EMV '96
Integrated Circuit Card Application Specification for Payment Systems

Version 3.1.1
May 31, 1998

Table of Contents

Tables

Figures

1. Scope

The *Integrated Circuit Card Application Specification for Payment Systems* (hereinafter referred to simply as 'the Application Specification') defines the terminal and integrated circuit card (ICC) procedures necessary to effect a payment system transaction in an international interchange environment.

In particular it covers:

- Mapping of data elements to files
- Transaction flow (the sequence of events and the commands issued to the card)
- Exception processing
- Coding of specific data objects described generally in the *Integrated Circuit Card Specification for Payment Systems* (hereinafter referred to simply as the 'Card Specification') (see Annex A)

The functions described are those necessary to ensure that payment system cards conforming to this specification can perform the set of common core functions in all terminals conforming to this specification. Application functions unique to individual payment systems and those functions not performed in interchange are not described, but are not precluded.

This specification does not address clearing and settlement or any transactions where the ICC is not present.

The Application Specification assumes familiarity with the Card Specification. The Card Specification describes functionality outside the application layer, including application selection. Both specifications are intended for use by payment system members, ICC and terminal manufacturers, and designers of applications using ICCs or interfacing to payment system applications that use ICCs.

1.1 EMV Specification Version Numbering

To facilitate future reference of the EMV specifications and to differentiate between technical updates and editorial clarifications, with the publication of this version of the EMV specifications, EMV has introduced the following version numbering scheme:

version X.Y.Z,

where:

X indicates the phase number of the specifications
Y indicates technical change(s) from the previous version
Z indicates editorial change(s) from the previous version

Therefore, this version of the EMV specifications is version 3.1.1, since the basis for this specification is version 3 and both technical and editorial changes have been made.

2. Normative References

The following standards contain provisions that are referenced in this specification:

Europay, MasterCard, and Visa (EMV):March 31, 1998	Integrated Circuit Card Specification for Payment Systems
Europay, MasterCard, and Visa (EMV):March 31, 1998	Integrated Circuit Card Terminal Specification for Payment Systems
ISO 8859:1987	Information processing - 8-bit single byte coded graphic character sets
ISO 9564:1991	Banking - Personal identification number management and security
ISO/IEC 7816-4:1995	Identification cards - Integrated circuit(s) cards with contacts - Part 4: Inter-industry commands for interchange
ISO/IEC 7816-5:1994	Identification cards - Integrated circuit(s) cards with contacts - Part 5: Numbering system and registration procedure for application identifiers

3. Definitions

The following terms are used in this specification.

Application - The protocol between the card and the terminal and its related set of data.

Byte - 8 bits.

Card - A payment card as defined by a payment system.

Command - A message sent by the terminal to the ICC that initiates an action and solicits a response from the ICC.

Concatenation - Two elements are concatenated by appending the bytes from the second element to the end of the first. Bytes from each element are represented in the resulting string in the same sequence in which they were presented to the terminal by the ICC, that is, most significant byte first. Within each byte bits are ordered from most significant bit to least significant. A list of elements or objects may be concatenated by concatenating the first pair to form a new element, using that as the first element to concatenate with the next in the list, and so on.

Cryptogram - Result of a cryptographic operation.

Financial Transaction - The act between a cardholder and a merchant or acquirer that results in the exchange of goods or services against payment.

Function - A process accomplished by one or more commands and resultant actions that are used to perform all or part of a transaction.

Integrated Circuit(s) - Electronic component(s) designed to perform processing and/or memory functions.

Integrated Circuit(s) Cards - A card into which one or more integrated circuits are inserted to perform processing and memory functions.

Message - A string of bytes sent by the terminal to the card or vice versa, excluding transmission control characters.

Online - When used in the Application Specification, online means online to the issuer or to a host system standing in for the issuer. Connections between the terminal and a merchant or acquirer host are not included in this definition.

Payment System - For the purposes of this specification, Europay International S.A., MasterCard International Incorporated, or Visa International Service Association.

PIN Pad - An arrangement of numeric and command keys to be used for PIN entry.

Response - A message returned by the ICC to the terminal after the processing of a command message received by the ICC.

Script - A command or a string of commands transmitted by the issuer to the terminal for the purpose of being sent serially to the ICC as commands.

Template - Value field of a constructed data object, defined to give a logical grouping of data objects.

Terminal - The device used in conjunction with the ICC at the point of transaction to perform a financial transaction. It incorporates the interface device and may also include other components and interfaces such as host communications.

4. Abbreviations and Notations

The following abbreviations and notations are used in this specification.

AAC	Application Authentication Cryptogram
AAR	Application Authorisation Referral
AC	Application Cryptogram
ADF	Application Definition File
AFL	Application File Locator
AID	Application Identifier
ANSI	American National Standards Institute
APDU	Application Protocol Data Unit
ARQC	Authorisation Request Cryptogram
ATC	Application Transaction Counter
b	Binary
BER	Basic Encoding Rules
CDOL	Card Risk Management Data Object List
CVM	Cardholder Verification Method
CVR	Cardholder Verification Rule
DDOL	Dynamic Data Authentication Data Object List
DF	Dedicated File, as defined by ISO 7816-4
FCI	File Control Information
hex.	Hexadecimal
ICC	Integrated Circuit Card
ID	Identifier
IEC	International Electrotechnical Commission
ISO	International Organisation for Standardisation

M	Mandatory
O	Optional
PAN	Primary Account Number
PDOL	Processing Options Data Object List
PIN	Personal Identification Number
RFU	Reserved for Future Use
RID	Registered Application Provider Identifier
SFI	Short File Identifier
SW1	Status Word 1
SW2	Status Word 2
TC	Transaction Certificate
TDOL	Transaction Certificate Data Object List
TLV	Tag Length Value
TSI	Transaction Status Information
TVR	Terminal Verification Results
var.	Variable

The following notations apply:

'0' to '9' and 'A' to 'F'	16 hexadecimal digits
#	Number
[...]	Optional part
xx	Any value

5. Files for Financial Transaction Interchange

The description of the file structure and commands for accessing the files may be found in the Card Specification, as may the definition of each of the data objects. The payment system or issuer will map the appropriate data objects to files according to their needs, subject to the following restrictions:

- All files accessible using the READ RECORD command as defined in the Card Specification containing data objects defined in the Card Specification shall use short file identifiers (SFIs) in the range 1 to 10. These files:

– Shall be linear files readable using the READ RECORD command as described in the Card Specification.

– May contain multiple records. Each record is limited to 254 bytes, including tag and length.

– Each record shall be coded as a constructed data object. The tag of the constructed data object shall be '70' indicating a template proprietary to this specification, and the length field shall contain the total length of the encapsulated data objects.

– Shall contain only data objects defined in this specification and coded in accordance with the Basic Encoding Rules - Tag Length Value (BER-TLV) described in the Card Specification.

– May have access conditions to be satisfied for updates, but must be readable unconditionally.

- Files with SFIs in the range 11 to 20 are reserved for proprietary data to be specified by the individual payment systems.

- Files with SFIs in the range 21 to 30 are reserved for proprietary data to be specified by the issuer.

- The Application File Locator (AFL) described in the Card Specification determines the files and records to be used for processing a transaction. The use of the AFL is described in section 7.2. The data objects listed in Table 1 are used by the offline data authentication algorithm and, when present, should be located in the first record referenced by the AFL.[1]

[1] This allows the terminal to optionally perform the hashing necessary for data authentication in parallel with reading and parsing of data for other purposes.

Tag	Value
'8F'	Certification Authority Public Key Index
'90'	Issuer Public Key Certificate

Table 1 - Data Objects Used by the Offline Data Authentication Algorithm

Additional information may be found in complementary payment system documentation.

5.1 Mandatory Data Objects

Table 2 lists the data objects that must be present in the ICC in files read using the READ RECORD command. All other data objects defined in this specification to be resident in such files in the card are optional.

Tag	Value	Presence
'5F24'	Application Expiration Date	M
'5A'	Application Primary Account Number (PAN)	M
'8C'	Card Risk Management Data Object List 1	M
'8D'	Card Risk Management Data Object List 2	M

Table 2 - Mandatory Data Objects

Table 3 lists the data objects that must be present if the ICC supports offline static data authentication. Table 4 lists the data objects that must be present if the ICC supports offline dynamic data authentication.[2] Offline data authentication is required to support offline transactions but is optional in cards that support only online transactions.

[2] The exception may be that the Issuer Public Key Remainder or the ICC Public Key Remainder could be absent. This is because if the public key modulus can be recovered in its entirety from the public key certificate there is no need for a remainder.

Tag	Value
'8F'	Certification Authority Public Key Index
'90'	Issuer Public Key Certificate
'93'	Signed Static Application Data
'92'	Issuer Public Key Remainder
'9F32'	Issuer Public Key Exponent

Table 3 - Data Required for Offline Static Data Authentication

Tag	Value
'8F'	Certification Authority Public Key Index
'90'	Issuer Public Key Certificate
'92'	Issuer Public Key Remainder
'9F32'	Issuer Public Key Exponent
'9F46'	ICC Public Key Certificate
'9F47'	ICC Public Key Exponent
'9F48'	ICC Public Key Remainder
'9F49'	Dynamic Data Authentication Data Object List (DDOL)

Table 4 - Data Required for Offline Dynamic Data Authentication

5.2 Data Retrievable by GET DATA Command

Data objects listed in Table 5 are not retrievable by the READ RECORD command but are retrieved by the terminal using the GET DATA command as described in the Card Specification.

Of the objects listed here, only the Application Transaction Counter (ATC) is a mandatory data object, and it can be retrieved by either the GET DATA command or in the response to a GENERATE APPLICATION CRYPTOGRAM (AC) command. The terminal retrieves the ATC via the GET DATA command only if the ICC contains the Lower Consecutive Offline Limit and Upper Consecutive Offline Limit

data objects. If the issuer does not wish terminal velocity checking to be performed and omits these data objects, the ICC does not need to support the GET DATA command.

