop out a warning window to enforce the whole chip erase. *Figure 4.2* shows the flow of how to set the chip locked.

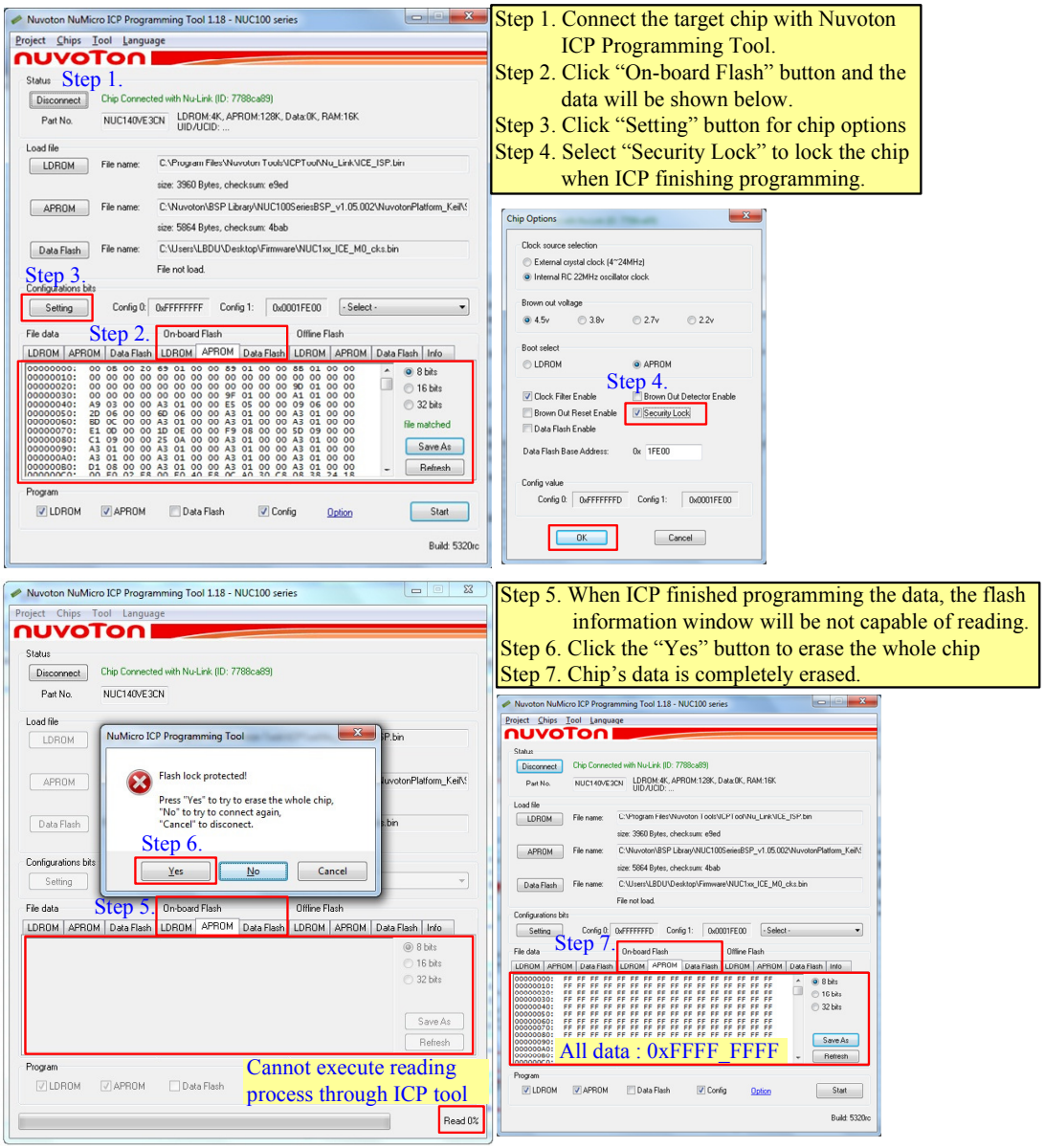

Fig 4.2 "Security Lock" Flow in the ICP Tool

4.4.1.3 Source Code Protection in ICP Offline Mode

Except enabling the **Security Lock** option to set the chip locked for source code protection as described in section *4.4.1.2*, *Nuvoton* provides another chip protection mechanism in Offline mode with the ICP tool.

