ICS 03.220.20
CCS R 80

团 体 标 准

T/ITS 0149—2021

智能交通运输系统 终端设备安全芯片 应用接口规范

Interface Specifications of Security Chip in the Terminal of the Intelligent Transport System

2021-12-07 发布 2022-03-01 实施

中国智能交通产业联盟 发布

目　　次

前　言

本部分按照 GB/T 1.1-2020 给出的规则起草。

本文件参考 GM/T 0018-2012 《公钥密码基础设施应用技术体系 密码设备应用接口规范》和 GM/T 0016-2012 《智能密码钥匙 密码应用接口规范》，与 GM/T 0018-2012 《公钥密码基础设施应用技术体系 密码设备应用接口规范》和 GM/T 0016-2012 《智能密码钥匙 密码应用接口规范》的一致程度为非等效。

本文件由中国智能交通产业联盟提出并归口。

请注意本文件的某些内容可能涉及专利。本文件的发布机构不承担识别这些专利的责任。

本文件主要起草单位：上海汽车集团股份有限公司、格尔软件有限公司、上海芯钛信息科技有限公司、联创汽车电子有限公司、北京中电华大电子设计有限责任公司、高通技术公司、郑州信大捷安信息技术股份有限公司、中国移动通信集团有限公司、东软集团股份有限公司。

本文件起草人：张显宏、高吉、张敏、张颖奇、李澜涛、赵炜铭、任佳憨、陈俊昌、周康乐、高珊、赵敬超、王睿、陈书平、刘献伦、康亮、彭金辉、田野、粟栗、李凤、祁帅、张骞。

智能交通运输系统 终端设备安全芯片 应用接口规范

1 范围

本文件规定了智能交通运输系统终端设备安全芯片应用接口，描述了应用接口的函数、数据类型、参数的定义和设备的安全要求。

本文件适用于智能交通运输系统终端设备安全芯片的研制、使用和检测。

2 规范性引用文件

下列文件中的内容通过文中的规范性引用而构成本文件必不可少的条款。其中注日期的引用文件，仅该日期对应的版本适用于本文件；不注日期的引用文件，其最新版本（包括所有的修改单）适用于本文件。

GM/T 0016-2012 智能密码钥匙 密码应用接口规范

GM/T 0018-2012 公钥密码基础设施应用技术体系 密码设备应用接口规范

3 术语和定义、缩略语

3.1 术语和定义

下列术语和定义适用于本文件。

3.1.1

算法标识 algorithm identifier

用于标明算法机制的数字化信息。

3.1.2

非对称密码算法 asymmetric cryptographic algorithm

加解密使用不同密钥的密码算法。其中一个密钥（公钥）可以公开，另一个密钥（私钥）必须保密，且由公钥求解私钥是计算不可行的。

3.1.3

解密 decryption

加密过程对应的逆过程。

3.1.4

加密 encryption

对数据进行密码变换以产生密文的过程。

3.1.5

密钥加密密钥 key-encrypting key

又称二级密钥(Secondary Key)或密钥传送密钥(key Transport key)，用于对密钥进行加解密。

3.2 缩略语

下列缩略语适用于本规范：

API：应用编程接口（Application Programming Interface）

PIN：个人身份识别码（Personal Identification Number)

MAC：消息鉴别码（Message Authentication Code）

ECC：椭圆曲线密码算法（Elliptic Curve Cryptography）

4 结构模型

终端设备安全芯片应用接口位于终端设备上位机应用程序与安全芯片之间，如图 1 所示。

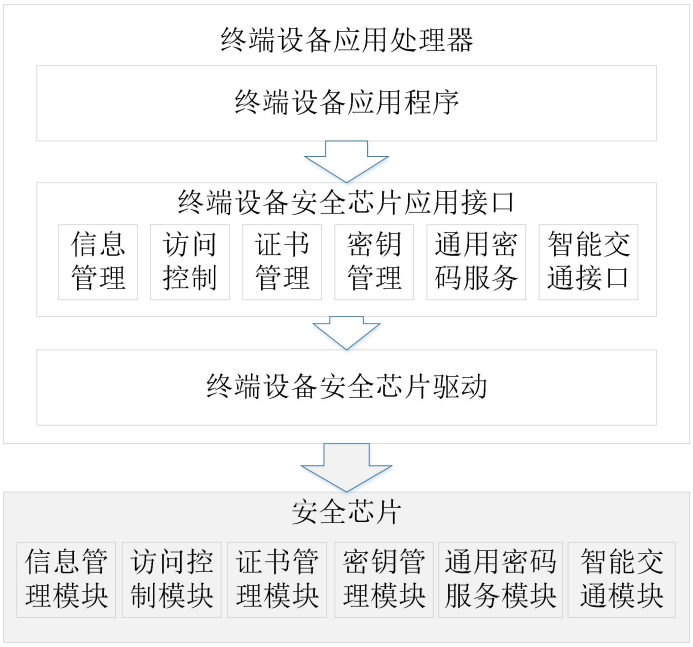

图 1 终端设备安全芯片应用调用模型

5 数据类型定义

5.1 算法标识

5.1.1 分组密码算法标识

分组密码算法标识包含密码算法的类型和加密模式。

分组密码算法标识的编码规则为：从低位到高位，第 0 位到第 7 位按位表示分组密码算法工作模式，第 8 位到第 31 位按位表示分组密码算法类型，分组密码算法的标识如表 1 所示。

表 1 分组密码算法标识表

标签	标识符	描述
ITS_SM4_ECB	0x00000401	SM4 算法 ECB 加密模式
ITS_SM4_CBC	0x00000402	SM4 算法 CBC 加密模式
ITS_SM4_CFB	0x00000404	SM4 算法 CFB 加密模式
ITS_SM4_OFB	0x00000408	SM4 算法 OFB 加密模式
ITS_SM4_MAC	0x00000410	SM4 算法 MAC 运算
ITS_SM4_CCM	0x00000416	SM4 算法 CCM 加密模式
ITS_AES_ECB	0x00002001	AES 算法 ECB 加密模式
ITS_AES_CBC	0x00002002	AES 算法 CBC 加密模式
ITS_AES_CFB	0x00002004	AES 算法 CFB 加密模式
ITS_AES_OFB	0x00002008	AES 算法 OFB 加密模式
ITS_AES_MAC	0x00002010	AES 算法 MAC 运算
ITS_AES_CCM	0x00002016	AES 算法 CCM 加密模式
0x00003000-0x80000000		为其他对称密码算法预留

5.1.2 非对称算法标识

非对称密码算法标识仅定义了密码算法的类型，在使用非对称算法进行数字签名运算时，可将非对称密码算法标识符与密码杂凑算法标识符进行“或”运算后使用，如“RSA with SHA1”可表示为ITS_RSA | ITS_SHA1，即0x00010002，“|”表示“或”运算。

非对称密码算法标识的编码规则为：从低位到高位，第0位到第7位为0，第8位到第15位按位表示非对称密码算法的算法协议，如果所表示的非对称算法没有相应的算法协议则为0，第16位到第31位按位表示非对称密码算法类型，非对称密码算法的标识如表2所示。

表2 非对称密码算法标识表

标签	标识符	描述
ITS_RSA	0x00010000	RSA算法
ITS_SM2_1	0x00020100	椭圆曲线签名算法
ITS_SM2_2	0x00020200	椭圆曲线密钥交换协议
ITS_SM2_3	0x00020400	椭圆曲线加密算法
ITS_ECC_1	0x00030100	NISTP256椭圆曲线签名算法
ITS_ECC_2	0x00030200	NISTP256椭圆曲线密钥交换协议
ITS_ECC_3	0x00030400	NISTP256椭圆曲线加密算法
ITS_ECC_4	0x00040100	BrainpoolP256椭圆曲线签名算法
ITS_ECC_5	0x00040200	BrainpoolP256椭圆曲线密钥交换协议
ITS_ECC_6	0x00040400	BrainpoolP256椭圆曲线加密算法
0x00040400-0x80000000		为其他非对称密码算法预留

5.1.3 杂凑算法标识

密码杂凑算法标识符可以在进行密码杂凑运算或计算MAC时应用，也可以与非对称密码算法标识符进行“或”运算后使用，表示签名运算前对数据进行密码杂凑运算的算法类型。

密码杂凑算法标识的编码规则为：从低位到高位，第0位到第7位表示密码杂凑算法，第8位到第31位为0，密码杂凑算法的标识如表3所示。

表3 杂凑算法标识表

标签	标识符	描述
ITS_SM3	0x00000001	SM3杂凑算法
ITS_SHA1	0x00000002	SHA1杂凑算法
ITS_SHA256	0x00000004	SHA256杂凑算法
ITS_SHA384	0x00000008	SHA384杂凑算法
0x0000010-0x000000FF		为其杂凑算法预留

5.2 复合数据类型

5.2.1 版本信息

类型定义：

```
typedef struct
{
    UCHAR ucHwVersion[32];
    UCHAR ucFwVersion[32];
    UCHAR ucSwVersion[32];
}Version_t;
```