Tag	Value	Presence
'9F36'	Application Transaction Counter (ATC)	M
'9F17'	PIN Try Counter	O
'9F13'	Last Online ATC Register	O

Table 5 - Data Objects Retrievable by GET DATA Command

5.3 Data Retrievable by Get Processing Options

Data objects listed in Table 6 are not retrievable by the READ RECORD command but are retrieved by the terminal using the GET PROCESSING OPTIONS command as described in the Card Specification. Table 6 defines the data returned, not the format of the response; the Card Specification describes the format of the data when returned by the GET PROCESSING OPTIONS command.

Tag	Value	Presence
'82'	Application Interchange Profile	M
'94'	Application File Locator	M

Table 6 - Data Retrievable by GET PROCESSING OPTIONS

6. Transaction Flow

The Application Interchange Profile specifies the application functions that are supported by the card. The terminal shall attempt to execute only those functions that the ICC supports.

6.1 Exception Handling

Exceptions to normal processing are described in the Application Specification for specific status codes returned in the status bytes (SW1, SW2) or for missing data. Unless otherwise specified in the Application Specification, any SW1 SW2 returned by the transport layer to the application layer other than '9000', '63Cx', or '6283' shall cause termination of the transaction.[3] This requirement applies to the Application Specification but does not apply to the application selection process.

6.2 Example Flowchart

The flowchart in Figure 1 gives an example of a transaction flow that may be used by a terminal for a normal purchase transaction. This flowchart is only an example, and the order of processing may differ from that given here. All restrictions on the order of processing are provided in section 7.

[3] Other actions may be taken by prior agreement but are outside the scope of this specification.

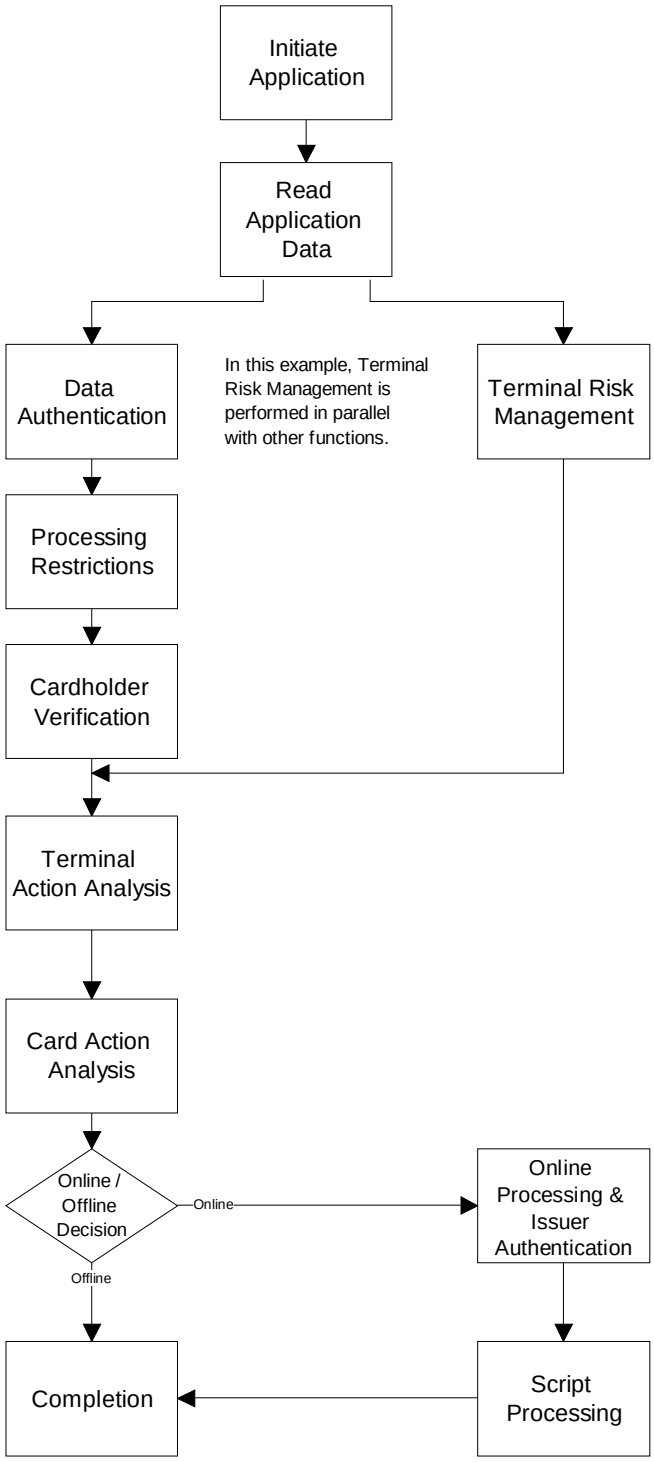

Figure 1 - Transaction Flow Example

6.3 Additional Functions

Provision has been made in this specification for additional functions beyond those described here. Such additional functions may be:

- Future additions to this specification
- Proprietary functions implemented by local or national agreement or by the individual payment systems

The Application Interchange Profile indicates the functions supported in the ICC according to this specification. Most of the bits in this data object are reserved for future use (RFU). When a new function is added, a bit in the Application Interchange Profile will be allocated to indicate support for the new function, and this specification will be updated to specify the new function and where it fits into the transaction flow.

Proprietary functions may be added to the terminal and the ICC application as long as they do not interfere with processing of terminals and ICCs not implementing the function. For example, offline dynamic data authentication based on symmetric keys may be added at local option. Such proprietary functions, while not described in this specification, are not precluded, as long as the functions specified herein continue to be supported for ICCs not implementing the proprietary functions.

7. Functions Used in Transaction Processing

The Card Specification describes all functionality outside the application layer, including the selection of the application. The functions described here begin after application selection has taken place.

The remainder of this section deals with the terminal-to-ICC dialogue on the level of the application logical functions.

7.1 Initiate Application Processing

Purpose: Inform the ICC that the processing of a new transaction is beginning and provide to the ICC the terminal information about the transaction. Obtain from the ICC the Application Interchange Profile and a list of files that contain the ICC data to be used in processing the transaction, and determine whether the transaction is allowed.

Conditions of Execution: This function shall always be executed by the terminal.

Sequence of Execution: This is the first function performed after application selection.

Description: The terminal sets all bits in the Transaction Status Information (TSI) and the Terminal Verification Results (TVR) to '0'.[4]

The Processing Options Data Object List (PDOL) is a list of tags and lengths of terminal-resident data elements needed by the ICC in processing the GET PROCESSING OPTIONS command. Only data elements identified in the Card Specification as having the terminal as the source of the data may be referenced in the PDOL.

If the PDOL does not exist, the GET PROCESSING OPTIONS command uses a command data field of '8300', indicating that the length of the value field in the command data is 0.

If the PDOL exists, the terminal extracts the PDOL from the FCI of the ADF and uses it to create a concatenated list of data elements without tags or lengths. The rules specified in Part II of the Card Specification (see 'Rules for Processing a Data Object List (DOL)') apply to processing of the PDOL. If an amount field (either Amount, Authorised or Amount, Other) is referenced in the PDOL and the terminal

[4] There may be some exceptions in the timing for this. For example, these bits could be set to '0' at the completion of the previous transaction or prior to application selection of this transaction. The intent here is that the processing steps as described in the Application Specification presume the bits have been initialised to '0'.

is unable to provide the amount at this point in transaction processing, the amount field in the data element list shall be filled with hexadecimal zeroes.

The terminal issues the GET PROCESSING OPTIONS command using either the command data field of '8300' (if there was no PDOL in the ICC) or a data object constructed with a tag of '83' and the appropriate length according to BER-TLV encoding rules and a value field that is the concatenated list of data elements resulting from processing the PDOL. The card returns either:

- The Application Interchange Profile, the Application File Locator (identifying the files and records containing the data to be used for the transaction), and status SW1 SW2 = '9000', or
- Status SW1 SW2 = '6985' ('Conditions of use not satisfied'), indicating that the transaction cannot be performed with this application.

The format of the response message is given in the Card Specification.

If the status words '6985' are returned, the terminal shall eliminate the current application from consideration and return to the application selection function to select another application.

7.2 Read Application Data

Purpose: Data contained in files in the ICC are required by the terminal to perform the various functions used in transaction processing as described in this section. The terminal must read this data from the ICC.

Conditions of Execution: This function shall always be executed by the terminal.

Sequence of Execution: The read application data function is performed immediately following the initiate application function.

Description: The terminal shall read the files and records indicated in the Application File Locator using the READ RECORD command identifying the file by its SFI. If an error prevents the terminal from reading data from the ICC, the transaction shall be terminated (see section 6.1).

The AFL is a list identifying the files and records to be used in the processing of a transaction. The terminal is to read only the records named in the AFL. Each element of the list corresponds to a file to be read and is structured as follows:

- The first byte codes the SFI in the five most significant bits. The three least significant bits of the first byte shall be set to zero.
- The second byte codes the first (or only) record number to be read for that SFI. The second byte shall never be set to zero.

- The third byte codes the last record number to be read for that SFI. Its value is either greater than or equal to the second byte. When the third byte is greater than the second byte, all the records ranging from the record number in the second byte to and including the record number in the third byte shall be read for that SFI. When the third byte is equal to the second byte, only the record number coded in the second byte shall be read for that SFI.

- The fourth byte codes the number of records involved in offline data authentication starting with the record number coded in the second byte. The fourth byte may range from zero to the value of the third byte less the value of the second byte plus 1.

The terminal shall process each entry in the AFL from left to right. A READ RECORD command as described in the Card Specification shall be issued for each record between the starting record number and the ending record number, inclusively. Any SW1 SW2 other than '9000' passed to the application layer as a result of reading any record shall cause the transaction to be terminated. Records specified in the AFL to be included in offline data authentication shall be processed as described in section 7.3.

The terminal shall store all recognised data objects read, whether mandatory or optional, for later use in the transaction processing. Data objects that are not recognised by the terminal (that is, their tags are unknown by the terminal) shall not be stored, but records containing such data objects may still participate in their entirety in offline data authentication, depending upon the coding of the AFL.