When using *Nuvoton's* "Nu-Link" or "Nu-Link Pro" programming tools (as shown in *Figure 4.3*) to program a chip in Offline mode, there are two ways to protect chips — "setting password for offline data" or "limiting the number of offline programming". *Figure 4.4* shows the steps of setting the password or the maximum number of programming in Offline mode.

Fig 4.3 Nu-Link and Nu-Link Pro Programming Tool

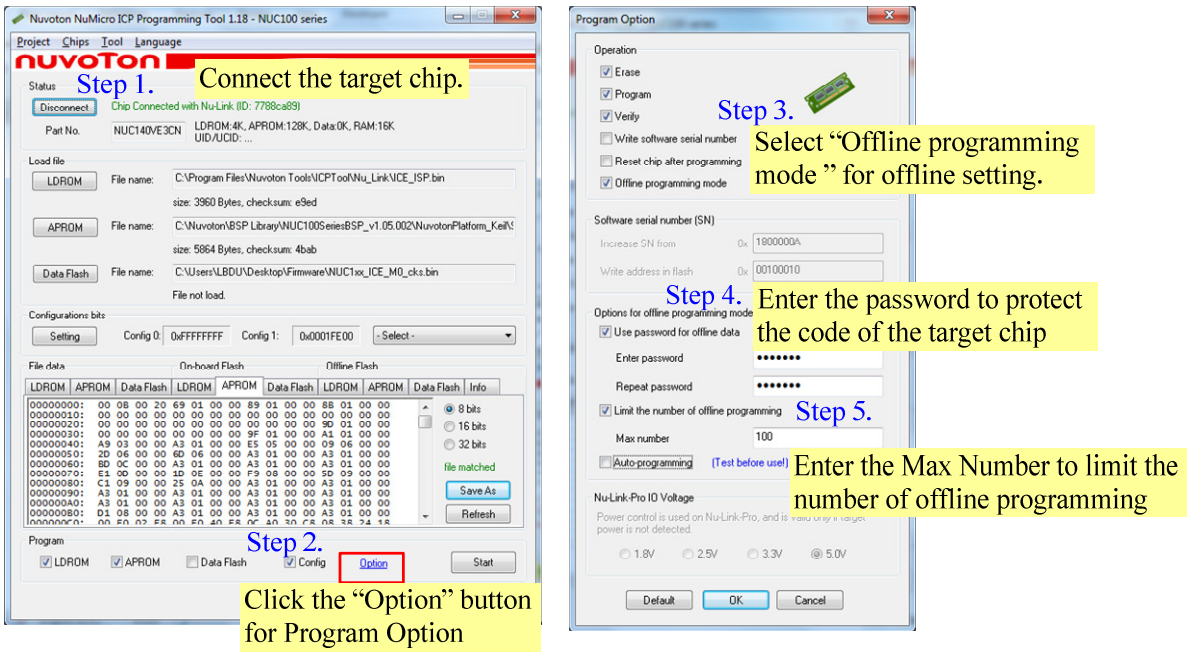

Fig 4.4 ICP Offline Mode

For the Step 4 in *Figure 4.4*, user can set any form of password and re-enter the password again to confirm the setting, and then click the **Start** button to execute the chip programming process. When the ICP tool detects the chip connected next time, a request form will appear to ask the user to enter the right password to unlock the chip. If the entered password does not match the preset password, the request window will not disappear untill the right one is entered. If user wants to remove the "Password" setting, just enter the right password and undo the click on step 4 or erase the whole chip data could achieve.