数据项描述参见表4。

表 4 版本信息描述

数据项	类型	含义	备注
ucHwVersion	SCHAR 数组	安全芯片硬件版本号	以'\0'为结束符的 ASCII 字符串
ucFwVersion	SCHAR 数组	安全芯片固件版本号	以'\0'为结束符的 ASCII 字符串
ucSwVersion	SCHAR 数组	安全芯片应用接口版本号	以'\0'为结束符的 ASCII 字符串

5.2.2 安全芯片信息

类型定义：

```
typedef struct
{
    Version_t Version;
    SCHAR scManufacturer[64];
    SCHAR scIssuer[64];
    SCHAR scLable[32];
    SCHAR scSerialNumber[32];
    BYTE  byReserved[64];
}ChipInfo_t;
```

数据项描述参见表 5。

表 5 设备信息描述

数据项	类型	含义	备注
Version	Version_t	版本号	
scManufacturer	SCHAR 数组	芯片生产厂商信息	以'\0'为结束符的 ASCII 字符串
scIssuer	SCHAR 数组	芯片发行厂商信息	以'\0'为结束符的 ASCII 字符串
scLable	SCHAR 数组	芯片标签	以'\0'为结束符的 ASCII 字符串
scSerialNumber	SCHAR 数组	序列号	以'\0'为结束符的 ASCII 字符串
byReserved	BYTE 数组	保留扩展	

5.2.3 证书类型数据结构

类型定义：

```
typedef enum
{
    ITS_CA = 1,
    ITS_E_CERT,
    ITS_BASE_SM2_CERT,
    ITS_BASE_RSA_CERT,
}ITSCertType_e;
```

数据项描述参见表 6。

表 6 证书类型描述

数据项	类型	含义	备注
ITS_CA	enum	智能交通 CA 证书	IEEE1609.2 或 CCSA YD/T 3957-2021 证书
ITS_E_CERT	enum	智能交通车载终端证书	IEEE1609.2 或 CCSA YD/T 3957-2021 证书
ITS_BASE_SM2_CERT	enum	基础的 SM2 证书	格式为 x.509 证书
ITS_BASE_RSA_CERT	enum	基础的 RSA 证书	格式为 x.509 证书

5.3 索引定义

索引定义是为了证书以及密钥的管理，密钥的生成、导入、导出、删除，证书的导入、导出、删除等操作均通过索引完成。

索引由终端设备安全芯片内部创建并管理，每个索引可以存储对称密钥、非对称密钥对、证书等，由索引号标识。

安全芯片可根据自身存储空间自行规划密钥、证书等存储的索引范围。

6 接口函数

6.1 安全芯片信息管理接口

6.1.1 概述

安全芯片管理主要完成读取安全芯片状态、读取安全芯片信息和读取安全芯片版本信息的操作。安全芯片管理系列函数如表 7 所示。

表 7 安全芯片管理系列函数

函数名称	功能
ITS_GetChipState	读取安全芯片状态
ITS_GetChipInfo	读取安全芯片信息
ITS_GetChipVersion	读取安全芯片版本信息

6.1.2 安全芯片状态查询

函数原型　INT32 ITS_ GetChipState (UINT32* pu32State)

功能描述　读取安全芯片状态

参数　　pu32State　　[OUT] 0-空闲；1-运算中；2-故障

返回值　　ITS_OK：成功

　　其他：错误码

6.1.3 安全芯片信息查询

函数原型　INT32 ITS_ GetChipInfo (ChipInfo_t *ptChipInfo)

功能描述　读取安全芯片信息

参数　　ptChipInfo　　[OUT] 安全芯片信息

返回值　　ITS_OK：成功

　　其他：错误码

6.1.4 安全芯片版本查询

函数原型　INT32 ITS_ GetChipVersion (Version_t *ptChipVersion)

功能描述　读取安全芯片版本信息

参数　　ptChipVersion　　[OUT] 安全芯片版本信息

返回值　　ITS_OK：成功

　　其他：错误码

6.2 访问控制接口

6.2.1 概述

访问控制主要完成安全芯片 PIN 码相关操作和主控密钥更新。访问控制系列函数如表 8 所示。

表 8 访问控制系列函数

函数名称	功能
ITS_VerifyPin	设备认证
ITS_ChangePin	修改 PIN
ITS_ReSetPin	重置 PIN
ITS_UpdateMasterKey	更新主控密钥

6.2.1 校验 PIN

函数原型　INT32 ITS_VerifyPin(CHAR8 *pcPin)

功能描述　验证 PIN

参数　pcPin　[IN] PIN 明文字符串，长度[4-16]

返回值　ITS_OK：　成功

其他：　错误码

6.2.2 修改 PIN

函数原型　INT32 ITS_ChangePin(CHAR8* pcOldPin, CHAR8* pcNewPin)

功能描述　修改 PIN

参数　pcOldPin　[IN] 旧的 PIN 明文字符串，长度[4-16]

pcNewPin　[IN] 新的 PIN 明文字符串，长度[4-16]

返回值　ITS_OK：　成功

其他：　错误码

6.2.3 重置 PIN

函数原型　INT32 ITS_ResetPin(UNIT8*　pu8Mkey, UINT8*　pu8Mac)

功能描述　重置 PIN

参数　pu8Mkey　[IN] 使用安全芯片主控密钥加密新的 PIN 密文

pu8Mac　[IN] 使用安全芯片主控密钥计算 Mkey 的 MAC 值

返回值　ITS_OK：　成功

其他：　错误码

6.2.4 更新主控密钥

函数原型　INT32 ITS_UpdateMasterKey (UNIT8* pu8Mkey,uint8* pu8Mac,UINT8* pu8CV)

功能描述　更新安全芯片主控密钥

参数　pu8Mkey　[IN] 新的安全芯片主控密钥密文，使用当前安全芯片主控密钥加密保护

pu8Mac　[IN] 使用当前的安全芯片主控密钥对 pu8MKey 计算 MAC

pu8CV　[IN] 新的安全芯片主控密钥校验值

返回值　ITS_OK：　成功

其他：　错误码

6.3 证书管理接口

6.3.1 概述

证书管理主要完成证书的导入、导出和删除。证书管理系列函数如表 9 所示。

表 9 证书管理系列函数

函数名称	功能
ITS_ImportCert	向安全芯片导入证书
ITS_ExportCert	从安全芯片导出证书
ITS_DeleteCert	删除安全芯片中的证书

6.3.2 证书导入

函数原型　INT32 ITS_ImportCert(UINT32 u32Index, ITSCertType_e eCertType, UINT32 u32Len, UCHAR * pucCert)

功能描述　向安全芯片导入证书，证书类型见 5.2.3

参数　u32Index　[IN] 索引

eCertType　[IN] 证书类型

u32Len　[IN] 证书长度

pucCert　[IN] 证书数据

返回值　ITS_OK：　成功

其他：　错误码

6.3.2 证书导出

函数原型　INT32 ITS_ExportCert(UINT32 u32Index, ITSCertType_e* eCertType, UINT32* pu32CertLen, UCHAR* pucCertData)

功能描述　从安全芯片导出证书

参数　u32Index　[IN] 索引

	eCertType	[OUT] 证书类型
	pu32CertLen	[OUT] 证书长度
	pu32CertData	[OUT] 证书数据
返回值	ITS_OK：	成功
	其他：	错误码

6.3.4 证书删除

函数原型　INT32 ITS_DeleteCert(UINT32 u32Index)

功能描述　删除安全芯片中的证书

参数	u32Index	[IN] 索引
返回值	ITS_OK：	成功
	其他：	错误码

6.4 密钥管理接口

6.4.1 概述

密钥管理主要完成非对称算法 SM2、ECC 和 RSA 的密钥对的生成、查询和删除，对称密钥的导入、导出和删除。密钥管理系列函数如表 10 所示。

表 10 密钥管理系列函数

函数名称	功能
ITS_GenSM2KeyIndex	SM2 密钥对生成
ITS_DeleteSM2Key	SM2 密钥删除
ITS_QuerySM2Key	SM2 密钥查询
ITS_GenECCKeyIndex	ECC 密钥对生成
ITS_DeleteECCKey	ECC 密钥删除
ITS_QueryECCKey	ECC 密钥查询
ITS_GenRSAKeyIndex	RSA 密钥对生成
ITS_DeleteRSAKey	RSA 密钥删除
ITS_QueryRSAKey	RSA 密钥查询
ITS_ImportSymmKey	导入对称密钥
ITS_DeleteSymmKey	删除对称密钥
ITS_EXportSymmKey	导出对称密钥

6.4.2 SM2 密钥对生成

函数原型　INT32 ITS_GenSM2KeyIndex(UINT32 u32KeyIndex, UINT8* pu8X, UINT8* pu8Y)

功能描述　产生 SM2 密钥对，密钥对安全芯片内部存储

参数	u32KeyIndex	[IN] 密钥存储的索引号
	pu8X	[OUT] 公钥 X，定长 32 字节
	pu8Y	[OUT] 公钥 Y，定长 32 字节

返回值	ITS_OK：	成功
	其他：	错误码

6.4.3 SM2 密钥删除

函数原型　INT32 ITS_DeleteSM2Key(UNIT32 u32KeyIndex);

