

© 2011 EMVCo, LLC ("EMVCo"). All rights reserved. Any and all uses of these Specifications is subject to the terms and conditions of the EMVCo Terms of Use agreement available at www.emvco.com. These Specifications are provided "AS IS" without warranties of any kind, and EMVCo neither assumes nor accepts any liability for any errors or omissions contained in these Specifications. EMVCO DISCLAIMS ALL REPRESENTATIONS AND WARRANTIES, EXPRESS OR IMPLIED, INCLUDING WITHOUT LIMITATION IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT, AS TO THESE SPECIFICATIONS.

EMVCo makes no representations or warranties with respect to intellectual property rights of any third parties in or in relation to the Specifications. EMVCo undertakes no responsibility to determine whether any implementation of these Specifications may violate, infringe, or otherwise exercise the patent, copyright, trademark, trade secret, know-how, or other intellectual property rights of third parties, and thus any person who implements any part of these Specifications should consult an intellectual property attorney before any such implementation.

Without limiting the foregoing, the Specifications may provide for the use of public key encryption and other technology, which may be the subject matter of patents in several countries. Any party seeking to implement these Specifications is solely responsible for determining whether its activities require a license to any such technology, including for patents on public key encryption technology. EMVCo shall not be liable under any theory for any party's infringement of any intellectual property rights in connection with these Specifications

EMV

Integrated Circuit Card

Specifications for Payment Systems

Common Payment Application Specification

Version 1.0
December 2005

EMV

Integrated Circuit Card

Specifications for Payment Systems

Common Payment Application

Specification

Version 1.0
December 2005

Contents

Part I – General

1	Scope	1-1
1.1	Underlying Standards	1-1
1.2	Audience	1-1
1.3	Contents	1-2
2	Normative References	2-1
2.1	EMV Documents	2-1
3	Definitions	3-1
4	Abbreviations, Notations, Conventions, Terminology, and Symbols	4-1
4.1	Abbreviations	4-1
4.2	Notations	4-5
4.3	Data Element Format Conventions	4-6
4.4	Terminology	4-8
4.5	Requirement Notation	4-10
4.6	Flow Chart Symbols	4-11

Part II – Introduction to Processing

5	Overview	5-1
5.1	Implementer-Options	5-2
5.2	Functional Overview	5-3
5.2.1	Application Selection (mandatory)	5-3
5.2.2	Initiate Application Processing (mandatory)	5-4
5.2.3	Read Application Data (mandatory)	5-4
5.2.4	Offline Data Authentication (optional)	5-5
5.2.5	Processing Restrictions (performed by the terminal)	5-6
5.2.6	Cardholder Verification (mandatory)	5-6
5.2.7	Terminal Risk Management (performed by the terminal)	5-7
5.2.8	Terminal Action Analysis (performed by the terminal)	5-7
5.2.9	First Card Action Analysis (mandatory)	5-8
5.2.10	Online Processing (conditional – if first AC type was ARQC)	5-9
5.2.11	Second Card Action Analysis (conditional – if first AC type was ARQC)	5-10
5.2.12	Issuer Script Command Processing (optional)	5-11
5.3	Sample Transaction Flow	5-12

5.4	Minimum Functionality	5-14
5.4.1	Card Functional Requirements	5-14
5.4.2	Command Support Requirements	5-16
6	General Command Information	6-1
6.1	Implementation of the CPA Specifications	6-1
6.2	State Machine	6-2
6.2.1	States	6-2
6.2.2	Sequence of Commands and Transition Between States	6-4
6.3	Command Validation	6-6
6.4	Exception Handling	6-6
6.4.1	Status Words	6-6
6.4.2	Missing or Invalid Resources	6-6

Part III - Function Processing

7	Application Selection	7-1
7.1	Purpose	7-1
7.2	Sequence of Execution	7-1
7.2.1	Subsequent Related Processing	7-1
7.3	Processing	7-2
7.3.1	Building the Candidate List	7-2
7.3.2	Identifying and Selecting the Application	7-3
8	Initiate Application Processing	8-1
8.1	Purpose	8-1
8.2	Sequence of Execution	8-2
8.2.1	Prior Related Processing	8-2
8.2.2	Subsequent Related Processing	8-2
8.3	Card Data	8-4
8.4	Terminal Data	8-8
8.5	GET PROCESSING OPTIONS Command	8-9
8.5.1	Command Coding	8-10
8.5.2	Processing	8-12
8.5.3	Profile Selection	8-13
8.5.4	Profile Behaviour	8-16
8.5.5	Respond to GET PROCESSING OPTIONS Command	8-17
8.6	Function Flow Charts	8-17

9	Read Application Data	9-1
9.1	Purpose	9-1
9.2	Sequence of Execution	9-2
9.2.1	Prior Related Processing	9-2
9.2.2	Subsequent Related Processing	9-2
9.3	Card Data	9-3
9.4	Terminal Data	9-3
9.5	READ RECORD Command	9-4
9.5.1	Command Coding	9-4
9.5.2	Processing	9-5
9.5.3	Respond to READ RECORD Command	9-6
9.6	Function Flow Charts	9-7
9.7	Additional File Requirements	9-8
9.7.1	VLP Data File	9-8
9.7.2	Transaction Log File	9-9
9.7.3	File Containing the Profile Selection Entries	9-10
10	Offline Data Authentication	10-1
10.1	Purpose	10-1
10.2	Sequence of Execution	10-2
10.2.1	Prior Related Processing	10-2
10.2.2	Subsequent Related Processing	10-3
10.3	Card Data	10-5
10.4	Terminal Data	10-9
10.5	Determining Whether to Perform SDA, DDA, or CDA	10-10
10.6	Static Data Authentication (SDA)	10-11
10.6.1	Commands	10-11
10.6.2	Processing	10-11
10.7	Offline Dynamic Data Authentication	10-12
10.7.1	INTERNAL AUTHENTICATE Command	10-12
10.7.2	GENERATE APPLICATION CRYPTOGRAM (AC) Command	10-15
11	Processing Restrictions	11-1
11.1	Purpose	11-1
11.2	Sequence of Execution	11-2
11.2.1	Prior Related Processing	11-2
11.2.2	Subsequent Related Processing	11-2
11.3	Card Data	11-3
11.4	Terminal Data	11-4

11.5	Processing	11-5
11.5.1	Application Version Number	11-5
11.5.2	Application Usage Control	11-5
11.5.3	Application Effective Date	11-6
11.5.4	Application Expiration Date	11-6
12	Cardholder Verification	12-1
12.1	Purpose	12-1
12.2	Sequence of Execution	12-2
12.2.1	Prior Related Processing	12-2
12.2.2	Subsequent Related Processing	12-2
12.3	Card Data	12-4
12.4	Terminal Data	12-7
12.5	GET DATA Command	12-8
12.5.1	Command Coding	12-8
12.5.2	Processing	12-8
12.5.3	GET DATA Flow Chart	12-10
12.6	GET CHALLENGE Command	12-12
12.6.1	Command Coding	12-12
12.6.2	Processing	12-12
12.6.3	GET CHALLENGE Response	12-13
12.6.4	GET CHALLENGE Flow Chart	12-13
12.7	VERIFY Command	12-14
12.7.1	Command Coding	12-15
12.7.2	Processing	12-17
12.7.3	VERIFY Flow Chart	12-21
13	Terminal Risk Management	13-1
13.1	Purpose	13-1
13.2	Sequence of Execution	13-2
13.2.1	Prior Related Processing	13-2
13.2.2	Subsequent Related Processing	13-2
13.3	Card Data	13-2
13.4	Terminal Data	13-2
13.5	Processing	13-3
13.5.1	Terminal Exception File	13-3
13.5.2	Merchant Forced Transaction Online	13-3
13.5.3	Floor Limits	13-3
13.5.4	Random Transaction Selection	13-3
13.5.5	Terminal Velocity Checking	13-3

14	Terminal Action Analysis	14-1
14.1	Purpose	14-1
14.2	Sequence of Execution	14-2
14.2.1	Prior Related Processing	14-2
14.2.2	Subsequent Related Processing	14-2
14.3	Card Data	14-3
14.4	Terminal Data	14-4
14.5	Processing	14-5
14.5.1	Review Offline Processing Results	14-5
14.5.2	Request Cryptogram Processing	14-6
15	First Card Action Analysis	15-1
15.1	Purpose	15-1
15.2	Sequence of Execution	15-2
15.2.1	Prior Related Processing	15-2
15.2.2	Subsequent Related Processing	15-2
15.3	Card Data	15-3
15.4	Terminal Data	15-13
15.5	First GENERATE AC Command	15-14
15.5.1	Command Coding	15-16
15.5.2	Profile Behaviour	15-19
15.5.3	Card Risk Management Processing	15-23
15.5.4	Determine Response Application Cryptogram Type	15-57
15.5.5	Application Approves Transaction Offline	15-60
15.5.6	Application Requests Online Processing	15-64
15.5.7	Application Declines Transaction Offline	15-65
15.5.8	Respond to GENERATE AC Command	15-68
15.6	Function Flow Charts	15-74
16	Online Processing	16-1
16.1	Purpose	16-2
16.2	Sequence of Execution	16-3
16.2.1	Prior Related Processing	16-3
16.2.2	Subsequent Related Processing	16-3
16.3	Card Data	16-4
16.4	Terminal Data	16-5
16.5	Commands	16-5
16.6	Processing	16-6
16.6.1	Online Request	16-6
16.6.2	Online Response	16-6
16.6.3	No Online Response	16-7
16.7	Function Flow Chart	16-8

17	Second Card Action Analysis	17-1
17.1	Purpose	17-2
17.2	Sequence of Execution	17-3
17.2.1	Prior Related Processing	17-3
17.2.2	Subsequent Related Processing	17-3
17.3	Card Data	17-4
17.4	Terminal Data	17-13
17.5	Second GENERATE AC Command	17-15
17.5.1	Command Coding	17-17
17.5.2	Configure Second Card Action Analysis	17-20
17.5.3	Online Authorisation Completed	17-24
17.5.4	Online Processing requested, Online Authorisation Not Completed	17-45
17.5.5	Application Approves Transaction (TC)	17-62
17.5.6	Application Declines Transaction (AAC)	17-63
17.5.7	CVR Updates	17-64
17.5.8	Respond to GENERATE AC Command	17-66
17.6	Function Flow Charts	17-71
18	Issuer Script Command Processing	18-1
18.1	Purpose	18-2
18.2	Sequence of Execution	18-2
18.2.1	Prior Related Processing	18-2
18.2.2	Subsequent Related Processing	18-3
18.3	Card Data	18-4
18.4	Terminal Data	18-6
18.4.1	Authorisation Response Data	18-7
18.5	Overview	18-8
18.5.1	Authorisation Response Message	18-8
18.5.2	Issuer-to-Card Script Processing	18-8
18.5.3	Card Secure Messaging	18-9
18.5.4	Resulting Indicators	18-10
18.5.5	Script Commands Supported	18-11
18.6	APPLICATION UNBLOCK Command	18-12
18.6.1	APPLICATION UNBLOCK Command Coding	18-13
18.6.2	APPLICATION UNBLOCK Command Processing	18-13
18.6.3	APPLICATION UNBLOCK Flow	18-14
18.7	PIN CHANGE/UNBLOCK Command	18-17
18.7.1	PIN CHANGE/UNBLOCK Command Coding	18-18
18.7.2	PIN CHANGE/UNBLOCK Processing	18-19
18.7.3	PIN CHANGE/UNBLOCK Flow	18-23

18.8	PUT DATA Command	18-29
18.8.1	PUT DATA Command Coding	18-29
18.8.2	PUT DATA Processing	18-31
18.8.3	PUT DATA Flow	18-34
18.9	UPDATE RECORD Command	18-39
18.9.1	UPDATE RECORD Command Coding	18-39
18.9.2	UPDATE RECORD Processing	18-41
18.9.3	UPDATE RECORD Flow	18-44

Part IV - Additional Topics

19	Additional Functions	19-1
19.1	Overview	19-1
19.2	Requirements for Additional Functionality	19-2
19.3	Examples of Additional Functionality	19-3
19.3.1	Additional Accumulators, Counters, and Cyclic Accumulators	19-3
19.3.2	Issuer-Discretionary Bits – CVR, ADR, and CIACs	19-4
19.3.3	Issuer-Discretionary Bits – Internal Application Data	19-4
19.3.4	Issuer-Discretionary Bytes – Issuer Application Data	19-4
19.3.5	Issuer-Discretionary Bytes – Issuer Authentication Data	19-5
19.3.6	Additional Issuer Script Commands	19-5
19.3.7	Support for Other MAC Lengths	19-6
20	Security and Key Management	20-1
20.1	Reference PIN Protection	20-2
20.2	Key Protection	20-3
20.3	Secure Messaging	20-3
20.4	Security Counters	20-4
20.5	Other Data Requirements	20-6
21	Personalisation	21-1
21.1	Data Elements to be Personalised	21-2
21.1.1	EMV Data in Records with SFI 1 through 10	21-2
21.1.2	CPA Data Elements Requiring Personalisation	21-3
21.1.3	CPA Default Implementation	21-12
21.2	EMV CPS	21-15
21.2.1	Personalisation Commands	21-15
21.2.2	Data Grouping Rules	21-17
21.2.3	Data Grouping Order	21-17
21.2.4	Grouped Data Groupings	21-18
21.2.5	DGIs for Record Data	21-19
21.2.6	Files with SFI between 1 and 10	21-19

21.2.7	Files with SFI between 11 and 30	21-20
21.2.8	CPA Recommended Data Group Indicators for Records	21-21
21.2.9	DGIs for Internal Application Data	21-31
21.2.10	DGIs for Command Response Data	21-33
21.2.11	DGIs for PIN and Key Related Data	21-33
21.2.12	RFU DGIs	21-34
21.3	Requirements for Data Element Values	21-34
21.3.1	Profile Selection Entries	21-34
21.3.2	AIP	21-34
21.3.3	AFL	21-34
21.3.4	VLP Profile Data	21-35
21.3.5	Token Authentication Profile Data	21-36
21.3.6	Offline Counters	21-37
21.3.7	CIACs	21-37
21.3.8	Signed Static Data	21-38
21.3.9	EMV Record Data	21-40
21.3.10	CDOLs	21-41
21.3.11	Keys	21-42
21.3.12	Previous Transaction History (PTH)	21-42
21.3.13	Log Entry	21-42
21.4	Missing Data Elements	21-43

Part V - Annexes

Annex A	Profile Selection File Processing	A-1
A1	Profile Selection Entry	A-1
A2	Processing	A-5
A3	Examples	A-8
Annex B	Additional Check Table Functionality	B-1
B1	Additional Check Table Content	B-1
B2	Processing Additional Check Table x	B-3
B3	Examples of Additional Check Table x Value	B-5
Annex C	Currency Conversion Functionality	C-1
C1	Currency Conversion Table Data Element	C-2
C2	Example	C-3
Annex D	Transaction Logging	D-1
D1	Transaction Log Entry Description	D-1
D2	Issuer Options for Transaction Logging	D-3
D3	Log Data Tables	D-4

D4	Processing Transaction Logging	D-7
D5	Example	D-9
Annex E	Management of Dates in Days	E-1
E1	Date Conversion	E-1
E2	Computation of Reference Day	E-2
Annex F	Security Counters	F-1
F1	Symmetric Keys	F-1
F2	PIN Encipherment Key	F-3
Annex G	Management of Profile Data	G-1
G1	Profile Resources	G-1
G2	Structure of Profile Resource Templates	G-2
G3	Profile Resource Templates for PUT DATA and GET DATA	G-3
G4	PUT DATA Command Applied to Templates	G-5
G5	GET DATA Command Applied to Templates	G-9
Annex H	Issuer Profile Options Specification and Processing	H-1
H1	Profile Selection	H-1
H2	Configuring Profile Resources	H-3
H3	Profile-specific Issuer Options Profile Control	H-8
H4	Profile-specific AIP and AFL	H-10
H5	Profile-specific CIACs	H-11
H6	Profile-specific Configuration of MTA Check	H-12
H7	Profile-specific Configuration of Offline Counters	H-14
H8	Profile-specific Configuration of Accumulators	H-16
H9	Profile-specific Configuration of Cyclic Accumulators	H-19
H10	Pre-defined Issuer profiles	H-21
H11	Example – Simple CCD-Compliant Profile	H-23
Annex I	Understanding Cyclic Accumulators	I-1
I1	Cyclic Accumulators	I-1
I2	Management of Cyclic Accumulators	I-7
I3	Management of Erroneous Transaction Dates	I-9
I4	Cyclic Accumulator Usage Scenarios	I-12
Annex J	GET DATA and PUT DATA Data Elements	J-1
Annex K	Data Element Tags	K-1
Annex L	Data Dictionary	L-1