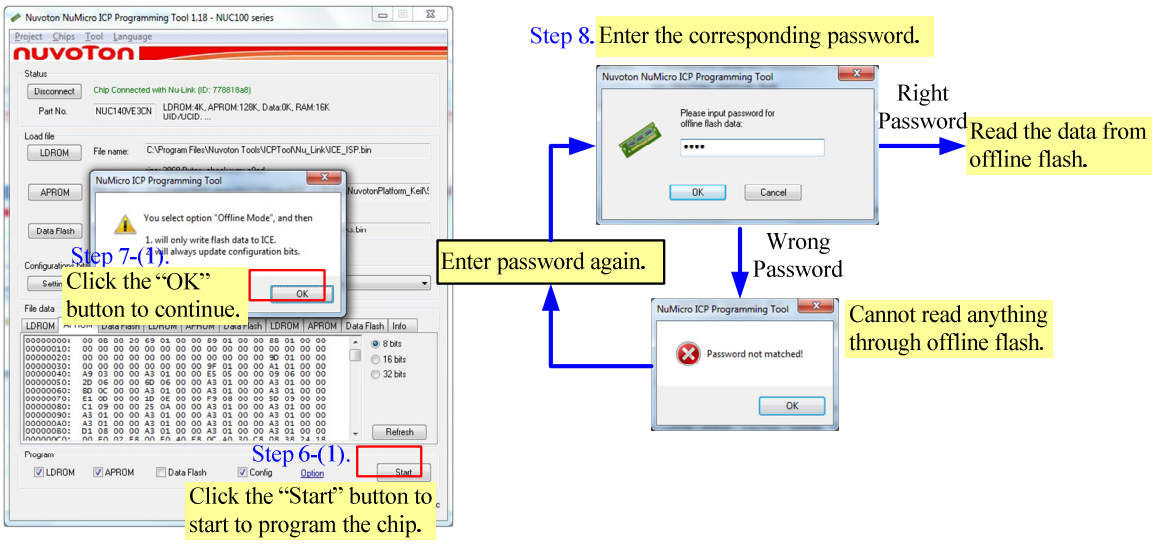

Fig 4.5 Password Setting in ICP Tool Offline Mode

Another offline protection mechanism is limit the number of programming chip when using the "Nu-Link" or "Nu-Link Pro" programming tools to program a chip. When the number of chip programming meets the limitation number, the "Nu-Link" or "Nu-Link Pro" will not be allowed to program any other chips. User needs to erase and re-program "Nu-Link" or "Nu-Link Pro" to continue chip programming again. User can get the flash data in "Nu-Link" or "Nu-Link Pro" from the *Offline Flash* window as shown in *Figure 4.6*.

