ICS 35.240.60
P 07

CJ

中华人民共和国城镇建设行业标准

CJ/T 304—2017
代替 CJ/T 304 2008

建设事业智能卡操作系统技术要求

Technical requirement for chip operating system of smart card in construction cause

2017-09-30 发布　　2018-05-01 实施

中华人民共和国住房和城乡建设部　　发布

目　　次

前　言

本标准按照 GB/T 1.1　2009 给出的规则起草。

本标准代替 CJ/T 304　2008《建设事业 CPU 卡操作系统技术要求》，与 CJ/T 304　2008 相比较，主要技术变化包括：

删除机电特性、逻辑接口与传输协议(2008 年版中第 5 章)；

删除密钥的推导方法和过程密钥的产生方法等相关内容(2008 年版中附录 C、E、F)；

增加记录格式说明(见 6.4)；

增加状态命令表(见 8.3.1)；

完善了相关指令的参数定义。

本标准做了下列编辑性修改：

修改了标准名称。

本标准由住房和城乡建设部标准定额研究所提出。

本标准由住房和城乡建设部信息技术应用标准化技术委员会归口。

本标准主编单位：中外建设信息有限责任公司。

本标准参编单位：住房和城乡建设部 IC 卡应用服务中心、北京亿速码数据处理有限责任公司、深圳市华旭科技开发有限公司、大唐微电子技术有限公司、武汉天喻信息产业股份有限公司、上海华虹集成电路有限责任公司、东信和平科技股份有限公司、捷德(中国)信息科技有限公司、上海复旦微电子集团股份有限公司、英飞凌科技(中国)有限公司、北京中电华大电子设计有限责任公司、上海雅斯拓智能卡技术有限公司、金邦达有限公司、北京中科软件有限公司、国民技术股份有限公司、广东楚天龙智能卡有限公司、河北一卡通电子支付服务有限公司、上海公共交通卡股份有限公司、郑州城市一卡通有限责任公司、南通公共交通总公司、青岛市琴岛通卡股份有限公司、深圳欧贝特卡系统科技有限公司。

本标准主要起草人：张永刚、尚治宇、樊静静、殷骏、张昕、王猛、张志红、李玉华、吴思凯、黄小鹏、丁晓明、段永刚、黄显明、刘觅、常小争、徐木平、刘春晓、柳建勇、刘典清、霍立民、胡振清、刘广乐、陆流、李妍、张鲲鹏。

本标准所代替标准的历次版本发布情况为：

CJ/T 304　2008。

建设事业智能卡操作系统技术要求

1 范围

本标准规定了建设事业智能卡文件和命令、应用选择、安全机制及安全要求、电子存折/电子钱包应用和相应的定义符号等。

本标准适用于建设事业智能卡操作系统的设计、管理以及应用。

2 规范性引用文件

下列文件对于本文件的应用是必不可少的。凡是注日期的引用文件，仅注日期的版本适用于本文件。凡是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。

CJ/T 166 建设事业集成电路(IC)卡应用技术

JR/T 0025 中国金融集成电路(IC)卡规范

3 术语和定义

下列术语和定义适用于本文件。

3.1

智能卡 smart card

一种具有微处理器芯片的IC卡，本文件中提到的IC卡特指智能卡。

3.2

接口设备 interface device

终端上插入IC卡的部分，包括其中的机械和电气部分。

3.3

响应 response

IC卡处理完终端收到的命令报文后，返回给终端的报文。

3.4

电子存折 electronic deposit

一种为持卡人进行消费、取现等交易而设计的使用个人密码(PIN)保护的金融IC卡应用。

注：它支持圈存、圈提、消费、取现等交易。

3.5

电子钱包 electronic purse

一种为方便持卡人小额消费而设计的金融IC卡应用。

注：它支持圈存、消费等交易。除圈存交易外，使用电子钱包进行的其他交易均无需提交个人密码(PIN)。

3.6

哈希函数 hash function

将位串映射为定长位串的函数。

3.7

报文 message

由终端向卡或卡向终端发出的，不含传输控制字符的字节串。

3.8

报文认证码　message authentication code

对交易数据及其相关参数进行运算后产生的代码。主要用于验证报文的完整性。

3.9

半字节　nibble

一个字节的高四位或低四位。

3.10

密钥　key

控制加密转换操作的符号序列。

3.11

数字签名　digital signature

一种非对称加密数据变换。

注：它使得接收方能够验证数据的原始性和完整性，保护发送和接收的数据不被第三方伪造，同时对于发送方来说，还可用以防止接收方的伪造。

3.12

加密算法　cryptographic algorithm

为了隐藏或揭露信息内容而变换数据的算法。

3.13

对称加密技术　symmetric cryptographic technique

发送方和接收方使用相同保密密钥进行数据变换的加密技术。

3.14

非对称加密技术　asymmetric cryptographic technique

采用两种相关变换进行加密的技术，一种是公开变换（由公共密钥定义），另一种是私有变换（由私有密钥定义）。

注：这两种变换具有以下属性，即私有变换不能通过给定的公开变换导出。

3.15

私有密钥　private key

一个实体的非对称密钥对中仅供实体自身使用的密钥，在数字签名模式中，私有密钥用于签名功能。

3.16

公共密钥　public key

一个实体的非对称密钥对中可以公开的密钥，在数字签名模式中，公共密钥用于验证功能。

3.17

灰锁　ash lock

应用临时锁定。

4　缩略语和符号

ADF	应用定义文件（Application Definition File）
AEF	应用基本文件（Application Elementary File）
AID	应用标识符（Application Identifier）
APDU	应用协议数据单元（Application Protocol Data Unit）
ATR	复位应答（Answer to Reset）

CLA	命令报文的类别字节(Class Byte of the Command Message)
COS	智能卡芯片操作系统(Chip operating system)
DDF	目录定义文件(Directory Definition File)
DEA	数据加密算法(Data Encryption Algorithm)
DES	数据加密标准(Data Encryption Standard)
DF	专用文件(Dedicated File)
DIR	目录(Directory)
EF	基本文件(Elementary File)
HHMMSS	时、分、秒(Hours, Minutes, Seconds)
INS	命令报文的指令字节(Instruction Byte of Command Message)
Lc	终端发出的命令数据的实际长度(Exact Length of Data Sent by the TAL IN A Case 3 or 4 Command)
Le	响应数据的最大期望长度(Maximum Length of Data Expected by the TAL in Response to a Case 2 or 4 Command)
Lr	响应数据域的长度(Length of Response Data Field)
MAC	报文认证码(Message Authentication Code)
MF	主控文件(Mater File)
NCA	认证机构公开密钥模数长度(Length of the Certification Authority Public Key Modulus)
NI	发卡方公开密钥模数长度(Length of the Issuer Public Key Modulus)
NIC	IC卡公开密钥模数长度(Length of the ICC Public Key Modulus)
P1	参数1(Parameter 1)
P2	参数2(Parameter 2)
P3	参数3(Parameter 3)
PIN	个人密码(Personal Identification Number)
PIX	专用应用标识符扩展码(Proprietary Application Identifier Extension)
PSA	支付系统应用(Payment System Application)
PSE	支付系统环境(Payment System Environment)
PSAM	消费安全存取模块(Purchase Secure Access Module)
RFU	保留为将来使用(Reserved for Future Use)
RID	已注册的应用提供者标识(Registered Application Provider Identifier)
RSA	一种非对称加密算法(Rivest,Shamir,Adleman)
SAM	安全存取模块(Secure Access Module)
SFI	短文件标识符(Short File Identifier)
SHA	安全哈希算法(Secure Hash Algorithm)
SW1	状态码1(Status Word One)
SW2	状态码2(Status Word Two)
TAC	交易验证码(Transaction Authorization Cryptogram)
YYYYMMDD	年、月、日(Year, Month, Day)

5 文件和命令

5.1 说明

IC卡中的每个应用都包括一系列信息项，在终端成功地完成应用选择后可以对这些信息进行访问。

一个信息项称为一个数据元，数据元是信息的最小单位，它用名称、逻辑内容说明、格式及代码来标识。

5.2 文件

5.2.1 文件结构

数据文件中数据元应以记录方式或二进制方式存储，文件结构及引用方式由文件的用途决定，并应符合下列要求：

a) 除目录文件外，数据文件的内容和布局应在应用规范中说明，也可由发卡方定义；

b) 描述符合本文件的应用文件结构被定义为支付系统应用(PSA)，不符合本文件的其他应用也可出现在IC卡上，并可使用本文件中定义的命令进行操作；

c) IC卡中PSA的路径可通过明确选择支付系统环境(PSE)来激活，一个成功的PSE选择能够对目录结构进行访问；

d) 应用定义文件(ADF)、应用基本文件(AEF)等要求应符合CJ/T 166的要求。

5.2.2 文件查询

5.2.2.1 通过文件名查询

卡中的任何ADF或目录定义文件(DDF)可通过其专用文件(DF)名查询，ADF的DF名对应其应用标识符(AID)，每个DF名在给定的IC卡中应是唯一的。

5.2.2.2 通过SFI查询

短文件标识符(SFI)用于选择AEF。对给定应用中的任何AEF，可以通过SFI(5位代码，取值范围从1～30)查询。SFI的编码在每个用到它的命令中描述。

在一个给定的应用中SFI应是唯一的，专用SFI的使用由应用决定。

5.3 命令

5.3.1 命令APDU格式

命令应用协议数据单元(APDU)由4字节长的必备头和一个可变长度的条件体组成，见图1。

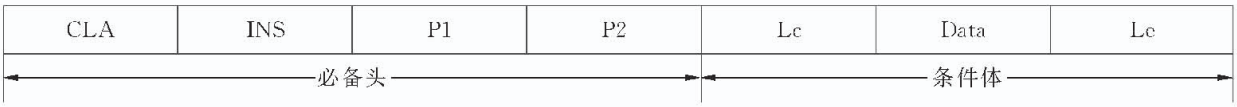

图1 命令APDU结构

命令APDU中发送的数据字节数用命令数据域的长度(Lc)表示。

响应APDU中期望返回的数据字节数用期望数据长度(Le)表示。当Le存在且值为0时，表示需要最大字节数(256字节)。在命令报文需要时，Le始终被设为'00'。

命令 APDU 报文的内容见表 1。

表 1　命令 APDU 的内容

代码	描　　述	长度(字节)
CLA	命令类别	1
INS	指令代码	1
P1	指令参数 1	1
P2	指令参数 2	1
Lc	命令数据域中存在的字节数	0 或 1
Data	命令发送的数据位串(=Lc)	可变
Le	响应数据域中期望的最大数据字节数	0 或 1

5.3.2　响应 APDU 格式

响应 APDU 格式由一个变长的条件体和后随两字节长的必备尾组成,见图 2。

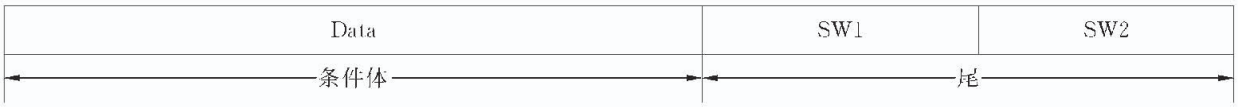

图 2　响应 APDU 的结构

APDU 响应报文的内容见表 2。

表 2　响应 APDU 的内容

代码	描　　述	长度(字节)
Data	响应中接收的数据位串(=Lr)	变长
SW1	命令处理状态	1
SW2	命令处理限定	1

响应 APDU 的命令如下:

APPLICATION BLOCK (应用锁定);
APPLICATION UNBLOCK (应用解锁);
CARD BLOCK (卡片锁定);
EXTERNAL AUTHENTICATION (外部认证);
GET CHALLENGE (产生随机数);
GET RESPONSE (取响应);
INTERNAL AUTHENTICATION (内部认证);
PIN CHANGE/UNBLOCK (个人密码修改/解锁);
READ BINARY (读二进制);
READ RECORD (读记录);
SELECT (选择);
UPDATE BINARY (修改二进制);

UPDATE RECORD（修改记录）；
VERIFY（校验）。

5.3.3 APDU 命令

5.3.3.1 APPLICATION BLOCK 命令

5.3.3.1.1 定义和范围

APPLICATION BLOCK 命令使当前选择的应用失效。

当 APPLICATION BLOCK 命令成功地完成后，用 SELECT 命令选择已失效的应用，将回送状态码“选择文件无效”。

对其他命令的影响根据不同应用而定。

5.3.3.1.2 命令报文

APPLICATION BLOCK 命令报文编码见表 3。

表 3 APPLICATION BLOCK 命令报文

代码	值	代码	值
CLA	‘84’	Lc	数据字节数
INS	‘1E’	Data	报文认证码(MAC)数据元
P1	‘00’，其他值保留为将来使用	Le	不存在
P2	‘00’或‘01’		
注：P2=‘00’：命令执行成功后可锁定应用，但该应用可以用 APPLICATION UNBLOCK 命令解锁。P2=‘01’：命令执行成功后将永久锁定应用。			

5.3.3.1.3 命令报文数据域

命令报文数据域包括根据本标准规定进行编码的报文认证码(MAC)数据元。

5.3.3.1.4 响应报文数据域

响应报文数据域不存在。

5.3.3.1.5 响应报文状态码

无论应用是否已经失效，APPLICATION BLOCK 命令执行成功的状态码是‘9000’。

IC 卡可能回送的 APPLICATION BLOCK 警告状态码见表 4。

表 4 APPLICATION BLOCK 警告状态

SW1	SW2	含　义
‘62’	‘00’	无信息提供
‘62’	‘81’	回送数据可能出错
‘62’	‘83’	选择文件无效

IC 卡可能回送的 APPLICATION BLOCK 错误状态码见表 5。

表 5　APPLICATION BLOCK 错误状态

SW1	SW2	含　义
‘64’	‘00’	状态标志位未变
‘65’	‘81’	内存失败
‘69’	‘82’	不满足安全状态
‘69’	‘84’	引用数据无效
‘69’	‘87’	安全报文数据项丢失
‘69’	‘88’	安全报文数据项不正确
‘6A’	‘86’	参数 P1、P2 不正确
‘6A’	‘88’	未找到引用数据

5.3.3.2　APPLICATION UNBLOCK 命令

5.3.3.2.1　定义和范围

APPLICATION UNBLOCK 命令用于恢复当前应用。

当 APPLICATION UNBLOCK 命令成功地完成后，由 APPLICATION BLOCK 命令产生的对应用命令响应的限制将被取消。

5.3.3.2.2　命令报文

APPLICATION UNBLOCK 命令报文编码见表 6。

表 6　APPLICATION UNBLOCK 命令报文

代码	值	代码	值
CLA	‘84’	Lc	数据字节数
INS	‘18’	Data	报文认证码(MAC)数据元
P1	‘00’，其他值保留为将来使用	Le	不存在
P2	‘00’，其他值保留为将来使用		

5.3.3.2.3　命令报文数据域

命令报文数据域的内容包括根据规定进行编码的 MAC 数据元。

5.3.3.2.4　响应报文数据域

响应报文数据域不存在。

5.3.3.2.5　响应报文状态码

不论应用是否已经失效，APPLICATION UNBLOCK 命令执行成功的状态码是‘9000’。

IC 卡可能回送的 APPLICATION UNBLOCK 错误状态码见表 7。

表 7　APPLICATION UNBLOCK 错误状态

SW1	SW2	含　义
‘64’	‘00’	标志状态位未变
‘65’	‘81’	内存失败
‘69’	‘82’	不满足安全状态
‘69’	‘87’	安全报文数据项丢失
‘69’	‘88’	安全报文数据项不正确
‘93’	‘03’	应用已被永久锁定

5.3.3.3　CARD BLOCK 命令

5.3.3.3.1　定义和范围

CARD BLOCK 命令使卡中所有应用永久失效。

当 CARD BLOCK 命令成功地完成后，所有后续的命令都将回送状态码“不支持此功能”，且不执行任何其他操作。

5.3.3.3.2　命令报文

CARD BLOCK 命令报文编码见表 8。

表 8　CARD BLOCK 命令报文

代码	值	代码	值
CLA	‘84’	Lc	数据字节数
INS	‘16’	Data	报文认证码(MAC)数据元
P1	‘00’，其他值保留为将来使用	Le	不存在
P2	‘00’，其他值保留为将来使用		

5.3.3.3.3　命令报文数据域

命令报文数据域包括根据本标准规定进行编码的 MAC 数据元。

5.3.3.3.4　响应报文数据域

响应报文数据域不存在。

5.3.3.3.5　响应报文状态码

CARD BLOCK 命令执行成功的状态码是‘9000’。

IC 卡可能回送的 CARD BLOCK 错误状态码见表 9。

表 9　CARD BLOCK 错误状态

SW1	SW2	含　义
‘64’	‘00’	标志状态位没变
‘65’	‘81’	内存失败
‘69’	‘87’	安全报文数据项丢失
‘69’	‘88’	安全报文数据项不正确

5.3.3.4　EXTERNAL AUTHENTICATION 命令

5.3.3.4.1　定义和范围

EXTERNAL AUTHENTICATION 命令要求 IC 卡中的应用验证密码。
IC 卡的响应包括命令处理状态的回送。

5.3.3.4.2　命令报文

EXTERNAL AUTHENTICATION 命令报文编码见表 10。

表 10　EXTERNAL AUTHENTICATION 命令报文

代码	值	代码	值
CLA	‘00’	Lc	08
INS	‘82’	Data	发卡方认证数据
P1	‘00’	Le	不存在
P2	‘00’		
注：EXTERNAL AUTHENTICATION 命令使用的算法参考值(P1)编码为‘00’表示无信息。算法参考值在命令发出之前是已知的，或者在数据域中提供。			

5.3.3.4.3　命令报文数据域

命令报文数据域中包含 8 字节的数据，该 8 字节数据为认证数据。

5.3.3.4.4　响应报文数据域

响应报文数据域不存在。

5.3.3.4.5　响应报文状态码

EXTERNAL AUTHENTICATION 命令执行成功的状态码是‘9000’。
IC 卡可能回送的 EXTERNAL AUTHENTICATION 警告状态码见表 11。

表 11　EXTERNAL AUTHENTICATION 警告状态

SW1	SW2	含　义
‘63’	‘00’	认证失败

IC 卡可能回送的 EXTERNAL AUTHENTICATION 错误状态码见表 12。

表 12　EXTERNAL AUTHENTICATION 错误状态

SW1	SW2	含　义
‘67’	‘00’	Lc 不正确
‘69’	‘83’	认证方法锁定
‘6A’	‘86’	参数 P1、P2 不正确

5.3.3.5　GET CHALLENGE 命令

5.3.3.5.1　定义和范围

GET CHALLENGE 命令请求一个用于安全相关过程的随机数。

该随机数只能用于下一条指令，无论下一条指令是否使用了该随机数，该随机数都将立即失效。

5.3.3.5.2　命令报文

GET CHALLENGE 命令报文编码见表 13。

表 13　GET CHALLENGE 命令报文

代码	值	代码	值
CLA	‘00’	Lc	不存在
INS	‘84’	Data	不存在
P1	‘00’	Le	‘04’
P2	‘00’		

5.3.3.5.3　命令报文数据域

命令报文数据域不存在。

5.3.3.5.4　响应报文数据域

响应报文数据域包括随机数，长度为 4 字节。

5.3.3.5.5　响应报文状态码

GET CHALLENGE 命令执行成功的状态码是‘9000’。

IC 卡可能回送的 GET CHALLENGE 错误状态码见表 14。

表 14　GET CHALLENGE 错误状态

SW1	SW2	含　义
‘6A’	‘81’	不支持此功能
‘6A’	‘86’	参数 P1、P2 不正确

5.3.3.6 GET RESPONSE 命令

5.3.3.6.1 定义和范围

该指令只用于 T=0 协议卡片。

当 APDU 不能用现有协议传输时，GET RESPONSE 命令提供了一种从卡片向接口设备传送 APDU(或 APDU 的一部分)的传输方法。

5.3.3.6.2 命令报文

GET RESPONSE 命令报文编码见表 15。

表 15 GET RESPONSE 命令报文

代码	值	代码	值
CLA	'00'	Lc	不存在
INS	'C0'	Data	不存在
P1	'00'	Le	响应的期望数据最大长度
P2	'00'		

5.3.3.6.3 命令报文数据域

命令报文数据域不存在。

5.3.3.6.4 响应报文数据域

响应报文数据域的长度由 Le 的值决定。

如果 Le 的值为零，在附加数据有效时，卡片应回送状态码'6CXX'，否则回送状态码'6F00'。

5.3.3.6.5 响应报文状态码

GET RESPONSE 命令执行成功的状态码是'9000'。

正常处理情况见表 16。

表 16 GET RESPONSE 正确响应命令报文

SW1	SW2	含　义
'61'	'XX'	正常处理 'XX'表示可以通过后续 GET RESPONSE 命令得到的额外数据长度

IC 卡可能回送的 GET RESPONSE 警告状态码见表 17。

表 17 GET RESPONSE 警告状态

SW1	SW2	含　义
'62'	'81'	回送的数据可能有错

IC 卡可能回送的 GET RESPONSE 错误状态码见表 18。

表 18　GET RESPONSE 错误状态

SW1	SW2	含　义
‘67’	‘00’	长度错误(Le 不正确)
‘6A’	‘86’	参数 P1、P2 不正确
‘6C’	‘XX’	长度错误(Le 不正确，‘XX’表示实际长度)
‘6F’	‘00’	数据无效

5.3.3.7　INTERNAL AUTHENTICATION 命令

5.3.3.7.1　定义和范围

INTERNAL AUTHENTICATION 命令提供了利用接口设备发来的随机数和自身存储的相关密钥进行数据认证的功能。

5.3.3.7.2　命令报文

INTERNAL AUTHENTICATION 命令报文编码见表 19。

表 19　INTERNAL AUTHENTICATION 命令报文

代码	值	代码	值
CLA	‘00’	Lc	认证数据的长度
INS	‘88’	Data	认证数据
P1	‘00’	Le	‘00’
P2	‘00’		
注：INTERNAL AUTHENTICATION 命令的参数 P1 为‘00’时的含义是无信息。P1 的值可事先得到，也可以在数据域中提供。INTERNAL AUTHENTICATION 命令的参数 P2 为‘00’时的含义是无信息。P2 的值可事先得到，也可以在数据域中提供。			

5.3.3.7.3　命令报文数据域

命令报文数据域的内容是应用专用的认证数据。

5.3.3.7.4　响应报文数据域

响应报文数据域内容是相关认证数据。

5.3.3.7.5　响应报文状态码

INTERNAL AUTHENTICATION 命令执行成功的状态码是‘9000’。

IC 卡可能回送的 INTERNAL AUTHENTICATION 警告状态码见表 20。

表 20　INTERNAL AUTHENTICATION 警告状态

SW1	SW2	含　义
‘62’	‘81’	回送的数据可能有错

IC 卡可能回送的 INTERNAL AUTHENTICATION 错误状态码见表 21。

表 21　INTERNAL AUTHENTICATION 错误状态

SW1	SW2	含　义
‘64’	‘00’	标志状态位未变
‘67’	‘00’	Lc 域不存在
‘68’	‘82’	不支持安全报文
‘69’	‘85’	不满足使用条件
‘6A’	‘80’	数据域参数不正确
‘6A’	‘86’	参数 P1、P2 不正确

5.3.3.8　PIN UNBLOCK 命令

5.3.3.8.1　定义和范围

PIN UNBLOCK 命令为发卡方提供了解锁个人密码的功能。

当 PIN UNBLOCK 命令成功完成后，卡将执行以下功能：

a）　重置个人密码尝试计数器的值；

b）　命令中个人密码的传递采用加密方式。

5.3.3.8.2　命令报文

PIN CHANGE/UNBLOCK 命令报文编码见表 22。

表 22　PIN CHANGE/UNBLOCK 命令报文

代码	值	代码	值
CLA	‘84’	Lc	数据字节数
INS	‘24’	Data	加密的个人密码数据元和 MAC 数据元
P1	‘00’	Le	不存在
P2	‘00’或‘01’		
注：当 P2＝‘00’时，Lc 应包括 MAC 数据元的长度，同时包括个人密码数据元和 MAC 数据元的长度。当 P2＝‘01’时，使用 DPUK 对 PIN 数据加密。			

5.3.3.8.3　命令报文数据域

命令报文数据域中个人密码数据元（如果存在）和其后的 MAC 数据元组成。

5.3.3.8.4　响应报文数据域

响应报文数据域不存在。

5.3.3.8.5　响应报文状态码

PIN CHANGE/UNBLOCK 命令执行成功的状态码是‘9000’。

IC 卡可能回送的 PIN CHANGE/UNBLOCK 警告状态码见表 23。

表 23　PIN CHANGE/UNBLOCK 警告状态

SW1	SW2	含　义
‘62’	‘00’	无信息提供
‘62’	‘81’	数据可能出错

IC 卡可能回送的 PIN CHANGE/UNBLOCK 错误状态码见表 24。

表 24　PIN CHANGE/UNBLOCK 错误状态

SW1	SW2	含　义
‘64’	‘00’	标志状态位没变
‘65’	‘81’	内存失败
‘69’	‘82’	不满足安全状态
‘69’	‘84’	引用数据无效
‘69’	‘87’	安全报文数据项丢失
‘69’	‘88’	安全报文数据项不正确
‘6A’	‘86’	参数 P1、P2 不正确
‘6A’	‘88’	未找到引用数据
‘93’	‘03’	应用被永久锁定

