SQFlash Utility

User's Manual

Version v2.0.4211.1

Advantech Co. Ltd.

No. 1, Alley 20, Lane 26,
Rueiguang Road, Neihu District,
Taipei 114, Taiwan, R. O. C.

www.advantech.com

Copyright Notice

Part No.
Version: v2.0.4211.1

Printed in Taiwan 2013-07-19

Version History

Date	Version	Remark
2008-11-20	0.1	First release
2008-12-03	0.2	Update Tool
2008-12-15	0.3	Update naming & PN
2009-01-06	1.0	Add UnInitializeCF API
2009-01-19	1.1	Update SMART attribute
2009-01-20	1.2	Update PN table
2009-02-12	1.3	Add Flash Lock Feature
2010-08-17	2.0.0812.1	1. Version 2.0 release
2011-05-19	2.0.1621.1	1. Add New Model Support
2011-06-10	2.0.1808.1	1. Temporarily remove SMART support on models with FW version “Ver7.M0K” for bug solving
2013-07-19	2.0.4211.1	1. Add New Model Support

Table of Contents

Introduction

SQFlash Utility software package is a flash management package contains utility and API to access Advantech flash storages. It supports Software Protection (Security ID Read/Write & Flash Lock) and Life Monitoring (S.M.A.R.T.) features. Access Code protected package provides user a safe environment not only protect application itself but also prevents Security ID not to be read without the same Access Code while writing. Based on CF 3.0 specification, user can “Lock” SQFlash via Flash Lock function and “Unlock” by BIOS while booting. A locked SQFlash cannot be read by any card reader or boot from other platforms without a BIOS with “Unlock” feature. S.M.A.R.T. attribute contains Max/Average Program and Erase Cycles, Power On Time, ECC count and Endurance Check. User can monitor it directly by SQFlash Utility or implement into application by SQFlash Utility API. Life-span detection mechanism can be designed through the information of Endurance Check.

Benefits

- **Faster Time to Market**
  The Utility is ready to run without modifications. System developers can use it to control the SQFlash without knowing the controller specs of the SQFlash. API and sample code ready for software developers to implement flash management mechanism into their applications.
- **Protect Your Intellectual Property**
  In order to help protect customers’ intellectual property, Advantech has designed in Security ID feature for Advantech SQFlash. Customers can easily implement security functions on their applications base on encrypted utility and library. Flash Lock feature helps user to lock SQFlash to protect flash data not to be read.
- **Monitor SQFlash Health**
  Advantech SQFlash management package provides utility and API to get Self-Monitoring, Analysis, and Reporting Technology (S.M.A.R.T.) information from SQFlash cards. Customer can monitor the flash storage health and design an early warning mechanism by life-span detection.

System Requirements

Hardware

Following hardware are required to run Advantech SQFlash Utility for Windows XP：

1. Processor

   Minimum Requirement： A 200 megahertz (MHz) processor, such as the Intel Pentium/Celeron family, AMD K6/Athlon/Duron family, or compatible processor.

2. RAM

   Minimum Requirement： RAM size is dependent on the running applications and using XPE features.

3. SQFlash: CompactFlash / PATA DOM / PATA SSD / CFast / SATA DOM / SATA Half-Slim Module / mSATA / SATA SSD

   Supported models and firmware requirement as shown below.

   CompactFlash
   (Firmware: 2.M0J/K/L/M, 2.A0K/L/M, 6.A1B~H, 6.A2A)

Part Number	Description
SQF-P10xx-xG-CTx	SQFlash P6 series normal temperature CF card (0~70°C)
SQF-P10xx-xG-ETx	SQFlash P6 series wide temperature CF card (-40~85°C)
SQF-P10xx-xG-P8C	SQFlash P8 series normal temperature CF card (0~70°C)
SQF-P10xx-xG-P8E	SQFlash P8 series wide temperature CF card (-40~85°C)

PATA DOM
(Firmware: 2.A1E/F/H)

Part Number	Description
SQF-PDMxx-xG-xxCE	SQFlash P6 series normal temperature PATA DOM (0~70°C)
SQF-PDMxx-xG-xxEE	SQFlash P6 series wide temperature PATA DOM (-40~85°C)
SQF-PDMxx-xG-xx8C	SQFlash P8 series normal temperature PATA DOM (0~70°C)
SQF-PDMxx-xG-xx8E	SQFlash P8 series wide temperature PATA DOM (-40~85°C)

2.5” PATA SSD
(Firmware: 3.A1A/B/C/D)

Part Number	Description
SQF-P25xx-xG-CTE	SQFlash P7 series normal temperature PATA SSD (0~70°C)
SQF-P25xx-xG-ETE	SQFlash P7 series wide temperature PATA SSD (-40~85°C)