All mandatory data objects shall be present in the card. If any mandatory data objects are not present, the terminal shall terminate the transaction.

Redundant primitive data objects are not permitted. If the terminal encounters more than one occurrence of a single primitive data object while reading data from the ICC, the transaction shall be terminated.

Proprietary data files may or may not conform to this specification. Records in proprietary files may be represented in the AFL and may participate in offline data authentication provided that they are readable without conditions by the READ RECORD command coded according to the Card Specification. Otherwise, the reading and processing of proprietary files is beyond the scope of this specification.

7.3 Offline Data Authentication

Purpose: Offline data authentication is performed as specified in the Card Specification. This specification describes how it is determined whether offline data authentication will be performed, what kind of authentication will be performed, and how the success or failure of authentication affects the transaction flow and data recorded in the TVR and TSI.

Conditions of Execution: Availability of data in the ICC to support offline data authentication is optional; its presence is indicated in the Application Interchange

Profile. If both the terminal and the ICC support offline data authentication, the terminal shall perform this function. Depending on the capabilities of the card and the terminal, either offline static data authentication or dynamic data authentication may be performed but not both.

If both of the following are true, the terminal shall perform offline dynamic data authentication as specified in the Card Specification:

- The Application Interchange Profile indicates that the card supports offline dynamic data authentication.
- The terminal supports offline dynamic data authentication.

If all of the following are true, the terminal shall perform offline static data authentication as specified in the Card Specification:

- The Application Interchange Profile indicates that the card supports offline static data authentication.
- The terminal supports offline static data authentication.
- Either the card or the terminal (or both) does not support offline dynamic data authentication.

If neither offline static data authentication nor offline dynamic data authentication is performed, the terminal shall set the 'Offline data authentication was not performed' bit to '1' in the TVR.

Sequence of Execution: The terminal shall perform offline data authentication in any order after application initiation but prior to completion of the terminal action analysis.

Description: Offline static data authentication authenticates static data put into the card by the issuer. Offline dynamic data authentication authenticates ICC-resident data, data from the terminal, and the card itself.

Input to the authentication process is formed from the records identified by the AFL, followed by the data elements identified by the optional Static Data Authentication Tag List (tag '9F4A').

Only those records identified in the AFL as participating in offline data authentication are to be processed. Records are processed in the same sequence in which they appear within AFL entries. The records identified by a single AFL entry are to be processed in record number sequence. The first record begins the input for the authentication process, and each succeeding record is concatenated at the end of the previous record.

The data from each record to be included in the offline data authentication input depends upon the SFI of the file from which the record was read.

- For files with SFI in the range 1 to 10, the record tag ('70') and the record length are excluded from the offline data authentication process. All other data in the data field of the response to the READ RECORD command (excluding SW1 SW2) is included.

- For files with SFI 11-30, all data in the data field (excluding SW1 SW2) is included.

The bytes of the record are included in the concatenation in the order in which they appear in the command response.

After all records identified by the AFL have been processed, the Static Data Authentication Tag List is processed, if it exists. Each entry in the list is processed, from left to right. Each entry consists of a tag representing a data element whose source is the ICC; data element lengths are not included in the tag list. The following rules apply to processing of the tag list:

- Only tags defined in the Card Specification and with the ICC as the source are permitted.

- All tags must represent data elements available in the current transaction.

- Tags must not refer to a constructed data object.

- The value field of the data object identified by the tag is to be concatenated to the current end of the input string. Tags and lengths of the data objects identified in the tag list are not included in the concatenation.

Building of the input list for offline data authentication is considered the first step in the offline data authentication process. If the input cannot be built because of a violation of one of the above rules but offline data authentication should be performed according to the 'Conditions of Execution' above, offline data authentication shall be considered to have been performed and to have failed; that is, the terminal shall set to '1' the 'Offline data authentication was performed' bit in the TSI and the appropriate 'Offline static data authentication failed' or 'Offline dynamic authentication failed' bit shall be set in the TVR.

See Part IV of the Card Specification for additional steps to be performed for offline data authentication. If offline static data authentication is performed but is unsuccessful, the 'Offline static data authentication failed' bit shall be set to '1' in the TVR, otherwise it shall be set to '0'. If offline dynamic data authentication is performed but is unsuccessful, the 'Offline dynamic data authentication failed' bit shall be set to '1' in the TVR, otherwise it shall be set to '0'.

Upon completion of the offline data authentication function, the terminal shall set to '1' the 'Offline data authentication was performed' bit in the TSI.

7.4 Processing Restrictions

Purpose: The purpose of the processing restrictions function is to determine the degree of compatibility of the application in the terminal with the application in the ICC and to make any necessary adjustments, including possible rejection of the transaction.

Conditions of Execution: This function shall always be executed by the terminal.

Sequence of Execution: Functions described here may be performed at any time after application selection and prior to completion of the terminal action analysis.

Description: The processing restrictions function is described as comprising the following compatibility checks:

- Application Version Number
- Application Usage Control
- Application Effective/Expiration Dates Checking

7.4.1 Application Version Number

The application within both the terminal and the ICC shall maintain an Application Version Number assigned by the payment system. The terminal shall use the version number in the ICC to ensure compatibility. If the Application Version Number is not present in the ICC, the terminal shall presume the terminal and ICC application versions are compatible, and transaction processing shall continue. If the Application Version Number is present in the ICC, it shall be compared to the Application Version Number maintained in the terminal. If they are different, the 'ICC and terminal have different application versions' bit shall be set to '1' in the TVR.

7.4.2 Application Usage Control

The Application Usage Control indicates restrictions limiting the application geographically or to certain types of transactions. If this data object is present, the terminal shall make the following checks:

- If the transaction is being conducted at an ATM, the 'Valid at ATMs' bit must be on in Application Usage Control.
- If the transaction is not being conducted at an ATM, the 'Valid at terminals other than ATMs' bit must be on in Application Usage Control.

If the Application Usage Control and Issuer Country Code are both present in the ICC, the terminal shall make the following checks:

- If the Transaction Type indicates that this is a cash transaction and the Issuer Country Code matches the Terminal Country Code, the 'Valid for domestic cash transactions' bit must be on in Application Usage Control.

- If the Transaction Type indicates that this is a cash transaction and the Issuer Country Code does not match the Terminal Country Code, the 'Valid for international cash transactions' bit must be on in Application Usage Control.

- If the Transaction Type indicates a purchase of goods and the Issuer Country Code matches the Terminal Country Code, the 'Valid for domestic goods' bit must be on in Application Usage Control.

- If the Transaction Type indicates a purchase of goods and the Issuer Country Code does not match the Terminal Country Code, the 'Valid for international goods' bit must be on in Application Usage Control.

- If the Transaction Type indicates a purchase of services and the Issuer Country Code matches the Terminal Country Code, the 'Valid for domestic services' bit must be on in Application Usage Control.

- If the Transaction Type indicates a purchase of services and the Issuer Country Code does not match the Terminal Country Code, the 'Valid for international services' bit must be on in Application Usage Control.

- If the transaction has a cashback amount[5] and the Issuer Country Code matches the Terminal Country Code, the 'Domestic cashback allowed' bit must be on in Application Usage Control.

- If the transaction has a cashback amount[5] and the Issuer Country Code does not match the Terminal Country Code, the 'International cashback allowed' bit must be on in Application Usage Control.

If any of the above tests fail, the 'Requested service not allowed for card product' bit shall be set to '1' in the TVR.

7.4.3 Application Effective/Expiration Dates Checking

If the Application Effective Date is present in the ICC, the terminal shall check that the current date is greater than or equal to the Application Effective Date. If it is not, the 'Application not yet effective' bit shall be set to '1' in the TVR.

[5] It is preferred that if the relevant 'Domestic cashback allowed' or 'International cashback allowed' bit is set to '0' in the Application Usage Control, the cashback option should not be offered to the cardholder. However, it is possible for some terminals that the decision regarding cashback may have been made before application selection.

The terminal shall check that the current date is less than or equal to the Application Expiration Date. If it is not, the 'Expired application' bit shall be set to '1' in the TVR.

7.5 Cardholder Verification

Purpose: Cardholder verification is performed to ensure that the person presenting the ICC is the person to whom the application in the card was issued.

Conditions of Execution: Ability of the ICC to support at least one cardholder verification method is indicated in the Application Interchange Profile. If this bit is set to '1', the terminal shall use the cardholder verification related data in the ICC to determine whether one of the issuer-specified cardholder verification methods (CVMs) shall be executed. This process is described below.

Sequence of Execution: This function may be performed any time after application selection and before completion of the terminal action analysis.

Description: The CVM List (tag '8E') is a composite data object consisting of the following:

1. An amount field (4 bytes, binary format), referred to as 'X' in the CVM Condition Codes (see Table A-4 in Annex A). 'X' is expressed in the Application Currency Code with implicit decimal point. For example, 123 (hexadecimal '7B') represents £1.23 when the currency code is '826'.

2. A second amount field (4 bytes, binary format), referred to as 'Y' in the CVM Condition Codes (see Table A-4 in Annex A). 'Y' is expressed in Application Currency Code with implicit decimal point. For example, 123 (hexadecimal '7B') represents £1.23 when the currency code is '826'.

3. A variable-length list of two-byte data elements called Cardholder Verification Rules (CVRs). Each CVR describes a CVM and the conditions under which that CVM should be applied (See Table A-3 and Table A-4 in Annex A).

If the CVM List is not present in the ICC, the terminal shall terminate cardholder verification without setting the 'Cardholder verification was performed' bit in the TSI. If the CVM List is present in the ICC, the terminal shall process each rule in the order in which it appears in the list according to the following specifications. Cardholder verification is completed when any one CVM is successfully performed or when the list is exhausted.

If any of the following are true:

- The conditions expressed in the second byte of a CVR are not satisfied, or

- The ICC data required by the condition (for example, the Application Currency Code) is not present, or

- The CVM Condition Code is outside the range of codes understood by the terminal (which might occur if the terminal application program is at a different version level than the ICC application),

the terminal shall bypass the rule and proceed to the next. If there are no more CVRs in the list, cardholder verification has not been successful, and the terminal shall set the 'Cardholder verification was not successful' bit to '1' in the TVR.