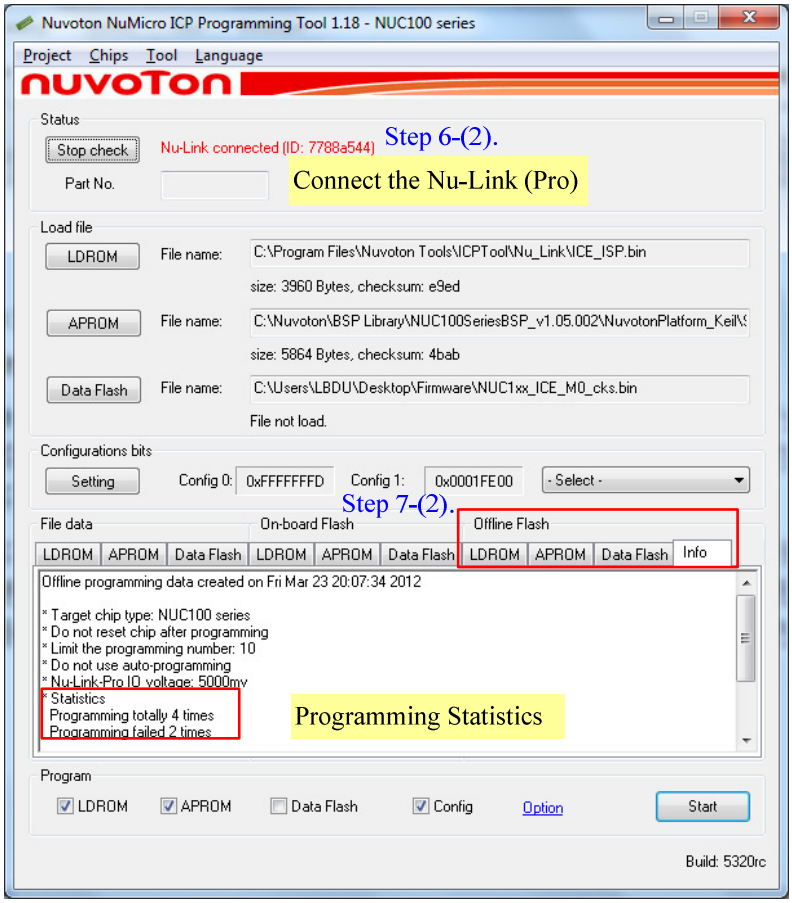

Fig 4.6 Offline Flash Data in the ICP Tool

Both setting the password or limiting the maximum number of chip programming in Offline mode provide further protection for chip programming.

4.4.2 UID Security Protection

4.4.2.1 What is UID?

UID (Unique Identification) stands for a specific code for every chip dispatched from Nuvoton, just like an identification card is unique for everyone as well. UID represents chip's part number and date of dispatching with length of 96 bits.

Why UID is so special for the chip's encryption issue? User may worry or concern if anyone industrial agent or person with intensions steals the source the code when a commodity is under development or developed stage. And then, they try to get the same IC to achieve the same function as user designed. Therefore Nuvoton provides a unique ID for every NuMicro™-M0 chip so that user could employ UID into the source code. The UID protection mechanism will be introduced in the next section.

4.4.2.2 UID Security Mechanism

To enhance the degree of safety for user source code, every NuMicro™-M0 chip dispatched from *Nuvoton* will be planted with a unique ID in the chip. User can put UID, which is dealt with DES (Data Encryption Standard), into DataFlash or some specified area. Also, users can design the standard of encryption by themselves to protect the source code in the chip from being stolen and produced by other intensive people.

Figure 4.7 shows the flow that UID needs to be dealt with DES and put into DataFlash. Then user can add a judging method in the firmware code to compare the current chip's UID with the previous one. If the result is not the same, the program will fall into dead loop. Consequently, it will greatly prevent user' commodity to be mass produced from source code being stolen, under this double protection mechanism from UID (*Nuvoton*) and DES (user). Moreover it also deeply increases the confidence and reliability for user to use Nuvoton IC.