5.3.3.9　READ BINARY 命令

5.3.3.9.1　定义和范围

READ BINARY 命令用于读取二进制文件的内容或部分内容。

5.3.3.9.2　命令报文

READ BINARY 命令报文编码见表 25。

表 25　READ BINARY 命令报文

代码	值	代码	值
CLA	‘00’或‘04’	Lc	不存在(CLA=‘04’时除外)
INS	‘B0’	Data	不存在(CLA=‘04’时，应包括 MAC)
P1	表 26		
P2	从文件中读取的第一个字节的偏移地址	Le	‘00’

命令报文中定义的 P1 引用控制参数见表 26。

表 26　READ BINARY 命令引用控制参数

b8	b7	b6	b5	b4	b3	b2	b1	含　义
X 1	—							读取模式： 用 SFI 方式
—	0	0	—					RFU(如果 b8=1)
—			X	X	X	X	X	SFI(取值范围 21～30)

5.3.3.9.3　命令报文数据域

一般情况下，命令报文数据域不存在。当使用安全报文时，命令报文数据域中应包含 MAC。MAC 的计算方法和长度由应用决定。

5.3.3.9.4　响应报文数据域

当 Le 的值为零时，只要文件的最大长度在 256(短长度)之内或 65536(扩展长度)之内，则其全部字节将被读出。

5.3.3.9.5　响应报文状态码

READ BINARY 命令执行成功的状态码是‘9000’。

IC 卡可能回送的 READ BINARY 警告状态码见表 27。

表 27　READ BINARY 警告状态

SW1	SW2	含　义
‘62’	‘81’	部分回送的数据可能有错
‘62’	‘82’	文件长度＜Le

IC 卡可能回送的 READ BINARY 错误状态错误状态码见表 28。

表 28　READ BINARY 错误状态

SW1	SW2	含　义
‘67’	‘00’	长度错误(Lc 域为空)
‘69’	‘81’	命令与文件结构不相容
‘69’	‘82’	不满足安全状态
‘69’	‘86’	不满足命令执行的条件(非当前 EF)
‘6A’	‘81’	不支持此功能
‘6A’	‘82’	未找到文件
‘6B’	‘00’	参数错误(偏移地址超出了 EF)
‘6C’	‘XX’	长度错误(Le 错误；‘XX’为实际长度)

5.3.3.10 READ RECORD 命令

5.3.3.10.1 定义和范围

READ RECORD 命令用于读取记录文件的内容。

IC 卡的响应由回送记录组成。

5.3.3.10.2 命令报文

READ RECORD 命令报文编码见表 29。

表 29 READ RECORD 命令报文

代码	值	代码	值
CLA	‘00’或’04’	Lc	不存在(CLA=‘04’时除外)
INS	‘B2’	Data	不存在(CLA=‘04’时除外)
P1	记录号或记录标识	Le	‘00’
P2	见表 30		

命令报文中定义的 P2 引用控制参数见表 30。

表 30 READ RECORD 命令引用控制参数

b8	b7	b6	b5	b4	b3	b2	b1	含　义
X	X	X	X	X	—			SFI
—					1	0	0	P1 为记录号
—					0	0	0	P1 为记录标识

5.3.3.10.3 命令报文数据域

当无安全报文使用时,命令报文数据域不存在。使用安全报文时,命令报文的数据域中应包含 MAC。MAC 的计算方法和长度由应用决定。

5.3.3.10.4 响应报文数据域

所有执行成功的 READ RECORD 命令的响应报文数据域由读取的记录组成。

5.3.3.10.5 响应报文状态码

READ RECORD 命令执行成功的状态码是‘9000’。

IC 卡可能回送的警告状态码见表 31。

表 31 READ RECORD 警告状态

SW1	SW2	含　义
‘62’	‘81’	回送的数据可能有错

IC 卡可能回送的错误状态码见表 32。

表 32　READ RECORD 错误状态

SW1	SW2	含　义
‘64’	‘00’	标志状态位没变
‘67’	‘00’	长度错误(Lc 域不存在)
‘69’	‘81’	命令与文件结构不相容
‘6A’	‘81’	不支持此功能
‘6A’	‘82’	未找到文件
‘6A’	‘83’	未找到记录

5.3.3.11　SELECT 命令

5.3.3.11.1　定义和范围

SELECT 命令通过文件名或 AID 来选择 IC 卡中的 PSE、DDF 或 ADF。命令执行成功后，PSE、DDF 或 ADF 的路径被设定。

应用到 AEF 的后续命令将采用 SFI 方式联系到所选定的 PSE、DDF 或 ADF。

从 IC 卡的响应报文应由回送 FCI 组成。

5.3.3.11.2　命令报文

SELECT 命令报文编码见表 33。

表 33　SELECT 命令报文

代码	值	代码	值
CLA	‘00’	Lc	‘05’～‘10’
INS	‘A4’	Data	文件名
P1	见表 34	Le	‘00’
P2	‘00’第一个或仅有一个 ‘02’下一个		

命令报文中定义的 P1 引用控制参数见表 34。

表 34　SELECT 命令引用控制参数

b8	b7	b6	b5	b4	b3	b2	b1	含　义
0	0	0	0	0	—			—
—					1	—		通过文件名选择
—						0	0	—

5.3.3.11.3　命令报文数据域

命令报文数据域应包括所选择的 PSE 名、DF 名或 AID。

5.3.3.11.4 响应报文数据域

响应报文中数据域应包括所选择的 PSE、DDF 或 ADF 的 FCI。成功选择 PSE 后回送的定义 FCI 见表 35。

表 35 SELECT PSE 的响应报文(FCI)

标志	值	存在方式
'6F'	FCI 模板	M
'84'	DF 名	M
'A5'	FCI 专用数据	M
'88'	目录基本文件的 SFI	M

成功选择 DDF 后回送 FCI 定义见表 36。

表 36 SELECT DDF 的响应报文(FCI)

标志	值	存在方式
'6F'	FCI 模板	M
'84'	DF 名	M
'A5'	FCI 专用模板	M
'88'	目录基本文件的 SFI	M

成功选择 ADF 后回送 FCI 的定义见表 37。

表 37 SELECT ADF 的响应报文(FCI)

标志	值	存在方式
'6F'	FCI 模板	M
'84'	DF 名	M
'A5'	FCI 专用数据	M
'BF0C'	发卡方自定数据的 FCI	O

5.3.3.11.5 响应报文状态码

SELECT ADF 命令执行成功的状态码是'9000'。

IC 卡可能回送的 SELECT 警告状态码见表 38。

表 38 SELECT 警告状态

SW1	SW2	含　义
'62'	'83'	选择的文件无效
'62'	'84'	FCI 格式与 P2 指定的不符

IC 卡可能回送的 SELECT 错误状态码见表 39。

表 39 SELECT 错误状态

SW1	SW2	含　义
‘64’	‘00’	标志状态位没变
‘67’	‘00’	P1、P2 与 Lc 不一致
‘6A’	‘81’	不支持此功能
‘6A’	‘82’	未找到文件
‘6A’	‘86’	参数 P1、P2 不正确
注：SW1 SW2＝‘6A82’用于表示当卡支持部分文件名选择时，没有与此部分文件名相匹配的文件。		

5.3.3.12 UPDATE BINARY 命令

5.3.3.12.1 定义和范围

UPDATE BINARY 命令报文使用命令 APDU 中给定的数据修改 EF 文件中已有的数据。

5.3.3.12.2 命令报文

UPDATE BINARY 命令报文编码见表 40。

表 40 UPDATE BINARY 命令报文

代码	值	代码	值
CLA	‘00’或‘04’	Lc	后续数据域的长度
INS	‘D6’	Data	修改用的数据＋报文认证码(MAC)数据元(4 字节)
P1	见表 41		
P2	要修改的第一个字节的偏移地址	Le	不存在
注：CLA ＝‘00’不需要安全报文；CLA ＝ ‘04’需要安全报文。			

命令报文中定义的引用控制参数见表 41。

表 41 UPDATE BINARY 命令引用控制参数

b8	b7	b6	b5	b4	b3	b2	b1	含　义
X 1	—							读取模式： 用 SFI 方式
—	0	0	—					RFU(如果 b8＝1)
—			X	X	X	X	X	SFI(取值范围 21～30)

5.3.3.12.3 命令报文数据域

命令报文数据域包括更新原有数据的新数据。MAC 数据元为 4 字节。

5.3.3.12.4 响应报文数据域

响应报文数据域不存在。

5.3.3.12.5 响应报文状态码

UPDATE BINARY 命令执行成功的状态码是‘9000’。

IC 卡可能回送的 UPDATE BINARY 警告状态码见表 42。

表 42 UPDATE BINARY 警告状态

SW1	SW2	含 义
‘63’	‘CX’	使用内部重试程序更新成功 X=‘0’表示不提供计数器 X≠‘0’表示允许重试次数

IC 卡可能回送的 UPDATE BINARY 错误状态码见表 43。

表 43 UPDATE BINARY 错误状态

SW1	SW2	含 义
‘65’	‘81’	内存失败(修改失败)
‘67’	‘00’	长度错误(Lc 域为空)
‘69’	‘81’	命令与文件结构不相容
‘69’	‘82’	不满足安全状态
‘69’	‘86’	不满足命令执行的条件(不是当前的 EF)
‘6A’	‘81’	不支持此功能
‘6A’	‘82’	未找到文件
‘6B’	‘00’	参数错误(偏移地址超出了 EF)

5.3.3.13 UPDATE RECORD 命令

5.3.3.13.1 定义和范围

UPDATE RECORD 命令报文用命令 APDU 中给定的数据更改指定的记录。

在使用当前记录地址时,该命令将在修改记录成功后重新设定记录指针。

5.3.3.13.2 命令报文

UPDATE RECORD 命令报文编码见表 44。

表 44 UPDATE RECORD 命令报文

代码	值	代码	值
CLA	‘00’或‘04’	P2	见表 45
INS	‘DC’	Lc	后续数据域的长度
P1	P1=‘00’表示当前记录 P1≠‘00’指定的记录号	Data	更新原有记录的新记录
		Le	不存在
注:CLA =‘00’ 不需要安全报文;CLA =‘04’ 需要安全报文。			

命令报文中定义的引用控制参数见表45。

表45 UPDATE RECORD命令引用控制参数

b8	b7	b6	b5	b4	b3	b2	b1	含　义
X	X	X	X	X	—			SFI
—					0	0	0	第一个记录
—					0	0	1	最后一个记录
—					0	1	0	下一个记录
—					0	1	1	上一个记录
—					1	0	0	记录号在P1中给出
其余值								RFU

5.3.3.13.3 命令报文数据域

命令报文数据域由更新原有记录的新记录和MAC数据元(4字节)组成。

5.3.3.13.4 响应报文数据域

响应报文数据域不存在。

5.3.3.13.5 响应报文状态码

UPDATE RECORD命令执行成功的状态码是'9000'。

IC卡可能回送的UPDATE RECORD警告状态码见表46。

表46 UPDATE RECORD警告状态

SW1	SW2	含　义
'63'	'CX'	使用内部重试程序更新成功 X='0'表示不提供计数器 X≠'0'表示重试次数

IC卡可能回送的UPDATE RECORD错误状态码见表47。

表47 UPDATE RECORD错误状态

SW1	SW2	含　义
'65'	'81'	内存失败(修改失败)
'67'	'00'	长度错误
'69'	'81'	命令与文件结构不相容
'69'	'82'	不满足安全状态
'69'	'86'	不满足命令执行的条件(不是当前的EF)
'6A'	'81'	不支持此功能
'6A'	'82'	未找到文件
'6A'	'83'	未找到记录
'6A'	'84'	文件中存储空间不够
'6D'	'00'	INS不支持或错误
'6E'	'00'	CLA不支持或错误

5.3.3.14 VERIFY 命令

5.3.3.14.1 定义和范围

VERIFY 命令用于校验命令数据域中的个人密码的正确性。

5.3.3.14.2 命令报文

VERIFY 命令报文编码见表 48。

表 48 VERIFY 命令报文

代码	值	代码	值
CLA	‘00’	Lc	可变
INS	‘20’	Data	外部输入的个人密码
P1	‘00’	Le	不存在
P2	‘00’		
注：P2=‘00’表示无特殊限定符被使用。在 IC 卡上，VERIFY 命令在处理过程中应明确知道如何去寻找个人密码。			

5.3.3.14.3 命令报文数据域

命令报文数据域由持卡者输入的个人密码组成。

5.3.3.14.4 响应报文数据域

响应报文数据域不存在。

5.3.3.14.5 响应报文状态码

VERIFY 命令执行成功的状态码是‘9000’。

当前的应用选择中，命令数据域中外部输入的个人密码与卡中存放的个人密码校验失败时，IC 卡将回送 SW2=‘CX’，‘X’表示个人密码允许重试的次数；当卡回送‘C0’时，表示不能重试个人密码。

IC 卡可能回送的 VERIFY 警告状态码见表 49。

表 49 VERIFY 警告状态

SW1	SW2	含　义
‘63’	‘CX’	校验失败，‘X’表示允许重试的次数

IC 卡可能回送的 VERIFY 警告状态错误状态码见表 50。

表 50 VERIFY 错误状态

SW1	SW2	含　义
‘64’	‘00’	标志状态位没变
‘69’	‘83’	认证方法(个人密码)锁定
‘69’	‘84’	引用数据无效
‘6A’	‘86’	参数 P1 P2 不正确
‘6A’	‘88’	未找到引用数据

6 应用选择

6.1 一般要求

卡片和终端的应用选择应符合下列要求：

a) 终端应使用 SELECT 命令选择一个应用数据文件(ADF)，根据协议使用 IC 卡上的数据来决定选择哪种支付应用进行交易，其过程分两个步骤：
   1) 建立卡片与终端两者共同支持的应用列表；
   2) 在生成的应用列表中选择一个将要运行的应用。

b) 应用选择通常是最先执行的应用功能。一种支付系统应用包括以下内容：
   1) IC 卡上一组已由发卡方进行过客户化处理的数据文件；
   2) 一组由收单行或商户提供的终端中的数据；
   3) 一套卡和终端共同遵守的应用协议。

c) 所有应用都唯一的由一个应用标识符(AID)标识。这里描述的支付系统所采用的技术在设计上应能满足下列主要目标：
   1) 能够支持多功能 IC 卡；
   2) 能够支持多功能终端，而这些终端能够支持符合本标准的 IC 卡；
   3) IC 卡支持多应用，但不要求所有的应用都是支付应用；
   4) 尽可能保护现存应用，使其与本标准定义的应用在卡中共存；
   5) 最小的存储开销和处理开销；
   6) 具有允许发卡方优化选择过程的能力。

6.2 应用标识符的编码

IC 卡上允许存在其他应用提供者的应用数据文件(ADF)，但是其 RID 的定义应避免与支付应用 RID 的范围发生重复。

应用标识符(AID)的结构包含两个部分：

a) 一个经过注册的应用提供者标识符(长度为 5 字节)，它唯一地标识应用提供者。

b) 一个可选域，由应用提供者定义，最长 11 字节。这个域被称为“专用应用标识符扩展码(PIX)”，其长度为 0 到 11 字节，其值由应用提供者确定。该域的含义只对应于特定的 RID，不同 RID 下的 PIX 不需要唯一。

6.3 支付系统环境结构

IC 卡的支付系统环境应起始于一个真实存在的目录数据文件(DDF)，该 DDF 文件应包含支付系统目录以及本标准定义的所有 DDF 信息。

初始 DDF 所附属的目录包含了 ADF 的入口地址，该目录也可以包含其他 DDF 的入口地址，但这些入口地址的格式应符合本标准。

6.4 支付系统目录编码

支付系统目录(下文简称目录)是一个记录文件，用 1 到 10 的短文件标识符(SFI)标识。该目录附属于 DDF，目录的 SFI 包含在 DDF 文件控制信息中。目录可以使用本标准定义的 READ RECORD 命令进行读取。目录中一个记录可以包含几个入口地址，但一个入口地址不能跨越多个记录存储。

支付系统目录中的每一个记录都是一个结构数据对象，其值由如下所示的一个或多个目录的入口组成。

每个记录的格式，见表 51。

表 51　支付系统目录记录格式

标签 ‘70’	数据域长度 (L)	标识符 ‘61’	目录入口 1 长度	目录入口 1 (ADF)	……	标识符 ‘61’	目录入口 n 长度	目录入口 n (ADF)

支付系统目录记录中应当不包含任何通往 DDF 的入口。如果终端在处理这些记录时遇到了 DDF 的入口，终端可以忽略这些入口或者处理这些入口。

支付系统目录中的每一个入口都是一个应用模板(标签‘61’)，它包含表 52 或表 53 所示的信息。

表 52　ADF 目录入口格式

标签	长度	值	存在方式
‘4F’	5～16	ADF 名称(AID)	M
‘50’	1～16	应用标签	M
‘9F12’	1～16	应用优先名称	O
‘87’	1	应用优先权标识符	O

表 53　应用优先权标识符格式

b8	b7～b5	b4～b1	定　义
1	—		需要持卡人确认方可选择应用
0			不需持卡人确认即可选择应用
	‘XXX’	—	预留
—		0 0 0 0	未指定优先权
		‘XXXX’ (0 0 0 0 除外)	应用的排列或选择顺序，从 1～15，其中最高优先权为 1

6.5　目录入口中执行命令的使用

一个目录入口地址总是与卡中的一个数据文件(DF)相对应。如果在目录入口地址中没有指定一个“执行的命令”，则需执行 SELECT 命令来选择入口地址中指定的 DF，并使用目录中 ADF 名或 DDF 名作为文件名。有些 IC 卡对 SELECT 命令的解释具有二义性，比如对于支持 DF 部分名的 IC 卡就有可能将其入口地址中指定的文件名当成另一 DF 文件的部分名而造成选择应用错误。

“执行的命令”作为一种机制提供给 IC 卡，使得 IC 卡可以利用这个机制准确地选择正确的 DF，即选择与目录入口地址对应的 DF。“执行的命令”可以是 SELECT 命令的变形，即不一定是“按名称选择”的形式(例如按路径或文件标识选择)；也可以是其他命令，通过这些命令也能实现正确选择 DF 的结果并返回 FCI。当“执行的命令”数据项存在时，终端会利用它代替“按名称选择”命令来选择相关的 DF。

6.6　其他目录

除了初始目录之外，其他目录在支付系统环境下都是可选的，对此类目录的存在数目没有明确限制。每一个目录由一个目录 SFI 定位，SFI 存放在每个 DDF 的 FCI 中。目录 SFI 包括执行 READ RECORD 命令读目录时所用的 SFI。当包含该目录的 DDF 为当前选定的文件时，SFI 可用来读此目录。

目录 SFI 数据应出现在一个 DDF(FCI 专用模板)的 FCI 专用数据区域内。一个 DDF 最多包含一

个目录，因此目录 SFI 数据只在 FCI 中出现一次。

除了初始目录之外，所有目录入口均为 ADF 文件，或以包含目录 DDF 名称开始的 DDF。

6.7 终端的应用选择

6.7.1 直接选择应用

终端可以简单地使用 SELECT 命令依次选择每个应用。如果 SELECT 命令执行成功（回送 SW1SW2＝'9000'），则该终端将它所支持的 AID 与被选择文件的 FCI 中的文件名进行比较，通过比较的结果来查证 IC 卡是否支持此应用。如果二者相匹配，IC 卡支持该应用；如果返回的文件名比 AID 长而 AID 与返回文件名的起始部分相符，终端则重新发送 SELECT 命令并再次对选择进行验证；如果 IC 卡回送 SW1SW2 不等于'9000'，或者即使 IC 卡回送 SW1SW2 等于'9000'，而 AID 与文件名不相符且与文件名起始部分也不相符，证明卡不支持此应用。

一但终端支持的应用都被选择出来，则 IC 卡和终端都支持的应用列表就可以确定。然后终端可以选择指定的应用来运行。

直接选择适用于那些仅支持较少应用的终端，并且不能支持持卡人潜在的应用。这种方式不支持终端访问应用标签或应用优先名称，这些名称仅存在于目录中。

6.7.2 通过支付系统目录选择应用

终端可通过使用 IC 卡的目录（或多个目录）来确定卡片所支持的应用。应保证 IC 卡目录的结构设计正确，以便终端可以按照本标准描述的过程正确地选择应用。终端正确使用目录的步骤如下：

a) 终端首先在支付系统环境下用“SELECT”命令对文件'1PAY.SYS.DDF01'直接选择。由此建立支付系统环境并进入初始目录。

b) 终端从第一条记录开始，连续读目录中的所有记录，直到卡回送 SW1SW2＝'6A83'，表示所需记录序号已不存在。在执行 READ RECORD 命令查找第一个记录时，如果卡回送 SW1SW2＝'6A83'，则表示目录为空，转至下面步骤 f。

c) 如果目录中某个 ADF 名与终端支持的一个应用名相符，则将该应用列入最终应用选择的“候选名单”中。

d) 如果目录中出现一个指向 DDF 的入口地址，且该 DDF 的名称至少与一个终端所支持的 AID 的前几位匹配（例如：一个名为 1234 的 DDF 可与一个名为 12345678 的 AID 匹配），则终端选择该 DDF。如果该入口包含一个“执行的命令”，则执行该命令完成选择；如果不存在“执行的命令”，终端发出带 DDF 名的“SELECT”命令。使用所选 DDF 的文件控制信息（FCI）中的目录短文件标识符（SFI），读出目录并按规则 c 处理，之后终端继续回到上一个目录处理。

e) 当终端处理完第一个目录的列表后，所有能够按此方式找到的 ADF 就确定了，查找完毕。

f) 终端也可以采用其他方式寻找卡内其他的专用应用（例如：用 AID 找出本地的或非支付应用的专用选择方式）。

6.7.3 选择应用并执行操作

当终端确定了卡与终端相互支持的应用列表之后，下一步即要选取某个应用进行操作。可通过如下方法实现：

a) 如果没有互相支持的应用，交易终止。

b) 如果只有一个相互支持的应用，终端核查应用优先表明符的 b8 位。如果 b8 等于'0'，终端选择该应用。如果 b8 等于'1'并且终端规定要有持卡人的确认，在这种情况下，终端需要向持卡人提出确认请求，如持卡人同意，即选择该应用。如果终端没有规定要有持卡人的确认，或者

终端请求确认被拒绝，终端终止该交易。

c) 宜采用应用列表请持卡人选择。将采用级别优先方式为持卡人提供应用列表目录，高优先级别的应用在先。如果卡中没有指定优先顺序，则以终端的应用优先顺序为准；如果终端也没有指定优先顺序，则按照应用在卡中出现的顺序为准。如果出现多个应用重复指定优先顺序，或个别入口地址缺少应用优先表明符的情况，也可采用类似的方法，也就是说，在这种情况下终端可使用自己的优先顺序，也可以按卡上顺序将有重复优先符或无优先符的应用显示出来。

d) 终端可在没有持卡人协助的情况下选择应用。在这种情况下，终端应从相互支持的应用列表中选择优先级别最高的应用，如果终端不能对选择的应用提供确认，则应用选择禁止（应用优先表明符的 b8 等于‘1’）。

e) 一旦终端或持卡人确定了待执行的应用，则该应用被选中。如果与应用相关的目录入口地址指定了一个“执行的命令”，终端执行该命令进行应用的选择。如果不存在“执行的命令”，终端发出一个“SELECT”命令进行应用的选择。无论使用哪种命令，如果命令回送的 SW1SW2 值不是‘9000’，则此应用将从候选列表中删除，之后再将删除后的列表显示给持卡人，或者选择下一个优先级高的应用，重新进行应用选择。在合适的情况下，终端要给持卡人以提示。

7 安全机制及安全要求

7.1 安全机制

应满足 CJ/T 166 中规定的要求。

7.2 安全要求

应满足 CJ/T 166 中规定的要求。

7.3 加密算法

7.3.1 对称算法

安全报文允许使用 64 位块加密算法。

7.3.2 非对称算法

算法被用来进行静态和动态数据验证以及数字签名。

在选择公开密钥模数的长度时，应该考虑到密钥的生命周期以及在此生命周期内被解密的可能性。每个密钥的长度范围（上、下限）在其相应的专用标准中规定。

发卡方公开密钥的指数长度与 IC 卡公开密钥的指数长度由发卡方决定。指数可以是预先约定的固定的数字如 2、3 或 $2^{16}+1$，但是它的长度不能超过其对应的密钥模数长度的四分之一。

该数字签名算法中的公开密钥算法的标志码为 16 进制数字‘01’。

7.3.3 安全哈希算法

安全哈希算法应符合 JR/T 0025 规定。

8 电子存折/电子钱包应用

8.1 应用说明

电子存折/电子钱包应用是为持卡人进行交易而设计的一种应用。对于一张 IC 卡来说，它可以同

时支持电子存折和电子钱包两种应用，也可以只支持其中的一种。卡片上两种应用的存在情况可以由应用类型标识来指明。

8.2 文件

8.2.1 文件结构

电子存折/电子钱包应用对应的专用文件(DF)与基本数据文件构成一个树状结构的分支。该专用文件是其下属的基本数据文件的入口点。

8.2.2 专用文件

电子存折/电子钱包应用所对应的专用文件(DF)包含一个文件控制信息(FCI)。通过该文件可以对基本数据文件(EF)进行访问。该专用文件的上一层专用文件是主控文件(MF)。