Tables

Table 4-1: Terminology for Required and Optional Functionality	4-9
Table 5-1: Implementer-options	5-2
Table 5-2: Card Functional Requirements	5-14
Table 5-3: Command Support Requirements	5-16
Table 6-1: Application States	6-2
Table 6-2: Sequence of Commands and State Transitions for Commands	6-5
Table 8-1: Initiate Application Processing – Card Data	8-4
Table 8-2: Initiate Application Processing – Terminal Data	8-8
Table 8-3: GET PROCESSING OPTIONS Command Message	8-10
Table 8-4: PDOL Related Data	8-10
Table 9-1: Read Application Data – Card Data	9-3
Table 9-2: READ RECORD Command Message	9-4
Table 9-3: Reference Control Parameter Coding for READ RECORD Command	9-4
Table 9-4: Profile Selection File Entry	9-10
Table 10-1: Offline Data Authentication – Card Data	10-5
Table 10-2: Offline Data Authentication – Terminal Data	10-9
Table 10-3: INTERNAL AUTHENTICATE Command Message	10-13
Table 11-1: Processing Restrictions – Card Data	11-3
Table 11-2: Processing Restrictions – Terminal Data	11-4
Table 12-1: Cardholder Verification – Card Data	12-4
Table 12-2: PIN Processing – Terminal Data	12-7
Table 12-3: GET DATA Command Message	12-8
Table 12-4: GET CHALLENGE Command Message	12-12
Table 12-5: VERIFY Command Message	12-15
Table 13-1: Terminal Risk Management – Card Data	13-2
Table 14-1: Terminal Action Analysis – Card Data	14-3
Table 14-2: Review Offline Processing Results—Terminal Data	14-4
Table 15-1: First Card Action Analysis – Card Data	15-3
Table 15-2: First Card Action Analysis – Terminal Data	15-13
Table 15-3: First GENERATE AC Command Message	15-16
Table 15-4: Reference Control Parameter Coding for First GENERATE AC	15-16
Table 15-5: First GENERATE AC Command Data	15-17
Table 15-6: Card Risk Management Checks	15-24
Table 15-7: Conditions for Accumulating Transaction in Accumulator y	15-45
Table 15-8: Card's Response to First GENERATE AC Command	15-58
Table 15-9: Issuer Application Data for Profile '7E' (<i>Authentication Token</i>)	15-68
Table 15-10: Issuer Application Data for Profile Not '7E'	15-70
Table 15-11: Data Appended to Transaction Log	15-74
Table 16-1: Online Processing – Card Data	16-4

Table 17-1: Second Card Action Analysis – Card Data	17-4
Table 17-2: Second Card Action Analysis – Terminal Data	17-13
Table 17-3: SECOND GENERATE AC Command Message	17-17
Table 17-4: Reference Control Parameter Coding for SECOND GENERATE AC	17-17
Table 17-5: SECOND GENERATE AC Command Data: Amounts in CDOL2	17-18
Table 17-6: SECOND GENERATE AC Command Data: No Amounts in CDOL2	17-18
Table 17-7: Bits to Reset if Non-required Issuer Authentication Fails	17-26
Table 17-8: Bits to Reset if Issuer Authentication Passes	17-30
Table 17-9: Conditions for Accumulating Transaction using CSU Approval	17-35
Table 17-10: Bits to Reset if Issuer Authentication Data Not Present	17-41
Table 17-11: Card Risk Management Checks	17-46
Table 17-12: Conditions for Accumulating Offline Approved Transaction	17-52
Table 17-13: Issuer Application Data for SECOND GENERATE AC	17-67
Table 17-14: Data Elements to Log	17-70
Table 18-1: Issuer-to-Card Script Processing – Card Data	18-4
Table 18-2: Issuer-to-Card Script Processing – Terminal Data	18-6
Table 18-3: Issuer-to-Card Script Processing – Authorisation Response Data	18-7
Table 18-4: APPLICATION UNBLOCK Command Message	18-13
Table 18-5: PIN CHANGE / UNBLOCK Command Message	18-18
Table 18-6: PUT DATA Command Message	18-29
Table 18-7: UPDATE RECORD Command Message	18-39
Table 18-8: Reference Control Parameter coding for UPDATE RECORD	18-39
Table 21-1: SELECT Command Response Data Elements – Mandatory	21-3
Table 21-2: Unique CPA Persistent Data Elements – Mandatory	21-4
Table 21-3: Unique CPA Persistent Data Elements – Issuer-optional	21-5
Table 21-4: Unique CPA Persistent Data Elements – Dynamic RSA Option Elements	21-6
Table 21-5: Unique CPA Persistent Data Elements – Issuer-optional Transaction Logging Elements	21-6
Table 21-6: Unique CPA Persistent Data Elements – Optional Security Limit Elements	21-6
Table 21-7: Unique CPA Persistent Data Elements – Conditional VLP Elements	21-7
Table 21-8: Unique CPA Persistent Data Elements – Issuer-Optional VLP Elements	21-7
Table 21-9: CPA Persistent Data Elements – Data Sets	21-8
Table 21-10: Currency Conversion Parameter	21-11
Table 21-11: Number of Each Type of Profile Resource in CPA	21-13
Table 21-12: DGI Summary for Record Data	21-21
Table 21-13: Data Content for DGI '0101'	21-22
Table 21-14: Data Content for DGI '0102'	21-22
Table 21-15: Data Content for DGI '0201'	21-22
Table 21-16: Data Content for DGI '0202'	21-23
Table 21-17: Data Content for DGI '0203'	21-24
Table 21-18: Data Content for DGI '0204'	21-24

Table 21-19: Data Content for DGI '0205'	21-24
Table 21-20: Data Content for DGI '020n'	21-25
Table 21-21: Data Content for DGI '0301'	21-27
Table 21-22: Data Content for DGI '0302'	21-28
Table 21-23: Data Content for DGI '0303'	21-28
Table 21-24: Data Content for DGI '0B01'	21-29
Table 21-25: Data Content for DGI 'ssrr'	21-30
Table 21-26: DGI Summary for Internal Application Data	21-31
Table 21-27: Data Content for DGI '3000'	21-32
Table 21-28: DGI Summary for Command Response Data	21-33
Table 21-29: Data Content for DGI '8301' through '8305'	21-33
Table 21-30: VLP Record Data (SFI 11)	21-35
Table 21-31: PDOL Data to Support VLP Transaction	21-36
Table 21-32: PDOL Data to Support Token Authentication	21-36
Table 21-33: Static Data to be Authenticated	21-39
Table A-1: Data Elements in Profile Selection Entry	A-2
Table A-2: Profile Selection File – Simple Example	A-8
Table A-3: Profiles for Complex Profile Selection File	A-9
Table A-4: Profile Selection File – Complex Example	A-10
Table A-5: PDOL Contents – Complex Example	A-11
Table A-6: Profile Selection Entries – Complex Example	A-11
Table B-1: Additional Check Table x	B-2
Table B-2: Example of Additional Check Table x Value	B-5
Table B-3: Example of Additional Check Table x Value	B-6
Table C-1: Currency Conversion Parameter	C-2
Table C-2: Example Currency Conversion Parameters	C-3
Table D-1: Structure of Log Format data element	D-2
Table D-2: Transaction Logging Options in Application Control byte 3	D-3
Table D-3: Log Data Table Format	D-4
Table D-4: Example of GEN AC Log Data Table	D-5
Table D-5: Data Logged at First GENERATE AC for a TC or ACC	D-7
Table D-6: Data Saved for Second GENERATE AC after an ARQC	D-8
Table D-7: Data Logged at Second GENERATE AC	D-8
Table D-8: Example – Transaction Logging Settings in Application Control	D-9
Table D-9: Example – First GENERATE AC Command Data	D-10
Table D-10: Example – Second GENERATE AC Command Data	D-10
Table D-11: Example – Data Referenced in First GEN AC Unchanging Log Data Table	D-11
Table D-12: Example – Data Referenced in First GEN AC Log Data Table	D-11
Table D-13: Example – Data Saved if First GENERATE AC Response is an ARQC	D-12
Table D-14: Example – Data Logged at First GENERATE AC for a TC or AAC	D-12
Table D-15: Example – Data Extracted from Second GENERATE AC Command Data	D-13
Table D-16: Example – Transaction Data Logged at Second GENERATE AC	D-13

Table G-1: Profile Resource Templates for PUT DATA and GET DATA	G-3
Table G-2: Counter Controls Contents before PUT DATA Command	G-5
Table G-3: Coding of Received PUT DATA Command	G-6
Table G-4: Counter Controls Contents after PUT DATA Command	G-6
Table G-5: Accumulator Profile Controls Contents before PUT DATA Command	G-8
Table G-6: Coding of Received PUT DATA Command	G-8
Table G-7: Accumulator Controls Contents after the PUT DATA Command	G-9
Table G-8: Coding of the Received PUT DATA Command	G-10
Table I-1: Cyclic Accumulator x Data	I-2
Table I-2: Cyclic Accumulator x Control	I-3
Table I-3: Cyclic Accumulator Profile Control y for Cyclic Accumulator x	I-4
Table I-4: Management of Erroneous Transaction Dates	I-10
Table I-5: Parameters for Cyclic Accumulator Usage Example 1	I-12
Table I-6: Parameters for Cyclic Accumulator Usage Example 2	I-14
Table I-7: Parameters for Cyclic Accumulator Usage Example 3	I-16
Table I-8: Parameters for Cyclic Accumulator Usage Example 4	I-18
Table I-9: Parameters for Cyclic Accumulator Usage Example 5	I-20
Table J-1: GET DATA Command Data Elements	J-1
Table K-1: Data Element Tags	K-1

Figures

Figure 4-1: Flow Chart Symbols	4-11
Figure 5-1: Sample Transaction Flow	5-13
Figure 8-1: Initiate Application Processing Flow	8-18
Figure 9-1: READ RECORD Command Processing Flow	9-7
Figure 10-1: INTERNAL AUTHENTICATE Flow	10-14
Figure 12-1: GET DATA Flow	12-11
Figure 12-2: GET CHALLENGE Flow	12-13
Figure 12-3: VERIFY Flow	12-22
Figure 15-1: First Card Action Analysis Processing Flow	15-15
Figure 15-2: Examples – Building the IAD	15-72
** First Card Action Analysis Flow Charts **	<i>see page xix</i>
Figure 16-1: Online Processing Flow for Online Response	16-8
Figure 16-2: Online Processing Flow for No Online Response	16-9
Figure 17-1: Second Card Action Analysis Processing Flow	17-16
** Second Card Action Analysis Flow Charts **	<i>see page xx</i>
Figure 18-1: Command Data Format if Only MAC Data is Present	18-13
Figure 18-2: APPLICATION UNBLOCK Flow	18-15
Figure 18-3: Recovering New PIN Block from PIN CHANGE/UNBLOCK Command Data	18-21
Figure 18-4: PIN CHANGE/UNBLOCK Flow	18-24
Figure 18-5: Command Data Format for PUT DATA	18-31
Figure 18-6: PUT DATA Flow	18-35
Figure 18-7: Command Data Format for UPDATE RECORD	18-41
Figure 18-8: UPDATE RECORD Flow	18-45
Figure A-1: Profile Selection Entry Format	A-1
Figure A-2: Profile Selection Algorithm	A-5
Figure A-3: Profile Selection Using Card Data	A-7
Figure B-1: Additional Check Table x	B-1
Figure B-2: Additional Check Table Usage	B-4
Figure D-1: Processing First GEN AC Log Data Table	D-6
Figure G-1: Profile Resource Template	G-2
Figure G-2: Counter Controls before PUT DATA Command	G-5
Figure G-3: Counter Controls Coding after PUT DATA Command	G-6
Figure G-4: Profile Resource Template Coding after Personalisation	G-7
Figure G-5: Accumulator Profile Controls before PUT DATA Command	G-8
Figure G-6: Accumulator Profile Controls Template Contents after PUT DATA Command	G-9
Figure G-7: Card Response to the GET DATA Command	G-10
Figure H-1: Profile Selection Process	H-2
Figure H-2: Profile Control	H-3
Figure H-3: Profile-specific Controls	H-6
Figure H-4: Issuer Options Profile Control	H-9

Figure H-5: AIP/AFL Entry	H-10
Figure H-6: CIACs Entry	H-11
Figure H-7: MTA Profile Control	H-13
Figure H-8: Counter Profile Control for Counter n – Example (n=3)	H-15
Figure H-9: Accumulator Profile Control for Accumulator n – Example (n=2)	H-17
Figure H-10: Cyclic Accumulator Control	H-20
Figure I-1: Cyclic Accumulator	I-6
Figure I-2: Cyclic Accumulator Behaviour	I-8
Figure I-3: Management of Erroneous Transaction Dates	I-11

Card Action Analysis Flow Charts

Flow 15-1	First GENERATE AC Initial Flow	15-75
Flow 15-2	ARQC Requested	15-82
Flow 15-3	TC Requested	15-86
Flow 15-4	TC Checks	15-91
Flow 15-5	ARQC Checks	15-95
Flow 15-6	AAC Checks	15-98
Flow 15-7	Authentication Token	15-103
Flow 15-8	AAC Accumulator Check	15-104
Flow 15-9	AAC Counter Check	15-105
Flow 15-10	Accumulator Cumulating	15-106
Flow 15-11	Accumulator x Non-cumulating Check	15-109
Flow 15-12	Additional Check Table Processing	15-110
Flow 15-13	Build Issuer Application Data	15-112
Flow 15-14	Counter x Cumulating Check	15-115
Flow 15-15	Counter Non-cumulating Check	15-117
Flow 15-16	Currency Conversion	15-118
Flow 15-17	Cyclic Accumulator Cumulating	15-120
Flow 15-18	Cyclic Accumulator Non-cumulating Check	15-122
Flow 15-19	Maximum Number of Days Offline Check	15-123
Flow 15-20	Maximum Transaction Amount Check	15-124
Flow 15-21	First GENERATE AC Transaction Logging	15-126
Flow 15-22	Verify/Reset Cyclic Period	15-130

Second Card Action Analysis Flow Charts

Flow 17-1	Second GENERATE AC Initial Flow	17-72
Flow 17-2	Transaction Authorization Online	17-74
Flow 17-3	Issuer Authentication Present	17-75
Flow 17-4	Issuer Authentication Not Present	17-76
Flow 17-5	Issuer Authentication Failed	17-77
Flow 17-6	Issuer Authentication Okay	17-78
Flow 17-7	Transaction Approved after Issuer Authentication Failed	17-84
Flow 17-8	Online TC	17-85
Flow 17-9	Unable To Go Online	17-86
Flow 17-10	AAC Unable To Go Online	17-93
Flow 17-11	TC	17-94
Flow 17-12	AAC	17-97
Flow 17-13	Add to Accumulator x	17-99
Flow 17-14	Add to Counter x	17-101
Flow 17-15	Add to Cyclic Accumulator x	17-102
Flow 17-16	Build Issuer Application Data	17-104
Flow 17-17	Check Cyclic Reference Date	17-107
Flow 17-18	Currency Conversion	17-109
Flow 17-19	CVR Updates	17-111
Flow 17-20	Cyclic Accumulator Check with Cumulating	17-116
Flow 17-21	Cyclic Accumulator Check Without Cumulating	17-119
Flow 17-22	Maximum Transaction Amount Check	17-120
Flow 17-23	Offline TC Accumulator Check with Cumulating	17-122
Flow 17-24	Offline TC Accumulator Check Without Cumulating	17-124
Flow 17-25	Offline TC Counter Check with Cumulating	17-125
Flow 17-26	Offline TC Counter Check Without Cumulating	17-127
Flow 17-27	Reset Maximum Number of Days Offline	17-128
Flow 17-28	Reset Non-velocity Checking Indicators	17-129
Flow 17-29	Reset Velocity Checking Indicators	17-130
Flow 17-30	Transaction Logging Second GENERATE AC	17-131
Flow 17-31	Verify/Reset Cyclic Period	17-133

Part I

General

1 Scope

This document, the *Common Payment Application Specification* (CPA), defines the data elements and functionality for an application that complies with the EMV Common Core Definitions (CCD). It focuses on the functions performed by the integrated circuit card (ICC) and the interaction between the ICC and terminal at the point of transaction.

The objectives of the Common Payment Application Specification are to:

- Describe the functionality of a CCD-compliant implementation of EMV to ease vendor development efforts
- Specify a core set of functionalities that issuers can rely on having available in every implementation of CPA
- Specify an implementation that can be personalised with the same data elements to meet the business requirements of multiple payment systems

Because CPA is based on EMV and CCD, the specifications should be used together for reference and development purposes.

1.1 Underlying Standards

This specification is based on the EMV standards and should be read in conjunction with those standards. However, if any of the provisions or definitions in this specification differs from those standards, the provisions herein shall take precedence.

1.2 Audience

This specification is intended for use by ICC application developers, manufacturers of ICCs, system designers in payment systems, and financial institution staff responsible for implementing financial applications in ICCs.

1.3 Contents

This section provides an overview of each section of the *Common Payment Application Specification*. To provide clarity, requirements from EMV (including CCD) may be restated or replicated in this specification to provide a comprehensive application specification.

This specification is divided into the following sections:

Part I – General

- **Section 1, Scope**
- **Section 2, Normative References** – Lists standards and specifications referenced by this document.
- **Section 3, Definitions** – Provides a glossary of terms used in this document.
- **Section 4, Abbreviations, Notations, Conventions, Terminology, and Symbols** – Lists the acronyms and describes the formats used throughout this document.