CFast
(Firmware: Ver6A01d)

Part Number	Description
SQF-S10xx-xG-S7C	SQFlash 520 series normal temperature CFast (0~70°C)
SQF-S10xx-xG-S7E	SQFlash 520 series wide temperature CFast (-40~85°C)

SATA DOM
(Firmware: Ver6A01d)

Part Number	Description
SQF-SDMxx-xG-S7Cx	SQFlash 520 series normal temperature SATA DOM (0~70°C)
SQF-SDMxx-xG-S7Ex	SQFlash 520 series wide temperature SATA DOM (-40~85°C)

SATA Half-Slim Module
(Firmware: S5FAMA09/11/14/18; Ver6A01d; S8FM05A3 and above)

Part Number	Description
SQF-SLMxx-xG-S5C	SQFlash 810 series normal temperature SATA Slim (0~70°C)
SQF-SLMxx-xG-S5E	SQFlash 810 series wide temperature SATA Slim (-40~85°C)

SQF-SLMxx-xG-S7C	SQFlash 520 series normal temperature SATA Slim (0~70°C)
SQF-SLMxx-xG-S7E	SQFlash 520 series wide temperature SATA Slim (-40~85°C)
SQF-SLMxx-xG-S8C	SQFlash 820 series normal temperature SATA Slim (0~70°C)
SQF-SLMxx-xG-S8E	SQFlash 820 series wide temperature SATA Slim (-40~85°C)

Half Size mSATA
(Firmware: Ver6A01d)

Part Number	Description
SQF-SHMxx-xG-S7C	SQFlash 520 series normal temperature H/S mSATA (0~70°C)
SQF-SHMxx-xG-S7E	SQFlash 520 series wide temperature H/S mSATA (-40~85°C)

mSATA
(Firmware: S5FAMA09/11/14/18; Ver6A01d; S8FM05A3 and above)

Part Number	Description
SQF-SMSxx-xG-S5C	SQFlash 810 series normal temperature mSATA (0~70°C)
SQF-SMSxx-xG-S5E	SQFlash 810 series wide temperature mSATA (-40~85°C)
SQF-SMSxx-xG-S7C	SQFlash 520 series normal temperature mSATA (0~70°C)
SQF-SMSxx-xG-S7E	SQFlash 520 series wide temperature mSATA (-40~85°C)
SQF-SMSxx-xG-S8C	SQFlash 820 series normal temperature mSATA (0~70°C)
SQF-SMSxx-xG-S8E	SQFlash 820 series wide temperature mSATA (-40~85°C)

2.5” SATA SSD
(Firmware: S5FAMA09/11/14/18; Ver6A01d; S8FM05A3 and above)

Part Number	Description
SQF-S25xx-xG-S5C	SQFlash 810 series normal temperature SATA SSD (0~70°C)
SQF-S25xx-xG-S5E	SQFlash 810 series wide temperature SATA SSD (-40~85°C)
SQF-S25xx-xG-S7C	SQFlash 520 series normal temperature SATA SSD (0~70°C)
SQF-S25xx-xG-S7E	SQFlash 520 series wide temperature SATA SSD (-40~85°C)
SQF-S25xx-xG-S8C	SQFlash 820 series normal temperature SATA SSD (0~70°C)
SQF-S25xx-xG-S8E	SQFlash 820 series wide temperature SATA SSD (-40~85°C)

Software

- SQFlash Utility **v2.0.4211.1** utility and API are required
- Dot Net Framework 2.0 required

Environments

Operating Systems that SQFlash Utility supports include:

- Windows XP Embedded
- Windows XP Professional or Home Edition
- Windows 7

Installation

SQFlash Utility software package doesn’t need to install into your operation system. However, you must have Advantech SQFlash installed in your system and a valid Access Code to access the utility or API. Please contact Advantech to get legal access code.

SQFlash Utility

SQFlash Utility is a utility to manage Advantech SQFlash card to access security zone and S.M.A.R.T. information. Key features:

- Access Code protected
- Security ID Read/Write
- Flash Lock
- Protecting sample code generator
- S.M.A.R.T. attribute

This utility package contains 8 files

File Name	Description
SQFlash Utility.exe	Main program
SQFlash_Dll.dll	SQFlash Utility Library
PieChartControls.dll	SQFlash Utility Library
Cpp-ProtCode.txt	Protected sample code in C++
SQFlash.lib	SQFlash Utility Static Library
SQFlash.h	SQFlash Utility Header File
ePFlash.sys	BIOS Flash Driver
SQFlash_PID.pdf	Access Code

For security concern, please import static library SQFlash.lib to implement your protecting mechanism. Every user who wants to adopt SQFlash Utility package should apply Access Code from Advantech first. Access code can be changed by Main program and you can manage multiple devices at the same time. However, it's not allowed to have different Access Code if more than one SQFlash be managed at the same time and the same Main program.