功能描述　删除 SM2 密钥对

参数	u32KeyIndex	[IN] 密钥存储的索引号
返回值	ITS_OK：	成功
	其他：	错误码

6.4.4 SM2 密钥查询

函数原型　INT32 ITS_QuerySM2Key(UINT32 u32KeyIndex, UINT32 * pu32Xlen, UINT8* pu8X, UINT32 * pu32Ylen, UINT8* pu8Y);

功能描述　查询 SM2 密钥对，读取公钥

参数	u32KeyIndex	[IN] 密钥存储的索引号
	pu32Xlen	[OUT] 公钥 X 长度
	pu8X	[OUT] 公钥 X，定长 32 字节
	pu32Ylen	[OUT] 公钥 Y 长度
	pu8Y	[OUT] 公钥 Y，定长 32 字节
返回值	ITS_OK：	成功
	其他：	错误码

6.4.5 ECC 密钥对生成

函数原型　INT32 ITS_GenEccKeyIndex(UINT32 u32KeyIndex, UINT32 u32GroupID, UINT32 * pu32Xlen, UINT8* pu8X, UINT32 * pu32Ylen, UINT8* pu8Y);

功能描述　产生 ECC 密钥对，密钥对安全芯片内部存储

参数	u32KeyIndex	[IN] 密钥存储的索引号
	u32GroupID	[IN] 椭圆曲率 ID。1-nistP256; 2-brainpoolP256r1，brainpoolP256r1 为非必实现算法，下同。
	pu32Xlen	[OUT] 公钥 X 长度
	pu8X	[OUT] 公钥 X
	pu32Ylen	[OUT] 公钥 Y 长度
	pu8Y	[OUT] 公钥 Y
返回值	ITS_OK：	成功

	其他：	错误码

6.4.6 ECC 密钥删除

函数原型	INT32 ITS_DeleteECCKey(UINT32 u32KeyIndex);	
功能描述	删除 ECC 密钥对	
参数	u32KeyIndex	[IN] 密钥存储的索引号
返回值	ITS_OK：	成功
	其他：	错误码

6.4.7 ECC 密钥查询

函数原型	INT32 ITS_QueryEccKey(UINT32 u32KeyIndex, UINT32 * pu32GroupID, UINT32 * pu32Xlen, UINT8* pu8X, UINT32 * pu32Ylen, UINT8* pu8Y);	
功能描述	查询 ECC 密钥对，读取公钥	
参数	u32KeyIndex	[IN] 密钥存储的索引号
	pu32GroupID	[OUT] 椭圆曲率 ID
	pu32Xlen	[OUT] 公钥 X 长度
	pu8X	[OUT] 公钥 X，定长 32 字节
	pu32Ylen	[OUT] 公钥 Y 长度
	pu8Y	[OUT] 公钥 Y，定长 32 字节
返回值	ITS_OK：	成功
	其他：	错误码

6.4.8 RSA 密钥对生成

函数原型	INT32 ITS_GenRsaKeyIndex(UINT32 u32KeyIndex, UINT32 u32Modlen, UINT32 u32Elen, UINT8* pu8E, UINT32 * pu32Nlen, UINT8* pu8N);	
功能描述	产生 RSA 密钥对，密钥对安全芯片内部存储	
参数	u32KeyIndex	[IN] 密钥存储的索引号
	u32Modlen	[IN] 模长
	u32Elen	[IN] 指数长度
	pu8E	[IN] 指数
	pu32Nlen	[OUT] 公钥 N 长度
	pu8N	[OUT] 公钥 N
返回值	ITS_OK：	成功
	其他：	错误码

6.4.9 RSA 密钥删除

函数原型	INT32 ITS_DeleteRSAKey(UINT32 u32KeyIndex);	
功能描述	删除 RSA 密钥对	
参数	u32KeyIndex	[IN] 密钥存储的索引号
返回值	ITS_OK：	成功
	其他：	错误码

6.4.10 RSA 密钥查询

函数原型	INT32 ITS_QueryRsaKey(UINT32 u32KeyIndex,UNIT32* pu32Nlen, UINT8* pu8N, UINT32 * pu32Elen, UINT8* pu8E);	
功能描述	查询 RSA 密钥对，读取公钥	
参数	u32KeyIndex	[IN] 密钥存储的索引号
	pu32Nlen	[OUT] 公钥 N 长度
	pu8N	[OUT] 公钥 N
	pu32Elen	[OUT] 公钥 E 长度
	pu8E	[OUT] 公钥 E
返回值	ITS_OK：	成功
	其他：	错误码

6.4.11 导入对称密钥

函数原型	INT32 ImportSymmKey(UINT32 u32Alg, UINT32 u32KeyIndex, UINT32 u32Lock, UINT32 u32KeyLen, UINT8* pu8Key, UINT8* pu8Mac);	
功能描述	导入对称密钥	
参数	u32Alg	[IN] 导入密钥算法: 0-ALL; 1-AES; 2-SM4
	u32KeyIndex	[IN] 密钥存储的索引号
	u32Lock	[IN] 密钥是否锁定，不可修改: 0-不锁定; 1-锁定
	u32KeyLen	[IN] 密钥的长度
	pu8Key	[IN] 密钥密文，使用安全芯片主控密钥加密密钥明文得到（明文:2 字节长度+密钥明文+8000 补位，不强制补位）
	pu8Mac	[IN] 校验值，使用安全芯片主控密钥对密钥信息计算的 MAC 值，4 字节
返回值	ITS_OK：	成功
	其他：	错误码
备注	MAC 以 SM4 CBC 方式计算。下同	

6.4.12 删除对称密钥

函数原型　INT32 ITS_DeleteSymmKey(UINT32 u32KeyIndex);

功能描述　删除对称密钥

参数　　u32KeyIndex　　[IN] 密钥存储的索引号

返回值　　ITS_OK:　　成功

　　其他:　　错误码

6.4.13 导出对称密钥

函数原型　INT32 ITS_ExportSymmKey(UINT32 u32KeyIndex, UINT32 u32KekIndex, UINT32 * pu32Alg, UINT32 * pu32KeyLen, UINT8* pu8Key, UINT8* pu8CV);

功能描述　导出对称密钥

参数　　u32KeyIndex　　[IN] 密钥存储的索引号

　　u32KekIndex　　[IN] 保护密钥索引号

　　pu32Alg　　[OUT] 密钥算法: 0-ALL; 1-AES; 2-SM4

　　pu32KeyLen　　[OUT] 密钥长度

　　pu8Key　　[OUT] 密钥密文，使用保护密钥加密密钥明文得到

　　（明文:2 字节长度+密钥明文+8000 补位，不强制补位）

　　pu8CV　　[OUT] 密钥的校验值

返回值　　ITS_OK:　　成功

　　其他:　　错误码

6.5 通用密码服务接口

6.5.1 概述

通用密码服务函数提供对称算法运算、非对称算法运算、密码杂凑运算等功能。通用密码服务系列函数如表 11 所示。

表 11 通用密码服务系列接口函数

函数名称	功能
ITS_GenRandom	产生随机数
ITS_SM2Sign	SM2 数据签名，使用外部密钥
ITS_SM2SignIndex	SM2 数据签名，使用内部密钥
ITS_SM2Verify	SM2 验证签名
表 11 （第 2 页/共 2 页）	
ITS_SM2PKEncrypt	SM2 公钥加密
ITS_SM2SKDecrypt	SM2 私钥解密，使用外部密钥
ITS_SM2SKDecryptIndex	SM2 私钥解密，使用内部密钥
ITS_SM3Init	SM3 初始化
ITS_SM3Update	SM3 数据哈希计算
ITS_SM3Final	SM3 数据哈希计算结束

ITS_SM4ECB	SM4 ECB 对称运算，使用外部密钥
ITS_SM4ECBIndex	SM4 ECB 对称运算，使用内部密钥
ITS_SM4CBC	SM4 CBC 对称运算，使用外部密钥
ITS_SM4CBCIndex	SM4 CBC 对称运算，使用内部密钥
ITS_SM4CCM	SM4 CCM 对称运算，使用外部密钥
ITS_SM4CCMIndex	SM4 CCM 对称运算，使用内部密钥
ITS_ECCSign	ECC 数据签名，使用外部密钥
ITS_ECCSignIndex	ECC 数据签名，使用内部密钥
ITS_ECCVerify	ECC 验证签名
ITS_ECCPKEncrypt	ECC 公钥加密
ITS_ECCSKDecrypt	ECC 私钥解密，使用外部密钥
ITS_ECCSKDecryptIndex	ECC 私钥解密，使用内部密钥
ITS_RsaSKEncrypt	RSA 私钥加密，使用外部密钥
ITS_RsaSKEncryptIndex	RSA 私钥加密，使用内部密钥
ITS_RsaPKDecrypt	RSA 公钥解密
ITS_RsaPKEncrypt	RSA 公钥加密
ITS_RsaSKDecrypt	RSA 私钥解密，使用外部密钥
ITS_RsaSKDecryptIndex	RSA 私钥解密，使用内部密钥
ITS_AESECB	AES ECB 对称运算，使用外部密钥
ITS_AESECBIndex	AES ECB 对称运算，使用内部密钥
ITS_AESCBC	AES CBC 对称运算，使用外部密钥
ITS_AESCBCIndex	AES CBC 对称运算，使用内部密钥
ITS_SHAInit	SHA 初始化
ITS_SHAUpdate	SHA 数据哈希计算
ITS_SHAFinal	SHA 数据哈希计算结束