Part II – Introduction to Processing

- **Section 5, Overview** – This section provides an overview of each function in transaction processing.

The Processing Overview is structured into the functional processes described in EMV, and has the following sub-sections:

- **Section 5.1, Implementer-Options** – Identifies functionality that is optional for the application vendor to implement.
 - **Section 5.2, Functional Overview**
 - General Description – Provides a high-level overview of the function in transaction processing.
 - Condition of Execution – Defines the conditions under which this process is executed.
 - **Section 5.3, Sample Transaction Flow** – Illustrates a sample flow for a CPA transaction at an EMV-compliant terminal.
 - **Section 5.4, Minimum Functionality** – Provides a high-level overview of the functionalities that are mandatory for all card implementations to support.
- **Section 6, General Command Information** – Provides general requirements for processing commands during an EMV transaction. General command processing includes the application state machine, command validation, and exception handling.

Part III - Function Processing

For ease of use, each function processing section (sections 7 through 18) is structured as follows:

- Purpose – Defines the functionality of the process.
- Sequence of Execution – Outlines prior processing to aid in understanding previous activities related to this function, and subsequent processing to aid in understanding future activities related to this function.
- Card Data – Provides a brief description of the data from the card used to support the function.
- Terminal Data – Provides a brief description of the data from the terminal used to support the function.
- Command Processing – Provides a brief description of the commands used to support the function, and the functionality of the functional process. If there are several commands or functions within a process, they may be listed separately.
- Function Flow Chart – Where appropriate, provides a sample flow to illustrate how the functionality might be implemented.

NOTE: Flow charts are representative of processing and may not include all steps that may be performed. Other processing flows with the same results are allowed.

- **Section 7, Application Selection** – This function determines which of the applications, supported by both the card and terminal, will be used to conduct the transaction.
- **Section 8, Initiate Application Processing** – During this function, the card receives any terminal data which the card requested during Application Selection and sends the terminal a list of the data to be read and functions to be supported for the transaction.
- **Section 9, Read Application Data** – During this function, the terminal reads the card data records necessary for the transaction.
- **Section 10, Offline Data Authentication** – During this function, the terminal authenticates data from the card using RSA public key technology.
- **Section 11, Processing Restrictions** – During this function, application version checks, effective and expiration date checks, and other checks are performed by the terminal.
- **Section 12, Cardholder Verification** – During this function, the terminal determines the cardholder verification method (CVM) to be used and performs the selected CVM.

- **Section 13, Terminal Risk Management** – During this function, the terminal ensures that higher-value transactions are sent online and that chip transactions go online periodically.
- **Section 14, Terminal Action Analysis** – During this function, the terminal applies rules set by the issuer in the card and by the acquirer in the terminal to the results of offline processing. This analysis determines whether the transaction should be approved offline, declined offline, or sent online for an authorisation.
- **Section 15, First Card Action Analysis** – During this function, velocity checking and other risk management, internal to the card, is performed. The application then determines whether to send the transaction online for authorisation or to approve or decline the transaction offline, and generates the response cryptogram.
- **Section 16, Online Processing** – During this function, the issuer's host computer (or a proxy for the issuer) reviews and authorises or declines transactions using the issuer's host-based risk parameters.
- **Section 17, Second Card Action Analysis** – During this function, additional card risk management is performed, and the card processes the results of the attempt to send the transaction online. The application then generates the second response cryptogram.
- **Section 18, Issuer-to-Card Script Processing** – During this function, the card applies post-issuance changes sent from the issuer.

Part IV – Additional Topics

- **Section 19, Additional Functions** – Discusses optional extensions to the application, including examples of how to support additional counters or incorporate additional functionality using issuer-discretionary bits in application data elements.
- **Section 20, Security and Key Management** – Provides requirements related to security and key management for a CPA implementation that are in addition to the specifications for CCD-compliant applications, as specified in EMV 4.1.
- **Section 21, Personalisation** – Describes the personalisation requirements for a CPA implementation.

Part V – Annexes

- **Annex A, Profile Selection File Processing** – Describes how the Profile Selection File can be used during Application Initiation to customise application behaviour based on transaction characteristics as specified by the issuer at the time of personalisation. Includes an example illustrating how the data element might be personalised to provide this functionality.

- **Annex B, Additional Check Table Functionality** – Describes how the Additional Check Tables can be used during Card Action Analysis to provide a card risk management test that can be customised by the issuer at the time of personalisation. Includes an example illustrating how the data element might be personalised to provide this functionality.
- **Annex C, Currency Conversion Functionality** – Describes how the Currency Conversion Table can be used to estimate the domestic currency value of a non-domestic currency transaction using Currency Conversion Parameters specified by the issuer at the time of personalisation. Includes an example illustrating how the data element might be personalised to provide this functionality.
- **Annex D, Transaction Logging** – Describes how the card can be configured to support flexible transaction logging specified at the time of personalisation. Includes an example illustrating how the data elements might be personalised to provide this functionality.
- **Annex E, Management of Dates in Days** – Describes how a date in the format YYMMDD can be converted to a count of days since a reference date, so that the application can perform card risk management based on number of days elapsed.
- **Annex F, Security Counters** – Describes an implementation of security counters for CPA if the security counters described in section 20 are implemented in the application.
- **Annex G, Management of Profile Data** – Explains the management of application resource data for the PUT DATA and GET DATA commands using a single template tag for each type of resource.
- **Annex H, Issuer Profile Options Specification and Processing** – Describes how the application processes Issuer-defined profile options and configures card behaviour based on these options.
- **Annex I, Understanding Cyclic Accumulators** – Explains the behaviour and management of Cyclic Accumulators in the application.
- **Annex J: GET DATA and PUT DATA Data Elements** – Lists the data elements that are supported for the GET DATA and PUT DATA commands.
- **Annex K, Data Element Tags** – Lists the data element tags and template tags used in the application and terminal.
- **Annex L, Data Dictionary** – Defines the data elements used in processing the application from a card and issuer perspective.

2 Normative References

The following standards contain provisions that are referenced in this specification. The latest version shall apply unless a publication date is explicitly stated.

2.1 EMV Documents

EMV documents are available on the EMVCo Website:

<http://www.emvco.com/specifications.cfm>.

<i>EMV Book 1</i>	<i>EMV Integrated Circuit Card Specifications for Payment Systems</i> , version 4.1, Book 1, Application Independent ICC to Terminal Interface Requirements, May 2004.
<i>EMV Book 2</i>	<i>EMV Integrated Circuit Card Specifications for Payment Systems</i> , version 4.1, Book 2, Security and Key Management, May 2004.
<i>EMV Book 3</i>	<i>EMV Integrated Circuit Card Specifications for Payment Systems</i> , version 4.1, Book 3, Application Specification, May 2004.
<i>EMV Book 4</i>	<i>EMV Integrated Circuit Card Specifications for Payment Systems</i> , version 4.1, Book 4, Cardholder, Attendant, and Acquirer Interface Requirements, May 2004.
<i>EMV CPS</i>	<i>EMV Card Personalization Specification</i> , version 1.0, June 2003.

3 Definitions

This section provides a glossary of terms used in this specification; it is not intended as a data dictionary. For descriptions of specific card and issuer data elements, refer to Annex L: Data Dictionary.

Acquirer	A Payment System member that has a contractual relationship with a merchant or that disburses currency to a cardholder in a cash disbursement, and directly or indirectly enters the resulting transaction into interchange.
Application	A program and associated data that reside on an integrated circuit chip and satisfy a business function. Examples of applications include payment, stored value, and loyalty.
Application Authentication Cryptogram (AAC)	An Application Cryptogram generated by the card for offline and online declined transactions.
Application Block	Instructions sent to the card by the issuer, to shut down the selected application on a card to prevent further use of that application. This process does not preclude the use of other applications on the card. A blocked application may be unblocked by the issuer.
Application Cryptogram	<p>A cryptogram generated by the card in response to a GENERATE AC command. See also:</p> <ul style="list-style-type: none">• Application Authentication Cryptogram (AAC)• Authorisation Request Cryptogram (ARQC)• Transaction Certificate (TC)
Asymmetric Cryptographic Technique	A cryptographic technique that uses two related transformations: a public transformation (defined by the public key) and a private transformation (defined by the private key). The two transformations have the property that, given the public transformation, it is computationally infeasible to derive the private transformation.
ATM	An unattended terminal that has electronic capability, accepts PINs, and disburses currency or cheques.
Authentication	A cryptographic process that validates the claimed origin of data or identity of an entity.

Authorisation	A process whereby an issuer or a representative of the issuer approves a transaction.
Authorisation Request	A merchant's or acquirer's request for authorisation of a transaction.
Authorisation Request Cryptogram (ARQC)	The Application Cryptogram generated by the card when requesting online authorisation for a transaction. It is sent to the issuer in the authorisation request. The issuer may validate the ARQC during the Online Processing function to ensure that the card is authentic.
Authorisation Response	The issuer's reply to an authorisation request.
Authorisation Response Cryptogram (ARPC)	A cryptogram that may be generated by the issuer and may be sent to the card in the authorisation response. This cryptogram is the result of the Authorisation Request Cryptogram (ARQC) and the Card Status Update enciphered with a session key. It is validated by the card during Issuer Authentication to ensure that the response came from a valid issuer.
Byte	8 bits of data.
Card	A payment card as defined by a payment system.
Card Block	Instructions, sent to the card by the Issuer, which shut down all proprietary and non-proprietary applications that reside on a card, to prevent further use of the card. A card that has been blocked cannot be unblocked.
Cardholder	An individual to whom a card is issued or who is authorised to use that card.
Cardholder Confirmation	Confirmation by the cardholder that the application selected by the terminal is to be used for processing the transaction.
Cardholder Selection	The process by which a cardholder may select one of multiple applications mutually supported by the card and terminal for use in processing the transaction.
Cardholder Verification	The process of determining that the presenter of the card is the valid cardholder.
Cardholder Verification Method (CVM)	A method used to confirm the identity of a cardholder.
Cash Disbursement	Currency, including traveller's cheques, paid to a cardholder using a card.

Cashback	Cash obtained in conjunction with, and processed as, a purchase transaction.
Certificate	The public key and identity of an entity together with some other information, rendered unforgeable by signing with the private key of the certification authority which issued that certificate.
Certification Authority (CA)	A trusted third party that establishes a proof that links a public key and other relevant information to its owner.
Ciphertext	Enciphered information.
Combined DDA/Application Cryptogram Generation (CDA)	A form of offline dynamic data authentication, combined with processing of the GENERATE AC command.
Command	A message sent by the terminal to the ICC that initiates an action and solicits a response from the ICC.
Common Core Definitions (CCD)	A minimum common set of card application implementation options, card application behaviours, and data element definitions sufficient to accomplish an EMV transaction, as defined in <i>EMV Book 3</i> .
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.
Cryptographic Algorithm	An algorithm operating under the control of a cryptographic key and used to protect the confidentiality and/or integrity of input data.
Cryptographic Key	A sequence of bits used by a cryptographic algorithm.
Cryptography	The art or science of keeping messages secret and/or secure.

CVM List	An issuer-defined list contained within a chip application establishing the hierarchy of methods for verifying the authenticity of a cardholder.
Data Authentication	Validation that data stored in the integrated circuit card has not been altered since card issuance. See also Offline Data Authentication.
Data Encryption Standard (DES)	The public domain symmetric key cryptography algorithm of the National Institute for Standards and Technology.
Data Integrity	A property of data that has not been altered or destroyed in an unauthorised manner.
Decipherment	The reversal of a corresponding encipherment.
DES key	A 64-bit secret parameter of the Data Encryption Standard algorithm, consisting of 56 bits that must be independent and random, and 8 error-detecting bits set to make the parity of each 8-bit byte of the key odd.
Digital Signature	An asymmetric cryptographic transformation of data that allows the recipient of the data to prove the origin and integrity of the data, and that protects the sender and the recipient of the data against forgery by third parties, and the sender against forgery by the recipient.
Dynamic Data Authentication (DDA)	A form of offline data authentication where the card generates a digital signature using transaction-specific data elements, for validation by the terminal to protect against skimming.
EMV Specifications	Technical specifications maintained by JCB International, MasterCard International, and Visa International to create standards and ensure global interoperability for use of chip technology in the payment industry.
Encipherment	The reversible transformation of data by a cryptographic algorithm to produce ciphertext.
Exclusive-OR	Binary addition with no carry, giving the following values: $0 \oplus 0 = 0$ $0 \oplus 1 = 1$ $1 \oplus 0 = 1$ $1 \oplus 1 = 0$
Expired Card	A card on which the embossed, encoded, or printed expiration date has passed.

Financial Transaction	The act between a cardholder and a merchant or acquirer that results in the exchange of goods or services for payment or in cash disbursement.
Floor Limit	A currency amount that the Payment System has established for single transactions at specific types of merchants, above which online authorisation is required.
Function	A process accomplished by one or more commands and resultant actions that are used to perform all or part of a transaction.
Hardware Security Module (HSM)	A secure module used to store cryptographic keys and perform cryptographic functions.
Hash Function	<p>A function that maps strings of bits to fixed-length strings of bits, satisfying the following two properties:</p> <ul style="list-style-type: none">• It is computationally infeasible to find for a given output an input which maps to this output.• It is computationally infeasible to find for a given input a second input that maps to the same output. <p>Additionally, if the hash function is required to be collision-resistant, it must also satisfy the following property:</p> <ul style="list-style-type: none">• It is computationally infeasible to find any two distinct inputs that map to the same output.
Hash Result	The string of bits that is the output of a hash function.
ICC Master Key (MK)	A key unique to the card, which is used to derive a session key.
Integrated Circuit(s)	Electronic component(s) designed to perform processing and/or memory functions.
Integrated Circuit(s) Card (ICC)	A card into which one or more integrated circuits are inserted to perform processing and memory functions.
International Organization for Standardization (ISO)	The specialised international agency that establishes and publishes international technical standards.
Interoperability	The ability of all card acceptance devices and terminals to accept and read all chip cards that are properly programmed and that support one of the applications supported by the terminal.
Issuer	A Payment System member that issues cards.

Issuer Action Code (IAC)	<p>Issuer-configured rules which the terminal uses to determine whether a transaction should be declined offline, sent online for an authorisation, or declined if online is not available. EMV defines the following, which reflect the issuer-selected action to be taken upon analysis of the TVR data element:</p> <ul style="list-style-type: none">• IAC - Default: rules that determine when a transaction should be declined offline if it cannot go online• IAC - Denial: rules that determine when a transaction should be declined offline• IAC - Online: rules that determine when a transaction should be sent online for authorisation
Issuer Authentication	<p>Validation of the issuer by the card to ensure the integrity of the authorisation response. See Authorisation Response Cryptogram (ARPC).</p>
Key	<p>A sequence of bits that controls the operation of a cryptographic transformation.</p>
Key Expiry Date	<p>The date after which a signature made with a particular key is no longer valid. Example: Issuer certificates signed by the CA key must expire on or before the CA Key Expiration date. CA keys may be removed from terminals after this date has passed.</p>
Key Generation	<p>The creation of a new key for subsequent use.</p>
Keypad	<p>Arrangement of numeric, command, and, where required, function and/or alphanumeric keys laid out in a specific manner.</p>
Message	<p>A string of bytes sent by the terminal to the card or vice versa, excluding transmission-control characters.</p>
Message Authentication Code (MAC)	<p>A digital code generated using a symmetric cryptographic algorithm that protects the sender and the recipient of data against forgery by third parties.</p>
Nibble	<p>The four most significant or least significant bits of a byte of data.</p>
Offline Approval	<p>A transaction that is approved (accepted) at the point of transaction between the card and terminal without an authorisation response from the issuer or from a proxy for the issuer.</p>

Offline Data Authentication	<p>A process whereby the card is validated at the point of transaction using RSA public key technology to protect against counterfeit or skimming. EMV includes three forms:</p> <ul style="list-style-type: none">• Static Data Authentication (SDA)• Dynamic Data Authentication (DDA)• Combined DDA/AC Generation (CDA)
Offline Decline	<p>A transaction that is declined (not accepted) at the point of transaction between the card and terminal without an authorisation response from the issuer or from a proxy for the issuer.</p>
Offline PIN	<p>A PIN value stored on the card that is validated at the point of transaction between the card and the terminal.</p>
Offline PIN Verification	<p>The process whereby a cardholder-entered PIN is passed to the card for comparison to a PIN value stored secretly on the card.</p>
Offline-only Terminal	<p>A card acceptance device that is not capable of sending transactions online for issuer authorisation.</p>
Online Authorisation	<p>A method of requesting an authorisation through a communications network other than voice to an issuer or issuer representative.</p>
Online Card Authentication	<p>A process performed by the issuer to validate that the card is authentic, and to protect against data manipulation.</p>
Online PIN	<p>A method of PIN verification whereby the PIN entered by the cardholder into the terminal PIN pad is enciphered and included in the online authorisation request message sent to the issuer.</p>
Online-capable Terminal	<p>A card acceptance device that is able to send transactions online to the issuer for authorisation. EMV describes such a terminal as “offline with online capability.”</p>
Padding	<p>Appending extra bits or bytes to either side of a data string.</p>
Payment System Environment	<p>The set of logical conditions established within the ICC when a payment system application conforming to this specification has been selected, or when a Directory Definition File (DDF) used for payment system application purposes has been selected.</p>

Personalisation	The process of populating a card with the application data that makes it ready for use.
PIN Pad	Arrangement of numeric and command keys to be used for personal identification number (PIN) entry.
Plaintext	Data in its original unenciphered form.
Point of Transaction	The physical location where a merchant or acquirer (in a face-to-face environment) or an unattended terminal (in an unattended environment) completes a transaction.
Post-issuance Update	A command sent by the issuer through the terminal via an authorisation response to update the electronically stored contents of a chip card.
Private Key	That key of an entity's asymmetric key pair that is kept secret and should only be used by that entity. In the case of a digital signature scheme, the private key defines the signature function.
Public Key	That key of an entity's asymmetric key pair that can be made public. In the case of a digital signature scheme, the public key defines the verification function.
Public Key Certificate	The public key information of an entity signed by the certification authority and thereby rendered unforgeable.
Public Key Cryptographic Algorithm	A cryptographic algorithm that allows the secure exchange of information, but does not require a shared secret key, through the use of two related keys: a public key which may be distributed in the clear and a private key which is kept secret.
Public Key Pair	The two mathematically related keys—a public key and a private key—which, when used with the appropriate public key cryptographic algorithm, can allow the secure exchange of information, without the secure exchange of a secret.
Purchase Transaction	A retail purchase of goods or services; a point-of-sale transaction.
Random Selection	An EMV online-capable terminal function that allows for the selection of transactions for online processing. Part of the Terminal Risk Management function.
Receipt	A paper record of a transaction generated for the cardholder at the point of transaction.