First Window – Access Code

Please input a valid Access Code into this dialog. You can find the Access Code in SQFlash_PID.pdf. Once you type valid access Code with this utility, system will keep record and you won't need to type access Code next time.

Figure-1

Figure-2

Main Window – SQFlash Utility

- Select Device
  Select specific SQFlash if you have more than one in the system.
- Software Protection
  - Select Security ID mode (I or II) to write ID into SQFlash
  - Detect max and user-defined length of Security ID

Mode	Max Length
Security ID I	256 bytes
Security ID II	*256 bytes

*** If you are using older version firmware, SID2 maximum length would be 30bytes.**

  - Read or Write Security ID into SQFlash
  - Get protection code into application (C++ sample)
- Life monitoring: Endurance Check
- Flash Lock: Enable/Disable Flash Lock feature

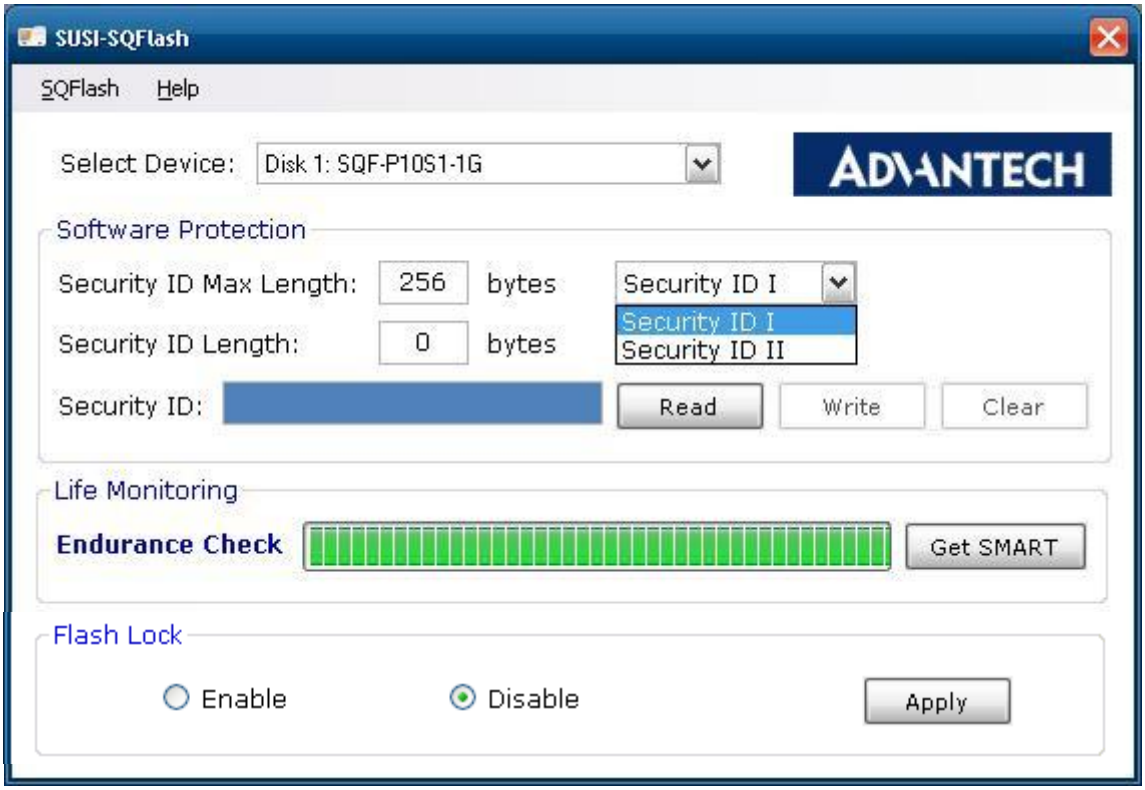
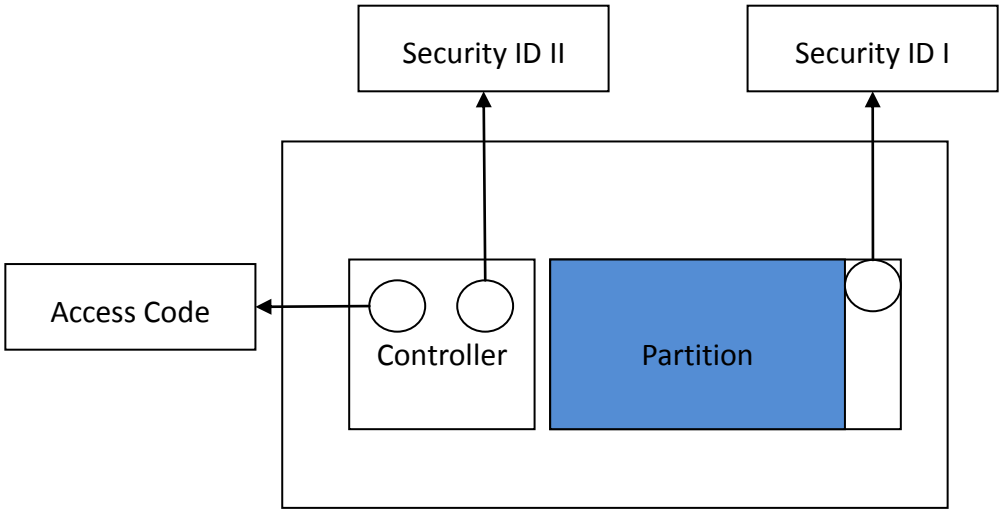