6.5.2 生成随机数

函数原型　INT32 ITS_GenRandom(UINT32 u32Len, UINT8* pu8RND);

功能描述　生成随机数

参数　u32Len　[IN] 需要产生的随机数长度

　　pu8RND　[OUT] 随机数

返回值　ITS_OK：　成功

　　其他：　错误码

6.5.3 SM2 数据签名（外部密钥）

函数原型　INT32 ITS_SM2Sign(UINT8* pu8SK, UINT32 u32HashFlag, UINT32 u32DataLen, UINT8* pu8Data, UINT8* pu8R, UINT8* pu8S);

功能描述　SM2 使用外部密钥签名

参数　pu8SK　[IN] 私钥

　　u32HashFlag　[IN] 哈希标志：0-不哈希；1-先哈希后签名

　　u32DataLen　[IN] 待签名数据长度

　　pu8Data　[IN] 待签名数据

　　pu8R　[OUT] 签名 R

	pu8S	[OUT] 签名 S
返回值	ITS_OK：	成功
	其他：	错误码
备注	SM2 一般对应使用 SM3 哈希算法，ECC 一般对应使用 SHA256 哈希算法。 哈希标志为 0 时，HASH 值由外部计算传入，IDA 不属于本接口处理范围。 哈希标志为 1 时，HASH 值计算无 IDA 参与。下同。	

6.5.4 SM2 数据签名（内部密钥）

函数原型	INT32 ITS_SM2SignIndex(UINT32 u32KeyIndex, UINT32 u32HashFlag, UINT32 u32DataLen, UINT8* pu8Data, UINT8* pu8R, UINT8* pu8S);	
功能描述	SM2 使用内部密钥签名	
参数	u32KeyIndex	[IN] 密钥对索引
	u32HashFlag	[IN] 哈希标志：0-不哈希；1-先哈希后签名
	u32DataLen	[IN] 待签名数据长度
	pu8Data	[IN] 待签名数据
	pu8R	[OUT] 签名 R
	pu8S	[OUT] 签名 S
返回值	ITS_OK：	成功
	其他：	错误码

6.5.5 SM2 验证签名

函数原型	INT32 ITS_SM2Verify(UINT32 u32HashFlag, UINT8* pu8X, UINT8* pu8Y, UINT8* pu8R, UINT8* pu8S, UINT32 u32DataLen, UINT8* pu8Data);	
功能描述	SM2 签名验证	
参数	u32HashFlag	[IN] 哈希标志：0-不哈希；1-先哈希后签名
	pu8X	[IN] 公钥 X
	pu8Y	[IN] 公钥 Y
	pu8R	[IN] 签名 R
	pu8S	[IN] 签名 S
	u32DataLen	[IN] 待验证签名数据长度
	pu8Data	[IN] 待验证签名数据
返回值	ITS_OK：	成功
	其他：	错误码

6.5.6 SM2 公钥加密

函数原型 INT32 ITS_SM2PKEncrypt(UINT8* pu8X, UINT8* pu8Y, UINT32 u32Datalen, UINT8* pu8Data, UINT32 * pu32Outlen, UINT8* pu8OutData);

功能描述 SM2 公钥加密

参数		
	pu8X	[IN] 公钥 X
	pu8Y	[IN] 公钥 Y
	u32DataLen	[IN] 待加密数据长度
	pu8Data	[IN] 待加密数据
	pu32Outlen	[OUT] 输出数据长度
	pu8OutData	[OUT] 输出数据
返回值	ITS_OK:	成功
	其他:	错误码

6.5.7 SM2 私钥解密（外部密钥）

函数原型 INT32 ITS_SM2SKDecrypt(UINT8* pu8Key,uint32 u32Datalen, UINT8* pu8Data, UINT32 * pu32Outlen, UINT8* pu8OutData);

功能描述 SM2 使用外部私钥解密

参数		
	pu8Key	[IN] 私钥
	u32Datalen	[IN] 待运算数据长度
	pu8Data	[IN] 待运算数据
	pu32Outlen	[OUT] 输出数据长度
	pu8OutData	[OUT] 输出数据
返回值	ITS_OK:	成功
	其他:	错误码

6.5.8 SM2 私钥解密（内部密钥）

函数原型 INT32 ITS_SM2SKDecryptIndex(UINT32 u32KeyIndex,uint32 u32Datalen, UINT8* pu8Data, UINT32 * pu32Outlen, UINT8* pu8OutData);

功能描述 SM2 使用内部私钥解密

参数		
	u32KeyIndex	[IN] 密钥对索引
	u32Datalen	[IN] 待运算数据长度
	pu8Data	[IN] 待运算数据
	pu32Outlen	[OUT] 输出数据长度
	pu8OutData	[OUT] 输出数据

返回值	ITS_OK：	成功
	其他：	错误码

6.5.9 SM3 初始化

函数原型　INT32 ITS_SM3Init(void);

功能描述　SM3 哈希初始化

参数	无	
返回值	ITS_OK：	成功
	其他：	错误码

6.5.10 SM3 数据哈希计算

函数原型　INT32 ITS_SM3Update(UINT32 u32Datalen, UINT8* pu8Data);

功能描述　SM3 数据哈希计算

参数	u32Datalen	[IN] 数据长度
	pu8Data	[IN] 数据
返回值	ITS_OK：	成功
	其他：	错误码

6.5.11 SM3 数据哈希计算结束

函数原型　INT32 ITS_SM3Final(UINT8* pu8Hash);

功能描述　SM3 数据哈希计算结束，得到哈希值

参数	pu8Hash	[OUT] 哈希结果，定长 32 字节
返回值	ITS_OK：	成功
	其他：	错误码

6.5.12 SM4 ECB 对称运算（外部密钥）

函数原型　INT32 ITS_SM4ECB(UINT32 u32Mode, UINT8* pu8Key, UINT32 u32Datalen, UINT8* pu8Data, UINT32 * pu32Outlen, UINT8* pu8OutData);

功能描述　SM4 使用外部密钥，加解密 ECB 模式

参数	u32Mode	[IN] 运算模式：0-加密；1-解密
	pu8Key	[IN] 运算密钥
	u32Datalen	[IN] 待运算数据长度
	pu8Data	[IN] 待运算数据，需要外部补位
	pu32Outlen	[OUT] 运算结果长度
	pu8OutData	[OUT] 运算结果

返回值	ITS_OK：	成功
	其他：	错误码

6.5.13 SM4 ECB 对称运算（内部密钥）

函数原型 INT32 ITS_SM4ECBIndex(UINT32 u32Mode, UINT32 u32KeyIndex, UINT32 u32Datalen, UINT8* pu8Data, UINT32 * pu32Outlen, UINT8* pu8OutData);

功能描述 SM4 使用内部密钥，加解密 ECB 模式

参数	u32Mode	[IN] 运算模式：0-加密；1-解密
	u32KeyIndex	[IN] 运算密钥索引
	u32Datalen	[IN] 待运算数据长度
	pu8Data	[IN] 待运算数据，需要外部补位
	pu32Outlen	[OUT] 运算结果长度
	pu8OutData	[OUT] 运算结果
返回值	ITS_OK：	成功
	其他：	错误码

6.5.14 SM4 CBC 对称运算 （外部密钥）

函数原型 INT32 ITS_SM4CBC(UINT32 u32Mode, UINT8* pu8Key, UINT8* pu8IV, UINT32 u32Datalen, UINT8* pu8Data, UINT32 * pu32Outlen, UINT8* pu8OutData);

功能描述 SM4 使用外部密钥，加解密 CBC 模式

参数	u32Mode	[IN] 运算模式：0-加密；1-解密
	pu8Key	[IN] 运算密钥
	pu8IV	[IN] 初始化向量
	u32Datalen	[IN] 待运算数据长度
	pu8Data	[IN] 待运算数据，需要外部补位
	pu32Outlen	[OUT] 运算结果长度
	pu8OutData	[OUT] 运算结果
返回值	ITS_OK：	成功
	其他：	错误码

6.5.15 SM4 CBC 对称运算 （内部密钥）

函数原型 INT32 ITS_SM4CBCIndex(UINT32 u32Mode, UINT32 u32KeyIndex, UINT8* pu8IV, UINT32 u32Datalen, UINT8* pu8Data, UINT32 * pu32Outlen, UINT8* pu8OutData);

功能描述　SM4 使用内部密钥，加解密 CBC 模式

参数		
	u32Mode	[IN] 运算模式：0-加密；1-解密
	u32KeyIndex	[IN] 运算密钥索引
	pu8IV	[IN] 初始化向量
	u32Datalen	[IN] 待运算数据长度
	pu8Data	[IN] 待运算数据，需要外部补位
	pu32Outlen	[OUT] 运算结果长度
	pu8OutData	[OUT] 运算结果
返回值	ITS_OK：	成功
	其他：	错误码