8.2.3 基本数据文件

电子存折/电子钱包应用下的基本数据文件有两种类型：记录文件类型和二进制文件类型。

8.2.4 文件选择

电子存折/电子钱包应用的专用文件采用应用标识符(AID)方式进行选择。成功选择了电子存折/电子钱包应用的专用文件后，该专用文件被设置为当前文件，并允许使用该应用的特殊命令对其进行操作。基本数据文件的选择是通过 READ 命令并采用 SFI 方式实现的。

8.3 命令

8.3.1 概述

描述电子存折/电子钱包应用(以下简称 ED/EP 应用)的命令和响应。

本标准详细定义了命令报文和响应报文的数据元。

在应用执行过程中，IC 卡总是处于以下状态之一，在一种状态下，只有某些命令能够被执行。IC 卡具有的状态如下：

空闲状态；

圈存状态；

消费/取现状态；

圈提状态；

修改状态。

应用选择完成后，IC 卡首先进入空闲状态。当 IC 卡从终端接收到一条命令时，它应首先检查当前状态是否允许执行该命令。在命令执行成功后，IC 卡将进入另一个状态(或同一个)。如果命令执行不成功，则 IC 卡进入空闲状态。ED/EP 应用的状态命令见表 54。

表 54 状态命令

状态命令	空闲	圈存	消费/取现	圈提	修改	CAPP1	CAPP2
CREDIT FOR LOAD	N/A	空闲	N/A	N/A	N/A	N/A	N/A
DEBITFOR PURCHASE/CASH WITHDRAW	N/A	N/A	空闲	N/A	N/A	N/A	N/A
DEBIT FOR UNLOAD	N/A	N/A	N/A	空闲	N/A	N/A	N/A
GET BALANCE	空闲	圈存	消费/取现	圈提	修改	CAPP1	CAPP2

表 54(续)

状态命令	空闲	圈存	消费/取现	圈提	修改	CAPP1	CAPP2
GET TRANSACTION PROVE	空闲	圈存	消费/取现	圈提	修改	CAPP1	CAPP2
INITIALIZE FOR LOAD	圈存	圈存	圈存	圈存	圈存	N/A	N/A
INITIALIZE FOR PURCHASE	消费/取现	消费/取现	消费/取现	消费/取现	消费/取现	N/A	N/A
INITIALIZE FOR CASH WITHDRAW	消费/取现	消费/取现	消费/取现	消费/取现	消费/取现	N/A	N/A
INITIALIZE FOR UNLOAD	圈提	圈提	圈提	圈提	圈提	N/A	N/A
INITIALIZE FOR UPDATE	修改	修改	修改	修改	修改	N/A	N/A
UPDATE OVERDRAW LIMIT	N/A	N/A	N/A	N/A	空闲	N/A	N/A
GET MESSAGE	空闲	圈存	消费/取现	圈提	修改	CAPP1	CAPP2
INITIALIZE FOR CAPP PURCHASE	CAPP1	N/A	N/A	N/A	N/A	N/A	N/A
UPDATE CAPP DATA CACHE	N/A	N/A	N/A	N/A	N/A	CAPP2	CAPP2
DEBIT FOR CAPP PURCHASE	N/A	N/A	N/A	N/A	N/A	N/A	空闲

8.3.2 CHANGE PIN 命令

8.3.2.1 定义和范围

CHANGE PIN 允许持卡人将当前个人密码修改为新的密码。CHANGE PIN 命令中的个人密码(PIN)值以明文方式传送。命令数据中个人密码(PIN)是以‘cn’格式存放的,它不需要整字节的填充,只有最低有效字节的低半字节可能需要填充,且填以‘F’。

当 CHANGE PIN 命令成功完成后,卡片要进行以下操作:

a) 密码尝试计数器复位至密码尝试次数的上限;

b) 将原个人密码置为新的个人密码。

8.3.2.2 命令报文

CHANGE PIN 命令报文见表 55。

表 55 CHANGE PIN 命令报文

代码	值	代码	值
CLA	‘80’	Lc	‘05’～‘0D’
INS	‘5E’	Data	当前 PIN ‖ ‘FF’ ‖ 新的 PIN
P1	‘01’	Le	不存在
P2	‘00’		

8.3.2.3 响应报文数据域

响应报文的数据域不存在。

8.3.2.4 响应报文的状态码

CHANGE PIN 命令执行成功的状态码是‘9000’。

IC卡可能回送的CHANGE PIN错误状态见表56。

表56 CHANGE PIN错误状态

SW1	SW2	含　义
‘63’	‘CX’	验证失败，还剩下X次尝试机会
‘65’	‘81’	内存错误
‘69’	‘01’	命令不接受(无效状态)
‘69’	‘83’	验证方法锁定
‘69’	‘85’	使用条件不满足
‘6A’	‘80’	数据域参数不正确
‘6A’	‘86’	P1、P2参数不正确

8.3.3 CREDIT FOR LOAD命令

8.3.3.1 定义和范围

CREDIT FOR LOAD命令用于圈存交易。

8.3.3.2 命令报文

CREDIT FOR LOAD命令报文见表57。

表57 CREDIT FOR LOAD命令报文

代码	值	代码	值
CLA	‘80’	Lc	‘0b’
INS	‘52’	Data	见表54
P1	‘00’	Le	‘04’或‘00’
P2	‘00’		

8.3.3.3 命令报文数据域

CREDIT FOR LOAD命令报文数据域见表58。

表58 CREDIT FOR LOAD命令报文数据域

说　明	长度(字节)
交易日期(主机)	4
交易时间(主机)	3
MAC2	4

8.3.3.4 响应报文数据域

CREDIT FOR LOAD响应报文数据域见表59。

表 59　CREDIT FOR LOAD 响应报文数据域

说　　明	长度(字节)
TAC	4
注：CREDIT FOR LOAD 命令执行不成功，则只在响应报文中回送 SW1 和 SW2。	

8.3.3.5　响应报文的状态码

CREDIT FOR LOAD 命令执行成功的状态码是‘9000’。

IC 卡可能回送的 CHANGE PIN 错误状态见表 60。

表 60　CREDIT FOR LOAD 错误状态

SW1	SW2	含　　义
‘65’	‘81’	内存错误
‘67’	‘00’	长度错误
‘69’	‘01’	命令不接受
‘69’	‘85’	使用条件不满足
‘93’	‘02’	MAC 无效

8.3.4　GET MESSAGE 命令

8.3.4.1　定义和范围

GET MESSAGE 命令用于消费/取现交易。读取 CPU 卡中的安全认证识别码，即芯片安全区域内的安全认证识别码或芯片中的 MID ‖ UID0UID1UID2UID3 ‖ 四字节认证码，将安全认证识别码发送给 PSAM 卡进行认证。该命令应在任意目录下都可以执行。

8.3.4.2　命令报文

GET MESSAGE 命令报文见表 61。

表 61　GET MESSAGE 命令

代码	值	代码	值
CLA	‘80’	P2	‘00’
INS	‘CA’	Le	‘09’
P1	‘0X’	Data	不存在
注：P1 中‘0X’，X=0 为 3DES 算法，X=3 为 SM1 算法，X=4 为 SM4 算法，算法标识说明应符合附录 A 的要求。			

8.3.4.3　命令报文数据域

GET MESSAGE 命令报文数据域不存在。

8.3.4.4 响应报文数据域

GET MESSAGE 命令执行成功的响应报文数据域见表 62。如果命令执行不成功，则只在响应报文中回送 SW1 和 SW2。响应报文数据回送 9 字节安全认证识别码。

表 62 GET MESSAGE 命令响应报文数据

说　　明	长度(字节)
安全认证识别码	9

8.3.4.5 响应报文状态码

GET MESSAGE 命令执行成功的状态码是‘9000’。

IC 卡可能回送的 GET MESSAGE 命令错误状态码警告状态码见表 63。

表 63 GET MESSAGE 命令错误状态码

SW1	SW2	含　　义
‘67’	‘00’	长度错误
‘6A’	‘86’	P1、P2 参数不正确
‘6D’	‘00’	INS 不支持或错误
‘6E’	‘00’	CLA 不支持或错误

8.3.5 DEBIT FOR PURCHASE/CASH WITHDRAW 命令

8.3.5.1 定义和范围

DEBIT FOR PURCHASE/CASH WITHDRAW 命令用于消费/取现交易。

8.3.5.2 命令报文

执行 INITIALIZE FOR PURCHASE 或 INITIALIZE FOR CASH WITHDRAW 后即选择了消费/取现交易。DEBIT FOR PURCHASE/CASH WITHDRAW 命令报文见表 64。

表 64 DEBIT FOR PURCHASE/CASH WITHDRAW 命令报文

代码	值	代码	值
CLA	‘80’	Lc	‘0F’
INS	‘54’	Data	见表 65
P1	‘01’	Le	‘08’或‘00’
P2	‘00’		

8.3.5.3 命令报文数据域

DEBIT FOR PURCHASE/CASH WITHDRAW 命令报文数据域见表 65。

表 65　DEBIT FOR PURCHASE/CASH WITHDRAW 命令报文数据域

说　　明	长度(字节)
终端交易序号	4
交易日期(终端)	4
交易时间(终端)	3
MAC1	4

8.3.5.4　响应报文数据域

命令执行成功的响应报文数据域见表 66。

表 66　DEBIT FOR PURCHASE/CASH WITHDRAW 响应报文数据域

说　　明	长度(字节)
TAC	4
MAC2	4
注:DEBIT FOR PURCHASE/CASH WITHDRAW 命令执行不成功,则只在响应报文中回送 SW1 和 SW2。	

8.3.5.5　响应报文的状态码

DEBIT FOR PURCHASE/CASH WITHDRAW 命令执行成功的状态码是‘9000’。

IC 卡可能回送的 DEBIT FOR PURCHASE/CASH WITHDRAW 错误状态见表 67。

表 67　DEBIT FOR PURCHASE/CASH WITHDRAW 错误状态

SW1	SW2	含　　义
‘65’	‘81’	内存错误
‘67’	‘00’	长度错误
‘69’	‘01’	命令不接受
‘69’	‘85’	使用条件不满足
‘93’	‘01’	金额不足
‘93’	‘02’	MAC 无效

8.3.6　DEBIT FOR UNLOAD 命令

8.3.6.1　定义和范围

DEBIT FOR UNLOAD 命令用于圈提交易。

8.3.6.2　命令报文

DEBIT FOR UNLOAD 命令报文见表 68。

表 68　DEBIT FOR UNLOAD 命令报文

代码	值	代码	值
CLA	‘80’	Lc	‘0B’
INS	‘54’	Data	见表 69
P1	‘03’	Le	‘04’或‘00’
P2	‘00’		

8.3.6.3　命令报文数据域

DEBIT FOR UNLOAD 命令报文数据域见表 69。

表 69　DEBIT FOR UNLOAD 命令报文数据域

说　　明	长度(字节)
交易日期(主机)	4
交易时间(主机)	3
MAC2	4

8.3.6.4　响应报文数据域

DEBIT FOR UNLOAD 命令执行成功的响应报文数据域见表 70。

表 70　DEBIT FOR UNLOAD 响应报文数据域

说　　明	长度(字节)
MAC3	4
注：DEBIT FOR UNLOAD 命令执行不成功，则只在响应报文中回送 SW1 和 SW2。	

8.3.6.5　响应报文的状态码

DEBIT FOR UNLOAD 命令执行成功的状态码是‘9000’。

IC 卡可能回送的 DEBIT FOR UNLOAD 错误状态见表 71。

表 71　DEBIT FOR UNLOAD 错误状态

SW1	SW2	含　　义
‘65’	‘81’	内存错误
‘67’	‘00’	长度错误
‘69’	‘01’	命令不接受
‘93’	‘02’	MAC 无效

8.3.7　GET BALANCE 命令

8.3.7.1　定义和范围

GET BALANCE 命令用于读取电子存折或电子钱包余额，实现查询余额交易。读取电子存折余

额需验证个人密码(PIN)。

8.3.7.2 命令报文

GET BALANCE 命令报文见表 72。

表 72 GET BALANCE 命令报文

代码	值	代码	值
CLA	'80'	Lc	不存在
INS	'5C'	Data	见表 73
P1	'00'	Le	'04'或'00'
P2	'01' 或 '02';'01'用于 ED,'02'用于 EP;其他值保留		

8.3.7.3 响应报文数据域

命令执行成功的响应报文数据域见表 73。

表 73 GET BALANCE 响应报文数据域

说 明	长度(字节)
ED 余额或 EP 余额	4
注:GET BALANCE 命令执行不成功,则只在响应报文中回送 SW1 和 SW2。	

8.3.7.4 响应报文的状态码

GET BALANCE 命令执行成功的状态码是'9000'。

IC 卡可能回送的 GET BALANCE 错误状态见表 74。

表 74 GET BALANCE 错误状态

SW1	SW2	含 义
'65'	'81'	内存错误
'67'	'00'	长度错误
'69'	'85'	使用条件不满足
'6A'	'86'	P1、P2 参数不正确
'93'	'02'	MAC 无效

8.3.8 GET TRANSACTION PROVE 命令

8.3.8.1 定义和范围

GET TRANSACTION PROVE 命令提供了一种在交易处理过程中的恢复机制。

8.3.8.2 命令报文

GET TRANSACTION PROVE 命令报文见表 75。

表 75　GET TRANSACTION PROVE 命令报文

代码	值	代码	值
CLA	‘80’	Lc	‘02’
INS	‘5A’	Data	见表 76
P1	‘00’	Le	‘08’
P2	要取的 MAC 或/和 TAC 所对应的交易类型标识		

8.3.8.3　命令报文数据域

GET TRANSACTION PROVE 命令报文数据域见表 76。

表 76　GET TRANSACTION PROVE 命令报文数据域

说　　明	长度(字节)
要取的 MAC 或/和 TAC 所对应的 ED/EP 联机或脱机交易序号	2

8.3.8.4　响应报文数据域

如果命令中指定的交易类型标识和 ED/EP 联机或脱机交易序号对应的 MAC 或 TAC 可用，则响应报文数据域见表 77。

表 77　GET TRANSACTION PROVE 响应报文数据域

说　　明	长度(字节)
MAC 或 TAC	4

8.3.8.5　响应报文的状态码

GET TRANSACTION PROVE 命令执行成功的状态码是‘9000’。

IC 卡可能回送的 GET TRANSACTION PROVE 错误状态见表 78。

表 78　GET TRANSACTION PROVE 错误状态

SW1	SW2	含　　义
‘65’	‘81’	内存错误
‘69’	‘01’	命令不接受
‘69’	‘85’	使用条件不满足
‘94’	‘06’	所需 MAC 不可用

8.3.9　INITIALIZE FOR CASH WITHDRAW 命令

8.3.9.1　定义和范围

INITIALIZE FOR CASH WITHDRAW 命令用于初始化取现交易。

8.3.9.2 命令报文

INITIALIZE FOR CASH WITHDRAW 命令报文见表 79。

表 79 INITIALIZE FOR CASH WITHDRAW 命令报文

代码	值	代码	值
CLA	'80'	Lc	'0B'
INS	'50'	Data	见表 80
P1	'02'	Le	'0E'或'00'
P2	'01'用于 ED 取现交易;其他值保留		

8.3.9.3 命令报文数据域

INITIALIZE FOR CASH WITHDRAW 命令报文的数据域见表 80。

表 80 INITIALIZE FOR CASH WITHDRAW 命令报文数据域

说　明	长度(字节)
密钥索引号	1
交易金额	4
终端机编号	6

8.3.9.4 响应报文数据域

INITIALIZE FOR CASH WITHDRAW 命令执行成功的响应报文数据域见表 81。

表 81 INITIALIZE FOR CASH WITHDRAW 响应报文数据域

说　明	长度(字节)
ED 余额	4
ED 脱机交易序号(IC 卡)	2
透支限额	3
密钥版本号(DPK)	1
算法标识(DPK)	1
伪随机数(IC 卡)	4
注:当 INITIALIZE FOR CASH WITHDRAW 命令执行不成功,则只在响应报文中回送 SW1 和 SW2。	

8.3.9.5 响应报文的状态码

INITIALIZE FOR CASH WITHDRAW 命令执行成功的状态码是'9000'。

IC 卡可能回送的 INITIALIZE FOR CASH WITHDRAW 错误状态见表 82。

表 82　INITIALIZE FOR CASH WITHDRAW 错误状态

SW1	SW2	含　义
‘65’	‘81’	内存错误
‘69’	‘85’	使用条件不满足
‘94’	‘01’	金额不足
‘94’	‘02’	交易计数器到达最大值
‘94’	‘03’	密钥索引不支持

8.3.10　INITIALIZE FOR LOAD 命令

8.3.10.1　定义和范围

INITIALIZE FOR LOAD 命令用于初始化圈存交易。

8.3.10.2　命令报文

INITIALIZE FOR LOAD 命令报文见表 83。

表 83　INITIALIZE FOR LOAD 命令报文

代码	值	代码	值
CLA	‘80’	Lc	‘0B’
INS	‘50’	Data	见表 84
P1	‘00’	Le	‘10’或‘00’
P2	‘01’ 或 ‘02’;‘01’用于 ED,‘02’用于 EP;其他值保留		

8.3.10.3　命令报文数据域

INITIALIZE FOR LOAD 命令报文数据域见表 84。

表 84　INITIALIZE FOR LOAD 命令报文数据域

说　明	长度(字节)
密钥索引号	1
交易金额	4
终端机编号	6

8.3.10.4　响应报文数据域

INITIALIZE FOR LOAD 命令执行成功的响应报文数据域见表 85。

表 85　INITIALIZE FOR LOAD 响应报文

说　　明	长度(字节)
ED 或 EP 余额	4
ED 或 EP 联机交易序号	2
密钥版本号(DLK)	1
算法标识(DLK)	1
伪随机数(IC 卡)	4
MAC1	4
注：当 INITIALIZE FOR LOAD 命令执行不成功，则只在响应报文中回送 SW1 和 SW2。	

8.3.10.5　响应报文的状态码

INITIALIZE FOR LOAD 命令执行成功的状态码是‘9000’。

IC 卡可能回送的 INITIALIZE FOR LOAD 错误状态见表 86。

表 86　INITIALIZE FOR LOAD 错误状态

SW1	SW2	含　　义
‘65’	‘81’	内存错误
‘67’	‘00’	长度错误
‘69’	‘85’	使用条件不满足
‘6A’	‘81’	功能不支持
‘6A’	‘86’	P1、P2 参数不正确
‘94’	‘02’	交易计数器达到最大值
‘94’	‘03’	密钥索引不支持

8.3.11　INITIALIZE FOR PURCHASE 命令

8.3.11.1　定义和范围

INITIALIZE FOR PURCHASE 命令用于初始化消费交易。

8.3.11.2　命令报文

INITIALIZE FOR PURCHASE 命令报文见表 87。

表 87　INITIALIZE FOR PURCHASE 命令报文

代码	值	代码	值
CLA	‘80’	Lc	‘0B’
INS	‘50’	Data	见表 88
P1	‘02’	Le	‘0F’或‘00’
P2	‘01’ 或 ‘02’；‘01’用于 ED，‘02’用于 EP；其他值保留		

8.3.11.3 命令报文数据域

INITIALIZE FOR PURCHASE 命令报文的数据域见表 88。

表 88 INITIALIZE FOR PURCHASE 命令报文数据域

说　　明	长度(字节)
密钥索引号	1
交易金额	4
终端机编号	6

8.3.11.4 响应报文数据域

命令执行成功的响应报文数据域见表 89。

表 89 INITIALIZE FOR PURCHASE 响应报文数据域

说　　明	长度(字节)
ED 或 EP 余额	4
ED 脱机交易序号或 EP 脱机交易序号	2
透支限额	3
密钥版本号(DPK)	1
算法标识(DPK)	1
伪随机数(IC 卡)	4
注：当 INITIALIZE FOR PURCHASE 命令执行不成功，则只在响应报文中回送 SW1 和 SW2。	

8.3.11.5 响应报文的状态码

INITIALIZE FOR PURCHASE 命令执行成功的状态码是‘9000’。

IC 卡可能回送的 INITIALIZE FOR PURCHASE 错误状态见表 90。

表 90 INITIALIZE FOR PURCHASE 错误状态

SW1	SW2	含　　义
‘65’	‘81’	内存错误
‘69’	‘85’	使用条件不满足
‘94’	‘01’	金额不足
‘94’	‘02’	交易计数器达到最大值
‘94’	‘03’	密钥索引不支持

8.3.12 INITIALIZE FOR UNLOAD 命令

8.3.12.1 定义和范围

INITIALIZE FOR UNLOAD 命令用于初始化圈提交易。

8.3.12.2 命令报文

INITIALIZE FOR UNLOAD 命令报文见表 91。

表 91 INITIALIZE FOR UNLOAD 命令报文

代码	值	代码	值
CLA	'80'	Lc	'0B'
INS	'50'	Data	见表 92
P1	'05'	Le	'10'或'00'
P2	'01'用于 ED 圈提交易;其他值保留		

8.3.12.3 命令报文数据域

INITIALIZE FOR UNLOAD 命令报文的数据域见表 92。

表 92 INITIALIZE FOR UNLOAD 命令报文数据域

说　明	长度(字节)
密钥索引号	1
交易金额	4
终端机编号	6

8.3.12.4 响应报文数据域

命令执行成功的响应报文数据域见表 93。

表 93 INITIALIZE FOR UNLOAD 响应报文数据域

说　明	长度(字节)
ED 余额	4
ED 联机交易序号	2
密钥版本号(DULK)	1
算法标识(DULK)	1
伪随机数(IC 卡)	4
MAC1	4
注:当 INITIALIZE FOR UNLOAD 命令执行不成功,则只在响应报文中回送 SW1 和 SW2。	

8.3.12.5 响应报文的状态码

INITIALIZE FOR UNLOAD 命令执行成功的状态码是'9000'。

IC 卡可能回送的 INITIALIZE FOR UNLOAD 错误状态见表 94。

表 94　INITIALIZE FOR UNLOAD 错误状态

SW1	SW2	含　义
‘65’	‘81’	内存错误
‘67’	‘00’	长度错误
‘69’	‘85’	使用条件不满足
‘6A’	‘86’	P1、P2 参数不正确
‘94’	‘01’	金额不足
‘94’	‘02’	交易计数器达到最大值
‘94’	‘03’	密钥索引不支持

8.3.13　INITIALIZE FOR UPDATE 命令

8.3.13.1　定义和范围

INITIALIZE FOR UPDATE 命令用于初始化修改透支限额交易。

8.3.13.2　命令报文

INITIALIZE FOR UPDATE 命令报文见表 95。

表 95　INITIALIZE FOR UPDATE 命令报文

代码	值	代码	值
CLA	‘80’	Lc	‘07’
INS	‘50’	Data	见表 96
P1	‘04’	Le	‘14’或‘00’
P2	‘01’		

8.3.13.3　命令报文数据域

INITIALIZE FOR UPDATE 命令报文的数据域见表 96。

表 96　INITIALIZE FOR UPDATE 命令报文数据域

说　明	长度(字节)
密钥索引号	1
终端机编号	6

8.3.13.4　响应报文数据域

命令执行成功的响应报文数据域见表 97。

表 97　INITIALIZE FOR UPDATE 响应报文数据域

说　明	长度(字节)
ED 余额	4
ED 联机交易序号	2
旧透支限额	3
密钥版本号(DUK)	1
算法标识(DUK)	1
伪随机数(IC 卡)	4
MAC1	4
注：当 INITIALIZE FOR UPDATE 命令执行不成功，则只在响应报文中回送 SW1 和 SW2。	

8.3.13.5　响应报文的状态码

INITIALIZE FOR UPDATE 命令执行成功的状态码是‘9000’。

IC 卡可能回送的 INITIALIZE FOR UPDATE 错误状态见表 98。

表 98　INITIALIZE FOR UPDATE 错误状态

SW1	SW2	含　义
‘65’	‘81’	内存错误
‘67’	‘00’	长度错误
‘69’	‘85’	使用条件不满足
‘6A’	‘86’	P1、P2 参数不正确
‘94’	‘02’	交易计数器达到最大值
‘94’	‘03’	密钥索引不支持

8.3.14　RELOAD PIN 命令

8.3.14.1　定义和范围

RELOAD PIN 命令用于发卡方重新给持卡人产生一个新的 PIN(可以与原 PIN 相同)。RELOAD PIN 只能在拥有或能访问到重装 PIN 子密钥(DRPK)的发卡方终端上执行。命令中的 PIN 数据以明文传送。

在成功执行 RELOAD PIN 命令后，IC 卡应完成以下操作：

a)　PIN 错误尝试计数器复位；

b)　IC 卡的原 PIN 应设置为新的 PIN 值。

8.3.14.2　命令报文

RELOAD PIN 命令报文见表 99。

表 99 RELOAD PIN 命令报文

代码	值	代码	值
CLA	‘80’	Lc	‘06’～‘08’
INS	‘5E’	Data	见表 100
P1	‘00’	Le	不存在
P2	‘00’		