Response	A message returned by the ICC to the terminal after the processing of a command message received by the ICC.
RSA (Rivest, Shamir, Adleman)	A public key cryptosystem developed by Rivest, Shamir, and Adleman, used for data encipherment and authentication.
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. Typically used to provide post-issuance updates to application data.
Secret Key	A key used with symmetric cryptographic techniques and usable only by a set of specified entities. It cannot be disclosed publicly without compromising the security of the system. This is not the same as the private key in a public/private key pair.
Secure Messaging	A process that enables messages to be sent from one entity to another, and protects against unauthorised modification or viewing.
Session Key	A temporary cryptographic key computed in volatile memory and not valid after a session is ended.
Static Data Authentication (SDA)	A type of Offline Data Authentication whereby the terminal validates a cryptographic value placed on the card during personalisation. This validation protects against some types of counterfeit, but does not protect against copying and replaying.
Symmetric Cryptographic Technique	A cryptographic technique that uses the same secret key for both the originator's and recipient's transformation. Without knowledge of the secret key, it is computationally infeasible to compute either the originator's or the recipient's transformation.
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. The terminal incorporates the interface device and may also include other components and interfaces such as host communications.

Terminal Action Code (TAC)	<p>Rules which the terminal uses to determine whether a transaction should be declined offline, sent online for an authorisation, or declined if online is not available. EMV defines the following, which reflect the acquirer-selected action to be taken upon analysis of the TVR data element:</p> <ul style="list-style-type: none">• TAC - Default: rules that determine when a transaction should be declined offline if it cannot go online• TAC - Denial: rules that determine when a transaction should be declined offline• TAC - Online: rules that determine when a transaction should be sent online for authorisation
Terminate Transaction	Stop the application processing for the current transaction and deactivate the card.
Token Authentication	A non-payment functionality supported in the application which allows the card to generate a token that can be used to authenticate that the card and cardholder are valid.
Transaction	An action taken by a terminal at the user's request. For a POS terminal, a transaction might be payment for goods, etc. A transaction selects among one or more applications as part of its processing flow.
Transaction Certificate (TC)	An Application Cryptogram generated when accepting a transaction
Transient Data	Data that is specific to the current transaction. This data is reset at the beginning of each transaction.
Velocity Checking	A card risk management function used to control how much is being spent offline or how many transactions are being conducted offline.
Visa Low-Value Payment (VLP)	Allows for quick low-value transactions at terminals that support the functionality.

4 Abbreviations, Notations, Conventions, Terminology, and Symbols

4.1 Abbreviations

a	Alphabetic
AAC	Application Authentication Cryptogram
AC	Application Cryptogram
ADR	Application Decisional Results
AEF	Application Elementary File
AFL	Application File Locator
AID	Application Identifier
AIP	Application Interchange Profile
an	Alphanumeric
ans	Alphanumeric Special
App.	Application
ARC	Authorisation Response Code
ARPC	Authorisation Response Cryptogram
ARQC	Authorisation Request Cryptogram
ATC	Application Transaction Counter
ATM	Automated Teller Machine
AUC	Application Usage Control
auth.	Authentication
b	Binary
BER	Basic Encoding Rules (defined in ISO/IEC 8825-1)
C	Conditional
CA	Certification Authority
CCD	Common Core Definitions
CCI	Common Core Identifier
CDA	Combined DDA/Application Cryptogram Generation

CDOL	Card Risk Management Data Object List
Cert.	Certificate
CIAC	Card Issuer Action Code
CID	Cryptogram Information Data
CLA	Class Byte of the Command Message
CPA	Common Payment Application
CPS	EMVCo Common Personalisation Specification
CRM	Card Risk Management
CSU	Card Status Update
CV	Cryptogram Version
CV Rule	Cardholder Verification Rule
CVM	Cardholder Verification Method
CVR	Card Verification Results
DDA	Dynamic Data Authentication
DDF	Directory Definition File
DDOL	Dynamic Data Authentication Data Object List
DES	Data Encryption Standard
DKI	Derivation Key Index
DOL	Data Object List
EMV	Europay, MasterCard, Visa
FC	Format Code
FCI	File Control Information
FIPS	Federal Information Processing Standard
GEN AC	GENERATE APPLICATION CRYPTOGRAM
GPO	GET PROCESSING OPTIONS
hex.	Hexadecimal
HSM	Hardware Security Module
IA	Issuer Authentication
IAC	Issuer Action Code (Denial, Default, Online)
IAD	Issuer Application Data
ICC	Integrated Circuit Card
IDN	ICC Dynamic Number

INS	Instruction Byte of the Command Message
Int'l	International
ISO	International Organization for Standardization
L	Length
Lc	Length of the Command Data Field
L _D	Length of the Plaintext Data in the Command Data Field
Le	Maximum Expected Length of the Response Data Field
LEN	Length
M	Mandatory
MAC	Message Authentication Code
MK	ICC Master Key for Session Key Generation
MK _{AC}	Master Key for Application Cryptogram Generation
MK _{SMC}	Master Key for Secure Messaging for Confidentiality
MK _{SMI}	Master Key for Secure Messaging for Integrity
MTA	Maximum Transaction Amount
n	Numeric
N/A	Not Applicable
N _{CA}	Length of the Certification Authority Public Key Modulus
N _I	Length of the Issuer Public Key Modulus
N _{IC}	Length of the ICC Public Key Modulus
N _{PE}	Length of the ICC PIN Encipherment Public Key Modulus
New Lc	Length of Command Data including Secure Messaging Components
P1	Parameter 1
P2	Parameter 2
PAN	Primary Account Number
PDOL	Processing Options Data Object List
PIN	Personal Identification Number
PK	Public Key
POS	Point Of Service
PTH	Previous Transaction History
Req	Requirement

4.1 Abbreviations

RFU	Reserved for Future Use
RID	Registered Application Provider Identifier
RSA	Rivest, Shamir, Adleman
SDA	Static Data Authentication
SFI	Short File Identifier
SMC	Secure Messaging for Confidentiality
SMI	Secure Messaging for Integrity
SSAD	Signed Static Application Data
SW1	Status Byte One
SW2	Status Byte Two
TAC	Terminal Action Code(s) (Default, Denial, Online)
TC	Transaction Certificate
TLV	Tag-Length-Value
TSI	Transaction Status Information
TVR	Terminal Verification Results
Txn	Transaction
var.	Variable
VLP	Visa Low-Value Payment
YYMM	year, month
YYMMDD	year, month, day

4.2 Notations

'0' to '9' and 'A' to 'F'	16 hexadecimal characters
'Bit Name'	The bit defined as "Bit Name" in a data element.
xb, xxb	Binary values
xx	Any value
$A := B$	A is assigned the value of B
$A = B$	The value of A is equal to the value of B
AND	Logical AND
OR	Logical OR
$X \oplus Y$	The symbol ' \oplus ' denotes bit-wise exclusive-OR and is defined as follows: $X \oplus Y$ The bit-wise exclusive-OR of the data blocks X and Y. If one data block is shorter than the other, then it is first padded to the left with sufficient binary zeros to make it the same length as the other.
[bx]	Bit x of the referenced data element
[x]	Byte x of the referenced data element
[x][by]	Bit y of byte x of the referenced data element
A / n	The integer division of A by n; that is, the unique integer d for which there exists an integer r, $0 \leq r < n$, such that: $A = dn + r$
$x * y$	Multiply x by y
$C := (A \parallel B)$	The concatenation of an n-bit number A and an m-bit number B, which is defined as $C = 2^m A + B$.
Element x	Instance x of data element Element (for example, Accumulator x could be Accumulator 1 or Accumulator 2).
Leftmost	Applies to a sequence of bits, bytes, or digits and used interchangeably with the term "most significant". If $C = (A \parallel B)$ as above, then A is the leftmost n bits of C.
Rightmost	Applies to a sequence of bits, bytes, or digits and used interchangeably with the term "least significant". If $C = (A \parallel B)$ as above, then B is the rightmost m bits of C.

4.3 Data Element Format Conventions

The EMV specifications use the following data element formats:

- a Alphabetic data elements contain a single character per byte. The permitted characters are alphabetic only (a to z and A to Z, upper and lower case).
- an Alphanumeric data elements contain a single character per byte. The permitted characters are alphabetic (a to z and A to Z, upper and lower case) and numeric (0 to 9).
- ans Alphanumeric Special data elements contain a single character per byte. The permitted characters and their coding are shown in the Common Character Set table in *EMV Book 4*, Annex B.
There is one exception: The permitted characters for Application Preferred Name are the non-control characters defined in the ISO/IEC 8859 part designated in the Issuer Code Table Index associated with the Application Preferred Name.
- b These data elements consist of either unsigned binary numbers or bit combinations that are defined elsewhere in the specification.
Binary example: The Application Transaction Counter (ATC) is defined as “b” with a length of two bytes. An ATC value of 19 is stored as Hex '00 13'.
Bit combination example: Processing Options Data Object List (PDOL) is defined as “b” with the format shown in *EMV Book 3*, section 5.4.
- cn Compressed numeric data elements consist of two numeric digits (having values in the range Hex '0'–'9') per byte. These data elements are left justified and padded with trailing hexadecimal 'F's.
Example: The Application Primary Account Number (PAN) is defined as “cn” with a length of up to ten bytes. A value of 1234567890123 may be stored in the Application PAN as Hex '12 34 56 78 90 12 3F FF' with a length of 8.
- n Numeric data elements consist of two numeric digits (having values in the range Hex '0' – '9') per byte. These digits are right justified and padded with leading hexadecimal zeroes. Other specifications sometimes refer to this data format as Binary Coded Decimal (“BCD”) or unsigned packed.
Example: Amount, Authorised (Numeric) is defined as “n 12” with a length of six bytes. A value of 12345 is stored in Amount, Authorised (Numeric) as Hex '00 00 00 01 23 45'.

var. Variable data elements are variable length and may contain any bit combination. Additional information on the formats of specific variable data elements is available elsewhere.

4.4 Terminology

This section clarifies several terms used throughout the specification.

Proprietary

“Proprietary” indicates concepts that are not defined in this specification and/or that are outside the scope of this specification.

Mandatory/Required/Recommended/Optional

One objective of CPA is to define a core set of functionality that must be available to issuers in every implementation of CPA. The minimum requirements and issuer-options reflect the EMV and CCD mandatory items in addition to requirements the payment systems determine must be available to all issuers. All other functionality is optional and not required.

Many features required to be supported by a CPA implementation are not mandated to be used by issuers. This document specifies requirements for an implementation of the CPA specification.

- Functionality that is an option for the application vendor to implement is called an *implementer-option*, and the functionality is characterised as *implementer-optional*. If implemented, it is the issuer’s choice whether to use the functionality.
- Functionality that is an option for the Issuer to use is called an *issuer-option* and the functionality is characterised as *issuer-optional*.

The following terminology is used to indicate these distinctions.

mandatory required shall issuer-option	minimum requirements for CPA
should	recommended functionality
optional may implementer-option	elective data elements and functions

Table 4-1: Terminology for Required and Optional Functionality

Markets can customise their card applications beyond the minimum requirements through adoption of the optional functions and through proprietary processing. Proprietary processing, however, must not interfere with global interoperability.

Card/Integrated Circuit

In general, the term “card” is used to describe functions performed by the CPA application on the card. When it is necessary to distinguish between the chip itself and another card feature such as the magnetic stripe, the term “integrated circuit” may be used.

4.5 Requirement Notation

Requirements for the CPA are identified and numbered in bold, and the required behaviour is written in italics to distinguish it from explanations of application behaviour.

Example:

Req x.x:

If the test is true, set the bit to the value 1b.

4.6 Flow Chart Symbols

This specification uses the following symbols in flow charts and in the diagrams in Annex H.

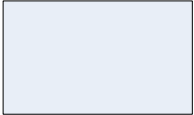
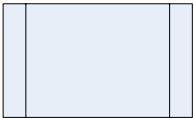
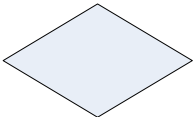
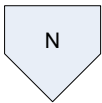




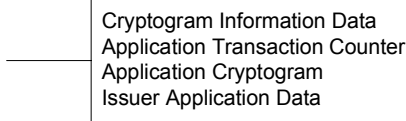

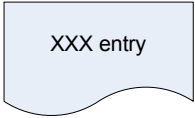
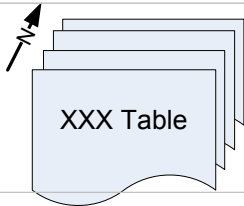
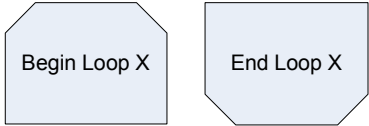
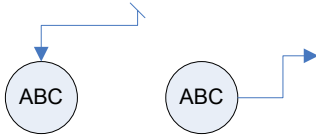
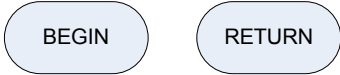
	Processing Step
	Sub-process
	Decision
	Off-page Connector (from N to N)
	Send Response to Terminal
	Out of Scope

Figure 4-1: Flow Chart Symbols
(continues)

	Predefined Process (Continue w/ Off-page Connector)
	Reference to Document Section
	Data Sub-elements
	Data Element
	Logical Data Element
	Set of Logical Data Elements
	Begin and End of a Repetitive Code Loop
	On-page Connectors
	Flow Chart Terminators

Part II

Introduction to Processing

5 Overview

This section describes the processing of an EMV transaction for a CPA-compliant application. It includes:

- a description of the CPA implementer-options
- an overview of each of the functional processes
- a sample transaction flow showing the order in which these functional processes may be performed and the commands sent by the terminal to the card for communications
- tables summarising functionality and command support requirements

5.1 Implementer-Options

As described in section 4.4, implementer-options denote functionality that is optional for the application vendor to implement. To minimise the options to be selected by an issuer when procuring CPA cards, this specification includes only the following implementer-options:

Implementer-option	Description
EMV CPS	A card that supports this implementer-option can be personalised using the personalisation method described in the <i>EMV Card Personalisation Specification</i> .
Dynamic-RSA	A card that supports this implementer-option is capable of DDA, CDA, and offline enciphered PIN verification.
VLP	A card that supports this implementer-option is capable of processing a transaction as a Visa Low-Value Payment.
Profile Selection Using Card Data	A card that supports this implementer-option is capable of associating card data such as the Application Identifier - card (AID, tag '4F') with application information that will allow the card data to be used during Application Initiation to configure application behaviour.
Application Security Counters	<p>A card that supports this implementer-option implements security counters within the application, as described in Annex F.</p> <p>NOTE: The security requirements in section 20 apply regardless of whether the application supports the Application Security Counters implementation-option.</p>

Table 5-1: Implementer-options

5.2 Functional Overview

This section describes functions used in processing of a CPA transaction by the card and terminal.

- Functions marked as *mandatory* are performed for all transactions, though some steps within the mandatory functions may be optional.
- Functions *not* marked *mandatory* are performed based upon parameters in the card, application, and/or terminal.

5.2.1 Application Selection (mandatory)

This section has optional additional functionality beyond that in EMV if the Profile Selection Using Card Data implementer-option is supported.