6.5.16 SM4 CCM 对称运算 （外部密钥）

函数原型　INT32 ITS_SM4CCM(UINT32 u32Mode, UINT8* pu8Key, UINT32 u32IVLen, UINT8* pu8IV, UINT32 u32Datalen, UINT8* pu8Data, UNIT32* pu32Outlen, UINT8* pu8OutData);

功能描述　SM4 使用外部密钥，加解密 CCM 模式

参数		
	u32Mode	[IN] 运算模式：0-加密；1-解密
	pu8Key	[IN] 运算密钥
	u32IVLen	[IN] 初始化向量长度
	pu8IV	[IN] 初始化向量
	u32Datalen	[IN] 待运算数据长度
	pu8Data	[IN] 待运算数据，如果 nMode = 1，数据格式：密文||标签(16 字节)
	pu32Outlen	[OUT] 运算结果长度
	pu8OutData	[OUT] 运算结果，如果 nMode = 0，数据格式：密文||标签(16 字节)
返回值	ITS_OK：	成功
	其他：	错误码

6.5.17 SM4 CCM 对称运算 （内部密钥）

函数原型　INT32 ITS_SM4CCMIndex(UINT32 u32Mode, UINT32 u32KeyIndex, UINT32 u32IVLen, UINT8* pu8IV, UINT32 u32Datalen, UINT8* pu8Data, UNIT32* pu32Outlen, UINT8* pu8OutData);

功能描述　SM4 使用内部密钥，加解密 CCM 模式

参数	u32Mode	[IN] 运算模式：0-加密；1-解密
	u32KeyIndex	[IN] 运算密钥索引
	u32IVLen	[IN] 初始化向量长度
	pu8IV	[IN] 初始化向量
	u32Datalen	[IN] 待运算数据长度
	pu8Data	[IN] 待运算数据，如果 nMode = 1，数据格式：密文||标签(16 字节)
	pu32Outlen	[OUT] 运算结果长度
	pu8OutData	[OUT] 运算结果，如果 nMode = 0，数据格式：密文||标签(16 字节)
返回值	ITS_OK：	成功
	其他：	错误码

6.5.18 ECC 数据签名（外部密钥）

函数原型 INT32 ITS_EccSign(UINT32 u32Group, UINT32 u32HashFlag, UINT32 pu32KeyLen, UINT8* pu8Key, UINT32 u32DataLen, UINT8* pu8Data, UINT32 * pu32Rlen, UINT8* pu8R, UINT32 * pu32Slen, UINT8* pu8S);

功能描述 ECC 使用外部私钥签名

参数	u32Group	[IN] 椭圆曲率 ID
	u32HashFlag	[IN] 哈希标志：0-不哈希；1-先哈希后签名
	pu32KeyLen	[IN] 私钥长度
	pu8Key	[IN] 私钥
	u32DataLen	[IN] 待签名数据长度
	pu8Data	[IN] 待签名数据
	pu32Rlen	[OUT] 签名 R 长度
	pu8R	[OUT] 签名 R
	pu32Slen	[OUT] 签名 S 长度
	pu8S	[OUT] 签名 S
返回值	ITS_OK：	成功
	其他：	错误码

6.5.19 ECC 数据签名（内部密钥）

函数原型 INT32 ITS_ECCSignIndex(UINT32 u32KeyIndex, UINT32 u32HashFlag, UINT32 u32DataLen, UINT8* pu8Data, UINT32 * pu32Rlen, UINT8* pu8R, UINT32 *

pu32Slen, UINT8* pu8S);

功能描述	ECC 使用内部私钥签名	
参数	u32KeyIndex	[IN] 密钥对索引
	u32HashFlag	[IN] 哈希标志: 0-不哈希；1-先哈希后签名
	u32DataLen	[IN] 待签名数据长度
	pu8Data	[IN] 待签名数据
	pu32Rlen	[OUT] 签名 R 长度
	pu8R	[OUT] 签名 R
	pu32Slen	[OUT] 签名 S 长度
	pu8S	[OUT] 签名 S
返回值	ITS_OK:	成功
	其他:	错误码

6.5.20 ECC 验证签名

函数原型 INT32 ITS_ECCVerify(UINT32 u32Group, UINT32 u32HashFlag, UINT32 u32Xlen, UINT8* pu8X, UINT32 u32Ylen, UINT8* pu8Y, UINT32 u32Rlen, UINT8* pu8R, UINT32 u32Slen, UINT8* pu8S, UINT32 u32DataLen, UINT8* pu8Data);

功能描述	ECC 签名验证	
参数	u32Group	[IN] 椭圆曲率 ID
	u32HashFlag	[IN] 哈希标志: 0-不哈希；1-先哈希后签名
	u32Xlen	[IN] 公钥 X 长度
	pu8X	[IN] 公钥 X
	u32Ylen	[IN] 公钥 Y 长度
	pu8Y	[IN] 公钥 Y
	u32Rlen	[IN] 签名 R 长度
	pu8R	[IN] 签名 R
	u32Slen	[IN] 签名 S 长度
	pu8S	[IN] 签名 S
	u32DataLen	[IN] 待验证签名数据长度
	pu8Data	[IN] 待验证签名数据
返回值	ITS_OK:	成功
	其他:	错误码

6.5.21 ECC 公钥加密

函数原型 INT32 ITS_ECCPKEncrypt(UINT32 u32Group, UINT32 u32Xlen, UINT8* pu8X, UINT32 u32Ylen, UINT8* pu8Y, UINT32 u32InLen, UINT8* pu8InData, UINT32 * pu32OutLen, UINT8* pu8OutData);

功能描述 ECC 公钥加密

参数		
	u32Group	[IN] 椭圆曲率 ID
	u32Xlen	[IN] 公钥 X 长度
	pu8X	[IN] 公钥 X
	u32Ylen	[IN] 公钥 Y 长度
	pu8Y	[IN] 公钥 Y
	u32InLen	[IN] 待运算数据长度
	pu8InData	[IN] 待运算数据
	pu32Outlen	[OUT] 输出数据长度
	pu8OutData	[OUT] 输出数据
返回值	ITS_OK：	成功
	其他：	错误码

6.5.22 ECC 私钥解密（外部密钥）

函数原型 INT32 ITS_ECCSKDecrypt(UINT32 u32Group, UINT32 u32Keylen, UINT8* pu8Key, UINT32 u32Datalen, UINT8* pu8Data, UINT32 * pu32OutLen, UINT8* pu8OutData);

功能描述 ECC 使用外部私钥解密

参数		
	u32Group	[IN] 椭圆曲率 ID
	u32Keylen	[IN] 私钥长度
	pu8Key	[IN] 私钥
	u32Datalen	[IN] 待运算数据长度
	pu8Data	[IN] 待运算数据
	pu32Outlen	[OUT] 输出数据长度
	pu8OutData	[OUT] 输出数据
返回值	ITS_OK：	成功
	其他：	错误码

6.5.23 ECC 私钥解密（内部密钥）

函数原型 INT32 ITS_ECCSKDecryptIndex(UINT32 u32KeyIndex,uint32 u32Datalen, UINT8* pu8Data, UINT32 * pu32OutLen, UINT8* pu8OutData);

功能描述　ECC 使用内部私钥解密

参数		
	u32KeyIndex	[IN] 密钥对索引
	u32Datalen	[IN] 待运算数据长度
	pu8Data	[IN] 待运算数据
	pu32Outlen	[OUT] 输出数据长度
	pu8OutData	[OUT] 输出数据
返回值	ITS_OK：	成功
	其他：	错误码

6.5.24 RSA 私钥加密（外部密钥）

函数原型　INT32 ITS_RsaSKEncrypt(UINT32 u32SKlen, UINT8* pu8SK, UINT32 u32Datalen, UINT8* pu8Data, UINT32 * pu32Outlen, UINT8* pu8OutData);

功能描述　RSA 使用外部私钥加密

参数		
	u32SKlen	[IN] 私钥长度
	pu8SK	[IN] 私钥
	u32Datalen	[IN] 待运算数据长度，该值必须等于私钥模长字节数
	pu8Data	[IN] 待运算数据
	pu32Outlen	[OUT] 运算结果长度
	pu8OutData	[OUT] 运算结果
返回值	ITS_OK：	成功
	其他：	错误码

6.5.25 RSA 私钥加密（内部密钥）

函数原型　INT32 ITS_RsaSKEncryptIndex(UINT32 u32KeyIndex, UINT32 u32Datalen, UINT8* pu8Data, UINT32 * pu32Outlen, UINT8* pu8OutData);

功能描述　RSA 使用内部私钥加密

参数		
	u32KeyIndex	[IN] 密钥对索引
	u32Datalen	[IN] 待运算数据长度，该值必须等于私钥模长字节数
	pu8Data	[IN] 待运算数据
	pu32Outlen	[OUT] 运算结果长度
	pu8OutData	[OUT] 运算结果
返回值	ITS_OK：	成功
	其他：	错误码