Figure-3

Security ID Function



1. Security ID I
   - This SID will need at least 256 bytes **UNPARTITIONED AREA** to allocate it. If this utility cannot search enough space to write SID, Security ID I user interface will be disabled.
   - Security ID I can be full run-time configuration. All read/write access can be performed immediately.
   - After you enable Security ID I, all disk management access would be disabled, ex. Format, fdisk, ghost, and etc. This is because SQFlash controller protects SID area and not allows any disk I/O to read or write this area. If any tool or command tries to access this area, it will return fail directly.
   - Since SID1 protected space will close to the end of current partition, re-partition or partition size reallocation is not allowed which might erase SID itself.
2. Security ID II
   - This SID will be located into controller and all disk management access will remain the same behavior.
   - If you are using older firmware, after you enable Security ID II, you will need to **COLD REBOOT** the system in order to write SID2 into controller.

Flash Lock Function

Flash Lock feature needs to be supported by Advantech BIOS. If target platform didn't support this feature, Flash Lock buttons will be gray out as Figure-4.

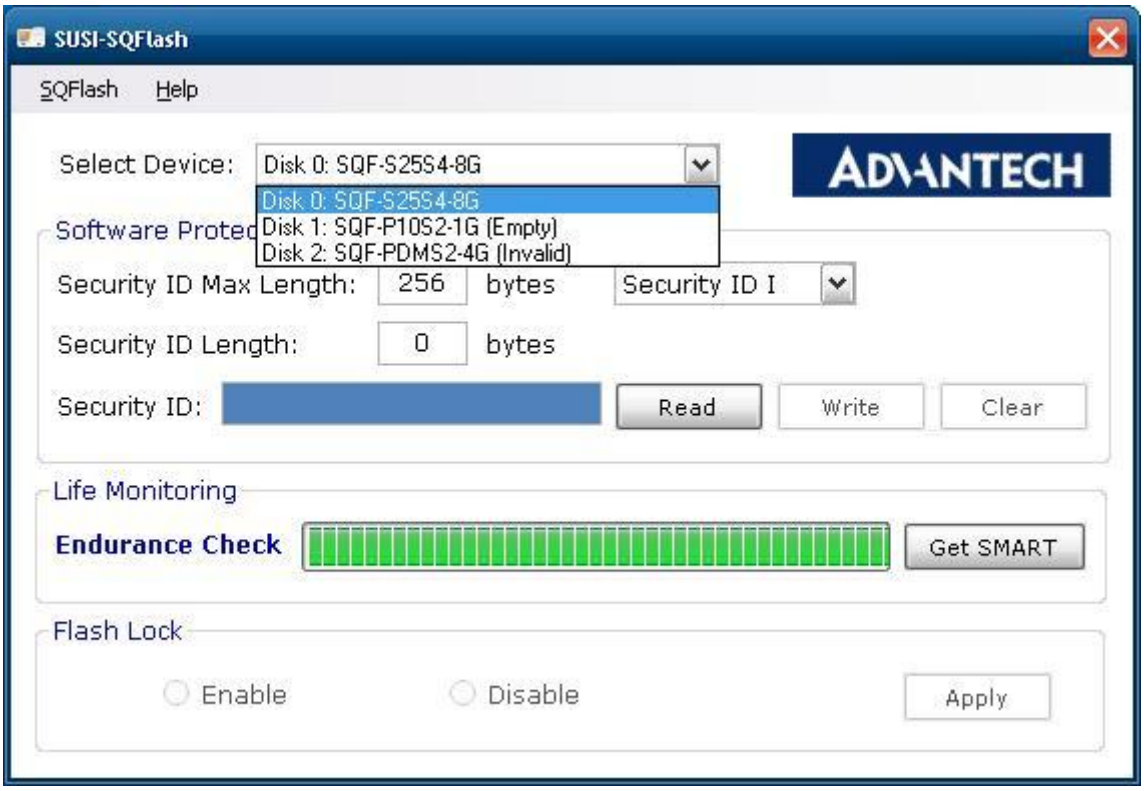


Figure-4

While user press Enable or Disable to change the status of Flash Lock, a progress dialog as Figure-5 will be pop-up.