8.3.14.3 命令报文数据域

RELOAD PIN 命令报文的数据域见表 100。

表 100 RELOAD PIN 命令报文数据域

说　　明	长度(字节)
重装的 PIN 值	2～6
MAC	4

8.3.14.4 响应报文数据域

响应报文的数据域不存在。

8.3.14.5 响应报文的状态码

RELOAD PIN 命令执行成功的状态码是‘9000’。
IC 卡可能回送的 RELOAD PIN 错误状态见表 101。

表 101 RELOAD PIN 错误状态

SW1	SW2	含　　义
‘65’	‘81’	内存错误
‘67’	‘00’	长度错误
‘69’	‘85’	使用条件不满足
‘69’	‘84’	引用数据无效
‘69’	‘88’	安全信息数据对象不正确
‘6A’	‘86’	P1、P2 参数不正确
‘6A’	‘88’	引用数据找不到
‘93’	‘03’	应用永久锁住

8.3.15 INITIALIZE FOR CAPP PURCHASE 命令

8.3.15.1 定义和范围

INITIALIZE FOR CAPP PURCHASE 命令用于初始化复合应用消费交易。

8.3.15.2 命令报文

INITIALIZE FOR CAPP PURCHASE 命令报文见表 102。

表 102 INITIALIZE FOR CAPP PURCHASE 命令报文格式

代码	值	代码	值
CLA	'80'	Lc	'0B'
INS	'50'	Data	见表 103
P1	'03'	Le	'0F'
P2	'02'		

8.3.15.3 命令报文数据域

INITIALIZE FOR CAPP PURCHASE 命令报文的数据域定义见表 103。

表 103 INITIALIZE FOR CAPP PURCHASE 命令报文的数据域定义

说　明	长度(字节)
密钥索引号	1
交易金额	4
终端机编号	6

8.3.15.4 响应报文数据域

INITIALIZE FOR CAPP PURCHASE 命令执行成功的响应报文数据域见表 104。

表 104 INITIALIZE FOR CAPP PURCHASE 命令执行成功的响应报文数据域

说　明	长度(字节)
电子钱包余额	4
电子钱包交易序号	2
透支限额	3
密钥算法版本号(DPK)	1
密钥标识(DPK)	1
伪随机数(IC 卡)	4
注：当 INITIALIZE FOR CAPP PURCHASE 命令执行不成功，则只在响应报文中回送 SW1 和 SW2。	

8.3.15.5 响应报文的状态码

INITIALIZE FOR CAPP PURCHASE 命令执行成功的状态码是'9000'。

IC 卡可能回送的 INITIALIZE FOR CAPP PURCHASE 错误状态见表 105。

表 105　INITIALIZE FOR CAPP PURCHASE 命令可能回送的错误状态

SW1	SW2	说　明
'65'	'81'	内存错误
'69'	'85'	使用条件不满足
'94'	'01'	金额不足
'94'	'03'	密钥索引不支持
'94'	'02'	交易计数器达到最大值
'94'	'08'	应用灰锁锁定

8.3.16　UPDATE CAPP DATA CACHE 命令

8.3.16.1　定义和范围

UPDATE CAPP DATA CACHE 命令用于复合应用消费交易中更新复合应用数据缓存，缓存数据将被 DEBIT FOR CAPP PURCHASE 命令用于改写复合应用专用文件中相关记录。

8.3.16.2　命令报文

UPDATE CAPP DATA CACHE 命令报文见表 106。

表 106　UPDATE CAPP DATA CACHE 命令报文

代码	值	代码	值
CLA	'80'	Lc	后续数据域的长度
INS	'DC'	Data	见 8.3.16.3
P1	复合应用类型标识符	Le	不存在
P2	见表 107		

UPDATE CAPP DATA CACHE 命令报文中的引用控制参数 P2 定义见表 107。

表 107　UPDATE CAPP DATA CACHE 命令报文中的引用控制参数 P2 定义

B8	B7	B6	B5	B4	B3	B2	B1	含　义
0	0	0	0	0	—	—	—	RFU
X	X	X	X	X	—	—	—	SFI
1	1	1	1	1	—	—	—	RFU
—	—	—	—	—	0	0	0	第一个标识符出现的记录
—	—	—	—	—	X	X	X	RFU
其他值								RFU

8.3.16.3　命令报文数据域

UPDATE CAPP DATA CACHE 命令报文数据域由更新原有记录的新记录组成。

8.3.16.4 响应报文数据域

响应报文数据域不存在。

8.3.16.5 响应报文的状态码

UPDATE CAPP DATA CACHE 命令执行成功的状态码是‘9000’。

IC 卡可能回送的 UPDATE CAPP DATA CACHE 错误状态码见表 108。

表 108 UPDATE CAPP DATA CACHE 可能回送的错误状态码

SW1	SW2	含 义
‘65’	‘81’	内存失败(修改失败)
‘67’	‘00’	长度错误(Lc 域为空)
‘69’	‘81’	命令与文件结构不相容
‘69’	‘82’	不满足安全状态
‘69’	‘86’	不满足命令执行的条件(不是当前的 EF)
‘6A’	‘81’	不支持此功能
‘6A’	‘82’	未找到文件
‘6A’	‘83’	未找到记录
‘6A’	‘84’	文件中存储空间不够
‘94’	‘07’	复合应用禁止

8.3.17 DEBIT FOR CAPP PURCHASE 命令

8.3.17.1 定义和范围

DEBIT FOR CAPP PURCHASE 命令用于复合应用消费交易。

8.3.17.2 命令报文

DEBIT FOR CAPP PURCHASE 命令报文见表 109。

表 109 DEBIT FOR CAPP PURCHASE 命令报文

代码	值	代码	值
CLA	‘80’	Lc	‘0F’
INS	‘54’	Data	见表 110
P1	‘01’	Le	‘08’
P2	‘00’		

8.3.17.3 命令报文数据域

DEBIT FOR CAPP PURCHASE 命令报文的数据域定义见表 110。

表 110　DEBIT FOR CAPP PURCHASE 命令报文的数据域定义

说　　明	长度(字节)
终端交易序号	4
交易日期	4
交易时间	3
MAC1	4

8.3.17.4　响应报文数据域

DEBIT FOR CAPP PURCHASE 命令执行成功的响应报文数据域见表 111。

表 111　DEBIT FOR CAPP PURCHASE 命令执行成功的响应报文数据域

说　　明	长度(字节)
TAC	4
MAC2	4
注：当 DEBIT FOR CAPP PURCHASE 命令执行不成功，则只在响应报文中回送 SW1 和 SW2。	

8.3.17.5　响应报文的状态码

DEBIT FOR CAPP PURCHASE 命令执行成功的状态码是‘9000’。

IC 卡可能回送的 DEBIT FOR CAPP PURCHASE 错误状态见表 112。

表 112　DEBIT FOR CAPP PURCHASE 可能回送的错误状态

SW1	SW2	说　　明
‘65’	‘81’	内存错误
‘67’	‘00’	长度错误
‘69’	‘01’	命令不接受(无效状态)
‘69’	‘85’	使用条件不满足
‘93’	‘01’	金额不足
‘93’	‘02’	MAC 无效

8.3.18　DEBIT FOR UNLOCK 命令

8.3.18.1　定义和范围

DEBIT FOR UNLOCK 命令用于对电子钱包进行解扣操作。

8.3.18.2　命令报文

DEBIT FOR UNLOCK 命令报文见表 113。

表 113 DEBIT FOR UNLOCk 命令报文

代码	值	代码	值
CLA	‘E0’	Lc	‘1B’
INS	‘7E’	Data	见表 114
P1	‘08’	Le	‘04’
P2	‘01’		

8.3.18.3 命令报文数据域

DEBIT FOR UNLOCK 命令报文的数据域定义见表 114。

表 114 DEBIT FOR UNLOCK 命令报文的数据域定义

说 明	长度(字节)
交易金额	4
交易序号	2
终端机编号	6
终端交易序号	4
交易日期(终端)	4
交易时间(终端)	3
GMAC	4

8.3.18.4 响应报文数据域

DEBIT FOR UNLOCK 命令执行成功的响应报文数据域见表 115。

表 115 DEBIT FOR UNLOCK 命令执行成功的响应报文数据域

说 明	长度(字节)
TAC	4
注：当 DEBIT FOR UNLOCK 命令执行不成功，则只在响应报文中回送 SW1 和 SW2。	

8.3.18.5 响应报文的状态码

DEBIT FOR UNLOCK 命令执行成功的状态码是‘9000’。

IC 卡可能回送的 DEBIT FOR UNLOCK 错误状态见表 116。

表 116 DEBIT FOR UNLOCK 错误状态

SW1	SW2	说 明
‘64’	‘00’	状态标志位未改变
‘65’	‘81’	内存错误
‘67’	‘00’	长度错误

表 116(续)

SW1	SW2	说　明
‘69’	‘01’	命令不接受(无效状态)
‘69’	‘85’	使用条件不满足
‘93’	‘01’	金额不足
‘93’	‘02’	MAC 无效
‘94’	‘06’	所需 MAC 和 TAC 不可用

8.3.19　GET LOCK PROOF 命令

8.3.19.1　定义和范围

GET LOCK PROOF 命令用于读取电子钱包应用的灰锁状态以及相关的证明码。

8.3.19.2　命令报文

GET LOCK PROOF 命令报文见表 117。

表 117　GET LOCK PROOF 命令报文

代码	值	代码	值
CLA	‘E0’	P2	‘00’
INS	‘CA’	Lc	不存在
P1	‘00’:普通读取; ‘01’:清除 TACUF(交易验证码待读标志)	Data	不存在
		Le	‘1E’或不存在

8.3.19.3　命令报文数据域

GET LOCK PROOF 命令报文的数据域不存在。

8.3.19.4　响应报文数据域

如果 GET LOCK PROOF 命令执行成功,根据 P1 的参数、电子钱包应用的灰锁状态和 TACUF,形成不同的响应报文数据域,其关系见表 118。

表 118　参数、状态与 GET LOCK PROOF 响应报文数据域的关系

P1 参数	TACUF	灰锁状态	响应报文数据域	
			数据域列表	报文中的状态字
‘00’	标志复位	无灰锁	见表 119	‘00’
		有灰锁	见表 120	‘01’
	标志置位	不影响	见表 121	‘10’
‘01’	将 TACUF 标志复位	不影响	不存在	

如果 GET LOCK PROOF 命令执行不成功,则只在响应报文中回送 SW1 和 SW2。数据域列表内容见表 119、表 120 和表 121。

表 119　正常状态 GET LOCK PROOF 命令执行成功的响应报文数据域

说　明	长度(字节)
状态字(='00'表示当前应用无灰锁)	1
上次发生的解扣、联机解扣的交易类型标识	1
上次解扣、联机解扣的电子钱包('01')	1
上次解扣、联机解扣的电子钱包余额	4
上次解扣、联机解扣的电子钱包的交易序号	2
上次解扣、联机解扣的终端机编号	6
上次解扣、联机解扣的日期	4
上次解扣、联机解扣的时间	3
上次解扣、联机解扣的交易金额	4
上次解扣的 TAC 或联机解扣的 MAC3	4

表 120　灰锁状态 GET LOCK PROOF 命令执行成功的响应报文数据域

说　明	长度(字节)
状态字(='01'表示当前应用已灰锁)	1
灰锁的交易类型标识	1
被灰锁的电子钱包('01')	1
被灰锁的电子钱包余额	4
被灰锁的电子钱包交易序号	2
执行 GREY LOCK 时的终端机编号	6
执行 GREY LOCK 时的日期	4
执行 GREY LOCK 时的时间	3
灰锁时的 MAC2	4
灰锁时的 GTAC	4

表 121　TAC 未读时 GET LOCK PROOF 命令执行成功的响应报文数据域

说　明	长度(字节)
状态字(='10'表示当前应用 TAC 未读)	1
上次解扣的交易类型标识	1
上次解扣的电子钱包('01')	1
上次解扣的电子钱包余额	4
上次解扣的电子钱包交易序号	2
上次执行解扣的终端机编号	6
上次执行解扣的日期	4
上次执行解扣的时间	3
解扣交易金额	4
上次解扣的 TAC	4

8.3.19.5 响应报文的状态码

GET LOCK PROOF 命令执行成功的状态码是‘9000’。
IC 卡可能回送的 GET LOCK PROOF 错误状态见表 122。

表 122 GET LOCK PROOF 错误状态

SW1	SW2	说 明
‘64’	‘00’	状态标志位未改变
‘65’	‘81’	内存错误
‘69’	‘85’	使用条件不满足
‘6A’	‘86’	P1、P2 参数不正确

8.3.20 GREY LOCK 命令

8.3.20.1 定义和范围

GREY LOCK 命令用于灰锁电子钱包。

8.3.20.2 命令报文

GREY LOCK 命令报文见表 123。

表 123 GREY LOCK 命令报文

代码	值	代码	值
CLA	‘E0’	Lc	‘13’
INS	‘7C’	Data	见表 124
P1	‘08’	Le	‘08’
P2	‘00’		

8.3.20.3 命令报文数据域

GREY LOCK 命令报文的数据域定义见表 124。

表 124 GREY LOCK 命令报文的数据域定义

说 明	长度(字节)
终端交易序号	4
终端随机数	4
交易日期(终端)	4
交易时间(终端)	3
MAC1	4

8.3.20.4 响应报文数据域

GREY LOCK 命令执行成功的响应报文数据域见表 125。

表 125 GREY LOCK 命令执行成功的响应报文数据域

说　明	长度(字节)
GTAC	4
MAC2	4
注：当 GREY LOCK 命令执行不成功，则只在响应报文中回送 SW1 和 SW2。	

8.3.20.5 响应报文的状态码

GREY LOCK 命令执行成功的状态码是‘9000’。

IC 卡可能回送的 GREY LOCK 错误状态见表 126。

表 126 GREY LOCK 错误状态

SW1	SW2	说明
‘64’	‘00’	状态标志位未改变
‘65’	‘81’	内存错误
‘67’	‘00’	长度错误
‘69’	‘01’	命令不接受(无效状态)
‘69’	‘85’	使用条件不满足
‘93’	‘02’	MAC 无效

8.3.21 GREY UNLOCK 命令

8.3.21.1 定义和范围

GREY UNLOCK 命令用于联机解扣交易。

8.3.21.2 命令报文

GREY UNLOCK 命令报文见表 127。

表 127 GREY UNLOCK 命令报文

代码	值
CLA	‘E0’
INS	‘7E’
P1	‘09’
P2	‘00’
Lc	‘0F’
Data	见表 128
Le	‘04’

8.3.21.3 命令报文数据域

GREY UNLOCK 命令报文的数据域定义见表 128。

表 128 GREY UNLOCK 命令报文的数据域定义

说明	长度(字节)
交易金额	4
交易日期(主机)	4
交易时间(主机)	3
MAC2	4

8.3.21.4 响应报文数据域

GREY UNLOCK 命令执行成功的响应报文数据域见表 129。

表 129 GREY UNLOCK 命令执行成功的响应报文数据域

说明	长度(字节)
MAC3	4

8.3.21.5 响应报文的状态码

GREY UNLOCK 命令执行成功的状态码是‘9000’。

IC 卡可能回送的 GREY UNLOCK 错误状态见表 130。

表 130 GREY UNLOCK 错误状态

SW1	SW2	说明
‘64’	‘00’	状态标志位未改变
‘65’	‘81’	内存错误
‘67’	‘00’	长度错误
‘69’	‘01’	命令不接受(无效状态)
‘69’	‘85’	使用条件不满足
‘93’	‘02’	MAC 无效
‘94’	‘01’	金额不足

8.3.22 INITIALIZE FOR GREY LOCK 命令

8.3.22.1 定义和范围

INITIALIZE FOR GREY LOCK 命令用于初始化灰锁操作。

8.3.22.2 命令报文

INITIALIZE FOR GREY LOCK 命令报文见表 131。

表 131 INITIALIZE FOR GREY LOCK 命令报文

代码	值
CLA	‘E0’
INS	‘7A’
P1	‘08’
P2	‘01’
Lc	‘07’
Data	见表 132
Le	‘0F’

8.3.22.3 命令报文数据域

INITIALIZE FOR GREY LOCK 命令报文的数据域定义见表 132。

表 132 INITIALIZE FOR GREY LOCK 命令报文的数据域定义

说明	长度(字节)
密钥索引号	1
终端机编号	6

8.3.22.4 响应报文数据域

INITIALIZE FOR GREY LOCK 命令执行成功的响应报文数据域见表 133。

表 133 INITIALIZE FOR GREY LOCK 命令执行成功的响应报文数据域

说明	长度(字节)
电子钱包余额	4
电子钱包交易序号	2
透支限额	3
密钥版本号	1
算法标识	1
伪随机数	4
注：当 INITIALIZE FOR GREY LOCK 命令执行不成功，则只在响应报文中回送 SW1 和 SW2。	

8.3.22.5 响应报文的状态码

INITIALIZE FOR GREY LOCK 命令执行成功的状态码是‘9000’。

IC 卡可能回送的 INITIALIZE FOR GREY LOCK 错误状态见表 134。

表 134 INITIALIZE FOR GREY LOCK 错误状态

SW1	SW2	说明
‘65’	‘81’	内存错误
‘69’	‘01’	命令不接受(无效状态)
‘69’	‘85’	使用条件不满足
‘94’	‘03’	密钥索引号不支持

8.3.23 INITIALIZE FOR GREY UNLOCK 命令

8.3.23.1 定义和范围

INITIALIZE FOR GREY UNLOCK 命令用于初始化联机解扣交易。

8.3.23.2 命令报文

INITIALIZE FOR GREY UNLOCK 命令报文见表 135。

表 135 INITIALIZE FOR GREY UNLOCK 命令报文

代码	值
CLA	‘E0’
INS	‘7A’
P1	‘09’
P2	‘01’
Lc	‘07’
Data	见表 136
Le	‘12’

8.3.23.3 命令报文数据域

INITIALIZE FOR GREY UNLOCK 命令报文的数据域定义见表 136。

表 136 INITIALIZE FOR GREY UNLOCK 命令报文的数据域定义

说明	长度(字节)
密钥索引号	1
终端机编号	6

8.3.23.4 响应报文数据域

INITIALIZE FOR GREY UNLOCK 命令执行成功的响应报文数据域见表 137。

表 137 INITIALIZE FOR GREY UNLOCK 命令执行成功的响应报文数据域

说明	长度(字节)
电子钱包余额	4
电子钱包脱机交易序号	2
电子钱包联机交易序号	2
密钥版本号	1
密钥算法	1
伪随机数	4
MAC1	4
注：当 INITIALIZE FOR GREY UNLOCK 命令执行不成功，则只在响应报文中回送 SW1 和 SW2。	

8.3.23.5 响应报文的状态码

INITIALIZE FOR GREY UNLOCK 命令执行成功的状态码是‘9000’。

IC 卡可能回送的 INITIALIZE FOR GREY LOCK 错误状态见表 138。

表 138 INITIALIZE FOR GREY LOCK 错误状态

SW1	SW2	说明
‘65’	‘81’	内存错误
‘67’	‘00’	长度错误
‘69’	‘01’	命令不接受(无效状态)
‘69’	‘85’	使用条件不满足
‘6A’	‘86’	P1、P2 参数不正确
‘94’	‘03’	密钥索引号不支持

8.3.24 UPDATE OVERDRAW LIMIT 命令

8.3.24.1 定义和范围

UPDATE OVERDRAW LIMIT 命令用于修改透支限额交易。

8.3.24.2 命令报文

UPDATE OVERDRAW LIMIT 命令报文见表 139。

表 139 UPDATE OVERDRAW LIMIT 命令报文

代码	值
CLA	‘80’
INS	‘58’
P1	‘00’

表 139（续）

代码	值
P2	‘00’
Lc	‘0E’
Data	见表 140
Le	‘04’

8.3.24.3 命令报文数据域

UPDATE OVERDRAW LIMIT 命令报文的数据域见表 140。

表 140 命令报文的数据域

说 明	长度(字节)
新透支限额	3
交易日期(发卡方)	4
交易时间(发卡方)	3
MAC2	4

8.3.24.4 响应报文数据域

UPDATE OVERDRAW LIMIT 命令执行成功的响应报文数据域见表 141。

表 141 UPDATE OVERDRAW LIMIT 响应报文数据域

说 明	长度(字节)
TAC	4
注：当 UPDATE OVERDRAW LIMIT 命令执行不成功，则只在响应报文中回送 SW1 和 SW2。	

8.3.24.5 响应报文的状态码

UPDATE OVERDRAW LIMIT 命令执行成功的状态码是‘9000’。

IC 卡可能回送的 UPDATE OVERDRAW LIMIT 错误状态见表 142。

表 142 UPDATE OVERDRAW LIMIT 错误状态

SW1	SW2	含 义
‘65’	‘81’	内存错误
‘67’	‘00’	长度错误
‘69’	‘00’	不能处理
‘69’	‘01’	命令不接受(无效状态)
‘69’	‘85’	使用条件不满足

表 142（续）

SW1	SW2	含　义
‘6D’	‘00’	INS不支持或错误
‘6E’	‘00’	CLA不支持或错误
‘93’	‘02’	MAC无效

8.4　交易流程

8.4.1　一般说明

描述了电子存折/电子钱包应用的交易流程。该流程描述的是卡片与终端相互作用后，所进行的交易处理过程。消费或充值交易要求终端应具有安全存取模块(SAM)。本文件要求终端和SAM之间是以安全方式进行通信。

交易流程中描述的数据应遵照附录B的规定，电子存折/电子钱包应用密钥关系应遵照附录C的规定。

电子存折/电子钱包应用密钥说明参见附录D。

8.4.2　交易预处理

8.4.2.1　标准的交易预处理(步骤1.1)

对于非接触式IC卡，终端应具有检测IC卡是否进入射频有效场强范围内的能力。一旦终端检测到有效IC卡进入，终端应具备分辨多张有效IC卡进入的情况，并依次逐卡自动选择或人工选择一张特定IC卡。当卡片选定后，终端将继续专项应用选择功能。终端应根据支持的应用自动选择卡片上的应用。

8.4.2.2　发出GET LOCK PROOF(P1＝‘00’)命令(步骤1.2)

终端发出GET LOCK PROOF(P1＝‘00’)命令对电子钱包的状态进行查询。

8.4.2.3　判断TACUF(交易验证码待读标志)(步骤1.3)

IC卡收到GET LOCK PROOF(P1＝‘00’)命令后，首先根据TACUF判断上次的解扣交易的TAC码是否未被终端正确读取。如果TACUF为1，即上次的解扣交易的TAC码有待读取，则进入返回TAC码(步骤1.4)节所描述的步骤；否则，进入判断灰锁标志(步骤1.5)节所描述的步骤进行灰锁标志的判断。

如果应用未发生过灰锁、解扣、联机解扣交易，则IC卡返回‘6985’出错信息给终端，但不回送其他数据，同时结束交易预处理流程。

8.4.2.4　返回TAC码(步骤1.4)

IC卡将上次的解扣交易的产生的未成功读取的TAC码返回给终端，以供终端形成补充交易数据包，进入进行补充交易所描述的步骤。

8.4.2.5　判断灰锁标志(步骤1.5)

IC卡对所选择的电子钱包应用进行灰锁判断，如果当前应用中的电子钱包应用无灰锁，则进入返回正常信息所描述的步骤，返回正常信息给终端。否则进入返回灰锁信息所描述的步骤，返回灰锁信息

给终端。

8.4.2.6 回送正常信息(步骤 1.6)

IC 卡将正常信息回送给终端,交易预处理流程完成。

8.4.2.7 回送灰锁信息(步骤 1.7)

IC 卡将上次的灰锁交易的产生日期、时间、MAC2、GTAC 等回送给终端,终端进入启动补扣交易流程。

8.4.2.8 进行补扣交易(步骤 1.8)

补扣交易的详细描述参见 8.4.10,完成后交易预处理流程结束。

8.4.2.9 进行补充交易(步骤 1.9)