5.2.1.1 Function General Description

For the terminal, Application Selection is an EMV function which has no additional functionality for CPA. When a CPA card is presented to an EMV terminal, the terminal determines which applications are supported by both the card and terminal. If there is only one mutually supported application and that application does not require cardholder confirmation, the terminal selects that application. If there are multiple mutually supported applications or a single application that requires cardholder confirmation, then the terminal displays all the mutually supported applications to the cardholder, and the cardholder selects the application to be used for payment. If these applications cannot be displayed, the terminal selects the highest priority eligible application not requiring cardholder confirmation as designated by the issuer during card personalisation.

If the Profile Selection Using Card Data implementer-option is not supported, then for the card application selection is an EMV function which has no additional functionality for CPA.

If the Profile Selection Using Card Data implementer-option is supported, then at the time of final selection of the application, the card uses card data such as the Application Identifier - card (AID, tag '4F') to determine which parameters will be used for the transaction during Application Initiation.

5.2.1.2 Condition of Execution

The terminal always executes this function, and the card responds.

5.2.2 Initiate Application Processing (mandatory)

This section has additional functionality beyond that in EMV.

5.2.2.1 Function General Description

When a CPA application is selected, the card may request data from the terminal. The requested data, the parameters selected during Application Selection (if the Profile Selection Using Card Data implementer-option is supported), and issuer-options are used to determine the list of functions supported by the card for this transaction, and the data to be read by the terminal for use in processing the transaction. The card may indicate different data or different support functions based upon characteristics of the transaction such as terminal country code.

5.2.2.2 Condition of Execution

The terminal always executes this function, and the card responds.

5.2.3 Read Application Data (mandatory)

This section has additional functionality beyond that in EMV.

5.2.3.1 Function General Description

The terminal reads the data indicated by the card.

5.2.3.2 Condition of Execution

The terminal always executes this function, and the card responds.

5.2.4 Offline Data Authentication (optional)

This section has additional functionality beyond that in EMV.

5.2.4.1 Function General Description

The terminal determines whether it should authenticate the card offline using either offline static or dynamic data authentication based upon the card and terminal support for these methods.

Offline Static Data Authentication (SDA) validates that important application data has not been altered since card personalisation. The terminal validates static (unchanging) data from the card using the Issuer Public Key Certificate and a digital signature. The Issuer Public Key is stored on the card as a public key certificate. The digital signature is produced by signing a hash of important application data using the Issuer Private Key. A match of the recovered hash with a generated hash of the actual application data proves that the data has not been altered.

Offline dynamic data authentication, like SDA, validates that the card data has not been fraudulently altered and additionally validates that the card is genuine. Offline dynamic data authentication has two forms: DDA and Combined DDA/Generate AC (CDA).

- For DDA, the terminal requests that the card generate a cryptogram over dynamic (transaction unique) data from the card and terminal using an ICC Private Key. The terminal verifies this dynamic signature using the ICC Public Key recovered from the ICC Public Key Certificate read from the card. The verification assures that the card is not a counterfeit card created with data skimmed (copied) from a legitimate card.
- For CDA, the generation of the dynamic signature is combined with the generation of the card's Application Cryptogram during Card Action Analysis to assure that the Application Cryptogram came from the valid card.

5.2.4.2 Condition of Execution

All cards support SDA if the issuer personalises the necessary data elements on the card in records read by the terminal using the READ RECORD command. In an application that supports the Dynamic-RSA implementer-option (described in section 4.4), support for both DDA and CDA is mandatory, but use of the DDA or CDA functionality is an issuer-option. Availability of the data to support Offline Data Authentication (SDA, DDA, or CDA) is an issuer-option and is indicated in the AIP.

5.2.5 Processing Restrictions (performed by the terminal)

This section has no change from EMV functionality.

5.2.5.1 Function General Description

In Processing Restrictions, the terminal uses data read from the card to determine if the transaction is allowed for the card. The terminal checks whether the effective date for the card has been reached, whether the card has expired, whether the application versions of the card and terminal match, and whether any Application Usage Control (AUC) restrictions are in effect. An issuer may use the Application Usage Control to restrict a card's use for domestic or international, ATM, cash, goods, services, or cashback.

5.2.5.2 Condition of Execution

The terminal always executes this function.

5.2.6 Cardholder Verification (mandatory)

This section has additional functionality beyond that in EMV.

5.2.6.1 Function General Description

Cardholder verification ensures that the cardholder is legitimate and the card is not lost or stolen. The terminal uses a Card Verification Method (CVM) List from the card to determine the type of verification to be performed. The CVM List establishes a priority of cardholder verification methods, which consider the capabilities of the terminal and characteristics of the transaction to prompt the cardholder for a specific method of cardholder verification. If the CVM is offline PIN, the terminal prompts the cardholder for a PIN and transmits the cardholder-entered PIN to the card, which compares it to a Reference PIN stored secretly in the card. The CVM List may also specify other methods of cardholder verification, such as online PIN, signature, or no cardholder verification required.

5.2.6.2 Condition of Execution

The terminal uses the AIP provided by the card in response to the GET PROCESSING OPTIONS command to determine that the application supports at least one cardholder verification method. The terminal shall use the cardholder verification related data in the ICC to determine which issuer-specified cardholder verification methods (CVMs) shall be executed.

5.2.7 Terminal Risk Management (performed by the terminal)

This section has no change from EMV functionality.

5.2.7.1 Function General Description

Terminal Risk Management checks whether the transaction amount is over the terminal floor limit, the account number is on an optional terminal exception file, or the merchant has forced the transaction online. Some transactions might be randomly selected for online processing.

5.2.7.2 Condition of Execution

The terminal uses the AIP provided by the card in response to the GET PROCESSING OPTIONS command to determine whether Terminal Risk Management must be performed. The terminal may choose to execute this function even if the AIP does not require the terminal to perform terminal risk management.

5.2.8 Terminal Action Analysis (performed by the terminal)

This section has no change from EMV functionality.

5.2.8.1 Function General Description

Terminal Action Analysis uses the results of Offline Data Authentication, Processing Restrictions, Terminal Risk Management, and Cardholder Verification, plus rules set in the card and terminal, to determine whether the transaction should be approved offline, sent online for authorisation, or declined offline. The card rules are set in fields called Issuer Action Codes (IACs) read by the terminal from the card; the terminal rules are set in Terminal Action Codes (TACs). After determining whether the transaction should be approved offline, sent online for authorisation, or declined offline; the terminal requests an application cryptogram from the card. The type of application cryptogram is one of the following: Transaction Certificate (TC) for an offline approval, Authorisation Request Cryptogram (ARQC) for an online request, or Application Authentication Cryptogram (AAC) for an offline decline.

5.2.8.2 Condition of Execution

The terminal always executes this function.

5.2.9 First Card Action Analysis (mandatory)

This section has additional functionality beyond that in EMV.

5.2.9.1 Function General Description

Upon receiving the first GENERATE AC command from the terminal, the card performs Card Action Analysis where Card Risk Management checks are performed to determine whether to change the transaction disposition requested by the terminal. These may include checks for count or amount velocity checking limits having been reached, prior incomplete online transactions, and failure of either Issuer Authentication or offline data authentication on a previous transaction. The card may convert a terminal request for an offline approval to an online transaction or an offline decline, and the card may convert an online request to an offline decline. The card must decline a transaction if the terminal decision is to decline the transaction.

After completion of the checks, the card generates the application cryptogram using application data and a secret key stored on the card. It returns this cryptogram to the terminal. For offline approved transactions, the cryptogram type is a TC. For offline declined transactions, the cryptogram type is an AAC. For transactions sent online, the cryptogram type is an ARQC.

5.2.9.2 Condition of Execution

The application always executes this function.

5.2.10 Online Processing (conditional – if first AC type was ARQC)

This section has no change from EMV functionality.

5.2.10.1 Function General Description

If the terminal is online-capable and the card determines that the transaction requires an online authorisation, then the terminal transmits an online authorisation request message to the issuer. This message includes the ARQC cryptogram, the data used to generate the ARQC, and indicators showing offline processing results. During online processing, the issuer validates the ARQC to authenticate the card in a process called Online Card Authentication. The issuer may consider the result of Online Card Authentication and the offline processing results in its authorisation decision.

The authorisation response message transmitted back to the terminal may include Issuer Authentication Data. Issuer Authentication Data contains an issuer-generated Authorisation Response Cryptogram (ARPC) (generated from the ARQC, the Card Status Update, and the card's secret key) and the Card Status Update. The response may also include post-issuance updates to the card called Issuer Script commands.

NOTE: Issuer Authentication Data may also contain Proprietary Authentication Data. Support for receipt of Proprietary Authentication Data and functionality associated with the additional data is allowed as additional functionality (see section 19.3.5).

NOTE: Issuer Authentication in CPA is performed as part of second GENERATE AC command processing instead of using an EXTERNAL AUTHENTICATE command.

5.2.10.2 Condition of Execution

The terminal executes this function if the application returns an ARQC in response to the first GENERATE AC command for the transaction.

5.2.11 Second Card Action Analysis (conditional – if first AC type was ARQC)

This section has additional functionality beyond that in EMV.

5.2.11.1 Function General Description

Upon receiving the second GENERATE AC command from the terminal, the card performs Second Card Action Analysis. An issuer-approved transaction may be converted to a decline based upon Issuer Authentication results and issuer-options in the card.

If the transaction was unable to go online, additional Card Risk Management checks are performed to determine whether to change the transaction disposition sent by the terminal. These include checks for count or amount velocity checking limits having been reached, prior incomplete online transactions, failure of Issuer Authentication or offline data authentication failure on a previous transaction.

If the authorisation response contains an ARPC, the application performs Issuer Authentication by validating the ARPC to verify that the response came from the genuine issuer (or its proxy). Successful Issuer Authentication may be required for resetting certain counters and indicators in the card. Successful Issuer Authentication is required for the application to process the Card Status Update. Requiring Issuer Authentication prevents circumventing the card's security features by simulating online processing and fraudulently approving a transaction to reset card counters and indicators.

If the authorisation response does not contain an ARPC, the card uses the transaction disposition and issuer-encoded rules to determine whether to update counters and indicators in the card.

The card generates a TC for approved transactions or an AAC for declined transactions.

After completion of the checks, the card generates the application cryptogram using application data and a secret key stored on the card. It returns this cryptogram to the terminal.

5.2.11.2 Condition of Execution

The application executes this function if the ICC returns an ARQC in response to the first GENERATE AC command for the transaction.

5.2.12 Issuer Script Command Processing (optional)

This section has additional functionality beyond that in EMV.

5.2.12.1 Function General Description

If the issuer includes a script in the authorisation response message, the terminal passes the script commands to the card. Prior to applying the updates, the card performs security checking to assure that the script came from the valid issuer and was not altered in transit. Supported script commands allow updating offline processing parameters, unblocking the application, and changing the offline PIN value.

5.2.12.2 Condition of Execution

The application executes this function if script commands are received from the terminal.

5.3 Sample Transaction Flow

The sample transaction flow shown in Figure 5-1 illustrates a possible flow for a CPA transaction at an EMV-compliant terminal. Note that the flow of processes is predominantly driven by the terminal, not the application on the ICC. The ICC and terminal communicate through commands sent from the terminal to the ICC, and responses received by the terminal from the ICC.

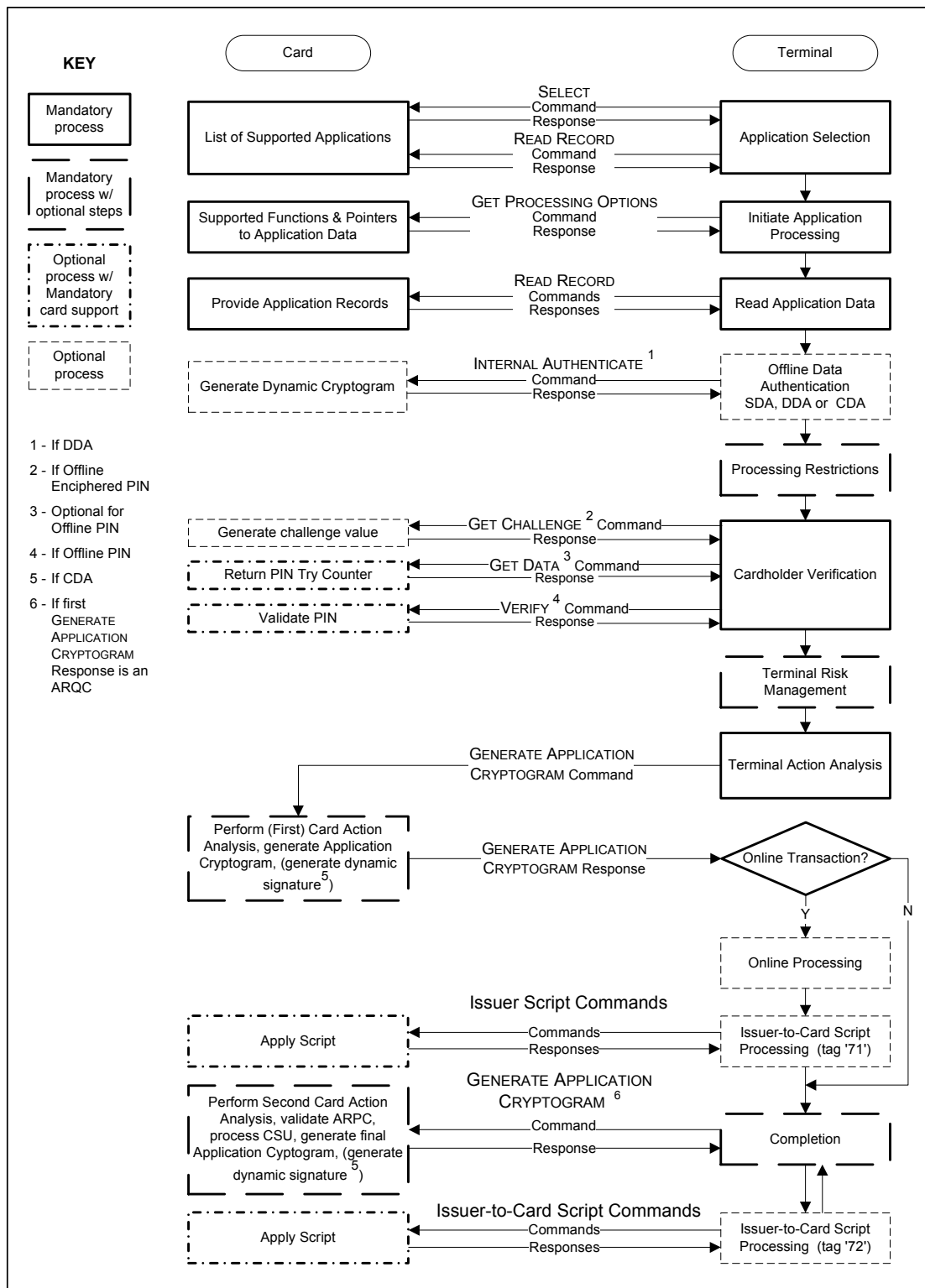


Figure 5-1: Sample Transaction Flow

5.4 Minimum Functionality

This section identifies the minimum functionality for the application. Terminal requirements are specified in EMV.

5.4.1 Card Functional Requirements

Req 5.1 (Mandatory functionality):

CPA cards shall support the mandatory functions listed in Table 5-2.

NOTE: Use of the supported functions may be an issuer-option. Table 5-2 indicates functionality that must be supported by the implementer so that the functionality is available for use by the issuer in all CPA implementations.

Req 5.2 (Conditional functionality):

CPA cards shall support the conditional functions listed in Table 5-2 if the associated condition is true.

NOTE: Use of the supported functions may be an issuer-option. Table 5-2 indicates functionality that must be supported by the implementer if the condition is true, so that the functionality is available for use by the issuer in all CPA implementations for which the condition is true.

Function	Card Support
<i>Application Selection</i>	<i>Mandatory</i>
▪ <i>Directory Method</i>	<i>Mandatory</i>
▪ <i>Partial AID Selection</i>	<i>Mandatory</i>
▪ <i>Explicit Selection Method</i>	<i>Mandatory</i>
<i>Initiate Application Processing</i>	<i>Mandatory</i>
<i>Read Application Data</i>	<i>Mandatory</i>
<i>Offline Data Authentication</i>	
▪ <i>SDA</i>	<i>Personalisation requirement — The card shall have the capability to be personalised with the data necessary for the terminal to perform SDA.</i>
▪ <i>DDA</i>	<i>Conditional — If Dynamic-RSA implementer-option is supported</i>
▪ <i>CDA</i>	<i>Conditional — If Dynamic-RSA implementer-option is supported</i>

Table 5-2: Card Functional Requirements
(continues)

Function	Card Support
<i>Cardholder Verification</i> <ul style="list-style-type: none"> ▪ <i>Offline Plaintext PIN</i> ▪ <i>Offline Enciphered PIN</i> 	<i>Mandatory</i> <i>Mandatory</i> <i>Conditional — If Dynamic-RSA implementer-option is supported</i>
<i>First Card Action Analysis</i> <ul style="list-style-type: none"> ▪ <i>Online/offline decision</i> ▪ <i>Card Risk Management</i> ▪ <i>Application Cryptogram</i> ▪ <i>Transaction Logging</i> 	<i>Mandatory</i> <i>Mandatory</i> <i>Mandatory</i> <i>Mandatory, algorithm specified by CCD for Cryptogram Version '5'</i> <i>Mandatory</i>
<i>Second Card Action Analysis</i> <ul style="list-style-type: none"> ▪ <i>Issuer Authentication</i> ▪ <i>CSU Processing</i> ▪ <i>Transaction Logging</i> 	<i>Mandatory</i> <i>Mandatory</i> <i>Mandatory</i> <i>Mandatory</i>
<i>Issuer-to-Card Script Processing</i> <ul style="list-style-type: none"> ▪ <i>Secure Messaging</i> 	<i>Mandatory</i> <i>Mandatory, form specified in CCD</i>

Table 5-2: Card Functional Requirements, continued

5.4.2 Command Support Requirements

Req 5.3 (Supported commands):

The application shall support the mandatory commands listed in Table 5-3. The application shall support the conditional commands listed in Table 5-3 if the associated condition is true.