If the conditions expressed in the second byte of a CVR are satisfied, the terminal shall attempt to perform the CVM if the CVM code is one of those listed in Annex A or is otherwise understood by the terminal. If the CVM is not among those listed and is not understood by the terminal, the terminal shall set the 'Unrecognised CVM' bit to '1' in the TVR.

If the CVM is performed successfully, cardholder verification is complete and successful. Otherwise, the terminal shall then examine b7 of byte 1 of the CVM field. If b7 is set to '1', processing continues with the next CVR, if one is present. If b7 is set to '0', or there are no more CVRs in the list, the terminal shall set the 'Cardholder verification was not successful' bit to '1' in the TVR and cardholder verification is complete.

When cardholder verification is completed, the 'Cardholder verification was performed' bit in TSI shall be set to '1'.

7.5.1 Offline PIN Processing

This section applies to the verification by the ICC of a plaintext or enciphered PIN presented by the terminal. cc>

cc>

If an offline PIN is the selected CVM as determined by the above process, offline PIN processing may not be successfully performed for any one of the following reasons:

- The terminal does not support offline PIN[6]. In this case, the terminal shall set to '1' the 'PIN entry required and PIN pad not present or not working' bit in the TVR. The terminal supports offline PIN, but the PIN pad is malfunctioning. In this case, the terminal shall set to '1' the 'PIN entry required and PIN pad not present or not working' bit in the TVR.

[6] This means that the terminal does not support either offline plaintext PIN verification or offline enciphered PIN verification. If the terminal supports at least one of these functions, it is considered to support offline PIN for the purposes of setting the TVR bits.

- The terminal bypassed PIN entry at the direction of either the merchant or the cardholder. In this case, the 'PIN entry required, PIN pad present, but PIN was not entered' bit shall be set to '1' in the TVR.

- The PIN is blocked upon initial use of the VERIFY command (the ICC returns SW1 SW2 = '6983' or '6984' in response to the VERIFY command). In this case, the 'PIN Try Limit exceeded' bit shall be set to '1' in the TVR.

- The number of remaining PIN tries is reduced to zero (indicated by an SW1 SW2 of '63C0' in the response to the VERIFY command). In this case, the 'PIN Try Limit exceeded' bit shall be set to '1' in the TVR.

The only case in which offline PIN processing is considered successful is when the ICC returns an SW1 SW2 of '9000' in response to the VERIFY command.

7.5.2 Online PIN Processing

If online PIN processing is a required CVM as determined by the above process, the processing may not be successfully performed for any one of the following reasons:

- The terminal does not support online PIN. In this case, the terminal shall set to '1' the 'PIN entry required and PIN pad not present or not working' bit in the TVR.

- The terminal supports online PIN, but the PIN pad is malfunctioning. In this case, the terminal shall set to '1' the 'PIN entry required and PIN pad not present or not working' bit in the TVR.

- The terminal bypassed PIN entry at the direction of either the merchant or the cardholder. In this case, the 'PIN entry required, PIN pad present, but PIN was not entered' bit shall be set to '1' in the TVR.

If the online PIN is successfully entered, the terminal shall set to '1' the 'Online PIN entered' bit in the TVR. In this case, cardholder verification is considered successful and complete.

7.5.3 Signature Processing

If a (paper) signature is a required CVM as determined by the above process, the terminal shall determine success based upon the terminal's capability to support the signature process (see complementary payment systems documentation for additional information). If the terminal is able to support signature, the process is considered successful, and cardholder verification is complete.

7.5.4 Combination CVMs

Some CVMs require multiple verification methods (for example, offline PIN plus signature). For these CVMs, all methods in the CVM must be successful for cardholder verification to be considered successful.

7.6 Terminal Risk Management

Purpose: Terminal risk management is that portion of risk management performed by the terminal to protect the acquirer, issuer, and system from fraud. It provides positive issuer authorisation for high-value transactions and ensures that transactions initiated from ICCs go online periodically to protect against threats that might be undetectable in an offline environment. The result of terminal risk management is the setting of appropriate bits in the TVR.

Conditions of Execution: Terminal risk management shall be performed if the 'Terminal risk management is supported' bit is set to '1' in the Application Interchange Profile. Random transaction selection need not be performed by a terminal with no online capability. If terminal risk management is not performed, sections 7.6.1 through 7.6.3 do not apply.

Sequence of Execution: Terminal risk management may be performed at any time prior to issuing the first GENERATE AC command.

Description: Terminal risk management consists of:

- Floor limit checking
- Random transaction selection
- Velocity checking

7.6.1 Floor Limits

To prevent split sales, the terminal may have a transaction log of approved transactions stored in the terminal consisting of at least the Application PAN and transaction amount and possibly the Application PAN Sequence Number and Transaction Date. The number of transactions to be stored and maintenance of the log is outside the scope of this specification, although to prevent split sales the number of transactions stored may be quite small.

During terminal risk management floor limit checking, the terminal checks the transaction log (if available) to determine if there is a log entry with the same Application PAN, and, optionally, the same Application PAN Sequence Number. If there are several log entries with the same PAN, the terminal selects the most recent entry. The terminal adds the Amount, Authorised for the current transaction to the amount stored in the log for that PAN to determine if the sum exceeds the Terminal Floor Limit. If the sum is greater than or equal to the Terminal Floor Limit, the terminal sets the 'Transaction exceeds floor limit' bit to '1' in the TVR.

If the terminal does not have a transaction log available or if there is no log entry with the same PAN, the Amount, Authorised is compared to the appropriate floor limit. If the amount authorised is equal to or greater than the floor limit, the 'Transaction exceeds floor limit' bit is set to '1' in the TVR.

7.6.2 Random Transaction Selection

For each application the relevant payment system specifies, in addition to the floor limit:

- Target Percentage to be Used for Random Selection (in the range of 0 to 99)
- Threshold Value for Biased Random Selection (which must be zero or a positive number less than the floor limit)
- Maximum Target Percentage to be Used for Biased Random Selection (also in the range of 0 to 99 but at least as high as the previous Target Percentage for Random Selection). This is the desired percentage of transactions 'just below' the floor limit that will be selected by this algorithm.

Any transaction with a transaction amount less than the Threshold Value for Biased Random Selection will be subject to selection at random without further regard for the value of the transaction. The terminal shall generate a random number in the range of 1 to 99. If this random number is less than or equal to the Target Percentage to be Used for Random Selection, the transaction shall be selected.

Any transaction with a transaction amount equal to or greater than the Threshold Value for Biased Random Selection but less than the floor limit will be subject to selection with bias toward sending higher value transactions online more frequently (biased random selection). For these transactions, the terminal shall compare its generated random number against a Transaction Target Percent, which is a linear interpolation of the target percentages provided by the payment system (Target Percentage to be Used for Random Selection, and Maximum Target Percentage to be Used for Biased Random Selection).[7] If the random number is less than or equal to the Transaction Target Percent, the transaction shall be selected.

The probability of selection as a function of the transaction amount may be charted as shown in Figure 2:

---

[7] The Transaction Target Percent is calculated as follows:

$$\text{Interpolation factor} = \frac{\text{Amount, Authorised} - \text{Threshold Value}}{\text{Floor Limit} - \text{Threshold Value}}$$

$$\text{Transaction Target Percent} = \left(\left(\text{Maximum Target Percent} - \text{Target Percent}\right) \times \text{Interpolation factor}\right) + \text{Target Percent}$$

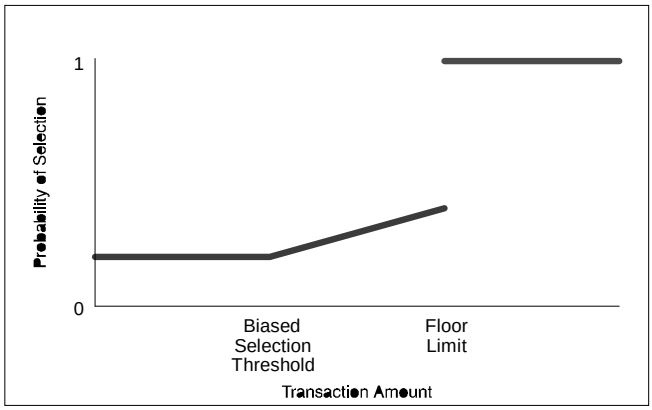

Figure 2 - Random Selection Probability (not to scale)

If the transaction is selected through the process described in this section, the 'Transaction selected randomly for online processing' bit shall be set to '1' in the TVR.

7.6.3 Velocity Checking[8]

If both the Lower Consecutive Offline Limit (tag '9F14') and Upper Consecutive Offline Limit (tag '9F23') exist, the terminal shall perform velocity checking as described in this section. If either of these data objects is not present in the ICC application, the terminal shall skip this section.

The ATC and Last Online ATC Register shall be read from the ICC using GET DATA commands. If either of the required data objects are not returned by the ICC in response to the GET DATA command, the terminal shall:

- Set both the 'Lower consecutive offline limit exceeded' and the 'Upper consecutive offline limit exceeded' bits to '1' in the TVR.
- Not set the 'New card' indicator in the TVR.
- End velocity checking for this transaction.

[8] The purpose of velocity checking is to allow an issuer to request that, after a certain number of consecutive offline transactions (the Lower Consecutive Offline Limit), transactions should be completed online. However, if the terminal is incapable of going online, transactions may still be completed offline until a second (Upper Consecutive Offline Limit) limit is reached. After the upper limit is reached, the recommendation of the issuer might be to reject any transaction that cannot be completed online. Once a transaction has been completed online with successful issuer authentication, the count begins anew, so that transactions may be processed offline until the lower limit is once again reached.