补充交易的详细描述参见 8.4.11,完成后交易预处理流程结束。

加入灰锁机制的交易预处理流程见图 3。

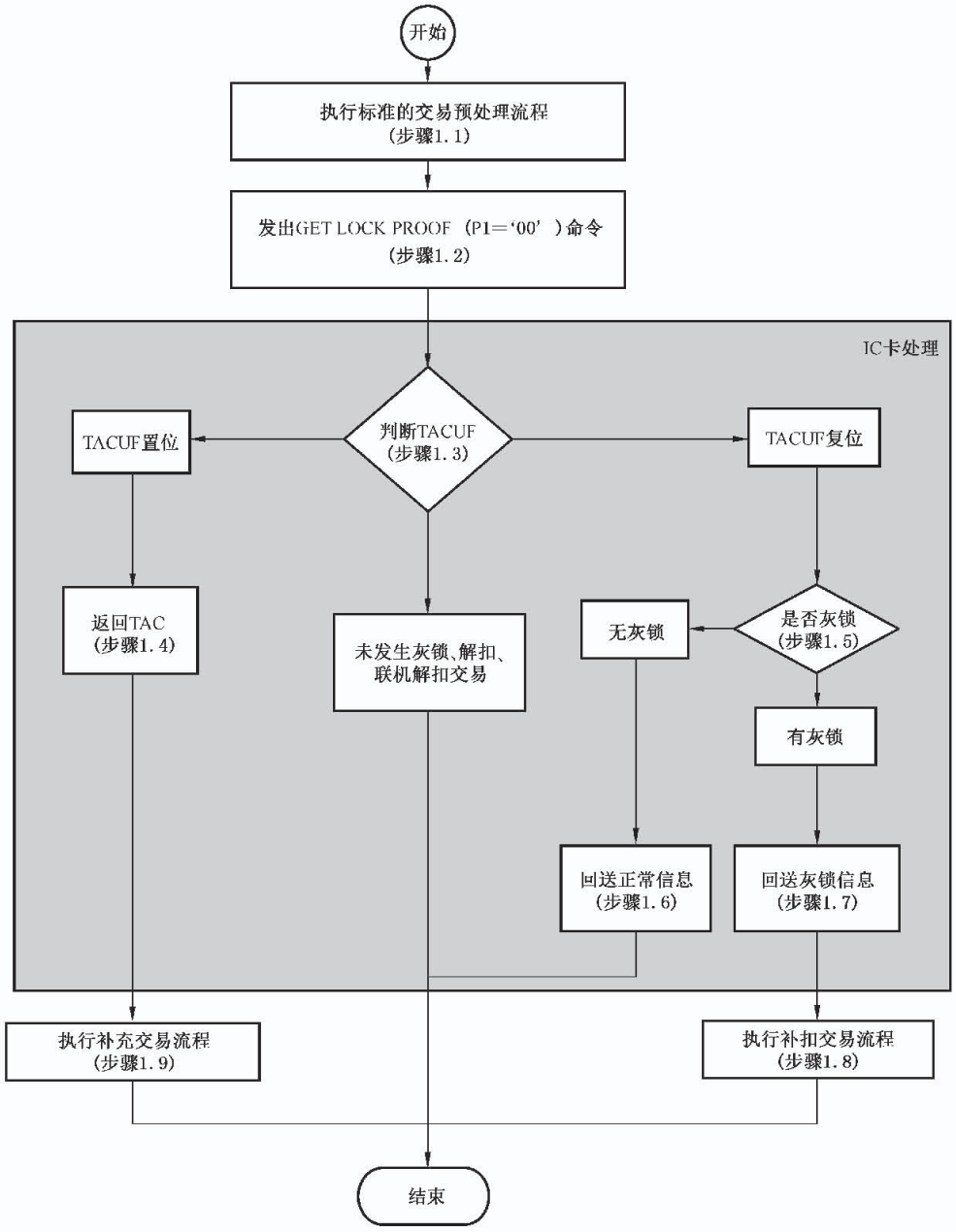

图 3　加入灰锁机制的交易预处理流程

8.4.3　圈存交易

8.4.3.1　一般说明

通过圈存交易,持卡人可将其相应账户上的资金划入电子存折或电子钱包中。这种交易应在终端上联机进行并要求提交个人密码(PIN)。

圈存交易流程见图 4。

补充定义如下。

a)　“交易处理”部分,交易明细定义为:

电子钱包交易序号;

交易金额；
交易类型标识；
终端机编号；
交易日期；
交易时间。

b) “处理 INITIALIZE FOR LOAD”部分，增加一检查过程：检查钱包是否被灰锁。如果灰锁，则回送状态码‘9408’(钱包灰锁锁定)，但不回送其他信息，同时终止命令的处理过程。

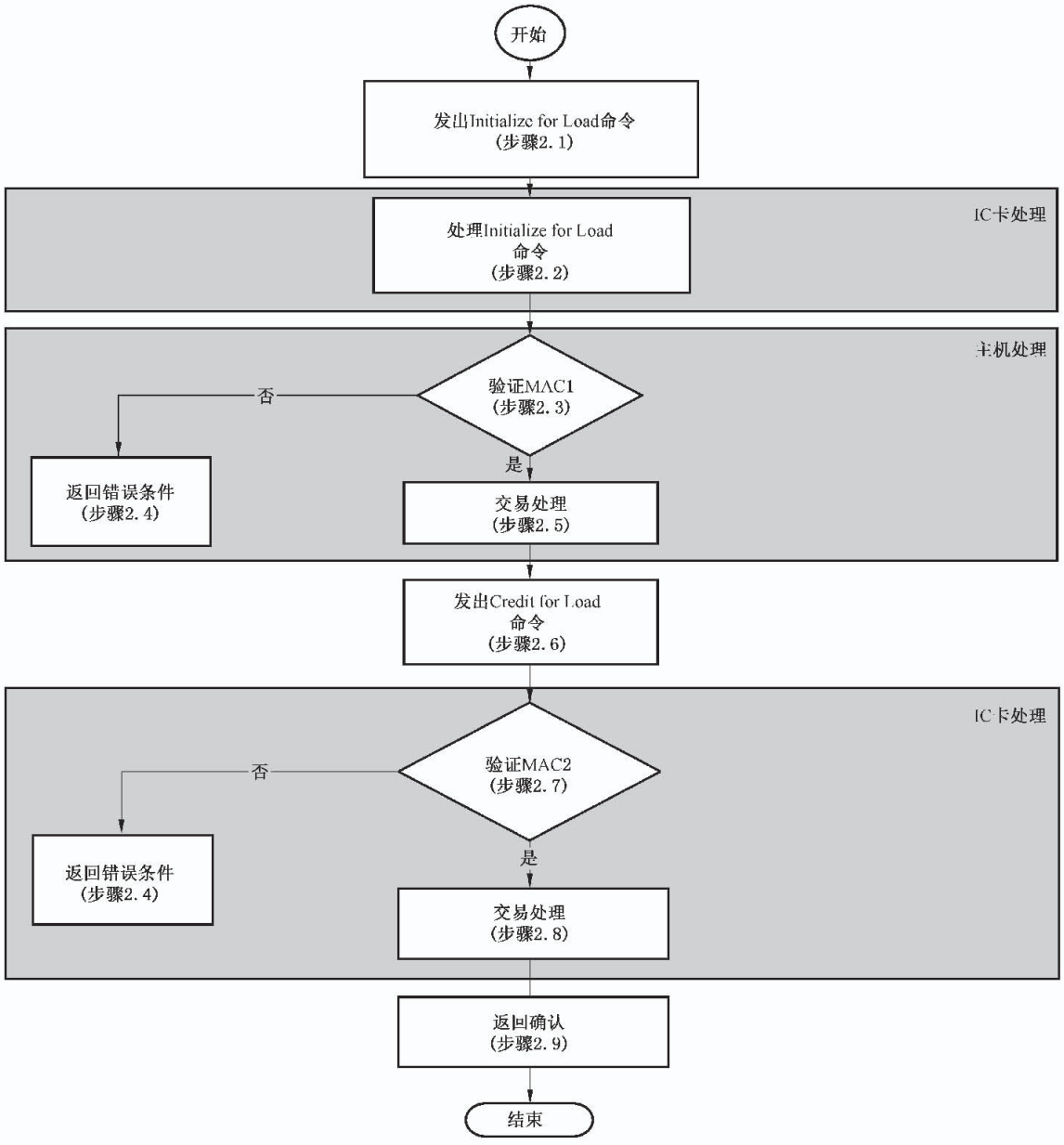

图 4 圈存交易处理流程

8.4.3.2 发出 INITIALIZE FOR LOAD 命令(步骤 2.1)

终端应发出 INITIALIZE FOR LOAD 命令启动圈存交易。

8.4.3.3 处理 INITIALIZE FOR LOAD 命令(步骤 2.2)

收到 INITIALIZE FOR LOAD 命令后,IC 卡将进行以下操作:

a) 检查是否支持命令中包含的密钥索引号。如果不支持,则回送状态码'9403'(不支持的密钥索引),但不回送任何其他数据,同时终止命令的处理过程。

b) 产生一个伪随机数,过程密钥 SESLK 和一个报文认证码(MAC1),用以供主机验证圈存交易及 IC 卡的合法性。SESLK 是用于电子存折或电子钱包圈存交易的过程密钥。用来产生过程密钥 SESLK 的输入数据如下:

1) SESLK:伪随机数(ICC)||电子存折联机交易序号或电子钱包联机交易序号||'8000'。

2) 用 SESLK 对以下数据加密产生 MAC1:

电子存折余额(交易前)或者电子钱包余额(交易前);

交易金额;

交易类型标识;

终端机编号。

c) IC 卡将 INITIALIZE FOR LOAD 响应报文回送给终端处理。如果 IC 卡回送的状态码不是'9000',则交易终止。

8.4.3.4 验证 MAC1(步骤 2.3)

收到 INITIALIZE FOR LOAD 命令响应报文后,终端把数据传给发卡方主机。主机将生成 SESLK 并确认 MAC1 是否有效。如果 MAC1 有效,交易处理将按交易处理(步骤 2.5)中描述的步骤继续执行。否则,交易处理将执行回送错误状态(步骤 2.4)中所描述的步骤。

8.4.3.5 回送错误状态(步骤 2.4)

如果不接受圈存交易,则主机应通知终端。

8.4.3.6 交易处理(步骤 2.5)

在确认能够进行圈存交易后,主机从持卡人在银行的相应账户中扣减圈存金额。

主机产生一个报文认证码(MAC2),用于 IC 卡对主机进行合法性检查。用 SESLK 对以下数据加密产生 MAC2(按所列顺序):

交易金额;

交易类型标识;

终端机编号;

交易日期(主机);

交易时间(主机)。

成功地进行了圈存交易后,主机将电子存折联机交易序号或电子钱包联机交易序号加 1,并向终端发送一个圈存交易接受报文,其中包括 MAC2、交易日期(主机)和交易时间(主机)。

8.4.3.7 发出 CREDIT FOR LOAD 命令(步骤 2.6)

终端收到主机发来的圈存交易接受报文后,发出 CREDIT FOR LOAD 命令更新卡上电子存折或电子钱包余额。

8.4.3.8 验证 MAC2(步骤 2.7)

收到 CREDIT FOR LOAD 命令后,IC 卡应确认 MAC2 的有效性。如果 MAC2 有效,交易处理将

执行交易处理(步骤 2.8)中描述的步骤。否则将向终端回送状态码‘9302’(MAC 无效)。

8.4.3.9 交易处理(步骤 2.8)

IC 卡将电子存折联机交易序号或电子钱包联机交易序号加 1,并且把交易金额加在电子存折或电子钱包的余额上。IC 卡应成功地完成以上所有操作或者一个也不完成。

在电子存折圈存交易或电子钱包圈存交易中,IC 卡用以下数据组成的一个记录更新交易明细:

电子存折联机交易序号或电子钱包联机交易序号;

交易金额;

交易类型标识;

终端机编号;

交易日期(主机);

交易时间(主机)。

TAC 的计算不采用过程密钥方式,它用 DTK 左右 8 位字节异或运算的结果对以下数据进行加密运算来产生(按所列顺序):

电子存折余额(交易后)或电子钱包余额(交易后);

电子存折联机交易序号(加 1 前)或电子钱包联机交易序号(加 1 前);

交易金额;

交易类型标识;

终端机编号;

交易日期(主机);

交易时间(主机)。

8.4.3.10 回送确认(步骤 2.9)

在成功完成步骤 2.8 后,IC 卡通过 CREDIT FOR LOAD 命令的响应报文将 TAC 回送给终端。主机可以不马上验证 TAC。

8.4.4 圈提交易

8.4.4.1 一般说明

通过圈提交易,持卡人可以把电子存折中的部分或全部资金划回到其在银行的相应账户上。这种交易应在金融终端上联机进行并要求提交个人密码(PIN)。只有电子存折应用支持圈提交易。

圈提交易处理流程见图 5。

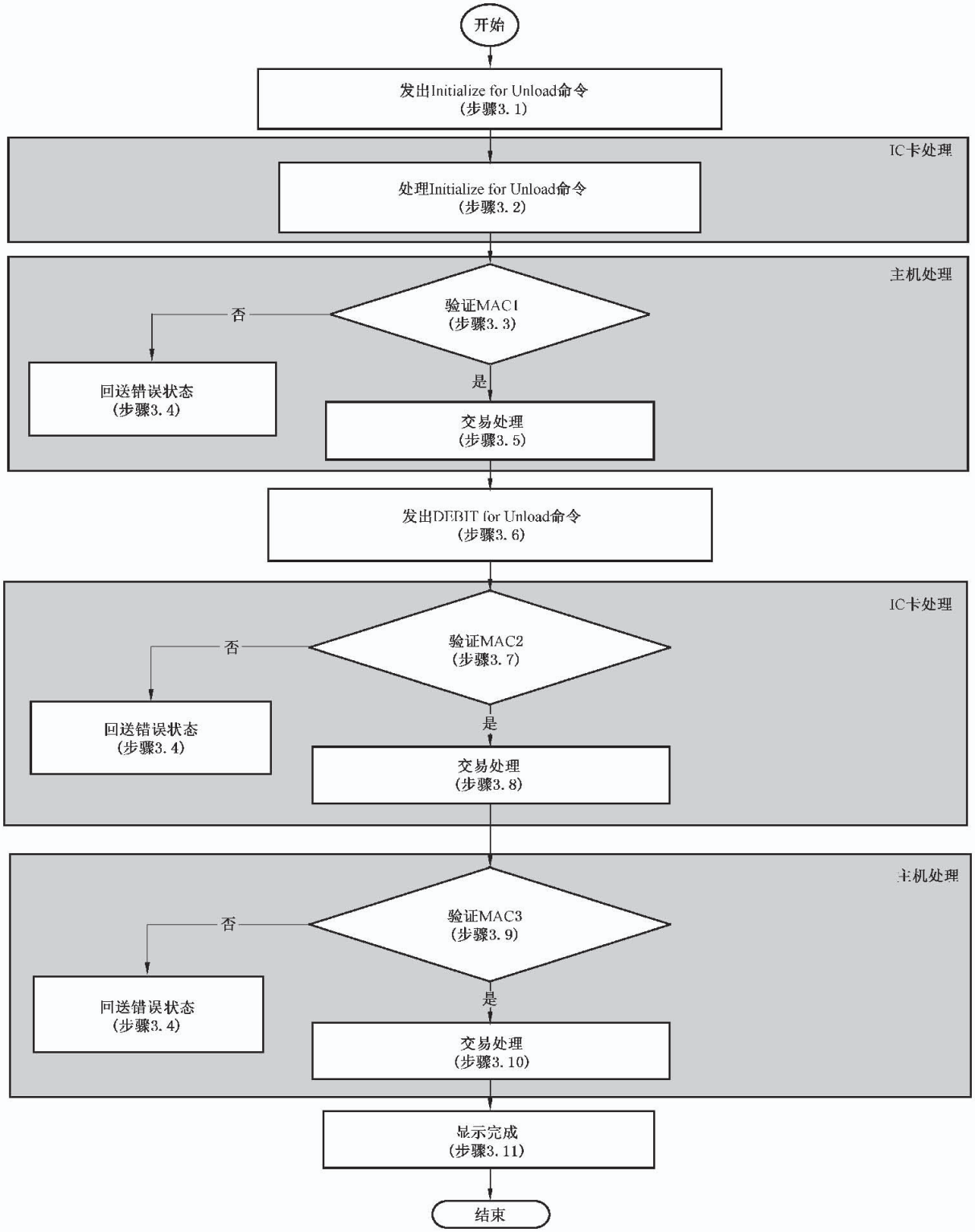
开始
发出Initialize for Unload命令
(步骤3.1)
IC卡处理
处理Initialize for Unload命令
(步骤3.2)
主机处理
验证MAC1
(步骤3.3)
否
回送错误状态
(步骤3.4)
是
交易处理
(步骤3.5)
发出DEBIT for Unload命令
(步骤3.6)
IC卡处理
验证MAC2
(步骤3.7)
否
回送错误状态
(步骤3.4)
是
交易处理
(步骤3.8)
主机处理
验证MAC3
(步骤3.9)
否
回送错误状态
(步骤3.4)
是
交易处理
(步骤3.10)
显示完成
(步骤3.11)
结束

图 5　圈提交易处理流程

8.4.4.2 发出 INITIALIZE FOR UNLOAD 命令(步骤 3.1)

终端发出 INITIALIZE FOR UNLOAD 命令启动圈提交易。

8.4.4.3 处理 INITIALIZE FOR UNLOAD 命令(步骤 3.2)

收到 INITIALIZE FOR UNLOAD 命令后,IC 卡将进行以下操作:

a) 检查是否支持命令中提供的密钥索引号。如果不支持,则回送状态码'9403'(不支持的密钥索引),但不回送任何其他数据,命令处理结束。

b) 检查命令中包含的交易金额是否超过电子存折余额。如果超过,则回送状态码'9401'(资金不足),但不回送其他数据。

c) 在通过以上检查后,IC 卡将产生一个伪随机数(ICC)、过程密钥 SESULK 和一个报文认证码(MAC1),供主机验证圈提交易及 IC 卡的合法性:

1) SESULK 是用于电子存折或电子钱包圈存交易的过程密钥。用来产生该过程密钥的输入数据如下:

SESULK:伪随机数(ICC)||电子存折联机交易序号||'8000'

2) 用 SESULK 对以下数据加密产生 MAC1(按所列顺序):

电子存折余额(交易前);

交易金额;

交易类型标识;

终端机编号。

d) IC 卡应向终端回送 INITIALIZE FOR UNLOAD 命令的响应报文和状态码'9000'。

e) 在收到 INITIALIZE FOR UNLOAD 的响应报文后,终端将圈提许可请求报文 MAC1 送往发卡方主机。

8.4.4.4 验证 MAC1(步骤 3.3)

主机将产生 SESULK 并验证 MAC1 是否有效。如果 MAC1 有效,将执行主机处理(步骤 3.5)中的步骤。否则终端应回送一个错误状态码,交易处理将转而执行回送错误状态(步骤 3.4)中所描述的步骤。

8.4.4.5 回送错误状态(步骤 3.4)

如果不接受圈提交易,主机应通知终端。

8.4.4.6 交易处理(步骤 3.5)

主机确认能够进行圈提交易后,将产生一个报文认证码(MAC2),以供 IC 卡对主机合法性进行检查。下面列出包含在 DEBIT FOR UNLOAD 命令中从主机经由终端传到 IC 卡的数据。主机向终端发送一个圈提交易接受报文,其中至少应包括交易日期(主机)、交易时间(主机)和 MAC2。

用 SESULK 对以下数据进行加密(按所列顺序)产生 MAC2:

交易金额;

交易类型标识;

终端机编号;

交易日期(主机);

交易时间(主机)。

8.4.4.7 发出 DEBIT FOR UNLOAD 命令(步骤 3.6)

终端收到主机的圈提交易接受报文后,向 IC 卡发出 DEBIT FOR UNLOAD 命令以更新卡上电子存折余额。

8.4.4.8 验证 MAC2(步骤 3.7)

IC 卡应确认 MAC2 是有效的。如果 MAC2 有效,交易处理将执行交易处理(步骤 3.8)中所描述的步骤。否则向终端返回状态码'9302'(MAC 无效)。

8.4.4.9 交易处理(步骤 3.8)

IC 卡将电子存折联机交易序号加 1,并从卡上的电子存折余额中扣减交易金额。IC 卡应成功地完成以上所有步骤或者一个也不完成。

IC 卡将产生一个报文认证码(MAC3)。并通过 DEBIT FOR UNLOAD 命令的响应报文将以下数据经终端送往主机。

用 SESULK 对以下数据加密产生 MAC3:

电子存折余额(交易后);
电子存折联机交易序号(加 1 前);
交易金额;
交易类型标识;
终端机编号;
交易日期(主机);
交易时间(主机)。

IC 卡用以下数据组成的一个记录更新交易明细:

电子存折联机交易序号;
交易金额;
交易类型标识;
终端机编号;
交易日期(主机);
交易时间(主机)。

8.4.4.10 验证 MAC3(步骤 3.9)

主机收到(经由终端)IC 卡返回的 MAC3 后,应确认 MAC3 是否有效。如果 MAC3 有效,交易处理将执行交易处理(步骤 3.10)中描述的步骤。否则将向终端返回一个错误状态码。

8.4.4.11 交易处理(步骤 3.10)

发卡方主机将交易金额加在持卡人的相应银行账户上,并将主机的电子存折联机交易序号加 1。

主机将向终端返回一个完成报文,表示持卡人的账户已更新。

8.4.4.12 显示完成(步骤 3.11)

在收到主机的完成报文后,终端将向持卡人显示交易完成信息。

如果需要,终端应能向持卡人提供交易纸凭证。

8.4.5 消费交易

8.4.5.1 一般说明

消费交易允许持卡人使用电子存折或电子钱包的余额进行购物或获取服务。此交易可以在销售点终端(POS)上脱机进行。使用电子存折进行的消费交易应提交个人密码(PIN),使用电子钱包则不需要。

消费交易处理流程见图 6。

补充定义如下:

a) “交易处理”:IC 卡从电子钱包余额中扣减消费的金额,电子钱包交易序号加 1,更新电子钱包消费交易记录。IC 卡应成功地完成以上所有步骤或者一个也不完成。

b) 对于电子钱包消费交易,IC 卡将用以下数据组成的一个记录更新交易明细:

交易金额;

交易类型标识‘06’;

卡片交易序号;

终端机编号;

交易日期(终端);

交易时间(终端)。

c) “处理 INITIALIZE FOR PURCHASE”部分,增加一检查过程:检查钱包是否被灰锁。如果灰锁,则返回状态码‘9408’(钱包灰锁锁定),但不返回其他信息,同时终止命令的处理过程。

d) “发出 GET MESSAGE 命令”部分,增加一认证过程:检查 PSAM 卡消费密钥权限状态。

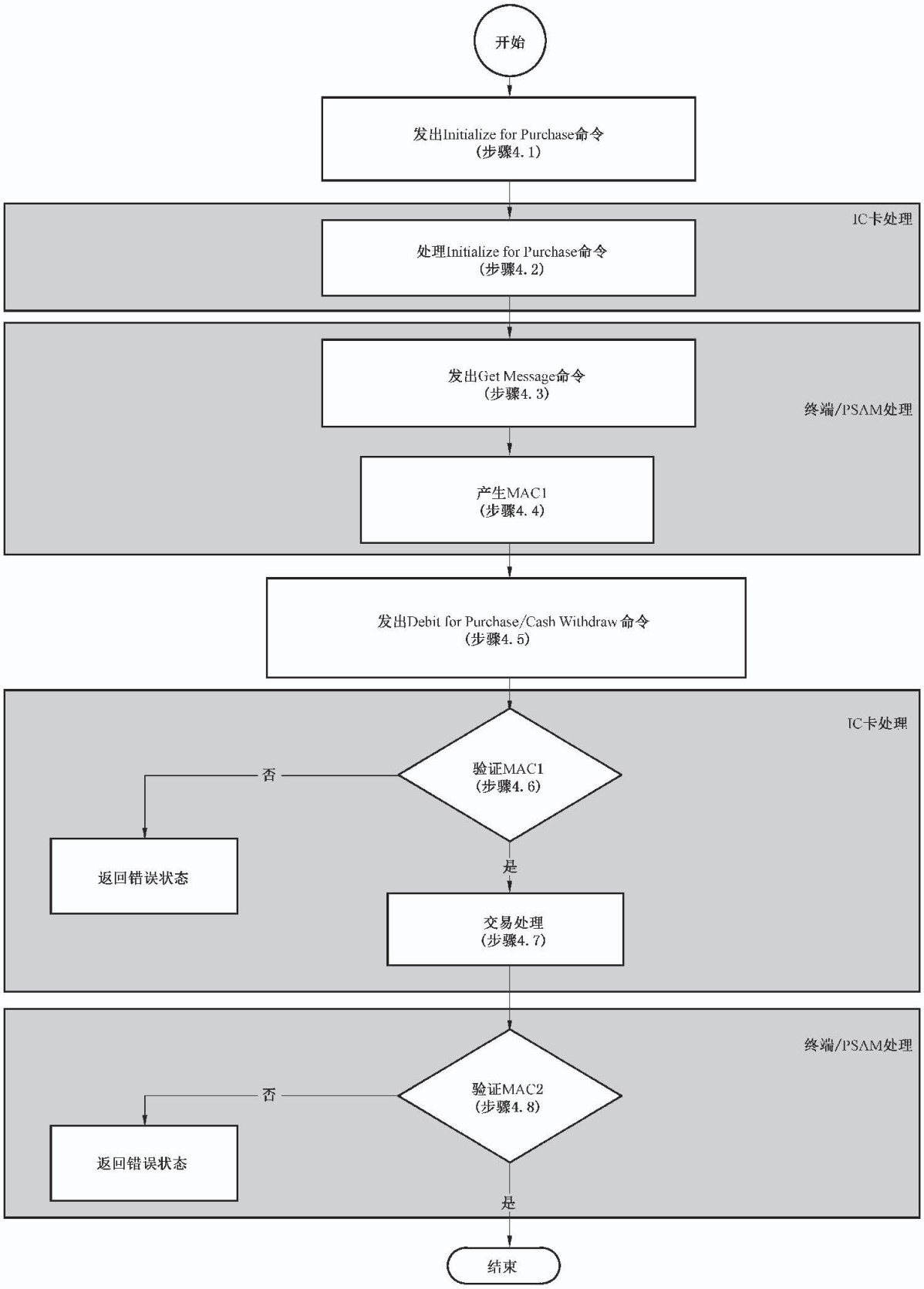
开始
发出Initialize for Purchase命令
(步骤4.1)
IC卡处理
处理Initialize for Purchase命令
(步骤4.2)
发出Get Message命令
(步骤4.3)
终端/PSAM处理
产生MAC1
(步骤4.4)
发出Debit for Purchase/Cash Withdraw命令
(步骤4.5)
IC卡处理
验证MAC1
(步骤4.6)
否
返回错误状态
是
交易处理
(步骤4.7)
终端/PSAM处理
验证MAC2
(步骤4.8)
否
返回错误状态
是
结束