Figure-5

Modify Access Code

If you manage more than one SQFlash at the same time, all devices should use the same Access Code. Otherwise, the device has no Access Code or different Access Code will remark as **Invalid** as Figure-6.

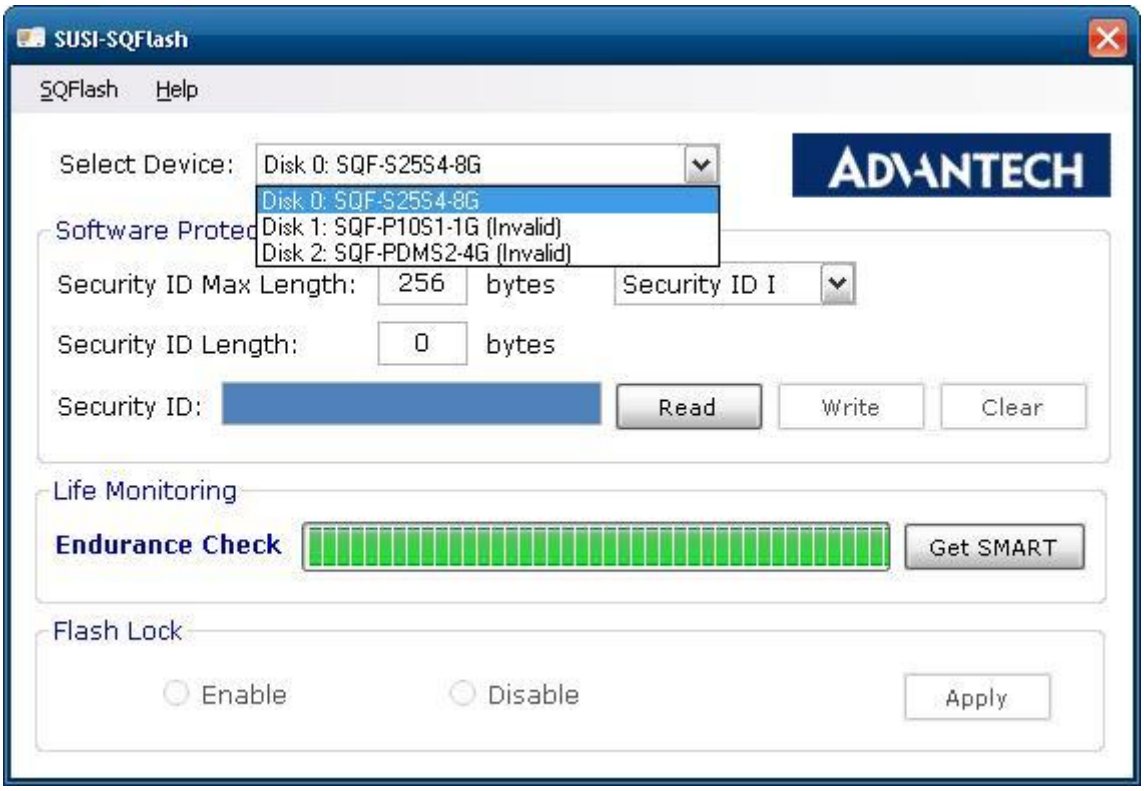


Figure-6

User can select invalid device to change Access Code. Click “Yes” In Figure-7 to change Access Code and you will get Figure-8 to input existing one and then user can change Access Code to new one.

Figure-7

If selected device already have Access Code inside, Main program will ask to check existing one in order to change it.

Figure-8

About Window – Advantech Copyright

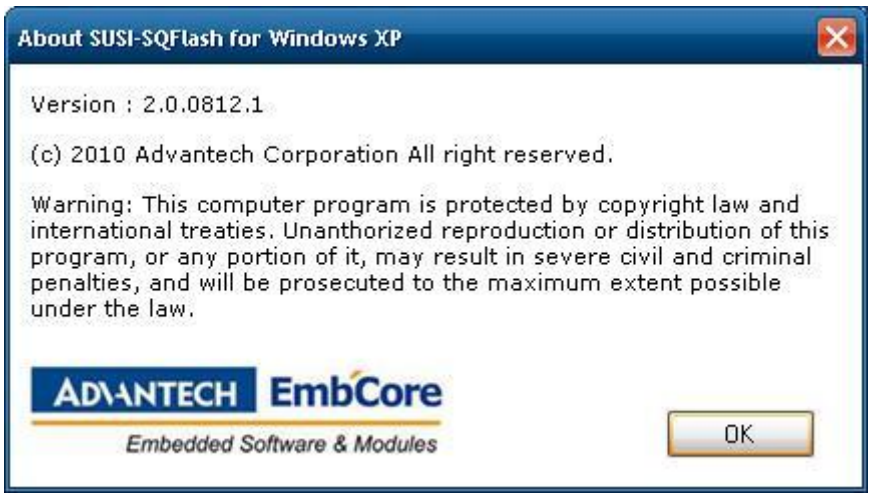


Figure-9

Get Code Window – Protecting Sample Code

Click “Get Code” to show C++ protected sample code in a new dialog.