If the required data objects are available, the terminal shall compare the difference between the ATC and the Last Online ATC Register with the Lower Consecutive Offline Limit to see if the limit has been exceeded. If the difference is equal to the Lower Consecutive Offline Limit, this means that the limit has not yet been exceeded. If the limit has been exceeded, the terminal shall set the 'Lower consecutive offline limit exceeded' bit to '1' in the TVR and also compare the difference with the Upper Consecutive Offline Limit to see if the upper limit has been exceeded. If it has, the terminal shall set the 'Upper consecutive offline limit exceeded' bit to '1' in the TVR.

The terminal shall also check the Last Online ATC Register for a zero value. If it is zero, the terminal shall set the 'New card' bit to '1' in the TVR.

Upon completion of terminal risk management, the terminal shall set to '1' the 'Terminal risk management was performed' bit in the TSI.

7.7 Terminal Action Analysis

Purpose: Once terminal risk management and application functions related to a normal offline transaction have been completed, the terminal makes the first decision as to whether the transaction should be approved offline, declined offline, or transmitted online. If the outcome of this decision process is to proceed offline, the terminal issues a GENERATE AC command to ask the ICC to return a TC. If the outcome of the decision is to go online, the terminal will issue a GENERATE AC command to ask the ICC for an Authorisation Request Cryptogram (ARQC). If the decision is to reject the transaction, the terminal will issue a GENERATE AC to ask for an Application Authentication Cryptogram (AAC).

An offline decision made here is not final. If the terminal asks for a TC from the ICC, the ICC, as a result of card risk management, may return an ARQC or AAC.

Conditions of Execution: The terminal action analysis function is always performed.

Sequence of Execution: The terminal action analysis function is performed after terminal risk management and cardholder- and merchant-entered transaction data has been completed. It shall be performed prior to the first use of the GENERATE AC command.

The Issuer Action Code - Default and Terminal Action Code - Default processing described below shall also be performed after online processing is attempted in the case where the terminal was unable to process the transaction online.

The terminal action analysis function may be executed at several places during a transaction to eliminate the need for unnecessary processing. If any processing results in the setting of a bit in the TVR (for example, failure of cardholder verification), it may be desirable to perform this function immediately to determine whether the transaction should be rejected offline based upon the issuer's parameters in the ICC or the acquirer's parameters in the terminal. Recognition of

such a decision early in processing may allow the terminal to avoid prolonging a transaction that will ultimately be rejected. Multiple executions of this decision process is optional on the part of the terminal.

Description: The terminal shall make a preliminary decision to reject the transaction, complete it online, or complete it offline based upon the TVR, issuer action preferences, and acquirer action preferences according to the method described in this section.

The ICC contains (optionally) three data elements to reflect the issuer's selected action to be taken based upon the content of the TVR. Each of the three data elements has defaults specified here in case any of these data elements are absent from the ICC. The three data elements are:

- Issuer Action Code - Denial
- Issuer Action Code - Online
- Issuer Action Code - Default

Collectively, these three data objects are termed the Issuer Action Codes. The purpose of each is described in this section below. The format of each is identical and mirrors the TVR. Each has one bit corresponding to each bit in the TVR, and the Issuer Action Code bit specifies an action to be taken if the corresponding bit in the TVR is set to '1'. Thus, the size and format of each of the Issuer Action Codes is identical to the TVR.

Similarly, the terminal may contain three data elements to reflect the acquirer's selected action to be taken based upon the content of the TVR. These data elements are:

- Terminal Action Code - Denial
- Terminal Action Code - Online
- Terminal Action Code - Default

Collectively, these three data objects are termed the Terminal Action Codes. The purpose of each is described in this section below. The format of each is identical and mirrors the TVR. Each has one bit corresponding to each bit in the TVR, and the Terminal Action Code bit specifies an action to be taken if the corresponding bit in the TVR is set to '1'. Thus, the size and format of each of the Terminal Action Codes is identical to the TVR and to the Issuer Action Codes.

The existence of each of the Terminal Action Codes is optional. In the absence of any Terminal Action Code, a default value consisting of all bits set to '0' is to be used in its place. However, it is strongly recommended that as a minimum, the Terminal Action Code - Online and Terminal Action Code - Default should be included with the bits corresponding to 'Offline data authentication was not performed', 'Offline

static data authentication failed', and 'Offline dynamic data authentication failed' set to '1'.[9]

Processing of the action codes is done in pairs, that is, the Issuer Action Code - Denial is processed together with the Terminal Action Code - Denial, the Issuer Action Code - Online is processed together with the Terminal Action Code - Online, and the Issuer Action Code - Default is processed together with the Terminal Action Code - Default. Processing of the action codes shall be performed in the order specified here.

If the Issuer Action Code - Denial does not exist, a default value with all bits set to '0' is to be used. Together, the Issuer Action Code - Denial and the Terminal Action Code - Denial specify the conditions that cause denial of a transaction without attempting to go online. If either data object exists, the terminal shall inspect each bit in the TVR. For each bit in the TVR that has a value of '1', the terminal shall check the corresponding bits in the Issuer Action Code - Denial and the Terminal Action Code - Denial. If the corresponding bit in either of the action codes is set to '1', it indicates that the issuer or the acquirer wishes the transaction to be rejected offline. In this case, the terminal shall issue a GENERATE AC command to request from the ICC an AAC. This AAC may be presented to the issuer to prove card presence during this transaction, but details of handling a rejected transaction are outside the scope of this specification.

If the Issuer Action Code - Online is not present, a default value with all bits set to '1' shall be used in its place. Together, the Issuer Action Code - Online and the Terminal Action Code - Online specify the conditions that cause a transaction to be completed online. These data objects are meaningful only for terminals capable of online processing. Offline-only terminals may skip this test and proceed to checking the Issuer Action Code - Default and Terminal Action Code - Default, described below. For a terminal capable of online processing, if the terminal has not already decided to reject the transaction as described above, the terminal shall inspect each bit in the TVR. For each bit in the TVR that has a value of '1', the terminal shall check the corresponding bits in both the Issuer Action Code - Online and the Terminal Action Code - Online. If the bit in either of the action codes is set to '1', the terminal shall complete transaction processing online and shall issue a GENERATE AC command requesting an ARQC from the ICC.

If the Issuer Action Code - Default does not exist, a default value with all bits set to '1' shall be used in its place. Together, the Issuer Action Code - Default and the Terminal Action Code - Default specify the conditions that cause the transaction to be rejected if it might have been approved online but the terminal is for any reason unable to process the transaction online. The Issuer Action Code - Default and the Terminal Action Code - Default are used only if the Issuer Action Code - Online and

[9] This protects against a fraudulent card with all the bits in the Issuer Action Code set to '0'. Without this protection, such a card could be created with no possibility of going online or declining transactions. All transactions would be approved offline.

the Terminal Action Code - Online were not used (for example, in case of an offline-only terminal) or indicated a desire on the part of the issuer or the acquirer to process the transaction online but the terminal was unable to go online. If the terminal has not already rejected the transaction and the terminal is for any reason unable to process the transaction online, the terminal shall use this code to determine whether to approve or reject the transaction offline. If any bit in Issuer Action Code - Default or the Terminal Action Code - Default and the corresponding bit in the TVR are both set to '1', the transaction shall be rejected and the terminal shall request an AAC to complete processing. If no such condition appears, the transaction may be approved offline, and a GENERATE AC command shall be issued to the ICC requesting a TC.

7.8 Card Action Analysis

Purpose: An ICC may perform its own risk management to protect the issuer from fraud or excessive credit risk. Details of card risk management algorithms within the ICC are specific to the issuer and are outside the scope of this specification, but as a result of the risk management process, an ICC may decide to complete a transaction online or offline or request a referral or reject the transaction. The ICC may also decide that an advice message should be sent to the issuer to inform the issuer of an exceptional condition.

Conditions of Execution: The card online/offline decision is specified by its response to the GENERATE AC command. Therefore, this section applies to all transactions. Whether the ICC performs any risk management tests is transparent to the terminal and outside the scope of this specification.

Sequence of Execution: The card action analysis process is performed when the terminal issues the GENERATE AC command for a given transaction.

Description: The result of risk management performed by the ICC is a decision for one of the following actions to be taken by the terminal:

- Approve the transaction offline. This option is available to the ICC only if the terminal has made a preliminary decision to complete the transaction offline, as described in section 7.7.
- Complete the transaction online.
- Request a referral.
- Reject the transaction.

The decision by the ICC is made known to the terminal by returning either a TC, an ARQC, an AAR, or an AAC to the terminal in response to a GENERATE AC command, as described in section 8.

Upon the completion of the card action analysis function, the terminal shall set to '1' the 'Card risk management was performed' bit in the TSI.

7.8.1 Terminal Messages for an AAC

An AAC returned by the card indicates either a rejection of the specific transaction or a restriction that disallows use of the card in the environment of the transaction (for example, the card application may be restricted only to specific merchant categories). In both cases, the card disapproves the transaction, but the terminal may choose to display different messages in the two cases. The card may optionally distinguish the cases by the use of the code returned in the Cryptogram Information Data (see the GENERATE AC command in the Card Specification). If an AAC is returned with b3-b1 = '001' in the Cryptogram Information Data, the AAC was returned due to card restrictions.

7.8.2 Advice Messages

The issuer may wish for an advice message, separate from either an authorisation request or a clearing message, to be sent in certain exception cases. (Currently, the only identified such case is 'PIN Try Limit exceeded', but allowance has been made for the addition of other cases later; see 'Coding of Cryptogram Information Data' in the GENERATE AC command in the Card Specification).

If b4 of the Cryptogram Information Data is '1', the terminal shall format and send an advice message. Further information may be found in complementary payment system documentation and in the Terminal Specification.

7.9 Online Processing

Purpose: Online processing is performed to ensure that the issuer can review and authorise or reject transactions that are outside acceptable limits of risk defined by the issuer, the payment system, or the acquirer.

Conditions of Execution: Online processing shall be performed if the ICC returns an ARQC in response to the first GENERATE AC command for the transaction.

Sequence of Execution: The online processing function is performed when the terminal receives an ARQC in response to the first GENERATE AC command.