图 6　消费交易处理流程

8.4.5.2 发出 INITIALIZE FOR PURCHASE 命令(步骤 4.1)

终端发出 INITIALIZE FOR PURCHASE 命令启动消费交易。

8.4.5.3 处理 INITIALIZE FOR PURCHASE 命令(步骤 4.2)

IC 卡收到 INITIALIZE FOR PURCHASE 命令后,将进行以下操作:

a) 检查是否支持命令中提供的密钥索引号。如果不支持,则返回状态码'9403'(不支持的密钥索引),但不返回其他数据。

b) 检查电子存折余额或电子钱包余额是否大于或等于交易金额。如果小于交易金额,则返回状态码'9401'(资金不足),但不返回其他数据。

c) 在通过以上检查之后,IC 卡将在发出 DEBIT FOR PURCHASE/CASH WITHDRAW 命令(步骤 4.5)中产生一个伪随机数(ICC)和过程密钥用于验证 MAC1。用于产生该过程密钥的输入数据如下:

SESPK:伪随机数(ICC)||电子存折脱机交易序号或电子钱包脱机交易序号||终端交易序号的最右两个字节。

8.4.5.4 发出 GET MESSAGE 命令(步骤 4.3)

检查 PSAM 卡消费密钥权限状态,得到安全认证识别码。

8.4.5.5 产生 MAC1(步骤 4.4)

使用伪随机数(ICC)和 IC 卡返回的电子存折脱机交易序号或电子钱包脱机交易序号,终端的安全存取模块(PSAM)将产生一个过程密钥(SESPK)和一个报文认证码(MAC1),供 IC 卡来验证 PSAM 的合法性。

用 SESPK 对以下数据进行加密产生 MAC1(按所列顺序):

交易金额;

交易类型标识;

终端机编号;

交易日期(终端);

交易时间(终端);

安全认证识别码。

注:安全认证识别码已在 8.3.4.1 中定义。

8.4.5.6 发出 DEBIT FOR PURCHASE/CASH WITHDRAW 命令(步骤 4.5)

终端发出 DEBIT FOR PURCHASE/CASH WITHDRAW 命令。

8.4.5.7 验证 MAC1(步骤 4.6)

在收到 DEBIT FOR PURCHASE/CASH WITHDRAW 命令后,IC 卡将验证 MAC1 的有效性。如果 MAC1 有效,交易处理将继续执行交易处理(步骤 4.7)中所描述的步骤。否则将向终端返回错误状态码'9302'(MAC 无效)。

8.4.5.8 交易处理(步骤 4.7)

IC 卡从电子存折余额或电子钱包余额中扣减消费的金额,并将电子存折或电子钱包脱机交易序号加 1。IC 卡应成功地完成以上所有步骤或者一个也不完成。只有余额和序号的更新均成功后,交易明

细才可更新。

IC卡产生一个报文认证码(MAC2)供PSAM对其进行合法性检查,并通过DEBIT FOR PURCHASE/CASH WITHDRAW命令响应报文返回以下数据,作为PASM产生MAC2的输入数据。用SESPK对以下数据进行加密产生MAC2:

交易金额。

用密钥DTK左右8位字节异或运算后的结果产生TAC。TAC将被写入终端交易明细,以便于主机进行交易验证。下面是用来生成TAC的数据,它们以明文形式通过CREDTE FOR PURCHASE/CASH WITHDRAW命令的响应报文从IC卡传送到终端:

交易金额;

交易类型标识;

终端机编号;

终端交易序号;

交易日期(终端);

交易时间(终端)。

对于电子存折消费交易和电子钱包消费交易(可选),IC卡将用以下数据组成的一个记录更新交易明细:

电子存折脱机交易序号或电子钱包脱机交易序号;

交易金额;

交易类型标识;

终端机编号;

交易日期(终端);

交易时间(终端)。

8.4.5.9 验证MAC2(步骤4.8)

在收到IC卡(经过终端)传来的MAC2后,PSAM要验证MAC2的有效性。MAC2验证的结果被传送到终端以便采取必要的措施。

8.4.6 复合应用消费交易

8.4.6.1 一般说明

复合应用消费交易允许持卡人使用电子钱包的余额进行购物或获取服务。此交易可以在终端设备或其他读卡设备上脱机进行。此交易无需提交个人密码(PIN)。

复合应用消费交易允许消费金额为0。

复合应用消费交易见图7。

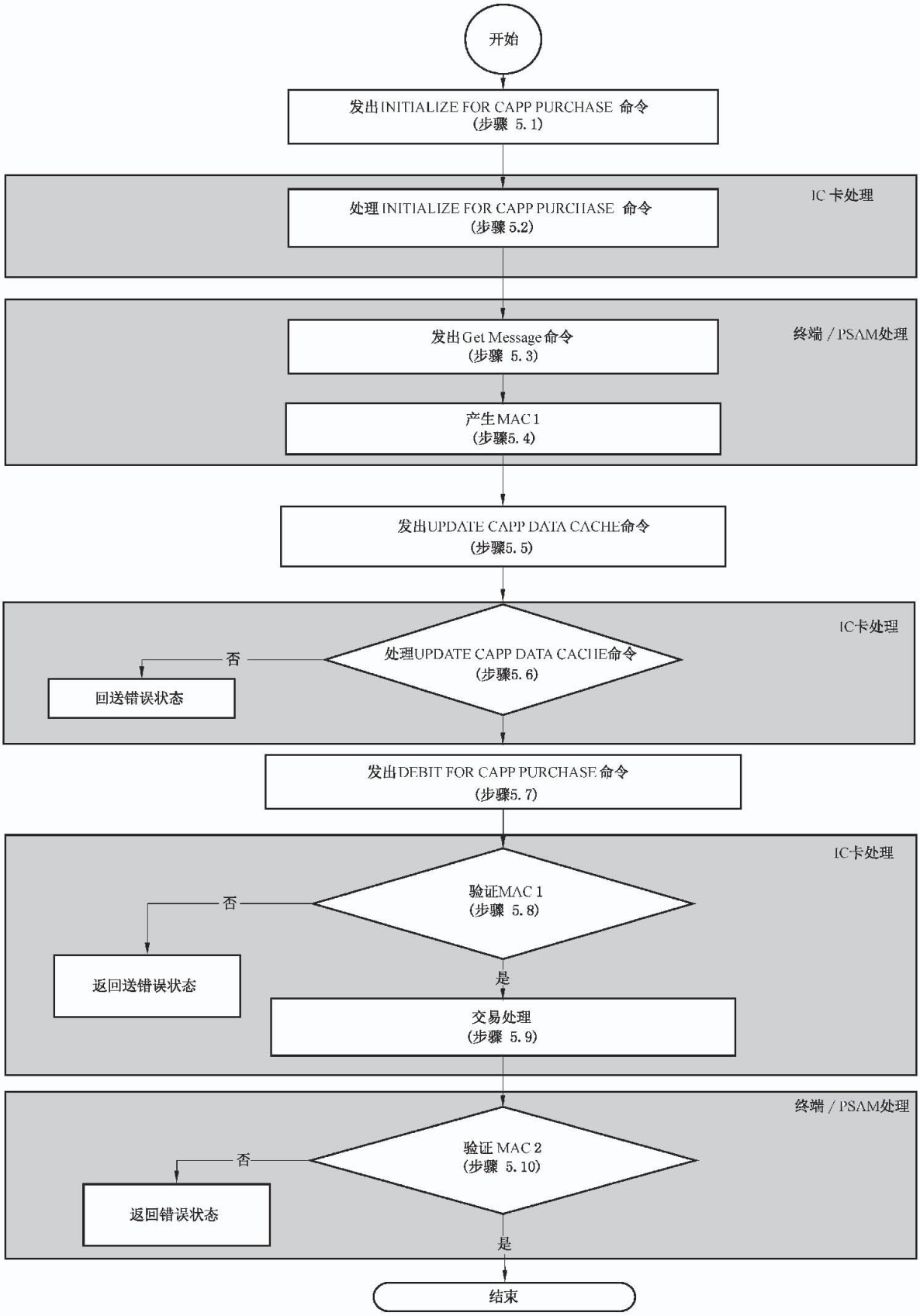
开始
发出INITIALIZE FOR CAPP PURCHASE 命令
(步骤 5.1)
IC卡处理
处理INITIALIZE FOR CAPP PURCHASE 命令
(步骤5.2)
终端/PSAM处理
发出Get Message命令
(步骤 5.3)
产生MAC 1
(步骤5.4)
发出UPDATE CAPP DATA CACHE命令
(步骤5.5)
IC卡处理
处理UPDATE CAPP DATA CACHE命令
(步骤5.6)
否
回送错误状态
发出DEBIT FOR CAPP PURCHASE命令
(步骤5.7)
IC卡处理
验证MAC 1
(步骤 5.8)
否
返回送错误状态
是
交易处理
(步骤 5.9)
终端/PSAM处理
验证 MAC 2
(步骤 5.10)
否
返回错误状态
是
结束

图 7　复合应用消费交易

8.4.6.2 发出 INITIALIZE FOR CAPP PURCHASE 命令(步骤 5.1)

终端发出 INITIALIZE FOR CAPP PURCHASE 命令启动复合应用消费交易。

8.4.6.3 处理 INITIALIZE FOR CAPP PURCHASE 命令(步骤 5.2)

IC 卡收到 INITIALIZE FOR CAPP PURCHASE 命令后,将进行以下操作:

a) 检查是否支持命令中提供的密钥索引号。如果不支持,则返回状态码'9403'(不支持的密钥索引),但不返回其他数据。

b) 检查钱包是否被灰锁,如果灰锁,则返回状态码'9408'(钱包灰锁锁定),但不返回其他信息,同时终止命令的处理过程。

c) 检查电子钱包余额是否大于或等于交易金额。如果小于交易金额,则返回状态码'9401',但不返回其他数据。

d) 在通过以上检查之后,IC 卡将产生一个伪随机数(ICC)和过程密钥。用于产生该过程密钥的输入数据如下:

SESPK:伪随机数(ICC)||电子钱包交易序号||终端交易序号的最右两个字节。

8.4.6.4 发出 GET MESSAGE 命令(步骤 5.3)

检查 PSAM 卡消费密钥权限状态。

8.4.6.5 产生 MAC1(步骤 5.4)

使用伪随机数(ICC)和 IC 卡返回的电子存折脱机交易序号或电子钱包脱机交易序号,终端的安全存取模块(PSAM)将产生一个过程密钥(SESPK)和一个报文认证码(MAC1),供 IC 卡来验证 PSAM 的合法性。

用 SESPK 对以下数据进行加密产生 MAC1(按所列顺序):

交易金额;

交易类型标识;

终端机编号;

交易日期(终端);

交易时间(终端);

安全认证识别码。

注:安全认证识别码已在 8.3.4.1 中定义。

8.4.6.6 发出 UPDATE CAPP DATA CACHE 命令(步骤 5.5)

终端发出 UPDATE CAPP DATA CACHE 命令。

8.4.6.7 处理 UPDATE CAPP DATA CACHE 命令(步骤 5.6)

IC 卡在收到 UPDATE CAPP DATA CACHE 命令后,将进行以下操作:

如果命令中存在 SFI 域,检查卡片当前应用下是否存在与命令中 SFI 值相同的文件。如果不存在,返回状态码'6A82'(未找到文件),但不返回其他数据。终端应终止此次复合应用消费交易。

根据命令中的复合应用类型标识符,查询复合应用专用文件中是否存在相同标识符的记录。如果不存在,则返回状态码'6A83'(未找到记录),但不返回其他数据。终端应终止此次复合应用消费交易。

检查复合应用专用文件中相应记录中的应用锁定标志字节。如果应用锁定标志为设置,则返回状态码'9407'(复合应用禁止),但不返回其他数据。终端应终止此次复合应用消费交易。

检查命令中的数据域长度是否大于复合应用专用文件中相应记录的长度。如果大于，则返回状态码‘6A84’(文件中存储空间不够)，但不返回其他数据。终端应终止此次复合应用消费交易。

在通过以上检查后，IC卡应暂存命令中的SFI、记录号、复合应用类型标识符和数据域。复合应用专用文件中相应记录中的数据不得通过此命令更新。

8.4.6.8 发出 DEBIT FOR CAPP PURCHASE 命令(步骤 5.7)

终端发出 DEBIT FOR CAPP PURCHASE 命令。

8.4.6.9 验证 MAC1(步骤 5.8)

在收到 DEBIT FOR CAPP PURCHASE 命令后，IC卡将验证 MAC1 的有效性。如果 MAC1 有效，交易处理将继续执行，否则将向终端返回错误状态码‘9302’(MAC 无效)。

8.4.6.10 交易处理(步骤 5.9)

IC卡从电子钱包余额中扣减消费的金额，电子钱包交易序号加 1，根据处理 UPDATE CAPP DATA CACHE 命令(步骤 5.6)中暂存的数据更新复合应用专用文件，更新电子钱包消费交易记录。IC卡应成功地完成以上所有步骤或者一个也不完成。

在根据中处理 UPDATE CAPP DATA CACHE 命令(步骤 5.6)暂存的数据更新复合应用专用文件时，如果更新数据长度小于记录长度，IC卡应在数据后自动填充‘00’至记录尾。

IC卡产生一个报文认证码 (MAC2)供 PSAM 对其进行合法性检查，并通过 DEBIT FOR CAPP PURCHASE 命令响应报文返回以下数据，作为 PSAM 产生 MAC2 的输入数据。用 SESPK 对以下数据进行加密产生 MAC2：

交易金额。

用密钥 DTK 左右 8 位字节异或运算后的结果产生 TAC。TAC 将被写入终端交易明细，以便于主机进行交易验证。下面是用来生成 TAC 的数据，它们以明文形式通过 DEBIT FOR CAPP PURCHASE 命令的响应报文从 IC卡传送到终端：

交易金额；

交易类型标识；

终端机编号；

终端交易序号；

交易日期(终端)；

交易时间(终端)。

对于电子钱包消费交易，IC卡将用以下数据组成的一个记录更新交易明细。

交易类型标识‘09’；

交易金额；

卡片交易序号；

终端机编号；

交易日期(终端)；

交易时间(终端)。

8.4.6.11 验证 MAC2 (步骤 5.10)

在收到 IC卡(经过终端)传来的 MAC2 后，PSAM 要验证 MAC2 的有效性。MAC2 验证的结果被传送到终端以便采取必要的措施。

8.4.7 修改透支限额交易

8.4.7.1 一般说明

“透支功能”从技术上支持的一种基于电子存折应用的有限信用功能。当电子存折中的实际金额不足时，它为持卡人提供了一种在发卡方所允许的透支额度内继续进行交易的方便性。修改透支限额交易应在金融终端上联机进行，且应提交个人密码(PIN)。

是否使用“透支功能”以及允许透支的额度由发卡方决定。

如果透支限额存在，电子存折的余额是实际圈存余额与透支限额之和。

修改透支限额交易流程见图 8。

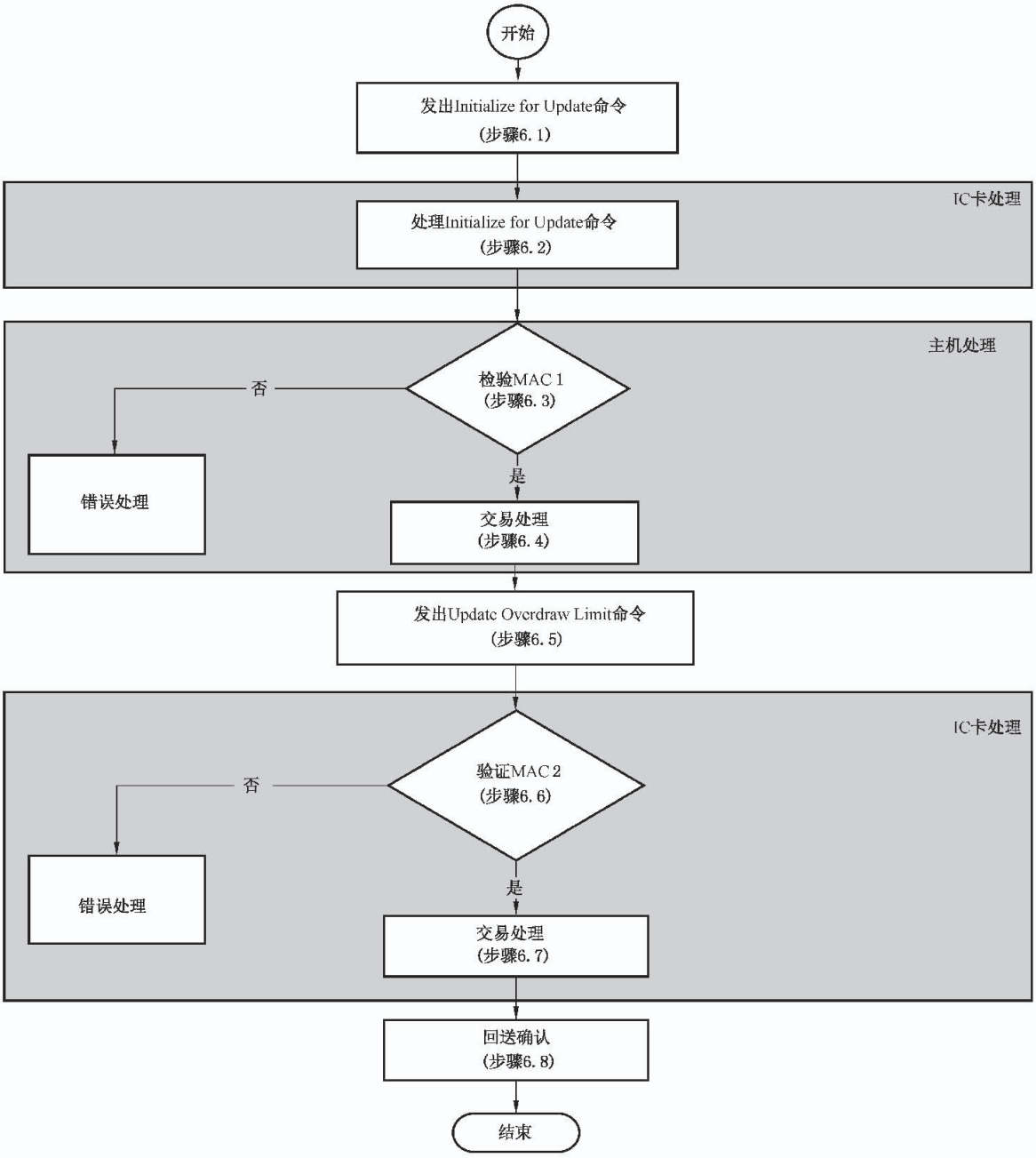

图 8 修改透支限额交易

8.4.7.2 发出 INITIALIZE FOR UPDATE 命令(步骤 6.1)

终端发出 INITIALIZE FOR UPDATE 命令启动修改透支限额交易。

8.4.7.3 处理 INITIALIZE FOR UPDATE 命令(步骤 6.2)

收到 INITIALIZE FOR UPDATE 命令后,IC 卡将进行以下操作:

a) 检查是否支持命令中提供的密钥索引号。如果不支持,则返回状态码'9403'(不支持的密钥索引)但不返回其他数据;

b) 在通过了以上检查之后,IC 卡将产生一个伪随机数(ICC)、一个过程密钥 SESUK 和一个报文认证码(MAC1)。用于产生过程密钥的输入数据如下:

SESUK:伪随机数(ICC)||电子存折联机交易序号||'8000'。

用 SESUK 对以下数据加密产生 MAC1(按所列顺序):

电子存折余额(交易前);

透支限额(交易前);

交易类型标识;

终端机编号。

8.4.7.4 验证 MAC1(步骤 6.3)

在收到 INITIALIZE FOR UPDATE 命令后,IC 卡将验证 MAC1 的有效性。如果 MAC1 有效,交易处理将继续执行,否则将向终端返回错误处理。

8.4.7.5 交易处理(步骤 6.4)

假定主机已经知道 IC 卡的透支限额。

基于 MAC1(或者其他由主机决定的验证标准)验证的结果,主机将决定是否允许修改透支限额。

如果主机拒绝交易,则应向终端发送一个拒绝报文,结束交易处理。

如果主机允许交易,则应生成一个报文认证码(MAC2),以供 IC 卡对主机合法性进行检查。

用 SESUK 对以下数据加密产生 MAC2(按所列顺序):

透支限额(交易后);

交易类型标识;

终端机编号;

交易日期(主机);

交易时间(主机)。

主机将电子存折联机交易序号加 1。

主机应向终端发送一个至少包括新透支限额、交易日期(主机)、交易时间(主机)和 MAC2 的许可信息。

8.4.7.6 发出 UPDATE OVERDRAWLIMIT 命令(步骤 6.5)

如果主机同意交易,终端将发出 UPDATE OVERDRAWLIMIT 命令。

8.4.7.7 验证 MAC2(步骤 6.6)

IC 卡将验证 MAC2 的有效性。如果 MAC2 有效,交易处理将执行交易处理(步骤 6.8)中的步骤。否则终端返回错误状态码'9302'(MAC 无效)。

8.4.7.8 交易处理(步骤 6.7)

直接用密钥 DTK 左右 8 字节异或后的结果对以下数据加密产生一个 TAC:

电子存折余额(交易后);

电子存折联机交易序号(加 1 前);

电子存折透支限额(交易后);

交易类型标识;

终端机编号;

交易日期(主机);

交易时间(主机)。

将当前电子存折余额置为新的电子存折余额,更新透支限额并使电子存折联机交易序号加 1。这三个修改应全部完成,或一个也不完成。

IC 卡通过响应报文将 TAC 和状态码'9000'传送给终端。

IC 卡用以下数据组成的一个记录更新交易明细:

电子存折联机交易序号;

透支限额;

交易类型标识;

终端机编号;

交易日期(主机);

交易时间(主机)。

8.4.7.9 回送确认(步骤 6.8)