6.5.26 RSA 公钥解密

函数原型　INT32 ITS_RsaPKDecrypt(UINT32 u32Nlen, UINT8* pu8N, UINT32 u32Elen, UINT8* pu8E, UINT32 u32Datalen, UINT8* pu8Data, UINT32 * pu32Outlen, UINT8* pu8OutData);

功能描述　RSA 公钥解密

参数		
	u32Nlen	[IN] 公钥 N 长度
	pu8N	[IN] 公钥 N
	u32Elen	[IN] 公钥 E 长度
	pu8E	[IN] 公钥 E
	u32Datalen	[IN] 待运算数据长度，该值必须等于公钥模长字节数
	pu8Data	[IN] 待运算数据
	pu32Outlen	[OUT] 运算结果长度
	pu8OutData	[OUT] 运算结果
返回值	ITS_OK:	成功
	其他:	错误码

6.5.27 RSA 公钥加密

函数原型　INT32 ITS_RsaPKEncrypt(UINT32 u32Nlen, UINT8* pu8N, UINT32 u32Elen, UINT8* pu8E, UINT32 u32Datalen, UINT8* pu8Data, UINT32 * pu32Outlen, UINT8* pu8OutData);

功能描述　RSA 公钥加密

参数		
	u32Nlen	[IN] 公钥 N 长度
	pu8N	[IN] 公钥 N
	u32Elen	[IN] 公钥 E 长度
	pu8E	[IN] 公钥 E
	u32Datalen	[IN] 待运算数据长度，该值必须等于公钥模长字节数
	pu8Data	[IN] 待运算数据
	pu32Outlen	[OUT] 运算结果长度
	pu8OutData	[OUT] 运算结果
返回值	ITS_OK:	成功
	其他:	错误码

6.5.28 RSA 私钥解密（外部密钥）

函数原型　INT32 ITS_RsaSKDecrypt(UINT32 u32SKlen, UINT8* pu8SK, uint32 u32Datalen, UINT8* pu8Data, UINT32 * pu32Outlen, UINT8* pu8OutData);

功能描述	RSA 使用外部私钥解密	
参数	u32SKlen	[IN] 私钥长度
	pu8SK	[IN] 私钥
	u32Datalen	[IN] 待运算数据长度，该值必须等于私钥模长字节数
	pu8Data	[IN] 待运算数据
	pu32Outlen	[OUT] 运算结果长度
	pu8OutData	[OUT] 运算结果
返回值	ITS_OK：	成功
	其他：	错误码

6.5.29 RSA 私钥解密（内部密钥）

函数原型	INT32 ITS_RsaSKDecryptIndex(UINT32 u32KeyIndex, uint32 u32Datalen, UINT8* pu8Data, UINT32 * pu32Outlen, UINT8* pu8OutData);	
功能描述	RSA 使用内部私钥解密	
参数	u32KeyIndex	[IN] 密钥对索引
	u32Datalen	[IN] 待运算数据长度，该值必须等于私钥模长字节数
	pu8Data	[IN] 待运算数据
	pu32Outlen	[OUT] 运算结果长度
	pu8OutData	[OUT] 运算结果
返回值	ITS_OK：	成功
	其他：	错误码

6.5.30 AES ECB 对称运算 （外部密钥）

函数原型	INT32 ITS_AESECB(UINT32 u32Mode, UINT32 u32Keylen, UINT8* pu8Key, UINT32 u32Datalen, UINT8* pu8Data,UINT32* pu32Outlen, UINT8* pu8OutData);	
功能描述	AES 使用外部密钥，加解密 ECB 模式	
参数	u32Mode	[IN] 运算模式：0-加密；1-解密
	u32Keylen	[IN] 密钥长度
	pu8Key	[IN] 密钥
	u32Datalen	[IN] 待运算数据长度
	pu8Data	[IN] 待运算数据，需要外部补位
	pu32Outlen	[OUT] 运算结果长度
	pu8OutData	[OUT] 运算结果
返回值	ITS_OK：	成功

其他：　　错误码

6.5.31 AES ECB 对称运算 （内部密钥）

函数原型　INT32 ITS_AESECBIndex(UINT32 u32Mode, UINT32 u32KeyIndex, UINT32 u32Datalen, UINT8* pu8Data,UINT32* pu32Outlen, UINT8* pu8OutData);

功能描述　AES 使用内部密钥，加解密 ECB 模式

参数

	u32Mode	[IN] 运算模式：0-加密；1-解密
	u32KeyIndex	[IN] 运算密钥索引
	u32Datalen	[IN] 待运算数据长度
	pu8Data	[IN] 待运算数据，需要外部补位
	pu32Outlen	[OUT] 运算结果长度
	pu8OutData	[OUT] 运算结果

返回值

ITS_OK：	成功
其他：	错误码

6.5.32 AES CBC 对称运算 （外部密钥）

函数原型　INT32 ITS_AESCBC(UINT32 u32Mode, UINT8* pu8IV, UINT32 u32Keylen, UINT8* pu8Key, UINT32 u32Datalen, UINT8* pu8Data, UINT32 * pu32Outlen, UINT8*pu8OutData);

功能描述　AES 使用外部密钥，加解密 CBC 模式

参数

	u32Mode	[IN] 运算模式：0-加密；1-解密
	pu8IV	[IN] 初始化向量
	u32Keylen	[IN] 密钥长度
	pu8Key	[IN] 密钥
	u32Datalen	[IN] 待运算数据长度
	pu8Data	[IN] 待运算数据，需要外部补位
	pu32Outlen	[OUT] 运算结果长度
	pu8OutData	[OUT] 运算结果

返回值

ITS_OK：	成功
其他：	错误码

6.5.33 AES CBC 对称运算 （内部密钥）

函数原型　INT32 ITS_AESCBCIndex(UINT32 u32Mode, UINT8* pu8IV, UINT32 u32KeyIndex, UINT32 u32Datalen, UINT8* pu8Data, UINT32 * pu32Outlen,

UINT8*pu8OutData);

功能描述	AES 使用内部密钥，加解密 CBC 模式	
参数	u32Mode	[IN] 运算模式：0-加密；1-解密
	pu8IV	[IN] 初始化向量
	u32KeyIndex	[IN] 运算密钥索引
	u32Datalen	[IN] 待运算数据长度
	pu8Data	[IN] 待运算数据，需要外部补位
	pu32Outlen	[OUT] 运算结果长度
	pu8OutData	[OUT] 运算结果
返回值	ITS_OK：	成功
	其他：	错误码

6.5.34 SHA 初始化

函数原型	INT32 ITS_SHAInit(UINT32 u32Alg);	
功能描述	SHA 哈希初始化	
参数	u32Alg	[IN] SHA 算法参数 0-SHA256，1-SHA384，2-SHA512
返回值	ITS_OK：	成功
	其他：	错误码

6.5.35 SHA 数据哈希计算

函数原型	UINT32 ITS_ShaUpdate(UINT32 u32Alg, UINT32 u32Datalen,UINT8* pu8Data);	
功能描述	SHA 数据哈希计算	
参数	u32Alg	[IN] SHA 算法参数 0-SHA256，1-SHA384，2-SHA512
	u32Datalen	[IN] 数据长度
	pu8Data	[IN] 数据
返回值	ITS_OK：	成功
	其他：	错误码

6.5.36 SHA 数据哈希计算结束

函数原型	UINT32 ITS_ShaFinal(UINT32 u32Alg, UINT32 * pu32Hashlen, UINT8* pu8Hash);	
功能描述	SHA 数据哈希计算结束，得到哈希值	
参数	u32Alg	[IN] SHA 算法参数 0-SHA256，1-SHA384，2-SHA512
	pu32Hashlen	[OUT] 哈希结果长度
	pu8Hash	[OUT] 哈希结果

返回值　ITS_OK：　成功

其他：　错误码

6.6 智能交通车辆服务接口

6.6.1 概述

智能交通车辆服务接口包含了智能交通终端证书密钥的产生、衍生、签名等操作。智能交通服务系列函数如表 12 所示。

表 12 智能交通服务系列函数描述

函数名称	功能
ITS_ECertGenu32KeyIndex	终端证书密钥对产生
ITS_ECertSignIndex	终端证书私钥 SM2 签名
ITS_ECertSM2SignIndexWithIDA	智能交通-终端证书私钥 SM2 签名带 IDA
ITS_ECertSM2SKDecryptIndex	智能交通-终端证书 SM2 私钥解密
ITS_GetPK	智能交通-获取终端证书的密钥对公钥
ITS_ECertSignIndexExp	智能交通-终端证书衍生私钥签名
ITS_ECertSM2SignIndexExpWithIDA	智能交通-终端证书衍生私钥 SM2 签名带 IDA
ITS_ECertSKDeriveIndex	智能交通-终端证书私钥衍生并存储
ITS_Verify	智能交通-验证签名(支持压缩公钥)

6.6.2 终端证书密钥对产生

函数原型　INT32 ITS_ECertGenu32KeyIndex(UINT32 u32Index, UINT8 u8Alg, UINT8* pu8X, UINT8* pu8Y);