Description: In general, online processing is the same as today's online processing of magnetic stripe transactions and is not described here. This section is limited to the additional online processing provided in an ICC environment that is not available in a magnetic stripe environment.

The ARQC may be sent in the authorisation request message.[10] The authorisation response message from the issuer may contain the Issuer Authentication Data (tag

[10] Actions performed by the acquirer or issuer systems are outside the scope of this specification. However, an explanation of what is expected to take place at the issuer may be useful for clarity. The ARQC is a cryptogram generated by the card from transaction

'91'). If the Issuer Authentication Data is received in the authorisation response message and the Application Interchange Profile indicates that the ICC supports issuer authentication, the Issuer Authentication Data shall be sent to the ICC in the EXTERNAL AUTHENTICATE command. If the ICC responds with SW1 SW2 other than '9000', the 'Issuer authentication was unsuccessful' bit shall be set to '1' in the TVR.

If the Issuer Authentication Data is received but the Application Interchange Profile indicates that the ICC does not support issuer authentication, this indicates that the ICC has combined the issuer authentication function with the GENERATE AC command. In this case, or if no Issuer Authentication Data is received, the terminal shall not execute the EXTERNAL AUTHENTICATE command.

The ICC shall permit at most one EXTERNAL AUTHENTICATE command in a transaction. If the terminal issues more than one, the second and all succeeding EXTERNAL AUTHENTICATE commands shall end with SW1 SW2 = '6985'.

Upon completion of online processing, if the EXTERNAL AUTHENTICATE command was sent to the card by the terminal, the terminal shall set to '1' the 'Issuer authentication was performed' bit in the TSI.

7.10 Issuer-to-Card Script Processing

Purpose: An issuer may provide command scripts to be delivered to the ICC by the terminal to perform functions that are not necessarily relevant to the current transaction but are important for the continued functioning of the application in the ICC. Multiple scripts may be provided with an authorisation response, and each may contain any number of Issuer Script Commands. Script processing is provided to allow for functions that are outside the scope of this specification but are nonetheless necessary.[11]

---

data using an issuer key stored in the card and known at the issuer authorisation system. The issuer uses this key to authenticate the ARQC and thereby authenticate the card. This process is termed 'online card authentication' or simply 'card authentication'.

Subsequent to card authentication, the issuer may generate a cryptogram on selected data included in the authorisation response or already known to the card. This cryptogram is sent to the terminal in the authorisation response as part of the Issuer Authentication Data. The terminal provides the Issuer Authentication Data to the ICC in the EXTERNAL AUTHENTICATE command or the second GENERATE AC command, as described in the Card Specification. The ICC may use the Issuer Authentication Data to authenticate that the response message originated from the issuer.

[11] An example might be unblocking of an offline PIN, which might be done differently by various issuers or payment systems.

A script may contain Issuer Script Commands not known to the terminal, but each command shall be delivered by the terminal to the ICC individually according to this specification.

Conditions of Execution: None.

Sequence of Execution: Two separate script tags are defined that are available for use by the issuer. Issuer scripts with tag '71' shall be processed prior to issuing the final GENERATE AC command. Issuer scripts using tag '72' shall be processed after issuing the final GENERATE AC command.

Description: An Issuer Script is a constructed data object (tag '71' or '72') containing (optionally) a Script Identifier and a sequence of Issuer Script Command APDUs to be delivered serially to the ICC. The Script Identifier is optional and is not interpreted by the terminal; it is meaningful only to the issuer. Figure 3 and Figure 4 illustrate an Issuer Script containing a Script Identifier and three commands.

T	L	T	L	Script ID	Commands
'71' or '72'	L(Σdata, including Script ID, tags, and lengths)	'9F18'	'04'	Identifier (4 bytes)	(see Figure 4)

Figure 3 - Issuer Script Format

T_1	L_1	V_1	T_2	L_2	V_2	T_3	L_3	V_3
'86'	$L(V_1)$	Command	'86'	$L(V_2)$	Command	'86'	$L(V_3)$	Command

Figure 4 - Issuer Script Command Format (Shown with Three Commands)

It is possible for multiple Issuer Scripts to be delivered with a single authorisation response. Each Issuer Script shall be processed by the terminal in the sequence in which it appears in the authorisation response according to the following rules:

- Issuer Script Commands shall be separated using the BER-TLV coding of the data objects defining the commands (tag '86').
- Each command will be delivered to the ICC as a command APDU in the sequence in which it appeared in the Issuer Script.
- The terminal shall examine only SW1 in the response APDU and perform one of the following actions:
  - If SW1 indicates either normal processing or a 'warning' according to the conventions described in the Card Specification, the terminal shall continue with the next command from the Issuer Script (if any).

– If SW1 indicates an 'error' condition, the processing of the Issuer Script shall be terminated.

If an Issuer Script is processed, the terminal shall set the 'Script processing was performed' bit in the TSI to '1'. If an error occurred in processing a script, the terminal shall set to '1' either the 'Script processing failed before final GENERATE AC' in the TVR if the identifying tag of the failing script was '71' or the 'Script processing failed after final GENERATE AC' in the TVR if the tag was '72'.

7.11 Completion

Purpose: The completion function closes processing of a transaction.

Conditions of Execution: This function is always performed by the terminal unless the transaction is terminated prematurely by error processing.

Sequence of Execution: The completion function is always the last function in the transaction processing. (Script processing may be performed after the completion function.)

Description: The ICC indicates willingness to complete transaction processing by returning either a TC or an AAC to either the first or second GENERATE AC command issued by the terminal. If the terminal decides to go online, completion shall be done when the second GENERATE AC command is issued.

See section 8 for additional information on the use of the GENERATE AC command.

8. GENERATE AC Command Coding

The GENERATE AC command format and response codes are described fully in the Card Specification. This section describes how the various options and data elements are used in transaction processing. Figure 5 and Figure 6 depict the interaction between the terminal decisions, ICC decisions, issuer approval, the GENERATE AC command, and the possible ICC responses. Figure 5 describes the overall flow; Figure 6 provides the additional logic for referral transactions.

The complete transaction flow is not shown in these charts, only the GENERATE AC commands, responses, and associated decisions.