IC 卡在 UPDATE OVERDRAWLIMIT 命令的响应报文中返回 TAC 和一个完成码,表明透支限额已经被成功更新。

8.4.8 灰锁消费交易

8.4.8.1 一般说明

灰锁消费交易允许持卡人使用电子钱包进行灰锁消费。此交易可以脱机进行。

灰锁消费交易流程见图 9。

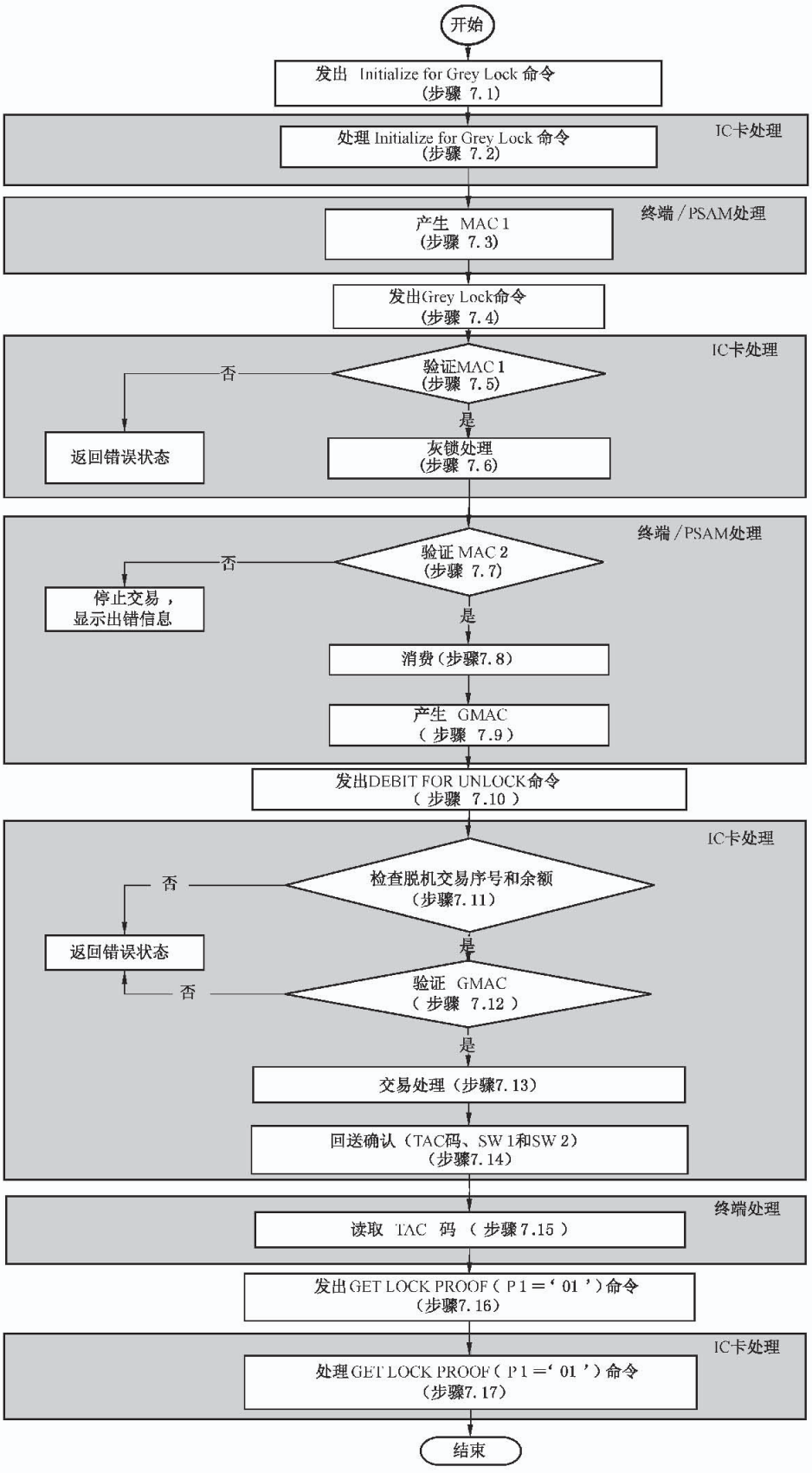
开始
发出 Initialize for Grey Lock 命令
(步骤 7.1)
IC卡处理
处理 Initialize for Grey Lock 命令
(步骤 7.2)
终端/PSAM处理
产生 MAC 1
(步骤 7.3)
发出Grey Lock命令
(步骤 7.4)
IC卡处理
验证MAC 1
(步骤 7.5)
否
是
返回错误状态
灰锁处理
(步骤 7.6)
终端/PSAM处理
验证 MAC 2
(步骤 7.7)
否
是
停止交易，
显示出错信息
消费(步骤7.8)
产生 GMAC
(步骤 7.9)
发出DEBIT FOR UNLOCK命令
(步骤 7.10)
IC卡处理
检查脱机交易序号和余额
(步骤7.11)
否
是
返回错误状态
验证 GMAC
(步骤 7.12)
否
是
交易处理(步骤7.13)
回送确认(TAC码、SW 1和SW 2)
(步骤7.14)
终端处理
读取 TAC 码 (步骤7.15)
发出GET LOCK PROOF(P1='01')命令
(步骤7.16)
IC卡处理
处理GET LOCK PROOF(P1='01')命令
(步骤7.17)
结束

图 9 灰锁消费交易流程

8.4.8.2 发出 INITIALIZE FOR GREY LOCK 命令(步骤 7.1)

终端发出 INITIALIZE FOR GREY LOCK 命令启动灰锁消费交易。

8.4.8.3 处理 INITIALIZE FOR GREY LOCK 命令(步骤 7.2)

IC 卡收到 INITIALIZE FOR GREY LOCK 命令后,将进行以下操作:

a) 检查命令中包含的密钥索引号是否被 IC 卡支持。如果不支持,返回状态码'9403'(不支持的密钥索引号)且不返回其他数据。

b) 在通过以上检查之后,IC 卡将产生一个伪随机数,这个伪随机数将包含在本命令的响应报文中返回终端,之后,IC 卡将内部的 TACUF 复位。

8.4.8.4 计算 MAC1(步骤 7.3)

使用 IC 卡返回的伪随机数和电子钱包脱机交易序号,终端的安全存取模块(PSAM)将产生一个终端随机数(TRAN),一个过程密钥(GSESPK)和一个报文认证码(MAC1),供 IC 卡来验证 PSAM 的合法性。

过程密钥 GSESPK 被用于电子钱包的灰锁消费交易。

过程密钥的产生分两步:即先是用 DPK 密钥产生的中间密钥,再用中间密钥采用下述的算法产生过程密钥。

用来产生中间密钥的输入数据如下:

TMPCK:伪随机数(ICC)|| 电子钱包脱机交易序号 || 终端交易序号的最右两个字节。

用中间密钥对终端随机数(TRAN)加密,运算的结果产生过程密钥:

GSESPK =DES(TMPCK,TRAN || '80000000')。

用 GSESPK 对以下数据进行加密产生 MAC1(按所列顺序):

交易类型标识;

终端机编号;

交易日期;

交易时间。

8.4.8.5 发出 GREY LOCK 命令(步骤 7.4)

终端发出 GREY LOCK 命令。

8.4.8.6 验证 MAC1(步骤 7.5)

IC 卡收到 GREY LOCK 命令后,将产生同样的过程密钥(GSESPK)并验证 MAC1 是否有效。如果 MAC1 是有效的,交易处理将继续。如果 MAC1 是无效的,IC 卡返回错误状态码'9302'(MAC 无效)给终端。

8.4.8.7 灰锁处理(步骤 7.6)

IC 卡将电子钱包脱机交易序号加 1,并将电子钱包应用灰锁。

IC 卡产生一个报文认证码(MAC2)供 PSAM 对 IC 卡合法性进行检查,并同时将 MAC2 写入内部文件。MAC2 将包含在从卡传送到 PSAM(通过终端)GREY LOCK 的命令响应报文和 GET LOCK PROOF 的命令响应报文中。

用 GSESPK 对以下这些数据进行加密产生 MAC2:

电子钱包余额;

电子钱包脱机交易序号(加1前)。

GTAC将包含在从卡传送到PSAM(通过终端)的GREY LOCK的命令响应报文和GET LOCK PROOF的命令响应报文中。如果之后出现交易异常中断等,使DEBIT FOR UNLOCK指令无法当时执行成功,GTAC可供终端纳入终端异常交易数据中,以便后来上传给主机进行灰锁验证。

下面是用来生成GTAC的要素:

交易类型标识;

终端机编号;

终端交易序号;

交易日期(终端);

交易时间(终端)。

IC卡应把GSESPK存贮到安全的内部文件中,(IC卡也可以将终端随机数、伪随机数(ICC)、终端交易序号等,写入内部文件,通过计算重新获得),以备交易中途IC卡掉电后,在后续交易流程中恢复过程密钥GSESPK。

IC卡将用以下数据组成的一个记录来更新内部专用明细。这个明细记录中的数据将包含在GET LOCK PROOF的命令响应报文中,由IC卡返回给终端。

交易类型标识('91'=电子钱包灰锁);

电子钱包代号('01'=电子钱包);

电子钱包余额;

电子钱包脱机交易序号;

终端机编号;

交易日期;

交易时间;

MAC2;

GTAC。

IC卡应全部成功地完成以上几个步骤或者一个也不完成,如果脱机交易序号的更新、电子钱包应用灰锁状态的设置没有成功,交易明细也不应更新。

8.4.8.8 验证MAC2(步骤7.7)

在收到IC卡(经终端)传来的MAC2后,PSAM要验证MAC2的有效性。MAC2如果有效,交易继续进行持卡人进行消费行为(步骤7.8)中所描述的步骤;如果MAC2是无效的,终端应停止交易并采取相应的措施。

8.4.8.9 持卡人进行消费行为(步骤7.8)

持卡人进行消费行为。在进行消费过程中,允许终端对IC卡下电。若下电以后,IC卡重新上电,经过交易预处理(选择应用、验证个人密码等)后应可以继续执行产生GMAC(步骤7.9)中所描述的步骤而不受影响。

8.4.8.10 产生GMAC(步骤7.9)

安全存取模块(PSAM)根据专用消费的金额,用过程密钥(GSESPK)产生一个报文认证码(GMAC),供IC卡来验证PSAM的合法性。

用GSESPK对以下数据进行加密产生GMAC:

交易金额;

安全认证识别码。

注：安全认证识别码已在 8.3.4.1 中定义。

8.4.8.11 发出 DEBIT FOR UNLOCK 命令(步骤 7.10)

终端发出 DEBIT FOR UNLOCK 命令。

8.4.8.12 检查脱机交易序号和余额(步骤 7.11)

收到 DEBIT FOR UNLOCK 命令后，IC 卡将进行以下操作：

a) 检查脱机交易序号是否匹配，如果脱机交易序号不匹配，IC 卡将返回‘9406’(脱机交易序号错)，但不返回其他数据；

b) 检查电子钱包余额是否大于或等于交易金额。如果小于交易金额，则返回状态码‘9401’(金额不足)，但不返回其他数据，IC 卡不操作内部出错计数器，终端应采取相应的措施；

c) 通过上面的检查后，IC 卡进入验证 GMAC(步骤 7.12)。

8.4.8.13 验证 GMAC(步骤 7.12)

IC 卡验证 GMAC 的有效性。如果 GMAC 是有效的，将 IC 卡内部的解扣出错计数器复位，交易处理将继续执行交易处理(步骤 7.13)。如果 GMAC 是无效的，IC 卡返回错误状态码‘9302’(MAC 无效)给终端，同时操作解扣出错计数器，3 次出错则临时锁住应用以防止恶意试探。该解扣出错计数器将在应用解锁命令执行成功后被复位。

8.4.8.14 交易处理(步骤 7.13)

IC 卡从卡上的电子钱包余额中减去灰锁消费的交易金额(如果交易金额为 0，则省略对余额的修改)、将电子钱包解锁、并将卡内的 TACUF(交易验证码待读标志)置位。

用密钥 DTK 产生一个 TAC。TAC 将被写入终端交易数据包，以便后来传给主机进行交易验证。

下面是用来生成 TAC 的要素(按所列顺序)：

交易金额；

交易类型标识(‘93’=电子钱包解扣)；

终端机编号(发出 DEBIT FOR UNLOCK 命令的终端)；

终端交易序号(发出 DEBIT FOR UNLOCK 命令的终端)；

交易日期(发出 DEBIT FOR UNLOCK 命令的日期)；

交易时间(发出 DEBIT FOR UNLOCK 命令的时间)。

对于电子钱包的灰锁消费交易，IC 卡将用以下数据组成的一个记录更新标准交易明细：

电子钱包脱机交易序号；

交易金额；

交易类型标识(‘93’=电子钱包解扣)；

终端机编号(发出 DEBIT FOR UNLOCK 命令的终端)；

交易日期(发出 DEBIT FOR UNLOCK 命令的日期)；

交易时间(发出 DEBIT FOR UNLOCK 命令的时间)。

对于电子钱包的灰锁消费交易，IC 卡将用以下数据组成的一个记录更新内部专用明细文件，以便以后终端可以通过 GET LOCK PROOF 命令得到：

交易类型标识；

电子钱包代号(‘01’=ET)；

电子钱包余额；

电子钱包脱机交易序号；

终端机编号(发出 DEBIT FOR UNLOCK 命令的终端);
交易日期(发出 DEBIT FOR UNLOCK 命令的日期);
交易时间(发出 DEBIT FOR UNLOCK 命令的时间);
交易金额;
TAC。

IC 卡应全部成功地完成以上几个步骤或者一个也不完成,如果余额的更新、TACUF 的置位、电子钱包应用的灰锁状态的恢复没有成功,标准交易明细和内部专用明细也不应被更新。

8.4.8.15 返回确认(步骤 7.14)

IC 卡在 DEBIT FOR UNLOCK 命令的响应报文中返回 TAC 码和 SW1SW2='9000',表明余额已被更新而且电子钱包应用已解锁。

8.4.8.16 读取 TAC 码(步骤 7.15)

终端读取由 IC 卡发来的 TAC 码,合成完整的交易成交数据包。

8.4.8.17 发出 GET LOCK PROOF(P1='01')命令(步骤 7.16)

终端发出 GET LOCK PROOF(P1='01')命令。

8.4.8.18 处理 GET LOCK PROOF(P1='01')命令(步骤 7.17)

IC 卡将内部的 TACUF(交易验证码待读标志)复位。

8.4.9 联机解扣交易

8.4.9.1 一般说明

联机解扣交易允许将 IC 卡上的电子钱包应用解锁并扣除相应的交易金额。本交易应在联机的终端上进行。

联机解扣交易流程见图 10。

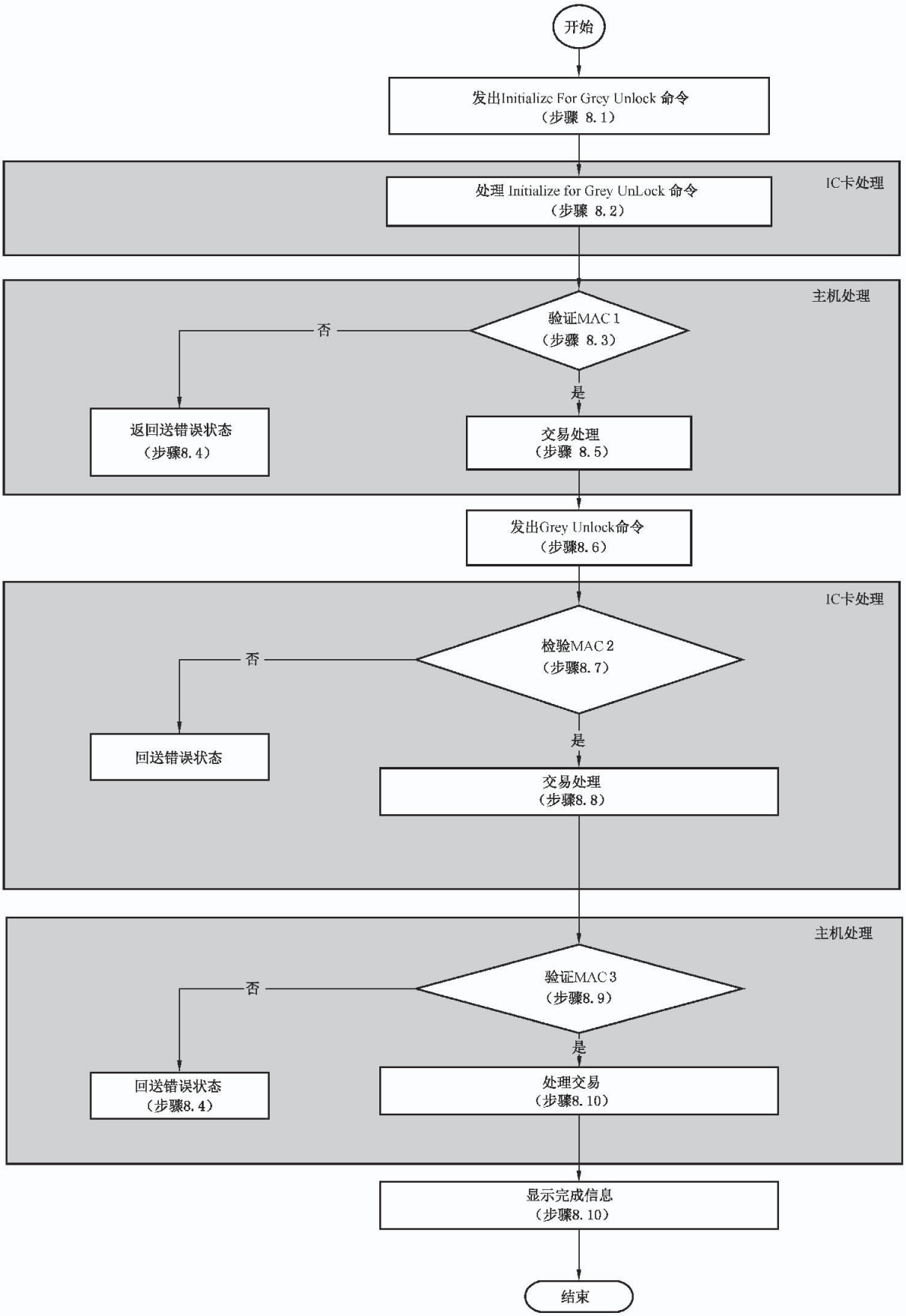

图 10　联机解扣交易流程

8.4.9.2 发出 INITIALIZE FOR GREY UNLOCK 命令(步骤 8.1)

终端发出 INITIALIZE FOR GREY UNLOCK 命令启动联机解扣交易。

8.4.9.3 处理 INITIALIZE FOR GREY UNLOCK 命令(步骤 8.2)

IC 卡收到 INITIALIZE FOR GREY UNLOCK 命令后,将进行以下操作:

a) 检查 IC 卡是否支持命令中包含的密钥索引号。如果不支持,返回状态码'9403'(不支持的密钥索引号),且不返回其他数据。

b) 在通过以上检查之后,IC 卡将产生一个伪随机数(ICC)、过程密钥 SESULK 和一个报文认证码(MAC1),供主机来验证联机解扣交易和 IC 卡的合法性。过程密钥 SESULK 被用于电子钱包的联机解扣交易。用来产生过程密钥的输入数据如下:

SESULK:伪随机数(ICC)|| 电子钱包联机交易序号 || '8000'

用 SESUlK 对以下数据加密产生 MAC1(按所列顺序):

电子钱包余额;

电子钱包脱机交易序号;

交易类型标识;

终端机编号。

IC 卡把 INITIALIZE FOR GREY UNLOCK 命令的响应报文送给终端处理。

如果 IC 卡返回的状态不是'9000',终端将终止交易。

在收到 INITIALIZE FOR GREY UNLOCK 命令的响应报文后,终端将联机解扣许可请求数据包送往发卡方主机。

8.4.9.4 验证 MAC1(步骤 8.3)

主机将生成 SESULK 并且确认 MAC1 是否有效。如果 MAC1 有效,交易处理将按主机处理(步骤 8.5)中描述的步骤继续执行:否则主机返回一个错误状态码,交易处理将转至返回错误状态(步骤 8.4)中描述的步骤。

8.4.9.5 返回错误状态(步骤 8.4)

如果出现使联机解扣交易不能被接受的情况,则主机应通知终端。终端将采取相应的措施。

8.4.9.6 交易处理(步骤 8.5)

在确认能够进行联机解扣交易后,主机将产生一个报文认证码(MAC2),供 IC 卡对主机合法性进行检查。主机发送一个联机解扣交易接受报文给终端,其中至少包括 MAC2、交易日期(主机)和交易时间(主机)。

用 SESULK 对以下数据进行加密产生 MAC2(按所列顺序):

应补扣的交易金额;

交易类型标识;

终端机编号;

交易日期(主机);

交易时间(主机)。

8.4.9.7 发出 GREY UNLOCK 命令(步骤 8.6)

终端收到主机的联机解扣交易响应报文后,向 IC 卡发出 GREY UNLOCK 命令,以更新卡上电子

钱包余额、并将电子钱包应用解锁。

8.4.9.8 验证 MAC2(步骤 8.7)

收到 GREY UNLOCK 命令后,IC 卡先检查电子钱包余额是否大于或等于交易金额。如果小于交易金额,则返回状态码'9401'(金额不足),但不返回其他数据。IC 卡还要验证 MAC2 的有效性。如果 MAC2 是有效的,交易处理将继续执行;否则 IC 卡返回错误状态码'9302'(MAC 无效)给终端。

8.4.9.9 交易处理(步骤 8.8)

IC 卡从卡上的电子钱包余额中减去交易金额(如果交易金额为 0,则省略对余额的修改),将电子钱包联机交易序号加 1,将内部的解扣出错计数器复位,并将电子钱包应用解锁。

IC 卡产生一个报文认证码(MAC3),包含在从卡传送到主机(通过终端)的 GREY UNLOCK 命令的响应报文中,以供主机对联机解扣交易的成功的合法性进行检查。

用 SESULK 对以下数据进行加密产生 MAC3(按所列的顺序):

电子钱包余额;
电子钱包联机交易序号(加 1 前);
交易金额;
交易类型标识;
终端机编号;
交易日期(主机);
交易时间(主机)。

在对电子钱包的联机解扣交易中,IC 卡用以下数据组成的一个记录来更新标准交易明细:

电子钱包联机交易序号;
交易金额;
交易类型标识('95'=电子钱包联机解扣);
终端机编号;
交易日期(主机);
交易时间(主机)。

对于电子钱包的联机解扣交易,IC 卡将用以下数据组成的一个记录更新内部专用明细文件,以便以后终端可以通过 GET LOCK PROOF 命令得到:

交易类型标识;
电子钱包代号('01'=电子钱包);
电子钱包余额;
电子钱包联机交易序号;
终端机编号;
交易日期;
交易时间;
交易金额;
MAC3。

IC 卡应全部成功地完成以上几个步骤或者一个也不完成,如果上述的操作没有成功,标准交易明细和内部专用明细也不应更新。

8.4.9.10 验证 MAC3(步骤 8.9)

主机收到从 IC 卡(经过终端)传来的 MAC3 后,应验证 MAC3 的有效性。

如果 MAC3 正确，则执行显示完成(步骤 8.10)中描述的步骤，否则主机发给终端错误状态码。

8.4.9.11 显示完成(步骤 8.10)

在收到主机的完成报文后，终端做相应的处理，显示完成信息。

8.4.10 补扣交易

8.4.10.1 一般说明

补扣交易允许持卡人在灰锁的电子钱包应用中，对电子钱包补扣上次消费交易未扣除的交易额，并将电子钱包应用解锁。

本交易应在拥有该电子钱包的上次异常交易记录的终端上进行。异常交易记录至少包括灰锁的电子钱包应用的应用序列号、灰锁的电子钱包脱机交易序号、应扣的交易金额、GMAC。拥有异常交易记录的终端可以是产生灰卡的终端，也可以是通过网络通讯得到异常交易记录的其他终端。

在交易预处理流程中发现电子钱包应用已灰锁时，进入本交易流程。

交易预处理流程中，终端收到 IC 卡返回的 GET LOCK PROOF(P1='00')的命令响应报文后，得到上次的灰锁操作的产生日期、时间、MAC2、GTAC 等数据。这些数据是终端进行补扣交易的依据。

补扣交易流程见图 11。

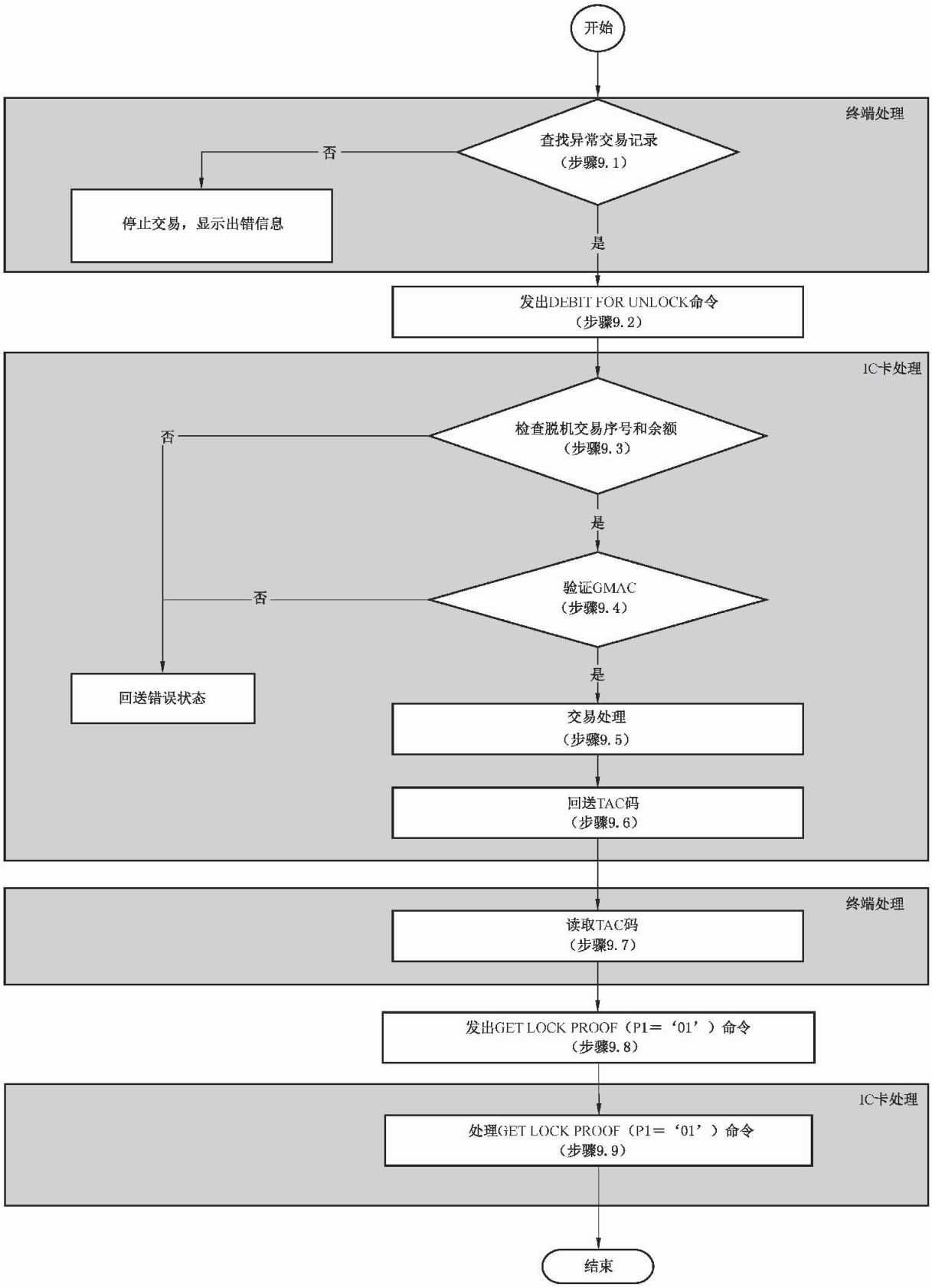
开始
终端处理
查找异常交易记录
（步骤9.1）
否
停止交易，显示出错信息
是
发出DEBIT FOR UNLOCK命令
（步骤9.2）
IC卡处理
检查脱机交易序号和余额
（步骤9.3）
否
是
验证GMAC
（步骤9.4）
否
回送错误状态
是
交易处理
（步骤9.5）
回送TAC码
（步骤9.6）
终端处理
读取TAC码
（步骤9.7）
发出GET LOCK PROOF（P1=‘01’）命令
（步骤9.8）
IC卡处理
处理GET LOCK PROOF（P1=‘01’）命令
（步骤9.9）
结束