功能描述　智能交通终端证书密钥对产生，密钥对内部存储

参数　u32Index　[IN] 证书索引

u8Alg　[IN] 算法: 0-SM2; 1-nistP256;2-brainpoolP256r1 nistP256 和 brainpoolP256r1 为非必实现算法

pu8X　[OUT] 公钥 X，定长 32 字节

pu8Y　[OUT] 公钥 Y，定长 32 字节

返回值　ITS_OK：　成功

其他：　错误码

6.6.3 终端证书私钥签名

函数原型　INT32 ITS_ECertSignIndex(UINT32 u32Index, UINT8* pu8Hash, UINT8* pu8R, UINT8* pu8S);

功能描述　使用终端私钥签名

参数　u32Index　[IN] 证书索引

pu8Hash　[IN] 待签名的 HASH 数据，定长 32 字节

pu8R　[OUT] 签名 R，定长 32 字节

	pu8S	[OUT] 签名 S，定长 32 字节
返回值	ITS_OK：	成功
	其他：	错误码

6.6.4 终端证书私钥 SM2 签名带 IDA

函数原型 INT32 ITS_ECertSM2SignIndexWithIDA(UINT32 u32Index, UINT32 u32Len, UINT8* pu8Data, UINT8* pu8X, UINT8* pu8Y, UINT8* pu8R, UINT8* pu8S);

功能描述 使用终端私钥签名 SM2 算法，数据先执行 SM3-E 哈希

参数	u32Index	[IN] 证书索引
	u32Len	[IN] 待签名数据长度
	pu8Data	[IN] 待签名数据
	pu8X	[IN] 参与签名的公钥 X
	pu8Y	[IN] 参与签名的公钥 Y
	pu8R	[OUT] 签名 R，定长 32 字节
	pu8S	[OUT] 签名 S，定长 32 字节
返回值	ITS_OK：	成功
	其他：	错误码

备注 非自定义 IDA 为固定值“china”。下同

6.6.5 终端证书 SM2 私钥解密

函数原型 INT32 ITS_ECertSM2SKDecryptIndex(UINT32 u32Index, UINT32 u32Datalen, UINT8* pu8Data, UINT32 * pu32Outlen, UINT8* pu8OutData);

功能描述 智能交通-终端证书使用内部私钥 SM2 解密

参数	u32Index	[IN] 证书索引
	u32Datalen	[IN] 待运算数据长度
	pu8Data	[IN] 待运算数据
	pu32Outlen	[OUT] 输出数据长度
	pu8OutData	[OUT] 输出数据
返回值	ITS_OK：	成功
	其他：	错误码

6.6.6 获取终端证书的密钥对公钥

函数原型 INT32 ITS_GetPK(UINT32 u32Index, UINT8* pu8X, UINT8* pu8Y);

功能描述 智能交通-获取密钥对公钥

参数	u32Index	[IN] 证书索引
	pu8X	[OUT] 公钥 X，定长 32 字节
	pu8Y	[OUT] 公钥 Y，定长 32 字节
返回值	ITS_OK：	成功
	其他：	错误码

6.6.7 智能交通-终端证书衍生私钥签名

函数原型 INT32 ITS_ECertSignIndexExp(UINT32 u32Index, UINT8* pu8Hash, UINT8* pu8Exp, UINT8* pu8C, UINT8* pu8R, UINT8* pu8S);

功能描述 使用终端衍生私钥签名 SM2 算法，直接对输入的 HASH 签名

车载硬件设备内部先执行 Private key = a + exp + c，然后再使用衍生私钥签名

参数	u32Index	[IN] 证书索引
	pu8Hash	[IN] 待签名的 HASH 数据，定长 32 字节
	pu8Exp	[IN] 衍生因子，定长 32 字节
	pu8C	[IN] 服务器返回的私钥因子，定长 32 字节
	pu8R	[OUT] 签名 R，定长 32 字节
	pu8S	[OUT] 签名 S，定长 32 字节
返回值	ITS_OK：	成功
	其他：	错误码
备注	对于密钥衍生公式 Private key = a + exp + c 公式，可以使用以下参数来对应：	
	a 为私钥	
	e=0000000000000000000000000000000000000000000000000000000000000001	
	k_new = (k_seed + f)* e + r = k_seed + f + r	
	关于衍生私钥详细内容，参照附录 B 密钥衍生函数说明。下同	

6.6.8 智能交通-终端证书衍生私钥 SM2 签名带 IDA

函数原型 INT32 ITS_ECertSM2SignIndexExpWithIDA(UINT32 u32Index, UINT32 u32Len, UINT8* pu8Data, UINT8* pu8X, UINT8* pu8Y, UINT8* pu8Exp, UINT8* pu8C, UINT8* pu8R, UINT8* pu8S);

功能描述 使用终端衍生私钥签名 SM2 算法，数据先执行 SM3-E 哈希

车载硬件设备内部先执行 Private key = a + exp + c，然后再使用衍生私钥签名

参数	u32Index	[IN] 证书索引

	u32Len	[IN] 待签名数据长度
	pu8Data	[IN] 待签名数据
	pu8X	[IN] 参与签名的公钥 X
	pu8Y	[IN] 参与签名的公钥 Y
	pu8Exp	[IN] 衍生因子，定长 32 字节
	pu8C	[IN] 服务器返回的私钥因子，定长 32 字节
	pu8R	[OUT] 签名 R，定长 32 字节
	pu8S	[OUT] 签名 S，定长 32 字节
返回值	ITS_OK：	成功
	其他：	错误码

6.6.9 智能交通-终端证书私钥衍生并存储

函数原型 INT32 ITS_ECertSKDeriveIndex(UINT32 u32Index1, UINT32 u32Index2, UINT8* pu8F, UINT8* pu8E, UINT8* pu8R, UINT8* pu8X, UINT8* pu8Y);

功能描述 内部私钥衍生后产生新的密钥对，并存入新的索引中

衍生计算公式：k_new = (k_seed + f)* e + r

参数	u32Index1	[IN] 用于衍生的私钥索引
	u32Index2	[IN] 用于存储的私钥索引
	pu8F	[IN] 计算因子 f
	pu8E	[IN] 计算因子 e
	pu8R	[IN] 计算因子 r
	pu8X	[OUT] 衍生后公钥 X，定长 32 字节
	pu8Y	[OUT] 衍生后公钥 Y，定长 32 字节
返回值	ITS_OK：	成功
	其他：	错误码

6.6.10 智能交通-验证签名(支持压缩公钥)

函数原型 INT32 ITS_Verify(UINT32 u32HashFlag, UINT32 u32Alg, UINT32 u32PKType , UINT8* pu8Pk, UINT8* pu8R, UINT8* pu8S, UINT32 u32IDALen, UINT8* pu8IDA, UINT32 u32DataLen, UINT8* pu8Data);

功能描述 智能交通签名验证，使用压缩公钥，支持 IDA 哈希

参数	u32HashFlag	[IN] 哈希标志：0-不哈希；1-先哈希后验证签名
	u32Alg	[IN] 算法标识：0-SM2; 1-nistP256; 2-

		brainpoolP256r1
	u32PKType	[IN] 公钥类型: 0-压缩公钥(公钥长度33字节，格式：ODD_FLAG||X) 1-非压缩公钥(公钥长度64字节，格式X||Y)
	pu8Pk	[IN] 公钥(根据nPKType，公钥格式不同)
	pu8R	[IN] 签名R
	pu8S	[IN] 签名S
	u32IDALen	[IN] IDA的长度，如果为0，表示无IDA参与HASH
	pu8IDA	[IN] IDA数据
	u32DataLen	[IN] 待验证签名数据长度
	pu8Data	[IN] 待验证签名数据
返回值	ITS_OK:	成功
	其他:	错误码

7 安全芯片的安全要求

7.1 安全芯片使用阶段

安全芯片须具备硬件层面的产品生命周期状态管理功能，并配合固件系统实现安全芯片全生命周期状态管理机制。生命周期应包含下面四个状态：

a) 生产状态。此状态下安全芯片内部所有密钥为空，可Debug调试。
b) 出厂状态。安全芯片交付给OEM厂商后的状态，此状态下安全芯片内部密钥已初始化完成，可Debug调试。
c) 安全状态。安全芯片中交付用户使用后的状态。此状态下安全芯片内部密钥都已初始化完成，不支持Debug调试。
d) 返厂状态。安全芯片故障返回设备生产厂家调试诊断的状态。此状态下安全芯片内部的核心安全密钥应该被销毁，安全芯片可Debug调试，处于该状态的安全芯片不能再恢复到生产/出厂/安全状态，不会再返回用户使用。

7.2 权限管理

7.2.1 权限分类

权限分安全芯片权限和用户权限。

安全芯片权限：安全芯片出厂时预置主控密钥，车辆激活时需更新安全芯片主控密钥，用于敏感数据的访问控制和用户PIN码的控制。

用户权限：用户PIN码验证通过后，获得用户权限，用户权限只作用于接口的调用。

7.2.2 权限使用

权限的使用应遵循以下要求：

a) 安全芯片权限用于更新认证密钥，更新主控密钥，重置用户PIN码和导入外部密钥。

b） 调用相关 API 接口需要用户权限。
c） 相关 API 接口调用是否需要用户权限在设计时指定。
d） 用户 PIN 码的修改需要用户权限，用户 PIN 码的重置需要安全芯片权限。