- C++ sample code
- Access Code and Security ID required

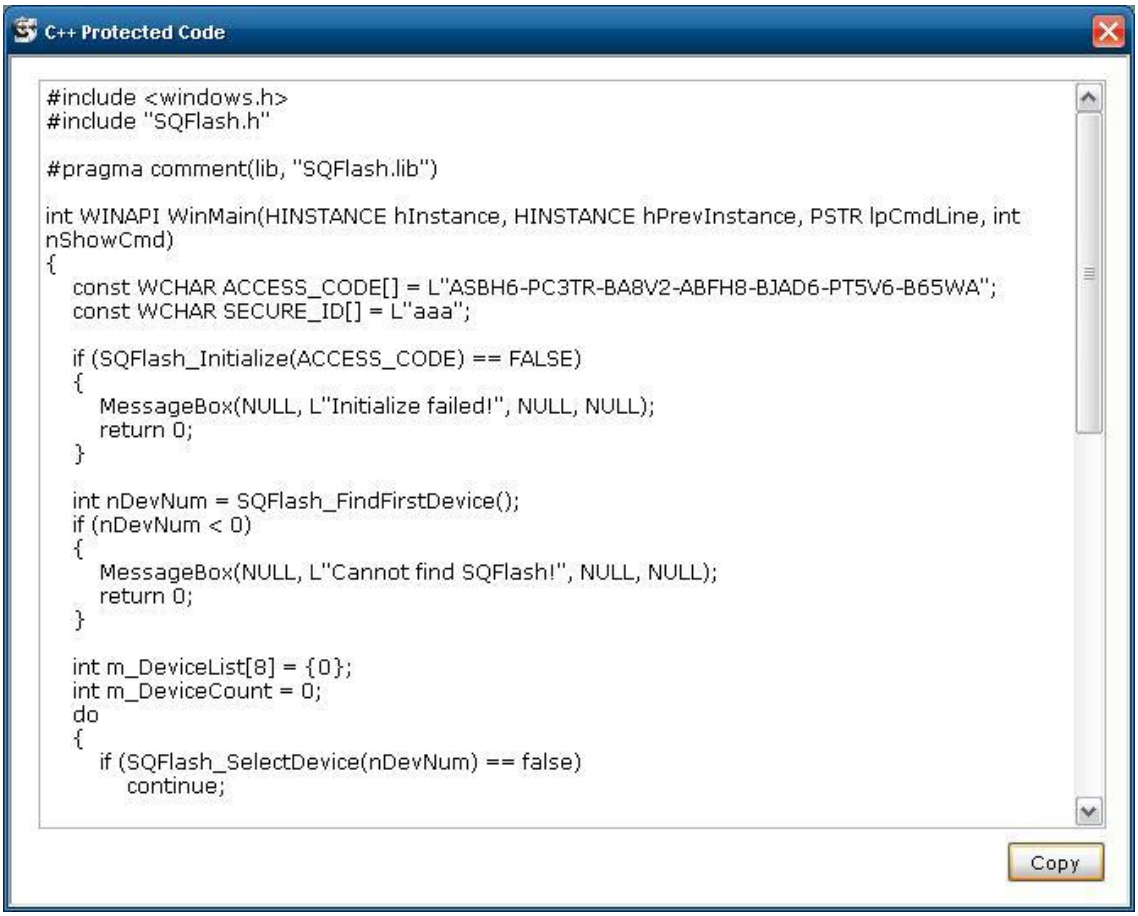


Figure-10

SMART Window – Get SMART attribute

Click “Get SMART” to S.M.A.R.T. attribute in a new dialog.

- Max Program
  In CF card max program and erase cycles.
- Average Program
  In CF card average program and erase cycles.
- Power On Time;
  Power on accumulates time.
- ECC Count;
  Error correct code number of times counting.
- Endurance Check;
  Endurance Check(%) is the result of (Average P/E cycles) / (Max P/E cycles).

Figure-11

SQFlash Utility API

Programming Overview

SQFlash-CF.lib is a static link library that exports all the API functions.