Command	CLA	INS	Card Support
<i>APPLICATION UNBLOCK</i>	'8C'	'18'	<i>Mandatory</i>
<i>GENERATE APPLICATION CRYPTOGRAM (GENERATE AC)</i>	'80'	'AE'	<i>Mandatory</i>
<i>GET CHALLENGE</i>	'00'	'84'	<i>Conditional — If Dynamic-RSA implementer-option is supported</i>
<i>GET DATA</i>	'80'	'CA'	<i>Mandatory</i>
<i>GET PROCESSING OPTIONS</i>	'80'	'A8'	<i>Mandatory</i>
<i>INTERNAL AUTHENTICATE</i>	'00'	'88'	<i>Conditional — If Dynamic-RSA implementer-option is supported</i>
<i>PIN CHANGE/UNBLOCK</i>	'8C'	'24'	<i>Mandatory</i>
<i>PUT DATA</i>	'0C'	'DA'	<i>Mandatory</i>
<i>READ RECORD</i>	'00'	'B2'	<i>Mandatory</i>
<i>SELECT</i>	'00'	'A4'	<i>Mandatory</i>
<i>UPDATE RECORD</i>	'0C'	'DC'	<i>Mandatory</i>
<i>VERIFY</i>	'00'	'20'	<i>Mandatory</i>

Table 5-3: Command Support Requirements

6 General Command Information

The requirements for command processing in this specification apply during the processing of an EMV-compliant financial transaction.

6.1 Implementation of the CPA Specifications

The purpose of this specification is to unambiguously define the external behaviour of the CPA application. In order to clearly specify this external behaviour it has been helpful to specify some internal processing and introduce some internal data elements (for example, internal flags, bits in the Previous Transaction History) that are one possible way to implement the required external behaviour. For example, in many of the flows the sequence of some actions during command processing does not make any difference to the external behaviour. It is nevertheless necessary to describe the internal processing flow and the internal data in order to clearly present application concepts. Implementations are not required to implement internal functionality using the data elements and processes strictly as defined in this specification. However, implementations are required to behave in a way that is indistinguishable (external to the application) from the behaviour that is described using the internal data elements and processes.

CPA is a CCD-compliant implementation of EMV.

Req 6.1 (Implementation of CCD):

The application shall meet the requirements for a CCD-compliant EMV application.

NOTE: CCD compliance is dependent on personalisation of parameters that configure application behaviour. See section 21.3 for personalisation requirements to comply with CCD.

Implementations are **not required** to strictly follow the flow diagrams describing the application behaviour. However, implementations are required to behave in a way that is indistinguishable (seen as a black box responding to commands) from the behaviour described by the flow diagrams. Whether or not an implementation internally follows the flow diagrams is not a condition for compliance with this specification.

Req 6.2 (Behaviour according to flow diagrams):

*Implementations must behave **as if** they follow the flow diagrams.*

In the sections describing the command processing, implementation notes identify where different implementations could lead to the same behaviour. This list is not exhaustive and implementers are welcome to use any platform-specific feature or optimisation that would facilitate the implementation while leaving the external card behaviour unchanged.

6.2 State Machine

The application operates by receiving a command from an EMV terminal, processing it, then generating and sending a response to the terminal. After this, the application is ready to process a new command.

In this document, the application is described as a state machine. As a result of the application receiving and processing a command, the state may change before the application accepts the next command from the terminal. The processing of commands leads to transitions within the state machine.

6.2.1 States

After the application has been personalised, the behaviour can be specified as a state machine. Table 6-1 provides the application states used in this description of CPA.

State	Description
SELECTED	Application is selected.
INITIATED	Transaction is initiated.
ONLINE	Application expects a connection with the issuer. It is also ready to accept a script command.
SCRIPT	Application is ready to accept a script command.

Table 6-1: Application States

SELECTED

Every transaction starts in the state **SELECTED**. The application goes to the state **SELECTED** upon receiving the **SELECT** command that selects the application to be used by the card and terminal for processing the transaction.

NOTE: When the card is in the **SELECTED** state:

- If the application has exceeded the maximum number of transactions allowed over the lifetime of the card (see section 8.5.2), then the card will respond to the **GET PROCESSING OPTIONS** command with an error status that allows the terminal to return to application selection for selection of a different application (if any). The state transitions for any other command are those indicated in Table 6-2.
- If the application is blocked, then the card responds with SW1 SW2 = '6285' (Selected file in termination state) in the **SELECT** command response and the application is either not added to, or removed from, the candidate list; depending on when the **SELECT** occurs.
- If the application is blocked, then the state transitions are those indicated in Table 6-2. This allows selection of a blocked application so that a script command may unblock the application. This functionality is done at special devices¹, and does not follow the EMV transaction flow.

INITIATED

The CPA application goes to state **INITIATED** after the processing of the **GET PROCESSING OPTIONS** command. In this state, a new transaction is initiated. Processing of the **GET DATA** and the **READ RECORD** commands does not cause the application to transition to a new state. Processing of the first **GENERATE AC** command causes the application to transition to either the **ONLINE** or the **SCRIPT** state.

NOTE: When the card is in the **INITIATED**, **ONLINE**, or **SCRIPT** state and the application is blocked, then the state transitions are those indicated in Table 6-2. The application will respond to the first **GENERATE AC** command with an AAC, as described in first Card Action Analysis chapter.

ONLINE

The CPA goes to state **ONLINE** after the processing of the first **GENERATE AC** command if the response is an **ARQC**. In this state, the application is expecting a response from the issuer. The application accepts script commands in this state. It also accepts a second **GENERATE AC** command, which may carry a response from the issuer. Processing of the second **GENERATE AC** command causes the application to transition to the **SCRIPT** state.

¹ An application is typically unblocked at a device that is designated or controlled by the issuer, and that does not process payment transactions.

SCRIPT

The CPA goes to state **SCRIPT** after responding with a TC or an AAC to a GENERATE AC command, in order to receive and process any script commands. (See Table 6-2 for state transitions.)

6.2.2 Sequence of Commands and Transition Between States

In Table 6-2, the column headings describe the current state of the application when a command is received from the terminal. The row headings describe the command received. Table 6-2 describes the transition between states that the application shall make upon the successful execution of CPA commands (that is, with SW1 SW2 = '9000' in the command response, or for the VERIFY command only, SW1 SW2 = '63Cx'). Each table entry gives the resulting state when execution of the command is successful.²

Conditions resulting in an error response to the terminal are described in the command processing sections in Part III of this document. If an error occurs in command processing for GENERATE AC or GET PROCESSING OPTIONS, in a state in which the command is allowed, the application shall transition to the **SELECTED** state. For an error in processing any other command, the application shall remain in the current state.

² “Not Allowed” indicates that the command is not permitted when the card is in the given state. The card shall discontinue processing the command, shall respond with an SW1 SW2 that indicates an error, should respond with SW1 SW2 = '6985' (Conditions of use not satisfied), and shall remain in the current state.

“Not Supported” indicates that the command is not applicable for an EMV transaction when the card is in the given state. The card should discontinue processing the command and respond with SW1 SW2 = '6985' (Conditions of use not satisfied), and remain in the current state.

State ⇒ Command ↓	SELECTED	INITIATED	ONLINE	SCRIPT
APPLICATION UNBLOCK	Not Allowed	Not Allowed	ONLINE	SCRIPT
GENERATE AC	Not Allowed	SCRIPT (if response is TC or AAC and SW1 SW2 = '9000') ONLINE (if response is ARQC and SW1 SW2 = '9000') SELECTED (if SW1 SW2 ≠ '9000')	SCRIPT (if SW1 SW2 = '9000') SELECTED (if SW1 SW2 ≠ '9000')	Not Allowed ³
GET CHALLENGE	SELECTED	INITIATED	ONLINE	SCRIPT
GET DATA	SELECTED	INITIATED	ONLINE	SCRIPT
GET PROCESSING OPTIONS	INITIATED (if SW1 SW2 = '9000') SELECTED (if SW1 SW2 ≠ '9000')	Not Allowed	Not Allowed	Not Allowed
INTERNAL AUTHENTICATE	Not Supported	INITIATED	Not Supported	Not Supported
PIN CHANGE/ UNBLOCK	Not Allowed	Not Allowed	ONLINE	SCRIPT
PUT DATA	Not Allowed	Not Allowed	ONLINE	SCRIPT
READ RECORD	SELECTED	INITIATED	ONLINE	SCRIPT
SELECT	SELECTED	SELECTED	SELECTED	SELECTED
UPDATE RECORD	Not Allowed	Not Allowed	ONLINE	SCRIPT
VERIFY	Not Supported	INITIATED	Not Supported	Not Supported

Table 6-2: Sequence of Commands and State Transitions for Commands

³ EMV requires that the card respond with SW1 SW2 = '6985' for a third and subsequent GENERATE AC commands received by a card in a single transaction.

6.3 Command Validation

Whenever a command is received by the application, the first step is to validate the command.

Req 6.3 (Command validation):

If the application receives an unknown command, it shall discontinue processing the command and shall respond with an SW1 SW2 indicating an error.

- *If the CLA is unknown, the card should respond with SW1 SW2 = '6E00' (CLA not supported).*
- *If the CLA is known, but the INS is unknown for the CLA, the card should respond with SW1 SW2 = '6D00' (Wrong INS).*
- *If both the CLA and INS are unknown, the card should respond with SW1 SW2 = either '6E00' or '6D00'.*

6.4 Exception Handling

6.4.1 Status Words

In addition to the command validation specified in section 6.3, each command processing section may identify further mandatory and recommended status words.

Application implementations may respond with other status words to indicate conditions in addition to those specified in CPA. For instance, applications may respond with implementation-specific status words when data required for the execution of a function is not present (that is, has not been personalised) or when data is corrupt. These additional status words are left to the implementation.

NOTE: The additional status words may cause termination of the transaction by the terminal.

6.4.2 Missing or Invalid Resources

During application processing, an application resource may be missing or invalid (for example, a resource referenced by the resource ID in one of the internal card data objects may be missing or incorrectly coded). In such a case, the action to be taken depends on the type of resource that is missing or invalid. Section 21.4 indicates the appropriate action to take if the resource is missing or invalid.

For the sake of clarity, flow charts in the following chapters do not contain error processing for all types of missing resources. Only those cases where command processing must be discontinued (and an error status returned to the terminal), or cases where a default value should be used, are illustrated.

Part III

Function Processing

7 Application Selection

This section is organised as follows:

- 7.1 Purpose
- 7.2 Sequence of Execution
 - 7.2.1 Subsequent Related Processing
- 7.3 Processing
 - 7.3.1 Building the Candidate List
 - 7.3.2 Identifying and Selecting the Application

7.1 Purpose

Application Selection is an application-independent function performed by the card and terminal to select which of the applications that are supported by both the card and terminal will be used to conduct the transaction.

7.2 Sequence of Execution

7.2.1 Subsequent Related Processing

Initiate Application Processing

If the Profile Selection Using Card Data implementer-option is supported, then the card identifies which parameters will be used for the transaction during Application Initiation.

If the card requires terminal data during Application Initiation, the Processing Options Data Object List (PDOL) was included in the SELECT response during Application Selection. The PDOL is a list of tags and lengths for terminal-resident data objects needed by the card in processing the GET PROCESSING OPTIONS command during Initiate Application Processing. The GET PROCESSING OPTIONS command sent to the card by the terminal includes any terminal data that was specified in the PDOL. If the card does not require terminal data during Application Initiation, the PDOL was not included in the SELECT response.

If the selected application cannot be initiated, the terminal terminates that application and returns to Application Selection for selection of another application.

7.3 Processing

The processing performed by the terminal and card to build the list of candidate applications is described in EMV. Application Selection takes place in two steps:

1. The terminal builds a candidate list of mutually supported applications.
2. A single application from this list is identified and selected to process the transaction.

The terminal sends the SELECT command to the card to obtain information on the applications supported by the card. The application information may include issuer preferences such as the priority in which the application is selected, application name, and language preference. The command either contains the name of the Payment Systems Environment directory (used for the PSE method) or a requested AID (used for the List of AIDs method and for final selection of the application).

Application Selection is performed as described for a CCD-compliant application in EMV Book 1, section 12 and EMV Book 4, section 11.3.

7.3.1 Building the Candidate List

There are two approaches used by the terminal to build a list of mutually supported applications.

- The PSE method is described in *EMV Book 1*, section 12.3.2. This method is optional for cards and terminals, but, if supported by the terminal, it is attempted first.

In the PSE method, the terminal selects the Payment System Environment and reads the Payment System Directory from the card using the READ RECORD command described in *EMV Book 1*, section 11.2. This file is a list of the payment applications supported by the card. The terminal adds any applications listed in both the card list and the terminal list to the candidate list.

- The List of AIDs method is described in *EMV Book 1*, section 12.3.3. Support of this method is mandatory for cards and terminals. In the List of AIDs method, the terminal issues a SELECT command for each terminal-supported application. If the card response indicates that the application is supported by the card, the terminal adds the application to the candidate list.

NOTE: Partial Selection is supported for CPA. See the CCD Part of *EMV Book 1* for application selection requirement.

7.3.2 Identifying and Selecting the Application

If there is at least one mutually supported application on the Candidate List, the terminal and cardholder determine which application to use.

NOTE: Issuers should be aware that setting their application to require confirmation by the cardholder (see *EMV Book 1*, section 12.4) will mean that the application cannot be selected by a terminal that does not support either Cardholder Confirmation or Cardholder Selection.

It is possible for a special terminal to select an application that was blocked in order to unblock the application. If this occurs, the card will return an AAC (offline decline) in response to a GENERATE AC command.

The card responds to the SELECT command with SW1 SW2 = '9000' if the card determines that the transaction can be performed with that application.

If the application is blocked, the card will discontinue processing the SELECT command and responds with SW1 SW2 = '6283' (Selected file invalidated).

If the card is blocked, the card will discontinue processing the SELECT command and respond with SW1 SW2 = '6A81' (Function not supported).

If the card contains a PDOL, it is part of the FCI data in the SELECT response. The terminal sends the data specified in the PDOL to the card during Initiate Application Processing.

Req 7.1 (Supported FCI length):

CPA shall be capable of responding with an FCI up to 240 bytes in length.

Personalisation requirements for the PDOL that are necessary to support CPA functionality are specified in section 21.1.

If the Profile Selection Using Card Data implementer-option is supported, then the card determines which entry in the GPO Parameters template will be used for the transaction during Application Initiation. The methods used to select the entry in GPO Parameters, and to indicate to the application which GPO Parameter x is used for the transaction is beyond the scope of the application.

Req 7.2 (Profile selection using card data implementer-option):

At the least, a card that supports the Profile Selection Using Card Data implementer-option shall be capable of associating each Application Identifier - card (AID, tag '4F') used to select the application with a different entry in the GPO Parameters template.

8 Initiate Application Processing

This section is organised as follows:

- 8.1 Purpose
- 8.2 Sequence of Execution
 - 8.2.1 Prior Related Processing
 - 8.2.2 Subsequent Related Processing
- 8.3 Card Data
- 8.4 Terminal Data
- 8.5 GET PROCESSING OPTIONS Command
 - 8.5.1 Command Coding
 - 8.5.2 Processing
 - 8.5.3 Profile Selection
 - 8.5.4 Profile Behaviour
 - 8.5.5 Respond to GET PROCESSING OPTIONS Command
- 8.6 Function Flow Charts

8.1 Purpose

During Initiate Application Processing, the terminal signals to the card that transaction processing is beginning. The terminal accomplishes this by sending the GET PROCESSING OPTIONS command to the card. When issuing this command, the terminal supplies the card with any data elements requested by the card in the Processing Options Data Objects List (PDOL). The PDOL (a list of tags and lengths of data elements) is optionally provided by the card to the terminal during Application Selection.

During processing of the GET PROCESSING OPTIONS command, the application performs Profile Selection, which enables the application to differentiate application functionality based on the transaction environment.

After Profile Selection, the card responds to the GET PROCESSING OPTIONS command with the Application Interchange Profile (AIP), a list of functions to be performed in processing the transaction. The card also provides the Application File Locator (AFL), a list of files and records that the terminal needs to read from the card.