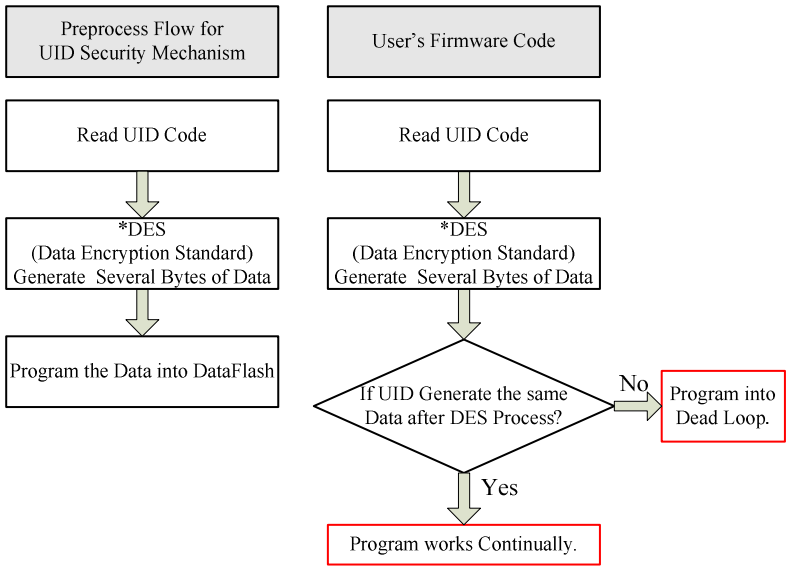

Fig 4.7 UID Protection Flow

4.4.3 UCID Security Protection

UCID (Unique Customer ID) is an initiative technique provided by *Nuvoton* to protect chip source code. In addition to the previous UID section, user can apply to *Nuvoton* for customized UCID to ensure chip safety.

First of all, user provides the product number or specific code to put into the UCID. User data will be encoded and the conversion result will be planted into the chip to become customized and highly protected. *Figure 4.8* shows the flow about how UCID protects the chip' source code, and *Figure 4.9* shows the information to be put into the UCID.

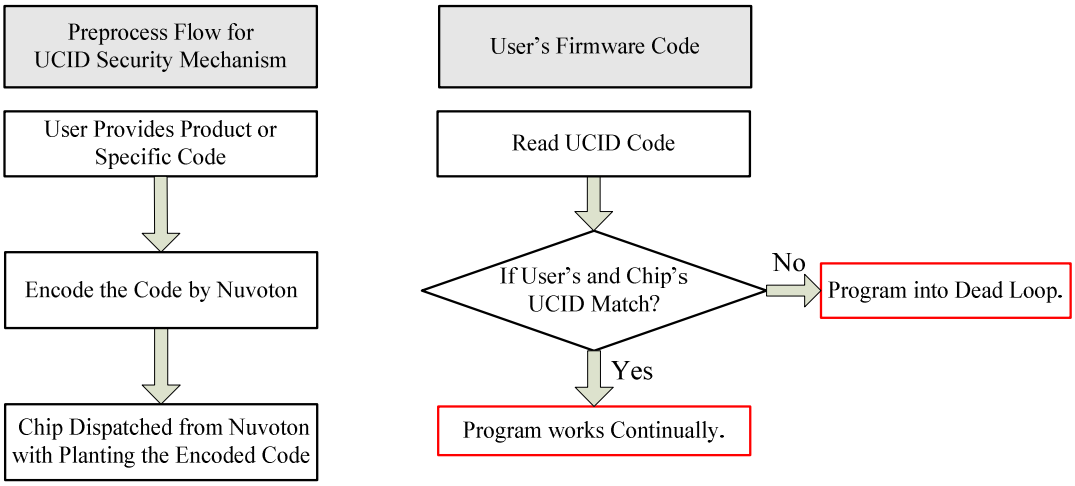

Fig 4.8 UCID Protection Flow

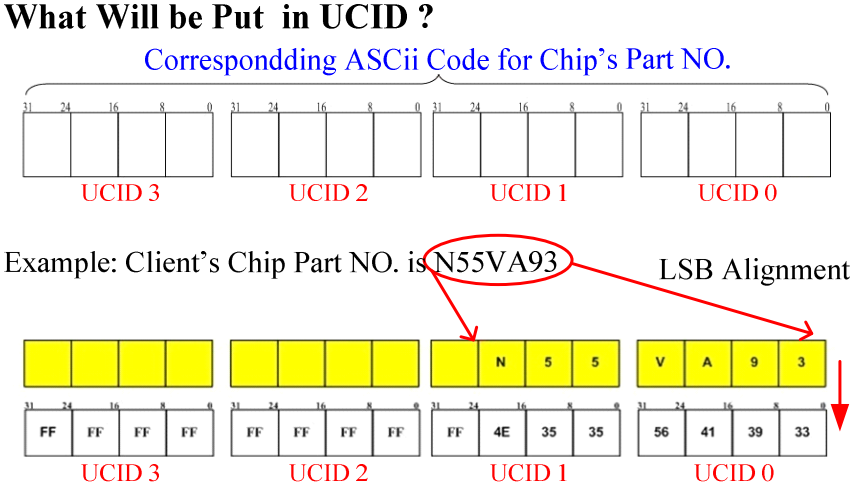

Fig 4.9 UCID Information

User can connect a chip with *Nuvoton's* "NuMicro ICP or ISP Programming Tool" to get the current UCID in the chip as shown in *Figure 4.10*.