1. **SQFlash_FindFirstDevice**
   Initial command to get first disk number of SQFlash device

2. **SQFlash_FindNextDevice**
   Initial command to get next disk number of SQFlash device

3. **SQFlash_GetDeviceModelName**
   Initial command to get SQFlash device model name

4. **SQFlash_SelectDevice**
   Initial command to select SQFlash device by disk number

5. **SQFlash_Initialize**
   Input valid access code in order to initial SQFlash Utility

6. **SQFlash_UnInitialize**
   Uninitialize SQFlash library

7. **SQFlash_GetSmartAttribute**
   S.M.A.R.T. attribute contains 5 data in Advantech SQFlash
   - Max Program
     In CF card max program and erase cycles.
   - Average Program
     In CF card average program and erase cycles.
   - Power On Time;
     Power on accumulates time.
   - ECC Count;
     Error correct code number of times counting.
   - Endurance Check;
     Endurance Check(%) is the result of (Average P/E cycles) / (Max P/E cycles).

8. **SQFlash_IsUnusedSectorExist**
   Find available sector to restore security ID

9. **SQFlash_SetSecurityID**
   Support maximum length to 256 bytes security string which will write into the hidden area of Advantech SQFlash.

10. **SQFlash_GetSecurityID**
    Read Security ID from hidden area

11. **SQFlash_ClearSecurityID**
    Clear Security ID from hidden area

12. **SQFlash_GetSecurityID2_MaxLen**
    Get maximum length security string which will write into Advantech SQFlash.

13. **SQFlash_SetSecurityID2**
    Write Security ID 2 into Advantech SQFlash

14. **SQFlash_GetSecurityID2**
    Read Security ID 2 from Advantech SQFlash

15. **SQFlash_ClearSecurityID2**
    Clear Security ID from Advantech SQFlash

16. **SQFlash_IsSupportFlashLock**
    Check target platform support Flash Lock or not.

17. **SQFlash_IsFlashLockEnable**
    Check target platform Flash Lock be enabled or not.

18. **SQFlash_EnableFlashLock**
    Enable Flash Lock feature.

19. **SQFlash_DisableFlashLock**
    Disable Flash Lock feature.

Initialize Function

Syntax:

int SQFlash_FindFirstDevice();

Parameters:

None

Return Value:

First SQFlash Device Disk Number

Syntax:

int SQFlash_FindNextDevice();

Parameters:

None

Return Value:

Next SQFlash Device Disk Number

Syntax:

BOOL SQFlash_GetDeviceModelName(TCHAR * pszModelName, DWORD cchBuffer);

Parameters:

pszModelName

[out] Pointer to the buffer that will receive the model name.

cchBuffer

[in] Specifies the maximum number of characters to copy to the buffer, including the NULL character.

Return Value:

If the function succeeds, the return value is TRUE.

If the function fails, the return value is FALSE.

Syntax:

BOOL SQFlash_SelectDevice(int nDeviceNum);

Parameters:

None

Return Value:

If the function succeeds, the return value is TRUE.

If the function fails, the return value is FALSE.

Syntax:

BOOL SQFlash_Initialize(LPCSTR pszPassword);

Parameters:

pszPassword

[in] Valid SN

Return Value:

If the library was initialized successfully, the return value is TRUE. Otherwise, the return value is FALSE.

Syntax:

Void SQFlash_UnInitialize( );

Parameters:

None

Return Value:

None

GetSmartAttribute Function

Syntax:

```
BOOL SQFlash_GetSmartAttribute(PATA_SMART_ATTR_TABLE pASAT);
```

Parameters:

pASAT

[in] Pointer to a ATA_SMART_ATTR_TABLE structure to receive the Smart Attribute.

ATA_SMART_ATTR_TABLE Structure

The ATA_SMART_ATTR_TABLE Structure contains Smart Attributes.

Syntax:

```
typedef struct _ATA_SMART_ATTR_TABLE
{
    DWORD dwMaxProgram;
    DWORD dwAverageProgram;
    DWORD dwEnduranceRemainLife;
    DWORD dwPowerOnTime;
    DWORD dwEccCount;
} ATA_SMART_ATTR_TABLE, *PATA_SMART_ATTR_TABLE;
```

Members:

dwMaxProgram

Max Program/Erase Cycles

dwAverageProgram

Average Program/Erase Cycles

dwEnduranceRemainLife

Endurance (%) Remain Life

dwPowerOnTime

Power On Time second)

dwEccCount

ECC Count

SetSecurityID Function

Syntax:

BOOL SQFlash_IsUnusedSectorExist();

Parameters:

None

Return Value:

If the function succeeds, the return value is TRUE.

If the function fails, the return value is FALSE.

Syntax:

BOOL SQFlash_SetSecurityID(LPCSTR pszSecurityID);

Parameters:

pszSecurityID

[in] Pointer to a null-terminated string to be used as the new Security ID.