Initiate Application Processing is performed as described in EMV Book 3, section 10.1.

8.2 Sequence of Execution

This shall be the first function performed after selection of the application.

8.2.1 Prior Related Processing

Application Selection

The card supplies the PDOL (if present) to the terminal as part of the FCI provided in response to the SELECT command.

If the Profile Selection Using Card Data implementer-option is supported, then the card identifies which parameters will be used for the transaction during Application Initiation.

8.2.2 Subsequent Related Processing

Application Selection

If GET PROCESSING OPTIONS command processing determined that the application is restricted from use, the card response allows the terminal to return to Application Selection for selection of another application.

Read Application Data

The AFL provided by the card in response to the GET PROCESSING OPTIONS command is used by the terminal to determine what application data to read from the card and which data is to be used in the Offline Data Authentication.

Offline Data Authentication

The terminal uses the AIP provided by the card in response to the GET PROCESSING OPTIONS command to determine the forms of Offline Data Authentication supported by the application.

Cardholder Verification

The terminal uses the AIP provided by the card in response to the GET PROCESSING OPTIONS command to determine whether the application supports Cardholder Verification. The sequence of verification methods is given in the CVM List contained in the record data read by the terminal.

Terminal Risk Management

The terminal uses the AIP provided by the card in response to the GET PROCESSING OPTIONS command to determine whether the terminal must perform Terminal Risk Management. The terminal may perform Terminal Risk Management regardless of the setting of the bit in the AIP.

First Card Action Analysis

The application uses options selected during Initiate Application Processing to control configurable card risk management functionality executed as part of processing the first GENERATE AC command. The terminal uses the AIP provided by the card in the GET PROCESSING OPTIONS response to determine whether the first GENERATE AC command may request CDA.

Online Processing

The terminal uses the AIP provided by the card in response to the GET PROCESSING OPTIONS command to determine whether the application supports Issuer Authentication using the EXTERNAL AUTHENTICATE command.

NOTE: CPA does not use the EXTERNAL AUTHENTICATE command. For CPA, Issuer Authentication is combined with second GENERATE AC command processing.

Second Card Action Analysis

The application uses options selected during Initiate Application Processing to control configurable card risk management functionality executed as part of processing the second GENERATE AC command. The terminal uses the AIP provided by the card in the GET PROCESSING OPTIONS response to determine whether the second GENERATE AC command may request CDA.

8.3 Card Data

The card data used in Initiate Application Processing are listed and described in Table 8-1. For a detailed description of these data and their usage, see Annex L: Data Dictionary.

Data	Description	Template	Tag
AIP/AFL Entry x	Consists of one AIP, the AFL Length, and one AFL, and can be selected using the Profile Selection Functionality. This allows for selection of some application functionality to be based on a test of transaction attributes.	'BF41'	'DF0x'
Application Control	Indicators used to activate or deactivate functions in the application.	—	'C1'
Application Decisional Results (ADR)	Indicators used internal to the application to identify exception conditions that occurred during the current and previous transactions. The Card Issuer Action Codes (CIACs) are compared to the Application Decisional Results to make decisions regarding whether to approve or decline the transaction offline, or request to go online.	—	—

Table 8-1: Initiate Application Processing – Card Data
(continues)

Data	Description	Template	Tag
Application File Locator (AFL)	<p>Indicates the file location and range of records which contain card data to be read by the terminal for use in transaction processing. For each file to be read, the AFL contains the following information:</p> <p>Byte 1 (b8-b4) Short File Identifier (a numeric label for a file)</p> <p>Byte 2 Record number of the first record to be read</p> <p>Byte 3 Record number of the last record to be read</p> <p>Byte 4 Number of consecutive records containing data to be used in Offline Data Authentication beginning with the first record to be read as indicated in Byte 2.</p>	—	'94'
Application Interchange Profile (AIP)	A list that indicates the capability of the application to support specific functions in the application (SDA, DDA, CDA, Terminal Risk Management, Cardholder Verification, and Issuer Authentication).	—	'82'
Application Transaction Counter (ATC)	Counter of transactions initiated for the application.	—	'9F36'
Card Verification Results (CVR)	Contains indicators that are set during the current transaction to indicate card risk management conditions that have occurred on the card. It is initialised to zero during Initiate Application Processing.	—	'9F52'

Table 8-1: Initiate Application Processing – Card Data, continued

Data	Description	Template	Tag
GPO Parameters x	Contains two parameters used in processing the GPO command: <ul style="list-style-type: none"> GPO Command Data Length Profile Selection Diversifier 	'BF3E'	'DF0x'
	GPO Command Data Length		
	Profile Selection Diversifier		
Previous Transaction History (PTH)	Contains indicators that identify for subsequent transactions that certain events have occurred; for example the application has been blocked, Issuer Authentication has failed, or the issuer has requested that the card go online on the next transaction.	—	'C7'
Processing Options Data Object List (PDOL)	The PDOL is a list of tags and lengths for terminal-resident data objects needed by the application in processing the GET PROCESSING OPTIONS command during Initiate Application Processing.	—	'9F38'

Table 8-1: Initiate Application Processing – Card Data, continued

Data	Description	Template	Tag
Profile Control x	Identifies which application resources are to be used when processing a transaction with Profile ID x. The only item in the Profile Control x data element used during Application Initiation processing is the following:	'BF3F'	'DFxx'
	AIP/AFL Entry ID	Identifies the AIP/AFL Entry x to be used when processing the transaction.	
Profile ID	Identifies the profile to be used for the transaction; associated with a description of application optional behaviours and data selected for use in processing a transaction.	—	—
Profile Selection File	A file personalised on the card that is used with PDOL-related data to determine which profile is to be used for the transaction. The application compares application data to PDOL-related data received in the GET PROCESSING OPTIONS command from the terminal to select the profile. The profile defines a number of card behaviours including the AIP and AFL to be returned in the GET PROCESSING OPTIONS response, counter and accumulator behaviour, and card decision parameters.	—	—
	Profile Selection Entries	Records in the Profile Selection File that contain the logic and data for one step in the profile selection process.	

Table 8-1: Initiate Application Processing – Card Data, continued

The application uses the following data elements in order to process and respond to the GET PROCESSING OPTIONS command: AIP/AFL Entries, Application Transaction Counter (ATC), GPO Command Data, GPO Command Data Length in GPO Parameters, Profile Controls, and Profile ID. If the application is to use terminal-sourced data in selection of the AIP and AFL for the GET PROCESSING OPTIONS command response, the application also uses the following data elements to perform Profile Selection File Processing: PDOL, Profile Selection Diversifier in GPO Parameters, and Profile Selection Entries in the Profile Selection File.

8.4 Terminal Data

The terminal data used in Initiate Application Processing are listed and described in Table 8-2. For a detailed description of these data and their usage, see Annex L: Data Dictionary.

Data	Description	Template	Tag
GPO Command Data	The application may use any of the data elements that can be requested from the terminal (using the PDOL) for the Profile Selection Processing.	—	—
VLP Terminal Support Indicator	A data element which, if present in the terminal, indicates that the terminal supports VLP processing.	—	'9F7A'

Table 8-2: Initiate Application Processing – Terminal Data

8.5 GET PROCESSING OPTIONS Command

The GET PROCESSING OPTIONS (GPO) command is used by the terminal to signal the application that transaction processing is beginning.

The command contains the value portion of terminal data elements requested by the application in the Processing Options Data Objects List (PDOL) that was optionally provided to the terminal by the card during Application Selection. The PDOL also specifies the number of bytes to be sent for each data element requested.

The card response is in Format 2 and is described in Section 6.5.8.4 of the CCD part of EMV Book 3. It contains only the Application Interchange Profile (AIP) specifying the functions supported by the card and the Application File Locator (AFL) specifying the files and records to be used for the transaction.

8.5.1 Command Coding

Code	Value
CLA	'80'
INS	'A8'
P1	'00'
P2	'00'
Lc	Var.
data	PDOL Related Data
Le	'00'

Table 8-3: GET PROCESSING OPTIONS Command Message

The command data contains the value in the terminal of data elements whose tags and lengths were listed in the PDOL, if a PDOL was sent to the terminal in the SELECT response.

PDOL Related Data follows command template '83' and is interpreted by the application as consisting of the data described in Table 8-4.

Name	Tag	Length	Value
PDOL Related Data	'83'	GPO Template Length	GPO Command Data

Table 8-4: PDOL Related Data

Req 8.1 (supported length for GPO command data):

At a minimum, the application shall support PDOL Related Data length in the range 2 to 128 bytes.

8.5.1.1 Command Format Validation

If the Profile Selection Using Card Data implementer-option is supported, then at the time of final selection of the application, the card selects the GPO Parameter x to be used for the transaction during Application Initiation. If the Profile Selection Using Card Data implementer-option is supported, the method used to indicate to the application which GPO Parameter x is used for the transaction is beyond the scope of the application.

Req 8.2 (Default GPO Parameter Entry):

If the Profile Selection Using Card Data implementer-option is not supported, then the GPO Parameter x to be used for the transaction during Application Initiation shall be GPO Parameter 1 (with tag 'DF01' in template 'BF3E').

Req 8.3 (Check P1 value for GPO command):

If the P1 parameter has a value other than '00', then the card shall discontinue processing the GPO command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6A86' (Incorrect Parameters, P1-P2).

Req 8.4 (Check P2 value for GPO command):

If the P2 parameter has a value other than '00', then the card shall discontinue processing the GPO command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6A86' (Incorrect Parameters, P1-P2).

Req 8.5 (Check length of GPO Command Data):

If the value of the GPO Template Length does not equal the value of the GPO Command Data Length parameter in the data element GPO Parameters x used for the transaction, then the card shall discontinue processing the GPO command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6700' (Wrong Length).

Req 8.6 (Minimum valid length for PDOL Related Data):

If the value of Lc is less than 2, then the card shall discontinue processing the GPO command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6700' (Wrong Length).

8.5.2 Processing

After the card receives the GET PROCESSING OPTIONS command from the terminal, the application begins transaction processing.

The application places a limit on the total number of transactions permitted to be processed by the application by not allowing the Application Transaction Counter (ATC) to roll over. When the limit ('FFFF') is reached, the application will no longer process transactions. If the limit is not reached, the application increments the Application Transaction Counter by one to include the current transaction and resets indicators used to indicate conditions and events that occur during the transaction.

NOTE: If an issuer chooses to limit the maximum number of transactions that may be processed by the application over the lifetime of the card to less than 65,535, the ATC may be personalised to a starting value other than zero. This is an issuer-option.

Req 8.7 (Check ATC):

If the value of the ATC is less than 'FFFF', then the application shall:

- *increment the ATC by one,*
- *reset transient transaction data, such as:*
 - *reset the Application Decisional Results (ADR) to '00 00 00 00 00 00'*
 - *reset the Card Verification Results (CVR) to '00 00 00 00 00'*
 - *reset Internal Flags (if implemented) to zero*

Otherwise (the ATC has the value 'FF FF'), the application shall discontinue processing the GET PROCESSING OPTIONS command and respond with SW1 SW2 = '6985' (Conditions of use not satisfied).

8.5.3 Profile Selection

Profile Selection uses information from the terminal to select application functionality based on a comparison of terminal data with personalised data. For example, it allows the application to select between multiple AIPs and AFLs personalised on the card when building the response to the GET PROCESSING OPTIONS command, and to configure the card risk management (such as which optional tests to perform, and which counters to use).

Profile Selection is a mechanism used to identify the transaction environment, and select a Profile to be used for the transaction. The information needed from the terminal for the application to identify the transaction environment is requested using the PDOL sent from the card to the terminal. The data is returned from the terminal to the card as the GPO command data field of the PDOL Related Data.

If the VLP implementer-option is not supported by the application or the transaction does not meet the conditions to use the VLP Profile, the application will check whether the issuer has requested to perform Profile Selection File Processing. If Profile Selection File Processing is not requested by the issuer, then the default Profile ID '01' will be used for the Transaction and the GET PROCESSING OPTIONS command will not use the PDOL-related data. Otherwise, the application will perform Profile Selection File Processing to determine the Profile used for the transaction.

Req 8.8 (Check whether VLP implementer-option supported):

If the VLP implementer-option is supported by the application, then the application shall perform VLP Profile Selection Processing, as described in section 8.5.3.1.

If the VLP implementer-option is not supported by the application:

- *if the 'Activate Profile Selection File' bit in Application Control has the value 0b, then the transaction shall be processed using the default Profile ID '01'.*
- *if the 'Activate Profile Selection File' bit in Application Control has the value 1b, then the application shall perform Profile Selection File Processing, as described in section 8.5.3.2.*

Additional functionality using data retrieved from the terminal using the PDOL is permitted, but is beyond the scope of this specification. See section 19 for further information regarding additional functionality.

8.5.3.1 VLP Profile Selection Processing

NOTE: Profile ID '7D' identifies the VLP profile, a special profile that if supported in the card and requested by the terminal, may be chosen to process the transaction. The purpose of the VLP profile is to have a fast offline transaction for very low value transactions. No Card Risk Management is performed in this profile. If the terminal requests an offline approval, the card will approve the transaction. If the terminal requests an offline decline, the card will decline the transaction.

If the VLP implementer-option is supported by the application, the application will first check whether the transaction is eligible for processing as a VLP transaction. If both the application and terminal support VLP, and the transaction meets the requirements for processing as VLP, then the VLP Profile will be used for the transaction.

If the transaction does not meet the conditions to use the VLP Profile, the application will check whether the issuer has requested to perform Profile Selection File Processing. If Profile Selection File Processing is not requested by the issuer, then the default Profile ID '01' will be used for the Transaction and the GET PROCESSING OPTIONS command will not use the PDOL-related data. Otherwise, the application will perform Profile Selection File Processing to determine the Profile used for the transaction.

NOTE: If VLP processing is allowed by the application, then the GPO Command Data begins with the VLP Terminal Support Indicator, Transaction Currency Code, and Amount Authorised. See section 21.3.4.

Req 8.9 (Check conditions for selection of VLP Profile):

If **all** of the following are true:

- the 'Activate VLP' bit in Application Control has the value 1b,
- **and** the following VLP conditions are met:
 - The VLP Terminal Support Indicator (byte 1 of GPO Command Data) has the value '01'
 - The Transaction Currency Code (bytes 2 and 3 of GPO Command Data) matches the Application Currency Code
 - Amount, Authorised (bytes 4-9 of GPO Command Data) is less than or equal to the VLP Available Funds
 - Amount, Authorised (bytes 4-9 of GPO Command Data) is less than or equal to the VLP Single Transaction Limit (if personalised on the card)⁴
 - PIN Try Counter does not equal 0
 - 'Issuer Authentication Failed' bit in PTH has the value 0b
 - 'Go Online on Next Transaction' bit in PTH has the value 0b
 - 'Last Online Transaction Not Completed' bit in PTH has the value 0b
 - 'Application Blocked' bit in PTH has the value 0b

then the transaction shall use Profile ID '7D' (VLP Profile) to process the transaction, continuing with section 8.5.4.

Otherwise, if the 'Activate Profile Selection File' bit in Application Control has the value 0b, then the transaction shall be processed using the default Profile ID '01'.

Otherwise the application shall perform Profile Selection File Processing, as described in section 8.5.3.2.

⁴ Support for VLP Single Transaction Limit is an issuer-option when the VLP implementer-option is supported.

8.5.3.2 Profile Selection File Processing

The application optionally uses the PDOL to request that the terminal send data in the PDOL Related Data. When the Profile Selection File functionality is activated, the application uses the content of the GPO Command Data and the Profile Selection Entries, to select the Profile ID of the Profile used for the transaction. If the Profile Selection Using Card Data implementer-option is supported, the application also uses the Profile Selection Diversifier in the GPO Parameter x used for the transaction in Profile Selection File Processing. The Profile ID identifies the Profile Control for the selected Profile. Based on the outcome of Profile Selection File Processing, the ICC may also discontinue processing the command and respond with a status word that causes the terminal to return to Application Selection to select another application.

Req 8.10 (Perform Profile Selection File Processing):

Profile Selection File Processing shall be performed as described in Annex A.

Req 8.11 (Profile Selection error):

If there is an error in Profile Selection File Processing, then the Profile ID shall be '7F'. If the Profile Selection File Processing results in selecting Profile ID '7F', then the card shall discontinue processing the GPO command and respond with SW1 SW2 = '6985' (Conditions of use not satisfied).

8.5.4 Profile Behaviour

The Profile ID selected during Profile Selection identifies the Profile Control to be used to configure the application behaviour for Initiate Application Processing and Card Action Analysis. The Profile Control identifies which AFL and AIP are returned to the terminal in the response to the GPO command (in addition to configuring other card behaviour for the transaction). For further explanation, see Annex H.

8.5.4.1 All Profiles

Req 8.12 (Perform Profile Selection File processing):

If Profile Selection processing results in selection of a Profile ID in the range '01' to '7E' for which an associated Profile Control x (where x is the value of the Profile ID selected for the transaction) is present, then the Profile Control selected for the transaction shall be Profile Control x.