7.2.3 PIN 码安全要求

PIN 码的使用应遵循以下要求：
a） PIN 码长度不少于 6 个字节。
b） PIN 码在安全芯片和本接口之间的传输过程中应采取保护措施，防止 PIN 码泄露。
c） PIN 码在安全芯片中应安全存储，不可从设备中导出。

7.3 密钥安全要求

推荐采用国家商用密码算法 SM2、SM3、SM4。若采用 RSA 算法，密钥长度至少 2048 位
密钥应遵循以下安全要求：
a） 安全芯片内产生的随机数应为真随机数，应符合随机性检测的要求。
b） 安全芯片内产生的会话密钥应使用随机数。
c） 安全芯片内的密钥应具备有效的密钥保护机制防止解剖、探测和读取。
d） 安全芯片内的密钥应按权限要求使用。
e） 除公钥外的密钥不能以明文形式出现在安全芯片外。
f） 签名私钥必须在安全芯片中产生。
g） 安全芯片失效时必须销毁该安全设内所有的密钥。

7.4 算法安全要求

推荐采用国家商用密码算法 SM2、SM3、SM4。若采用 RSA 算法，密钥长度至少 2048 位。

7.5 安全芯片抗攻击要求

安全芯片应具有以下抗攻击要求：
a） 安全芯片需要达到 GM/T 0008-2012《安全芯片密码检测准则》的安全防护要求。
b） 安全芯片硬件电路需要具备主动保护的屏蔽层，并有完整的物理攻击检测报警逻辑电路。
c） 安全芯片内部非易失性存储器需要有硬件实现的加密存储机制，对敏感数据（如密钥、根证书等）必须密文存储，并具备对敏感数据的访问控制机制。
d） 安全芯片密码算法硬件单元必须具备专业的抗攻击能力。抗攻击能力包括但不限于：
- 防止基于功耗的旁路窥探能力
- 防止基于时间的旁路窥探能力
- 防止故障注入攻击（频率，电压，温度等）的能力
- 防止物理攻击（探针，激光，电磁等）的能力

e） 安全芯片真随机数发生器（TRNG）电路必须进行冗余设计，应至少具备 4 个独立的物理发生源。

附录 A

（规范性附录）

错误代码定义和说明

错误代码定义和说明见表 A.1

表 A.1 错误代码定义和说明

错误代码标识		
宏描述	预定义值	说明
ITS_OK	0x00000000	成功
ITS_FAIL	0x00000001	失败
ITS_ERR_UNKNOWN	0x00000002	异常错误
ITS_ERR_ ACCESS_DENY	0x00000003	无执行权限
ITS_ERR_ KEY_NOTFOUND	0x00000005	密钥不存在
ITS_ERR_ KEY_INVALID	0x00000006	密钥不合法
ITS_ERR_ CALC_FAIL	0x00000008	密码运算失败
ITS_ ERR_FLASH_CHECK	0x00000009	FLASH 读写失败
ITS_ERR_SIG_INVALID	0x0000000A	签名不合法
ITS_ERR_DATA_INVALID	0x0000000B	数据不合法
ITS_ERR_AUTH	0x0000000C	设备认证失败
ITS_ERR_ALG_INVALID	0x0000000D	算法不合法
ITS_ERR_CERT_NOTFOUND r	0x0000000E	证书不存在
ITS_ERR_INDEX_INVALID	0x0000000F	索引不合法
ITS_ERR_LEN_INVALID	0x00000012	输入参数中的长度不合法
ITS_ERR_TYPE_INVALID	0x00000013	类型不合法
ITS_ERR_VERSION_INVALID	0x00000014	版本检查失败
ITS_ERR_VERIFY_PIN	0x00000017	PIN 验证失败
ITS_ERR_LOCK_PIN	0x00000018	PIN 锁定
ITS_ERR_GEN_RND	0x00000019	产生随机数失败
ITS_ERR_PARAMETER	0x000000E1	输入参数不合法
ITS_ERR_CMD_SEND	0x000000E2	数据发送失败
ITS_ERR_CMD_TIMEOUT	0x000000E3	超时
ITS_ERR_COMM	0x000000E4	数据接收失败
表 A.1（第 2 页/共 2 页）		

ITS_ERR_DATA_LEN	0x000000ED	输入的数据长度不合法
ITS_ERR_KEY_LEN	0x000000F1	密钥长度不合法
ITS_ERR_DECOM_FAIL	0x000000F2	公钥解压缩失败
ITS_ERR_NOTSUPPORT	0x000000F3	不支持的功能

附录 B

（规范性附录）

密钥衍生函数说明

本附录是描述 6.6.6-6.6.8 密钥衍生流程所使用的计算公式，附录内容引自 CCSA YD/T 3957-2021 规范 H.2。

B1.1 符号表达说明

以下符号表达说明适用于附录 B：

a） 有一个公认的“基点”，表示为 G；
b） 椭圆曲线的阶数用 l 表示；
c） 对于一个比特串 b 和数字 n，b^n表示通过将比特串 b 重复 n 次而形成的比特串；
d） 对于比特串 x，x^{INT}表示 x 转换为整数，例如：如果 x = 0101，则 x^{INT} = 5；
e） 对于比特串 x 和数字 n，x + n 是 x^{INT} + n 产生的比特串的简写，例如：如果 x = 0100，则 x + 1 = 0101，x + 2 = 0110，x + 3 = 0111，依此类推；
f） 对于比特串 x 和 y，x ⊕ y 表示它们的按位异或，而 x || y 表示串联的比特串，例如：如果 x = 0110 且 y = 1010，则 x ⊕ y = 1100 且 x || y = 01101010；
g） 对于 128 比特字符串 k 和 m，*Symm* (*k*, *m*)表示使用对称加密算法得到的 128 比特密文，*m* 为被加密的明文数据，*k* 为对称密钥，使用 128 比特的分组大小；
h） 对于数字 m 和 n，m mod n 表示对具有模数 n 的 m 进行模运算的结果，例如：如果 m = 9 且 n = 2，则 m mod n = 1。

B1.2 密钥衍生函数

两个密钥衍生函数：签名函数f_S和加密函数f_E。两个函数的参数如下：

——对称加密算法 Symm，具有 128 比特的输入和输出以及 128 比特的密钥，采用 SM4 分组密码算法，ECB/NoPadding 工作模式。
——一个 128 比特的密钥，分别表示为 f_S 函数的 k_S，和 f_E 函数的 k_E。
——l 是一个 256 比特的整数，表示进行密钥衍生的椭圆曲线的基点的阶。

两个函数的输入为(iINT, jINT)两个整数，其范围是(0 , 232-1)。

两个函数的输出为一个256比特的整数*o*，范围为(0 , 1)。

两个函数过程如下：

a） 将输入 i^{INT} 和 j^{INT} 转换为 32 比特的比特串 i 和 j。
b） b）拼接下列 128 位的比特串 x_S（用于 f_S）或 x_E（用于 f_E）。

1) $x_S = (0^{32} \,||\, i \,||\, j \,||\, 0^{32})$

2) $x_E = (1^{32} \,||\, i \,||\, j \,||\, 0^{32})$

c） 创建如下的临时输出 y_S（用于 f_S）或 y_E（用于 f_E）。输出为 3×128 = 384 比特的字符串。

1) $y_S = (Symm(k_S, x_S+1) \oplus (x_S+1)) \,||\, (Symm(k_S, x_S+2) \oplus (x_S+2)) \,||\, (Symm(k_S, x_S+3) \oplus (x_S+3))$

2) $y_E = (Symm(k_E, x_E+1) \oplus (x_E+1)) \,||\, (Symm(k_E, x_E+2) \oplus (x_E+2)) \,||\, (Symm(k_E, x_E+3) \oplus (x_E+3))$

这里，“x + i”表达了“将八位位组字符串**x**转换为整数，该整数加i，然后将加了i之后的整数转换回八位位组字符串”。例如，如果xS = (032 || i || j || 032) = (0000 0000 0000 0000 0000 0000 0000 0000b || i || j || 0000 0000 0000 0000 0000 0000 0000 0000)，则xS + 2 = (0000 0000 0000 0000 0000 0000 0000 0000 b || i || j || 0000 0000 0000 0000 0000 0000 0000 0010)

d） f_S和 f_E的最终输出如下：

1) $f_S(k_S, i^{INT}, j^{INT}) = y_S^{INT} \bmod 1$

2) $f_E(k_E, i^{INT}, j^{INT}) = y_E^{INT} \bmod 1$

T/ITS 0149-2021

中国智能交通产业联盟

标准

智能交通运输系统 终端设备安全芯片 应用接口规范

T/ITS 0149-2021

北京市海淀区西土城路 8 号（100088）

中国智能交通产业联盟印刷

网址：http://www.c-its.org.cn

2021 年 12 月第一版　2021 年 12 月第一次印刷