图 11　补扣交易流程

8.4.10.2 查找异常交易记录(步骤 9.1)

终端得到 IC 卡的响应报文后,在异常交易记录中进行查找,如果有符合条件的异常交易记录,就进入发出 DEBIT FOR UNLOCK 命令(步骤 9.2)节所描述的步骤;否则停止交易,显示相应的提示信息。

8.4.10.3 发出 DEBIT FOR UNLOCK 命令(步骤 9.2)

终端发出 DEBIT FOR UNLOCK 命令。

8.4.10.4 检查脱机交易序号和余额(步骤 9.3)

IC 卡收到 DEBIT FOR UNLOCK 命令后,将进行以下操作:

a) 检查脱机交易序号是否匹配,如果脱机交易序号不匹配,IC 卡将返回'9406'(脱机交易序号错),但不返回其他数据。

b) 检查电子钱包余额是否大于或等于交易金额。如果小于交易金额,则返回状态码'9401'(金额不足),但不返回其他数据,IC 卡不操作内部出错计数器,终端应采取相应的措施。

8.4.10.5 验证 GMAC(步骤 9.4)

IC 卡验证 GMAC 的有效性。如果 GMAC 是有效的,将 IC 卡内部的解扣出错计数器复位,交易处理将继续执行交易处理(步骤 9.5)节。如果 GMAC 是无效的,IC 卡返回错误状态码'9302'(MAC 无效)给终端,同时操作内部的解扣出错计数器,出错达到 3 次则临时锁住应用以防止恶意试探。

8.4.10.6 交易处理(步骤 9.5)

IC 卡从卡上的电子钱包余额中减去灰锁消费的交易金额(如果交易金额为 0,则省略对余额的修改)、将电子钱包应用解锁,并将卡内的 TACUF(交易验证码待读标志)置位。

用密钥 DTK 产生一个 TAC,TAC 将被写入终端交易数据包,以便后来传给主机进行交易验证。

下面是用来生成 TAC 的要素:

交易金额;

交易类型标识;

终端机编号(发出 DEBIT FOR UNLOCK 命令的终端);

终端交易序号(发出 DEBIT FOR UNLOCK 命令的终端);

交易日期(发出 DEBIT FOR UNLOCK 命令的日期);

交易时间(发出 DEBIT FOR UNLOCK 命令的时间)。

对于电子钱包的补扣交易,IC 卡将用以下数据组成的一个记录更新标准交易明细:

电子钱包脱机交易序号;

交易金额;

交易类型标识(电子钱包解扣);

终端机编号(发出 DEBIT FOR UNLOCK 命令的终端);

交易日期(发出 DEBIT FOR UNLOCK 命令的日期);

交易时间(发出 DEBIT FOR UNLOCK 命令的时间)。

对于电子钱包的补扣交易,IC 卡将用以下数据组成的一个记录更新内部专用明细文件,以便以后终端可以通过 GET LOCK PROOF 命令得到:

交易类型标识;

电子钱包代号('01'=电子钱包);

电子钱包余额;

电子钱包脱机交易序号码；
终端机编号(发出 DEBIT FOR UNLOCK 命令的终端)；
交易日期(发出 DEBIT FOR UNLOCK 命令的日期)；
交易时间(发出 DEBIT FOR UNLOCK 命令的时间)；
交易金额；
TAC。

IC 卡应全部成功地完成以上几个步骤或者一个也不完成，如果余额的更新、TACUF 的置位、电子钱包应用解锁未成功，标准交易明细和内部专用明细也不应被更新。

8.4.10.7 返回 TAC 码(步骤 9.6)

IC 卡在 DEBIT FOR UNLOCK 命令的响应报文中返回 TAC 码，表明余额已被更新而且电子钱包应用已解锁。

8.4.10.8 读取 TAC 码(步骤 9.7)

终端读取由 IC 卡发来的 TAC 码，合成完整的交易成交数据包。

8.4.10.9 发出 GET LOCK PROOF(P1='01')命令(步骤 9.8)

终端发出 GET LOCK PROOF(P1='01')命令。

8.4.10.10 处理 GET LOCK PROOF(P1='01')命令(步骤 9.9)

IC 卡将内部的交易验证码待读标志(TACUF)复位。

8.4.11 补充交易

8.4.11.1 一般说明

补充交易允许终端读取上次灰锁消费交易或补扣交易中未获得的 TAC，上送主机以供验证。

在交易预处理流程中发现电子钱包解扣操作已完成而 TAC 未成功读取，则进入本交易流程。

交易预处理流程中，终端收到 IC 卡返回的 GET LOCK PROOF(P1='00')的命令响应报文后，得到上次的灰锁操作的产生日期、时间、TAC 等数据。这些数据是终端进行补充交易的依据。

补充交易流程见图 12。

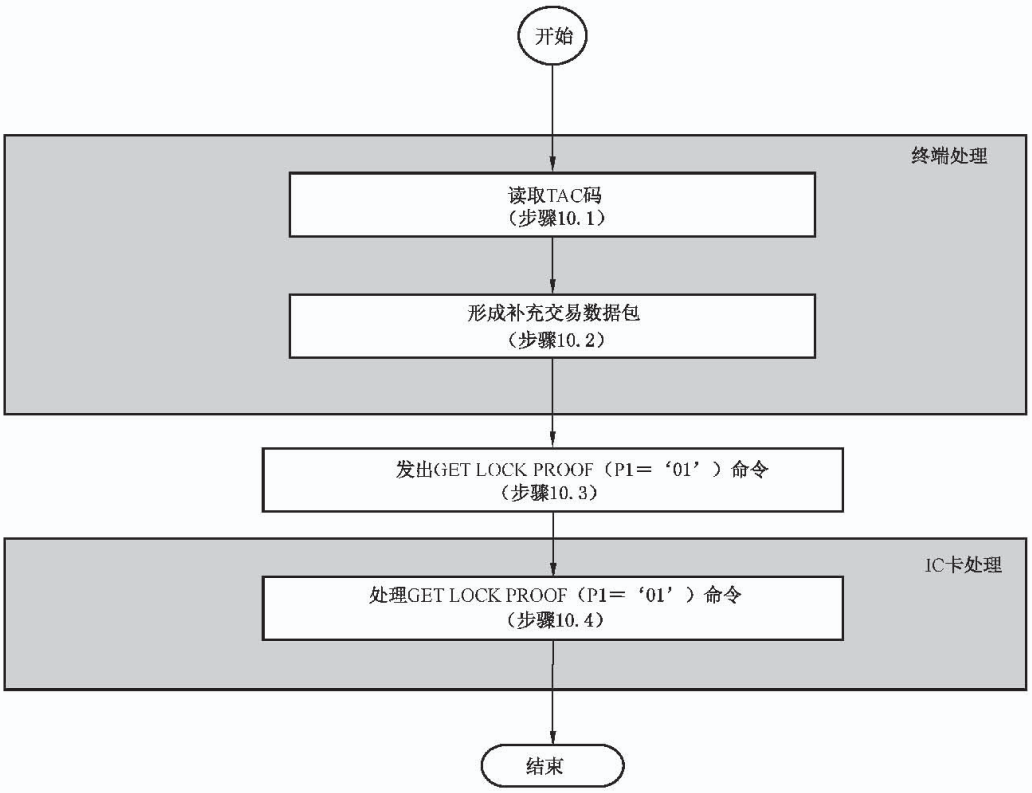

图 12　补充交易流程

8.4.11.2　读取 TAC(步骤 10.1)

终端通过 GET LOCK PROOF 命令中的响应报文,获得上次灰锁消费或补扣交易的 TAC。

8.4.11.3　形成补充交易数据包(步骤 10.2)

终端将读到的 TAC,和其他的相关数据合成一条补充交易数据包,以便上送主机。

8.4.11.4　发出 GET LOCK PROOF(P1='01')命令(步骤 10.3)

终端发出 GET LOCK PROOF(P1='01')命令。

8.4.11.5　处理 GET LOCK PROOF(P1='01')命令(步骤 10.4)

IC 卡将内部的 TACUF 复位。

8.4.12　查询余额交易

持卡人可以通过终端或其他读卡设备读取电子存折中的余额。此交易一般脱机进行。在电子存折应用中进行此交易应提交个人密码(PIN)。电子钱包则不需要。

终端利用 GET BALANCE 命令实现查询余额交易。

8.4.13　查询明细交易

持卡人可以通过终端或其他读卡设备读取 IC 卡中的交易明细记录。此交易一般采用脱机方式处

理。交易时需提交个人密码(PIN)。

终端发出一个 READ RECORD 命令来获得交易明细。这个命令会返回某个交易明细记录中所含的所有数据。交易明细文件为循环记录文件,且至少应包含 10 条记录。

交易明细中的记录使用记录号寻址。记录号范围从 1 到 n,n 是文件中记录的最大个数。最近写入的记录号为 1,前一记录号为 2,如此类推直到 n。n 代表文件中最早写入的记录。

IC 卡应支持在以下交易中记录明细:电子钱包圈存交易、电子存折圈存交易、电子存折圈提交易、电子存折消费交易、电子钱包消费交易、电子存折取现交易、电子存折修改透支限额交易。

8.4.14 应用维护功能

8.4.14.1 安全报文

电子存折/电子钱包应用涉及的安全机制,应按照“安全机制”部分的规定进行,并作如下改动和增补:

a) 在传送一个包含安全报文的命令前,主机向终端发送一个报文,要求从 IC 卡获得一个随机数。终端向 IC 卡发出一个 GET CHALLENGE 命令。从 IC 卡返回的随机数被送往主机以用于安全报文处理;

b) 从 IC 卡返回的 4 字节随机数后缀以‘00 00 00 00’,所得到的结果作为初始值,用以代替电子钱包/电子存折卡片标准中定义的初始化值;

c) 不采用过程密钥。除去 UNBLOCK PIN 命令外,均使用导出的应用维护密钥(DAMK)来计算 MAC。UNBLOCK PIN 命令采用导出的 PIN 解锁密钥来产生 MAC;

d) 全部采用双倍长密钥的 3DEA 算法。

8.4.14.2 卡片锁定

终端发出 CARD BLOCK 命令来锁定卡片。

CARD BLOCK 命令参照“文件和命令”部分。命令的成功执行使得 IC 卡中的所有应用无效。在这种情况下,进行应用选择将会返回状态码‘6A81’(功能不被支持)。

8.4.14.3 应用锁定

终端发出 APPLICATION BLOCK 命令来锁定应用。

命令的用法由发卡方自行决定。

命令参照“命令”部分。命令的成功执行导致 IC 卡中的电子存折/电子钱包应用无效。在这种状态下:

a) 选择此应用时,对 SELECT 命令 IC 卡返回状态码‘6283’(选择文件无效)和文件控制信息(FCI),在 T=0 协议时,卡片 FCI 需用 GET RESPONSE 命令取回。

b) 在应用被选择后,除以下情况外,IC 卡对其他命令只返回状态码‘6985’(使用的条件不满足):
   1) 当用 SELECT 命令选择此应用或其他应用时;
   2) 当用 GET CHALLANGE 命令时;
   3) APPLICATION BLOCK 命令;
   4) CARD BOLCK 命令;
   5) APPLICATION UNBLOCK 命令。

c) 如果在命令参数 P2 中指明永久性锁定此应用,IC 卡将设置一个内部标志以表明不允许执行 APPLICATION UNBLOCK 命令。

d) 此命令的执行并不改变电子存折联机交易序号和电子钱包联机交易序号的值。

8.4.14.4 应用解锁

终端发出 APPLICATION UNBLOCK 命令来对应用解锁。

如果对某应用连续三次解锁失败，则 IC 卡将永久锁定此应用并返回状态码‘9303’(应用永久锁定)。

如果在 APPLICATION UNBLOCK 命令中使用了永久锁定的选项，IC 卡将返回状态码‘9303’(应用永久锁定)且不再对应用解锁。

APPLICATION UNBLOCK 命令的成功执行使应用重新恢复成有效状态。在此之后，该应用对所有命令的响应就像应用锁定和应用解锁没有执行过一样。

此命令的执行并不改变电子存折联机交易序号和电子钱包联机交易序号的值。

8.4.14.5 PIN 解锁

终端发出 UNBLOCK PIN 命令对 PIN 解锁。

在命令报文中，P2 取‘01’值。使用 DPUK 对 PIN 数据加密。

如果 PIN 连续三次解锁失败，则 IC 卡将永久锁定此应用并返回状态码‘9303’(应用永久锁定)。

8.4.14.6 二进制形式修改

终端发出 UPDATE BINARY 指令。

如果三次执行此命令均告失败，则 IC 卡将永久锁定此应用并返回状态码‘9303’(应用永久锁定)。

8.4.14.7 更改 PIN

更改 PIN 功能不需要 MAC，它可以在任意支持该命令的终端上执行。当 IC 卡接到此命令时，它将进行以下操作：

a) 检查 PIN 尝试计数器。如果为 0，表明 PIN 已锁定，此命令不能执行。在这种情况下，IC 卡返回状态码‘6983’(认证方式锁定)。

b) 如果 PIN 没有锁定，则命令中的‘当前 PIN’会和 IC 卡上存放的 PIN 比较。如果二者相同，IC 卡将进行以下操作：

   1) 将 IC 卡上的 PIN 改为命令中的新 PIN；

   2) 将 PIN 尝试计数器置为 PIN 重试的最大次数。

c) 如果卡上的 PIN 和命令中的‘当前 PIN’并不相同，IC 卡将进行以下操作：

   1) 将 PIN 尝试计数器减 1；

   2) 返回状态码‘63Cx’，这里 x 是 PIN 尝试计数器的新值。如达到零，则卡片自动锁定 PIN。

8.4.14.8 重装 PIN

终端发出 RELOAD PIN 命令来重装 PIN。用密钥 DRPK 产生一个 MAC。当此命令失败三次之后，应用被永久锁定。

8.5 防拔

IC 卡应能够在交易处理中的任何情况下，甚至是在更新 EEPROM 过程中卡片掉电的情况下，保持数据的完整性。这就需要在每次更新数据前对数据进行备份，并且在卡片重新加电后自动地触发恢复机制。

在终端发给 IC 卡一个命令以更新电子存折余额或电子钱包余额时，卡片返回一个 MAC 或/和

TAC,以证明更新已经发生。这样的情况有圈存(TAC),圈提(MAC3)、消费/取现(TAC)和修改透支限额(TAC)。

IC卡应在更新余额前计算MAC或TAC,一旦余额更新成功,应保证可以通过GET TRANSCTION PROVE命令获得此MAC或TAC。如果防拔恢复已使余额恢复到更新前的数值,那么有关的加密数据不必再保留。接到更改ED或EP余额的命令,如Debit、Credit命令时,这些加密数据可能被丢弃。

如果在命令已执行结束,而终端还未收到响应之前,卡片突然离开,终端将会处于不知卡片是否更新的不定状态。这种情况下,终端应负责用GET TRANSACTION PROVE命令进行恢复。

如果卡片正在处理时突然离开,终端应提醒持卡人重新操作卡片,之后终端将检查发卡方标识和应用序列号以确认进入的卡片和前面离开的卡片是否同一张卡。如果是同一张卡,终端发出GET TRANSACTION PROVE命令,假如MAC和TAC返回,终端即完成交易处理。如果MAC和TAC无法返回,则说明IC卡中的余额没有被修改。交易可以用适当的初始化命令重新开始。

8.6 交易处理性能

交易处理性能要求针对非接触式IC卡。

交易处理性能要求主要体现在消费交易,从卡被选择到交易处理结束允许离开磁场感应区的时间宜限制为不大于300 ms。

附 录 A
（规范性附录）
算法标识文件 ADEE 说明

A.1 说明

本附录描述了算法标识文件 ADEE 的文件结构及算法支持说明。见表 A.1 和表 A.2。

表 A.1 ADEE 文件结构说明

文件标识(FID)	0xADEE		
文件类型	二进制文件		
文件大小	0010H		
文件存取控制	读＝自由	改写 ＝ DAMK 线路保护	
字节	数据元	长度	格式
01～01	算法标识 1	1	HEX
02～02	算法标识 1 取反	1	HEX
03～03	算法标识 2	1	HEX
04～04	算法标识 2 取反	1	HEX
05～16	预留(预留算法标识与上述算法使用方式相同)	12	HEX
注：先用 00A4 0000 02 ADEE 指令选中，再用 00B0 0000 10 读取；请勿使用带 SFI 的 00B0 指令进行读取。			

表 A.2 算法标识说明

算法标识 1	b7	b6	b5	b4	b3	b2	b1	b0	说 明
	—	—	—	—	—	—	0	0	代表支持 3DES 算法
	—	—	—	—	—	—	1	1	代表支持 SM1 算法
	—	—	—	—	0	1	—	—	代表支持 SM4 算法
	—	—	1	1	—	—	—	—	预留算法
算法标识 2	b7	b6	b5	b4	b3	b2	b1	b0	说 明
	—	—	—	—	—	—	0	0	代表支持 3DES 算法
	—	—	—	—	—	—	1	1	代表支持 SM1 算法
	—	—	—	—	0	1	—	—	代表支持 SM4 算法
	—	—	1	1	—	—	—	—	预留算法

A.2 文件结构及算法支持说明

ADEE 文件位于 MF 下公共钱包应用目录下。

附 录 B
（规范性附录）
数据元解释

本标准所使用的数据元定义见表 B.1。

表 B.1 数据元解释

数据域	说明	来源	长度（字节）	值
算法标识(DLK)	用来标识圈存交易的加密算法	IC 卡 PSAM 卡	1	
算法标识(DPK)	用来标识消费和取现交易的加密算法	IC 卡 PSAM 卡	1	
算法标识(DTK)	用来标识在交易中计算 TAC 使用的加密算法	IC 卡 PSAM 卡	1	
算法标识(DUK)	用来标识在修改透支限额交易中使用的加密算法	IC 卡 PSAM 卡	1	
算法标识(DULK)	用来标识在圈提交易中使用的加密算法	IC 卡 PSAM 卡	1	
应用类型标识	IC 卡支持的表示卡存在的应用类型(ED 或 EP)的标识	IC 卡	1	值： 01——只有 ED 02——只有 EP 03——ED 和 EP 都存在 所有其他值保留为将来使用
密钥索引号	为了唯一标识在一个密钥版本中的密钥索引号而分配的一个数字	IC 卡 PSAM 卡	1	
密钥版本号(DLK)	用来唯一标识圈存交易的密钥版本	IC 卡	1	
密钥版本号(DPK)	用来唯一标识一个消费或取现交易的密钥版本	IC 卡	1	
密钥版本号(DTK)	用来唯一标识计算 TAC 所用的密钥版本	IC 卡	1	
密钥版本号(DUK)	用来唯一标识一个修改透支限额交易的密钥版本	IC 卡	1	
密钥版本号(DULK)	用来唯一标识一个圈提交易的密钥版本	IC 卡	1	
交易时间	交易发生时间	终端	3	
交易类型标识(TTI)	用于标识持卡人选择的交易类型(例如：圈存、圈提、消费等)而分配的一个值	PSAM 卡 IC 卡	1	值： 01——ED 圈存 02——EP 圈存 03——圈提 04——ED 取款 05——ED 消费 06——EP 消费 07——ED 修改透支限额 08——信用消费

当为数据定义的长度超过实际数据长度，而位数没有占满时，补位规则如下：

格式 n 的数据元右靠齐并且左补十六进制‘0’。

格式 cn 的数据元左靠齐并且右补十六进制‘F’。

格式 an 的数据元左靠齐并且右补十六进制‘0’。

格式 ans 的数据元左靠齐并且右补十六进制‘0’。

当数据从一个实体移动到另一个时(例如：卡到终端)，不管其内部如何存放，都是按照由高到低的顺序传送。数据的连接也同样符合这个原则。

附 录 C
（规范性附录）
ED/EP 应用的密钥关系

C.1 说明

本附录描述了与 ED/EP 应用相关的设备实体之间的密钥关系，此处还描述了 IC 卡密钥的推导方法和过程密钥的产生方法。

C.2 密钥关系

为确保密钥的安全，密钥的产生和存放都应由一个专用的安全模块来处理。支持 ED 和 EP 应用的 CPU 卡、POS 设备之间的密钥关系见表 C.1 和表 C.2。

表 C.1 共用于电子存折和电子钱包应用的密钥

密钥	密钥说明	CPU 卡	POS (PSAM)
消费主密钥(MPK)	用于消费/取现交易的密钥	消费子密钥(DPK)，由 MPK 用应用序列号推导获得	消费主密钥(MPK)
圈存主密钥(MLK)	用于圈存交易的密钥	圈存子密钥(DLK)，由 MLK 用应用序列号推导获得	N/A
TAC 主密钥(MTK)	消费/取现交易中用于产生 TAC 的密钥	TAC 子密钥(DTK)，由 MTK 用应用序列号推导获得	N/A
PIN 解锁主密钥(MPUK)	用于解锁 PIN 的密钥	PIN 解锁子密钥(DPUK)，由 MPUK 用应用序列号推导获得	由发卡方考虑决定
PIN 重装主密钥(MRPK)	用于重装 PIN 的密钥	PIN 重装子密钥(DRPK)，由 MRPK 用应用序列号推导获得	N/A
应用主控密钥(MAMK)	用于应用维护功能的密钥	应用主控子密钥(DAMK)，由 MAMK 用应用序列号推导获得	N/A

表 C.2　用于电子存折应用的密钥

密钥	密钥说明	CPU 卡	POS（PSAM）
圈提主密钥（MULK）	用于圈提交易的密钥	圈提子密钥（DULK），由 MULK 用应用序列号推导获得	N/A
修改主密钥（MUK）	用于修改透支限额交易的密钥	子修改（透支限额）密钥（DUK），由 MUK 用应用序列号推导获得	N/A

附 录 D
（资料性附录）
应用密钥说明

D.1 密钥存储说明

见表 D.1。

表 D.1 密钥存储说明

密钥名称	密钥版本数量	密钥索引数量	密钥数量	密钥分散级别	主密钥存放位置	子密钥存放位置
卡片主控密钥	01	01	1	1	UM/J	U
卡片维护密钥	01	01	1	1	UM/J	U
应用主控密钥	01	01	1	1	UM/J	U
应用维护密钥	01	01	1	1	UM/J	U
复合消费维护密钥	01	01	1	1	UM/J/PM/P	U
消费密钥	01	10	10	2	UM/J/PM/P	U
圈存密钥	01	02	2	1	UM/J/IM/I	U
PIN 解锁密钥	01	01	1	1	UM/J/PM/P	U
PIN 重置密钥	01	01	1	1	UM/J/PM/P	U
TAC 密钥	01	01	1	2	UM/J/PM/P	U
内部认证密钥	01	01	1	1	UM/J/PM/P	U
PSAM/ISAM 卡卡片主控密钥	01	01	1	1	PM/IM	P/I
PSAM/ISAM 卡卡片维护密钥	01	01	1	1	PM/IM	P/I
PSAM/ISAM 卡应用主控密钥	01	01	1	1	PM/IM	P/I
PSAM/ISAM 卡应用维护密钥	01	01	1	1	PM/IM	P/I

D.2 主密钥存放位置

主密钥存放的位置说明应符合下列要求：

UM 用户卡发卡母卡；

PM PSAM 卡发卡母卡；

IM ISAM 卡发卡母卡；

J 后台认证加密机；

P　　PSAM 卡；
I　　ISAM 卡；
U　　用户卡。

中华人民共和国城镇建设
行 业 标 准
建设事业智能卡操作系统技术要求
CJ/T 304—2017
*
中国标准出版社出版发行
北京市朝阳区和平里西街甲2号(100029)
北京市西城区三里河北街16号(100045)
网址 www.spc.net.cn
总编室:(010)68533533 发行中心:(010)51780238
读者服务部:(010)68523946
中国标准出版社秦皇岛印刷厂印刷
各地新华书店经销
*
开本 880×1230 1/16 印张 6.5 字数 193 千字
2018年2月第一版 2018年2月第一次印刷
*
书号:155066·2-32574 定价 84.00 元

CJ/T 304-2017

中国标准出版社

建设事业智能卡操作系统技术要求