Return Value:

If the function succeeds, the return value is TRUE.

If the function fails, the return value is FALSE.

GetSecurityID Function

Syntax:

```
BOOL SQFlash_GetSecurityID(PCHAR pszSecurityID, DWORD
cchBuffer);
```

Parameters:

pszSecurityID

[out] Pointer to the buffer that will receive the Security ID.

cchBuffer

[in] Specifies the maximum number of characters to copy to the buffer, including the NULL character.

Return Value:

If the function succeeds, the return value is TRUE.

If the function fails, the return value is FALSE.

ClearSecurityID Function

Syntax:

BOOL SQFlash_ClearSecurityID();

Parameters:

None

Return Value:

If the function succeeds, the return value is TRUE.

If the function fails, the return value is FALSE.

SetSecurityID2 Function

Syntax:

int SQFlash_GetSecurityID2_MaxLen();

Parameters:

None

Return Value:

Maximum Length to support Security ID 2

Syntax:

BOOL SQFlash_SetSecurityID2(LPCSTR pszSecurityID);

Parameters:

pszSecurityID

[in] Pointer to a null-terminated string to be used as the new Security ID 2.

Return Value:

If the function succeeds, the return value is TRUE.

If the function fails, the return value is FALSE.

GetSecurityID2 Function

Syntax:

BOOL SQFlash_GetSecurityID2(PCHAR pszSecurityID, DWORD cchBuffer);

Parameters:

pszSecurityID

[out] Pointer to the buffer that will receive the Security ID2.

cchBuffer

[in] Specifies the maximum number of characters to copy to the buffer, including the NULL character.

Return Value:

If the function succeeds, the return value is TRUE.

If the function fails, the return value is FALSE.

ClearSecurityID2 Function

Syntax:

```
BOOL SQFlash_ClearSecurityID2();
```

Parameters:

None

Return Value:

If the function succeeds, the return value is TRUE.

If the function fails, the return value is FALSE.

IsSupportFlashLock Function

Syntax:

```
BOOL SQFlash_IsSupportFlashLock(HWND hWnd, UINT msgID);
```

Parameters:

hWnd

[in] Handle to a window. This window will receive processing progress information by window message.

msgID

[in] User defined window message for processing progress information. The wParam with this message represents current progress and lParam is the maximum progress value.

Return Value:

If the function succeeds, the return value is TRUE.

If the function fails, the return value is FALSE.

IsFlashLockEnable Function

Syntax:

BOOL SQFlash_IsFlashLockEnable(HWND hWnd, UINT msgID);

Parameters:

hWnd

[in] Handle to a window. This window will receive processing progress information by window message.

msgID

[in] User defined window message for processing progress information. The wParam with this message represents current progress and lParam is the maximum progress value.

Return Value:

If the function succeeds, the return value is TRUE.

If the function fails, the return value is FALSE.

EnableFlashLock Function

Syntax:

BOOL SQFlash_EnableFlashLock(LPCSTR pszPassword, HWND hWnd, UINT msgID);

Parameters:

pszPassword

[in] Valid SN

hWnd

[in] Handle to a window. This window will receive processing progress information by window message.

msgID

[in] User defined window message for processing progress information. The wParam with this message represents current progress and lParam is the maximum progress value.

Return Value:

If the function succeeds, the return value is TRUE.

If the function fails, the return value is FALSE.

DisableFlashLock Function

Syntax:

BOOL SQFlash_DisableFlashLock(HWND hWnd, UINT msgID);

Parameters:

hWnd

[in] Handle to a window. This window will receive processing progress information by window message.

msgID

[in] User defined window message for processing progress information. The wParam with this message represents current progress and lParam is the maximum progress value.

Return Value:

If the function succeeds, the return value is TRUE.

If the function fails, the return value is FALSE.

About

Advantech's Embedded Core Service (Emb'Core) is an open business model that provides integrated boards, modules and software services to speed up application development. The key proficiencies of Embedded Core Services are software, firmware, module, and board integration capability that offer customized solutions.
For further information please email: EmbCore@advantech.com

About Advantech
Founded in 1983, Advantech is a global leading ePlatform services provider of web-based technology, computing platforms and customization services to empower innovations in the connected eWorld. Advantech cooperates closely with partners to help provide complete solutions for a wide array of applications in various industries. Advantech delivers more than a thousand products and platform solutions in 5 main categories: Industrial & Network Computing, Embedded Computing, Applied Computing, eVideo Solutions, and eAutomation. With more than 2,700 talented people, Advantech operates an extensive support, sales and marketing network in 18 countries and 36 major cities. Advantech delivers efficient time-to-market services to all worldwide customers. (Corporate Website: www.advantech.com).