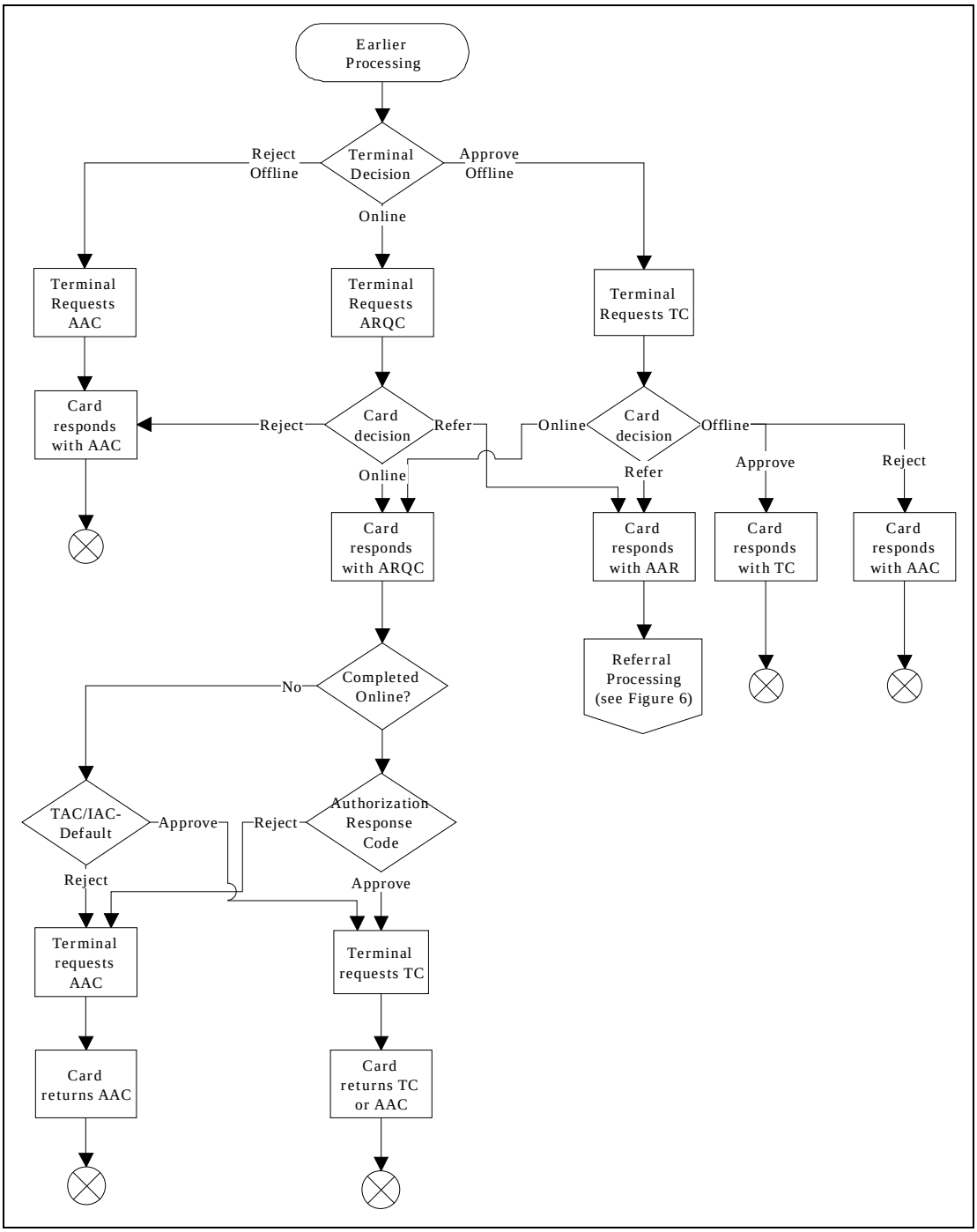

Figure 5 - Use of GENERATE AC Options cc>

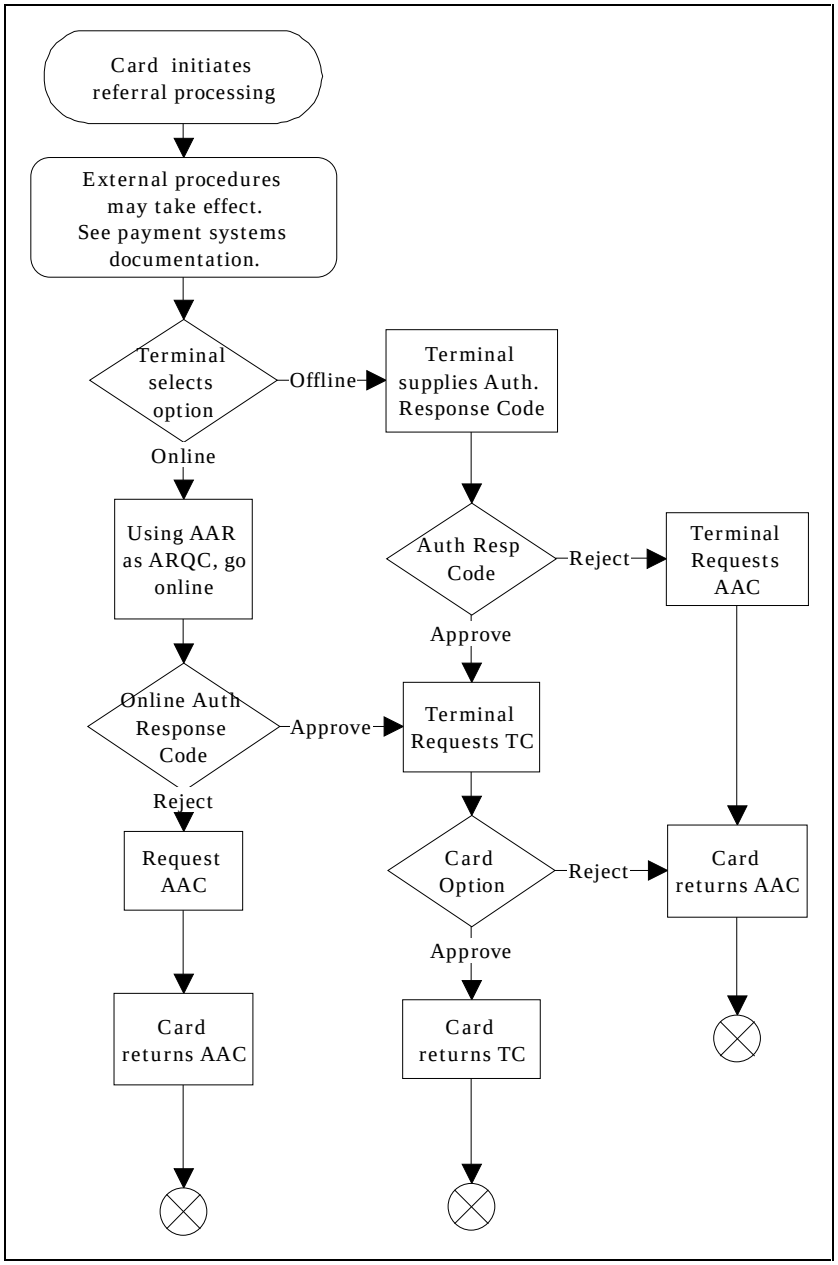

Figure 6 - Use of GENERATE AC with Referrals

8.1 Command Parameters

The GENERATE AC command parameters provide three different options to the terminal:

- Request for the generation of a TC
- Request for the generation of an ARQC
- Request for the generation of an AAC

8.2 Command Data

The data field of the GENERATE AC command is not TLV encoded, so it is imperative that the ICC know the format of this data when the command is received. This is achieved by allowing the ICC to specify the format of the data to be included in the command. This section describes how the ICC makes that specification.

8.2.1 Card Risk Management Data

The Card Risk Management Data Object List (CDOL) is a data object in the ICC that provides to the terminal a list of data objects that must be passed from the terminal to the ICC in the GENERATE AC command. There shall be two CDOLs in the ICC, referred to as CDOL1 (tag '8C') and CDOL2 (tag '8D'). CDOL1 is used with the first GENERATE AC command, and CDOL2 is used with the second GENERATE AC command (if used). The terminal uses the appropriate CDOL and the rules specified in the Card Specification (see 'Rules for Processing Data Object Lists (DOLs)') to build the appropriate data field for the command. It is the responsibility of the terminal to ensure that current data is used in building the GENERATE AC command. To this end, the command data should be built immediately prior to issuing the GENERATE AC command.

8.2.2 Transaction Certificate Data

A CDOL may request a TC Hash Value to be included in the command data of a GENERATE AC command. The TDOL is a data object that provides to the terminal a list of data objects to be used in generating the TC Hash Value. The ICC may contain the TDOL, but there may be a default TDOL in the terminal, specified by the payment system, for use in case the TDOL is not present in the ICC. To create the hash value, the terminal shall use the TDOL (if present) or the default TDOL to create the appropriate value for input to the hash algorithm, applying the rules specified in the Card Specification (see 'Rules for Processing Data Object Lists (DOLs)'). If the default TDOL is used, the terminal shall set to '1' the 'Default TDOL used' bit in the TVR. If a default TDOL is required but is not present in the terminal, a default TDOL with no data objects in the list shall be assumed.

If the terminal issues a second GENERATE AC command during the processing of a transaction, the terminal shall ensure that the data provided in the TC Hash Value is current at the time the command is issued.

8.3 Command Use

Either one or two GENERATE AC commands are issued during the processing of a transaction according to this specification.

The ICC shall respond to the first GENERATE AC command with any of the following:

- TC
- ARQC
- AAR
- AAC

The ICC shall respond to a second GENERATE AC command with either a TC or an AAC.

The possible responses listed above are in hierarchical order, with a TC being the highest and an AAC being the lowest. The terminal may request a TC, an ARQC, or an AAC. If the ICC responds with a cryptogram at a higher level, this indicates a logic error in the ICC. If this occurs after the first GENERATE AC command in a transaction, the transaction shall be terminated. If it occurs after the second GENERATE AC command, all processing for the transaction has been completed, and the cryptogram returned shall be treated as an AAC.

8.3.1 GENERATE AC (First Issuance)

The terminal completes its online/offline decision process with a GENERATE AC command (see Figure 5). The form of the command depends upon the decision made by the terminal:

- If the terminal decides the transaction might be completed offline, it requests a TC from the ICC. The ICC shall reply with a TC, an ARQC, an AAR, or an AAC, depending upon its own analysis of the transaction.
- If the terminal decides the transaction should go online, it requests an ARQC from the ICC. The ICC shall reply with an ARQC, an AAR, or an AAC.
- If the terminal decides to reject the transaction, it requests an AAC from the ICC. The ICC shall reply with an AAC.

If the ICC responds with a TC or an AAC, the terminal completes the transaction offline.

If the ICC responds with an AAR, the terminal either provides an Authorisation Response Code and proceeds to the completion function or uses the AAR to go online. See Figure 6 for referral processing logic.

If the ICC responds with an ARQC, the terminal attempts to go online, sending an authorisation request message to the issuer. Included in the authorisation request message is the ARQC for online card authentication.

8.3.2 GENERATE AC (Second Issuance)

Whether the terminal receives an authorisation response message as a result of online processing or an approval or rejection by using the Issuer Action Code - Default, it completes the transaction by requesting either a TC (in the case an approval was obtained) or an AAC (in case the issuer's instruction is to reject the transaction) from the ICC. If a TC was requested, the ICC shall reply with either a TC or an AAC. If an AAC was requested, the card shall reply with an AAC.

The ICC shall permit at most two GENERATE AC commands in a transaction. If the terminal issues more than two, the third and all succeeding GENERATE AC commands shall end with SW1 SW2 = '6985', and no cryptogram shall be returned.

9. Erroneous or Missing Data in the ICC

Data objects in the card are classified in section 5 as either mandatory or optional. However, some optional data objects must be present to support optional functions selected by the issuer or must be present to avoid inconsistencies if other related data objects are present.

When any mandatory data object is missing, the terminal terminates the transaction. When an optional data object is required because of the existence of other data objects or to support functions that must be performed due to the setting of bits in the Application Interchange Profile, the terminal sets to '1' the 'ICC data missing' indicator in the TVR.

Table 7 summarises the conditions under which this bit should be set to '1'. If none of the conditions in Table 7 apply, the bit shall be set to '0'. The setting of this bit is in addition to any other actions specified in other sections of this document.

Name	Tag	'ICC Data Missing' Shall Be Set If ...
Application Transaction Counter (ATC)	'9F36'	ATC is not returned by GET DATA command and both Lower and Upper Consecutive Offline Limit data objects are present
Cardholder Verification Method (CVM) List	'8E'	Not present and AIP indicates that cardholder verification is supported
Certification Authority Public Key Index	'8F'	Not present and AIP indicates offline static or dynamic data authentication is supported
Dynamic Data Authentication Data Object List (DDOL)	'9F49'	Not present and AIP indicates offline dynamic data authentication is supported
Issuer Public Key Certificate	'90'	Not present and AIP indicates offline static or dynamic data authentication is supported
Issuer Public Key Exponent	'9F32'	Not present and AIP indicates offline static or dynamic data authentication is supported
Issuer Public Key Remainder	'92'	Not present and AIP indicates offline static or dynamic data authentication is supported and the 'length of the Issuer Public Key', as recovered from the Issuer Public Key Certificate, indicates that there was insufficient space for the entire Issuer's Public Key in the certificate
Last Online Application Transaction Counter (ATC) Register	'9F13'	Last Online ATC Register is not returned by GET DATA command and both Lower and Upper Consecutive Offline Limits are present

Signed Static Application Data	'93'	Not present and AIP indicates offline static data authentication is supported
ICC Public Key Certificate	'9F46'	Not present and AIP indicates offline dynamic data authentication is supported
ICC Public Key Exponent	'9F47'	Not present and AIP indicates offline dynamic data authentication is supported
ICC Public Key Remainder	'9F48'	Not present and AIP indicates offline dynamic data authentication is supported and the 'length of the ICC Public Key', as recovered from the ICC Public Key Certificate, indicates that there was insufficient space for the entire ICC's Public Key in the certificate

Table 7 - ICC Data Missing Indicator Setting

It is up to the issuer to ensure that data in the card is of the correct format, and no format checking is mandated on the part of the terminal. However, if in the course of normal processing the terminal recognises that data is incorrectly formatted, the terminal shall terminate processing. This rule includes (but is not limited to):

I. Constructed data objects that do not parse correctly.

II. Dates that are out of range (for example, months that are not in the range 1 to 12).

III. Other data that must be in a specific range of values but are not.

IV. CVM Lists with no CVRs.

V. Multiple occurrences of a data object that should only appear once.

VI. An AFL with no entries.

VII. An AFL entry with invalid syntax, that is, any one or more of the following:

A. An SFI of 0 or 31.

B. A starting record number of 0.

C. An ending record number less than the starting record number (byte 3 < byte 2).

D. Number of records participating in offline data authentication greater than the number of records (byte 4 > byte 3 - byte 2 + 1).