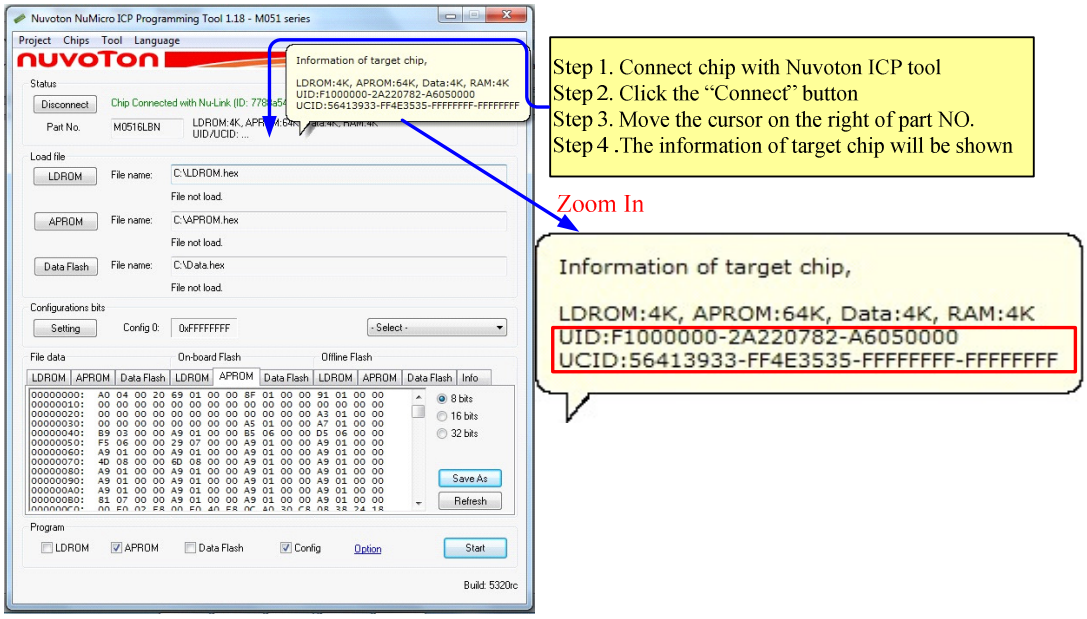

Fig 4.10 UID and UCID Data i ICP Tool

As to the protection mechanism for UCID, user can add a judging method in the firmware code to compare the current chip's UCID with the one encoded by Nuvoton. If the result is not the same, the program will fall into dead loop. Even user connects the chip with the ICP tool, an error window will appear to enforce the whole chip erase. According to this particular protection mechanism, the possibility of source code being stolen will be decreased, and product competition will be enhanced.

The NuMicro™-M0 series chips from *Nuvoton* are normally dispatched without UCID and top-printed on chips. If user wants to add this protection mechanism to the purchased chips, please contact numicro@nuvoton.com to get further information.

5 REVISION HISTORY

Date	Revision	Description
Dec.14, 2011	V1.0	First edition
Dec.14, 2011	V2.0	Added UCID flow
Dec.15, 2011	V3.0	Added ICP Offline mode
Dec.20, 2011	V4.0	Revised by CA10 CLYU2
Mar.24, 2012	V5.0	Changed into ENG version
Mar.26, 2012	V6.0	Revised by Sylvia