THIS PAGE LEFT INTENTIONALLY BLANK

Annexes

Annex A - Coding of Data Elements

This annex provides the coding for specific data elements used to control the transaction flow in the terminal or to record the status of processing for the transaction. In the tables, a '1' means that if that bit has the value '1', the corresponding 'Meaning' applies. An 'x' means that the bit does not apply.

Data (bytes or bits) indicated as RFU shall be set to '0'.

A1. Application Interchange Profile

Byte 1 (Leftmost):

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
1	x	x	x	x	x	x	x	Initiate[12]
x	1	x	x	x	x	x	x	Offline static data authentication is supported
x	x	1	x	x	x	x	x	Offline dynamic data authentication is supported
x	x	x	1	x	x	x	x	Cardholder verification is supported
x	x	x	x	1	x	x	x	Terminal risk management is to be performed
x	x	x	x	x	1	x	x	Issuer authentication is supported
x	x	x	x	x	x	0	x	RFU
x	x	x	x	x	x	x	0	RFU

Byte 2 (Rightmost):

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
0	x	x	x	x	x	x	x	RFU
x	0	x	x	x	x	x	x	RFU
x	x	0	x	x	x	x	x	RFU
x	x	x	0	x	x	x	x	RFU
x	x	x	x	0	x	x	x	RFU
x	x	x	x	x	0	x	x	RFU
x	x	x	x	x	x	0	x	RFU
x	x	x	x	x	x	x	0	RFU

Table A-1 - Application Interchange Profile

[12] This version of this specification does not describe processing in the case where this bit is set to '1'.

A2. Application Usage Control

Byte 1 (Leftmost):

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
1	x	x	x	x	x	x	x	Valid for domestic cash transactions
x	1	x	x	x	x	x	x	Valid for international cash transactions
x	x	1	x	x	x	x	x	Valid for domestic goods
x	x	x	1	x	x	x	x	Valid for international goods
x	x	x	x	1	x	x	x	Valid for domestic services
x	x	x	x	x	1	x	x	Valid for international services
x	x	x	x	x	x	1	x	Valid at ATMs
x	x	x	x	x	x	x	1	Valid at terminals other than ATMs

Byte 2 (Rightmost):

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
1	x	x	x	x	x	x	x	Domestic cashback allowed
x	1	x	x	x	x	x	x	International cashback allowed
x	x	0	x	x	x	x	x	RFU
x	x	x	0	x	x	x	x	RFU
x	x	x	x	0	x	x	x	RFU
x	x	x	x	x	0	x	x	RFU
x	x	x	x	x	x	0	x	RFU
x	x	x	x	x	x	x	0	RFU

Table A-2 - Application Usage Control

A3. Cardholder Verification Rule Format

Byte 1 (Leftmost): Cardholder Verification Method (CVM) Codes

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
0								RFU
	0							Fail cardholder verification if this CVM is unsuccessful
	1							Apply succeeding CVR if this CVM is unsuccessful
		0	0	0	0	0	0	Fail CVM processing
		0	0	0	0	0	1	Plaintext PIN verification performed by ICC
		0	0	0	0	1	0	Enciphered PIN verified online
		0	0	0	0	1	1	Plaintext PIN verification performed by ICC and signature (paper)
		0	0	0	1	0	0	Enciphered PIN verification performed by ICC
		0	0	0	1	0	1	Enciphered PIN verification performed by ICC and signature (paper)
		0	x	x	x	x	x	Values in the range 000110-011101 reserved for future use by this specification
		0	1	1	1	1	0	Signature (paper)
		0	1	1	1	1	1	No CVM required
		1	0	x	x	x	x	Values in the range 100000-101111 reserved for use by the individual payment systems
		1	1	x	x	x	x	Values in the range 110000-111110 reserved for use by the issuer
		1	1	1	1	1	1	This value is not available for use

Table A-3 - CVM Codes

Value	Meaning
'00'	Always
'01'	If cash or cashback
'02'	If not cash or cashback
'03'	If terminal supports the CVM[13]
'04' - '05'	RFU
'06'	If transaction is in the application currency[14] and is under X value
'07'	If transaction is in the application currency and is over X value
'08'	If transaction is in the application currency and is under Y value
'09'	If transaction is in the application currency and is over Y value
'0A' - '7F'	RFU
'80' - 'FF'	Reserved for use by individual payment systems

Table A-4 - CVM Condition Codes

[13] In the case of offline PIN CVM, this means 'If offline PIN pad present'. In the case of online PIN CVM, this means 'If online PIN pad present'.

[14] That is, Transaction Currency Code = Application Currency Code.

A4. Issuer Code Table Index

Value	Refers to
'01'	Part 1 of ISO 8859
'02'	Part 2 of ISO 8859
'03'	Part 3 of ISO 8859
'04'	Part 4 of ISO 8859
'05'	Part 5 of ISO 8859
'06'	Part 6 of ISO 8859
'07'	Part 7 of ISO 8859
'08'	Part 8 of ISO 8859
'09'	Part 9 of ISO 8859
'10'	Part 10 of ISO 8859

Table A-5 - Issuer Code Table Index

A5. Terminal Verification Results

Byte 1: (Leftmost)

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
1	x	x	x	x	x	x	x	Offline data authentication was not performed
x	1	x	x	x	x	x	x	Offline static data authentication failed
x	x	1	x	x	x	x	x	ICC data missing
x	x	x	1	x	x	x	x	Card appears on terminal exception file[15]
x	x	x	x	1	x	x	x	Offline dynamic data authentication failed
x	x	x	x	x	0	x	x	RFU
x	x	x	x	x	x	0	x	RFU
x	x	x	x	x	x	x	0	RFU

Byte 2:

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
1	x	x	x	x	x	x	x	ICC and terminal have different application versions
x	1	x	x	x	x	x	x	Expired application
x	x	1	x	x	x	x	x	Application not yet effective
x	x	x	1	x	x	x	x	Requested service not allowed for card product
x	x	x	x	1	x	x	x	New card
x	x	x	x	x	0	x	x	RFU
x	x	x	x	x	x	0	x	RFU
x	x	x	x	x	x	x	0	RFU

[15] There is no requirement in this specification for an exception file, but it is recognised that some terminals may have this capability.

Byte 3:

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
1	x	x	x	x	x	x	x	Cardholder verification was not successful
x	1	x	x	x	x	x	x	Unrecognised CVM
x	x	1	x	x	x	x	x	PIN Try Limit exceeded
x	x	x	1	x	x	x	x	PIN entry required and PIN pad not present or not working
x	x	x	x	1	x	x	x	PIN entry required, PIN pad present, but PIN was not entered
x	x	x	x	x	1	x	x	Online PIN entered
x	x	x	x	x	x	0	x	RFU
x	x	x	x	x	x	x	0	RFU

Byte 4:

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
1	x	x	x	x	x	x	x	Transaction exceeds floor limit
x	1	x	x	x	x	x	x	Lower consecutive offline limit exceeded
x	x	1	x	x	x	x	x	Upper consecutive offline limit exceeded
x	x	x	1	x	x	x	x	Transaction selected randomly for online processing
x	x	x	x	1	x	x	x	Merchant forced transaction online
x	x	x	x	x	0	x	x	RFU
x	x	x	x	x	x	0	x	RFU
x	x	x	x	x	x	x	0	RFU

Byte 5 (Rightmost):

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
1	x	x	x	x	x	x	x	Default TDOL used
x	1	x	x	x	x	x	x	Issuer authentication was unsuccessful
x	x	1	x	x	x	x	x	Script processing failed before final GENERATE AC
x	x	x	1	x	x	x	x	Script processing failed after final GENERATE AC
x	x	x	x	0	x	x	x	RFU
x	x	x	x	x	0	x	x	RFU
x	x	x	x	x	x	0	x	RFU
x	x	x	x	x	x	x	0	RFU

Table A-6 - Terminal Verification Results

A6. Transaction Status Information

Byte 1 (Leftmost):

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
1	x	x	x	x	x	x	x	Offline data authentication was performed
x	1	x	x	x	x	x	x	Cardholder verification was performed
x	x	1	x	x	x	x	x	Card risk management was performed
x	x	x	1	x	x	x	x	Issuer authentication was performed
x	x	x	x	1	x	x	x	Terminal risk management was performed
x	x	x	x	x	1	x	x	Script processing was performed
x	x	x	x	x	x	0	x	RFU
x	x	x	x	x	x	x	0	RFU

Byte 2 (Rightmost):

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
0	x	x	x	x	x	x	x	RFU
x	0	x	x	x	x	x	x	RFU
x	x	0	x	x	x	x	x	RFU
x	x	x	0	x	x	x	x	RFU
x	x	x	x	0	x	x	x	RFU
x	x	x	x	x	0	x	x	RFU
x	x	x	x	x	x	0	x	RFU
x	x	x	x	x	x	x	0	RFU

Table A-7 - Transaction Status Information