If Profile Control x is not present in the application (where x is the value of the Profile ID selected for the transaction), then the application shall discontinue processing the GPO command, and shall respond with SW1 SW2 = '6985' (Conditions of use not satisfied).

The application contains sets of AIP and AFL in AIP/AFL Entries that may be selected for use within a Profile. An issuer can only rely on support for up to six AIP/AFL Entries. It is allowed for an implementation to support more than six AIP/AFL Entries, but it is not required in all implementations.

Req 8.13 (Minimum number of AIP/AFL Entries supported):

At a minimum, the application shall be able to support up to six AIP/AFL Entries.

Req 8.14 (Select AIP/AFL for response):

If AIP/AFL Entry x is present; then the application shall use the AIP and AFL in AIP/AFL Entry x to generate the GPO command response, where x is the value of AIP/AFL ID in the Profile Control for the transaction.

If AIP/AFL Entry x is not present, then the application shall discontinue processing the GPO command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6985' (Conditions of use not satisfied).

8.5.4.2 VLP Profile (Profile ID = '7D')

Req 8.15 (Decrement VLP Available Funds):

If the Profile ID selected for the transaction is '7D' (that is, the VLP Profile is used for the transaction), then the application shall deduct the Amount, Authorised from the VLP Available Funds.

8.5.5 Respond to GET PROCESSING OPTIONS Command

The 'Issuer Authentication Is Supported' bit in the AIP sent in the response is set to 0b to indicate that the card supports Issuer Authentication as part of processing the Second GENERATE AC command rather than using the EXTERNAL AUTHENTICATE command.

The 'Cardholder verification is supported' bit in the AIP sent in the response is set to 1b to indicate that the card requires cardholder verification to be performed by the terminal.

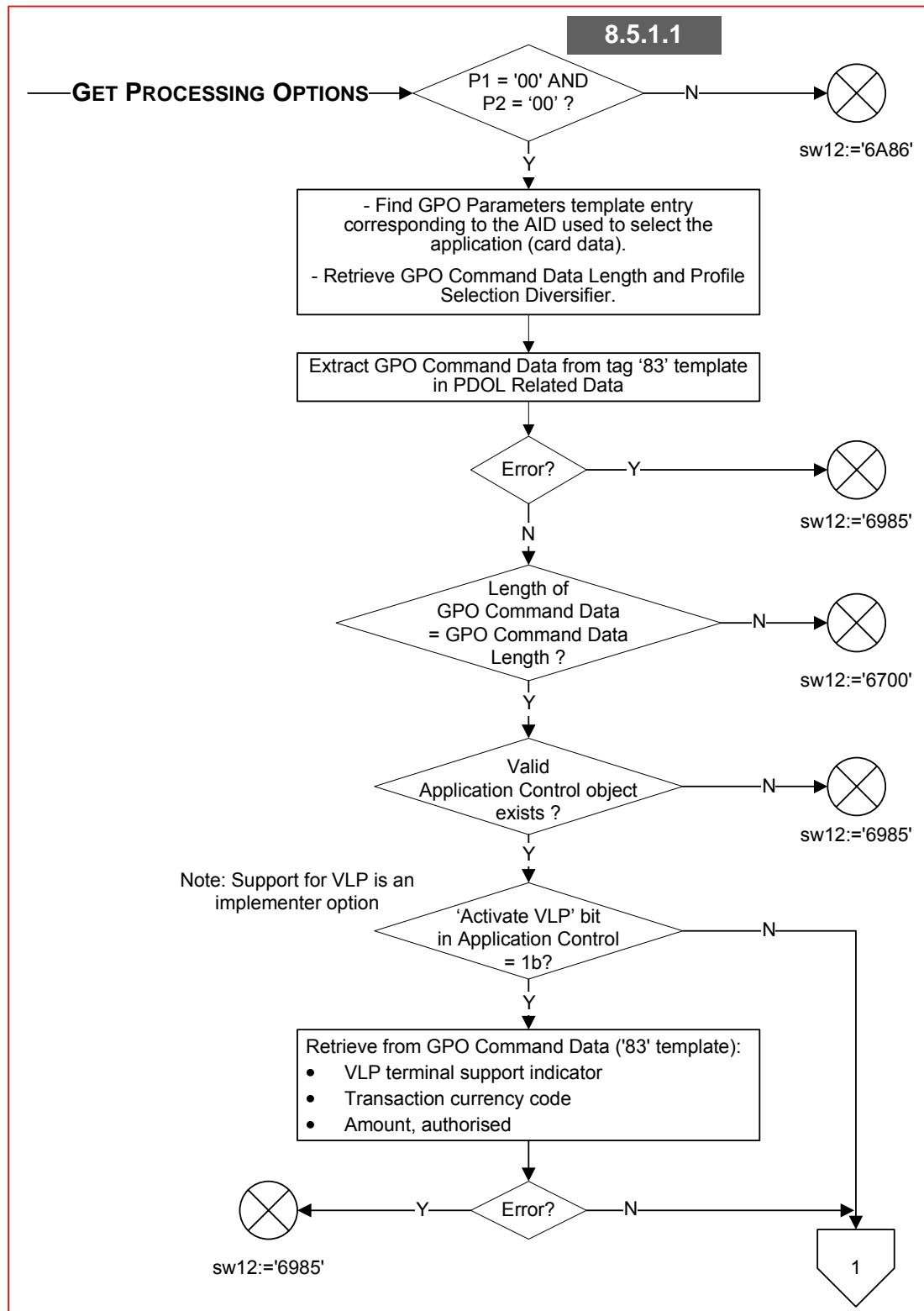
Req 8.16 (Format GPO response):

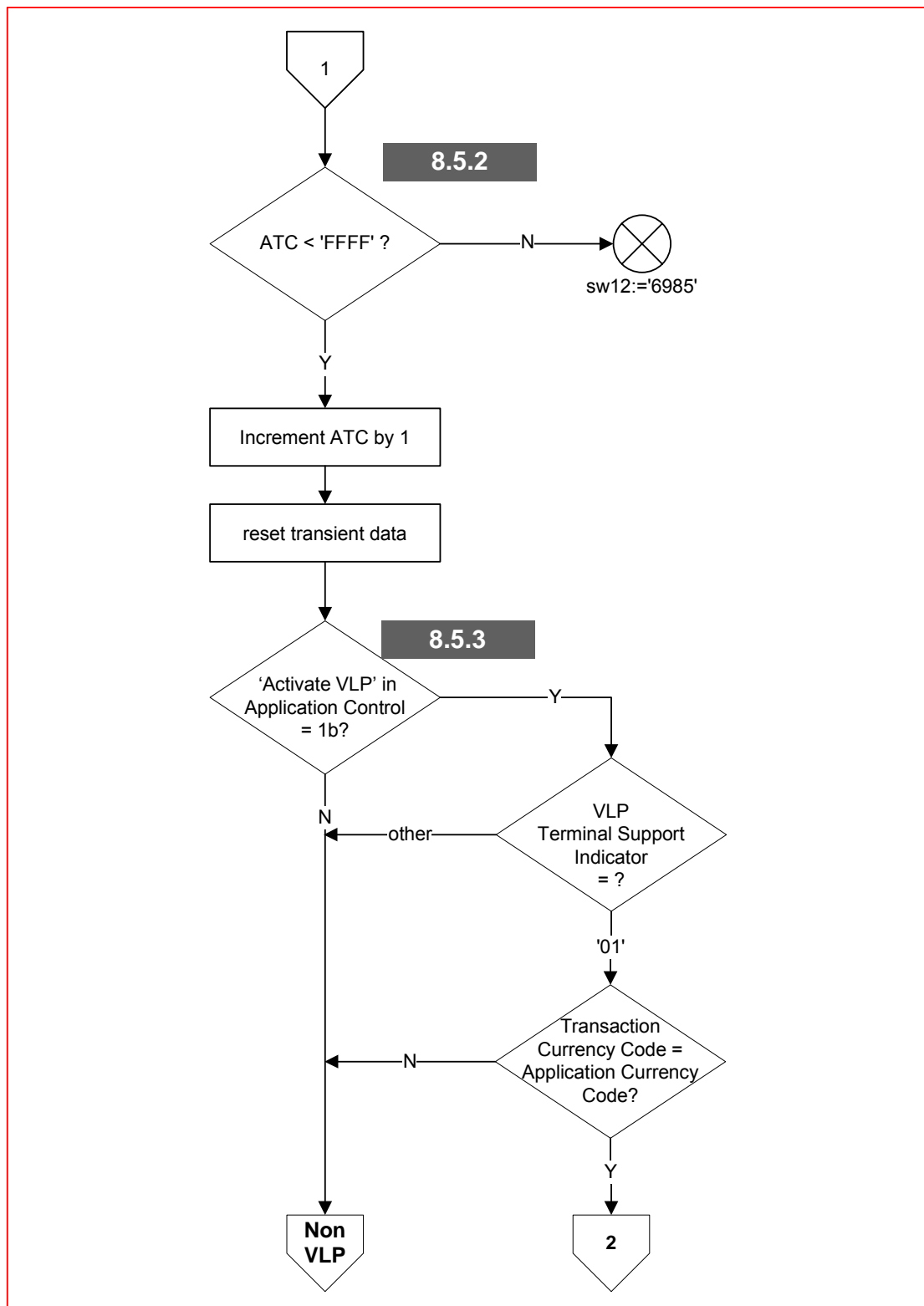
Response to the GPO command shall be formatted as specified in EMV Book 3, section 6.5.8.4 for a CCD-compliant card.

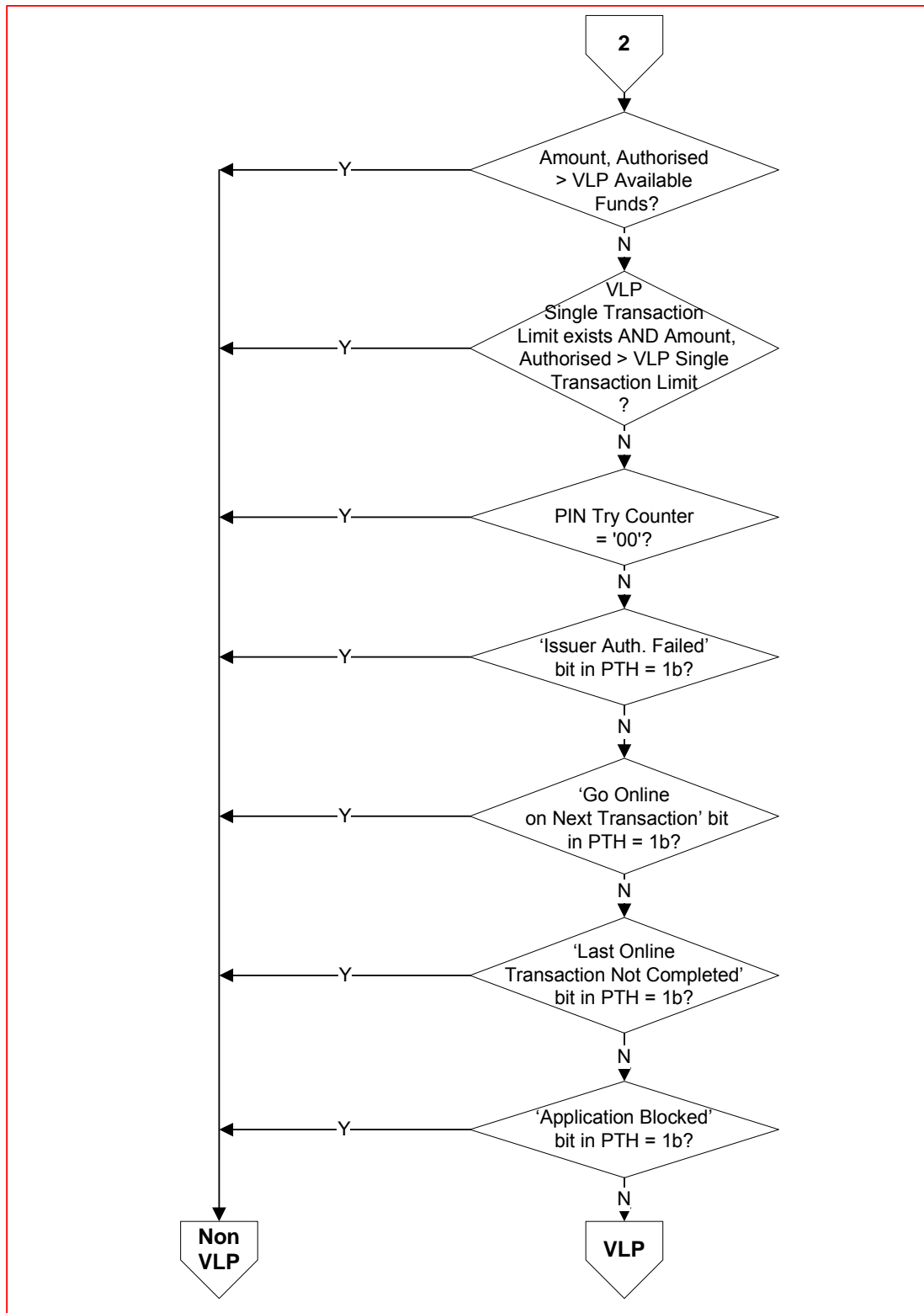
8.6 Function Flow Charts

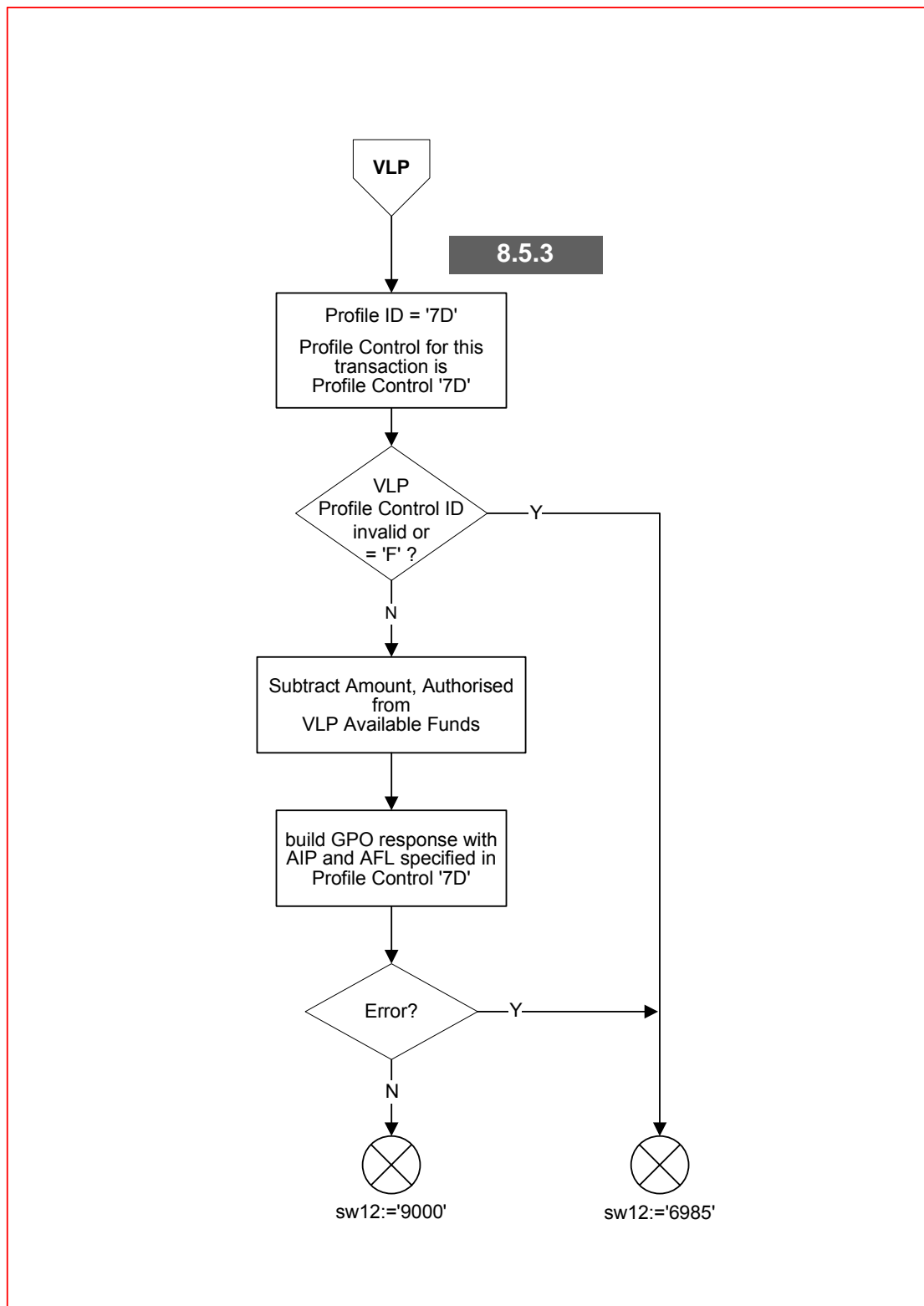
Figure 8-1 shows the Initiate Application Processing flow.

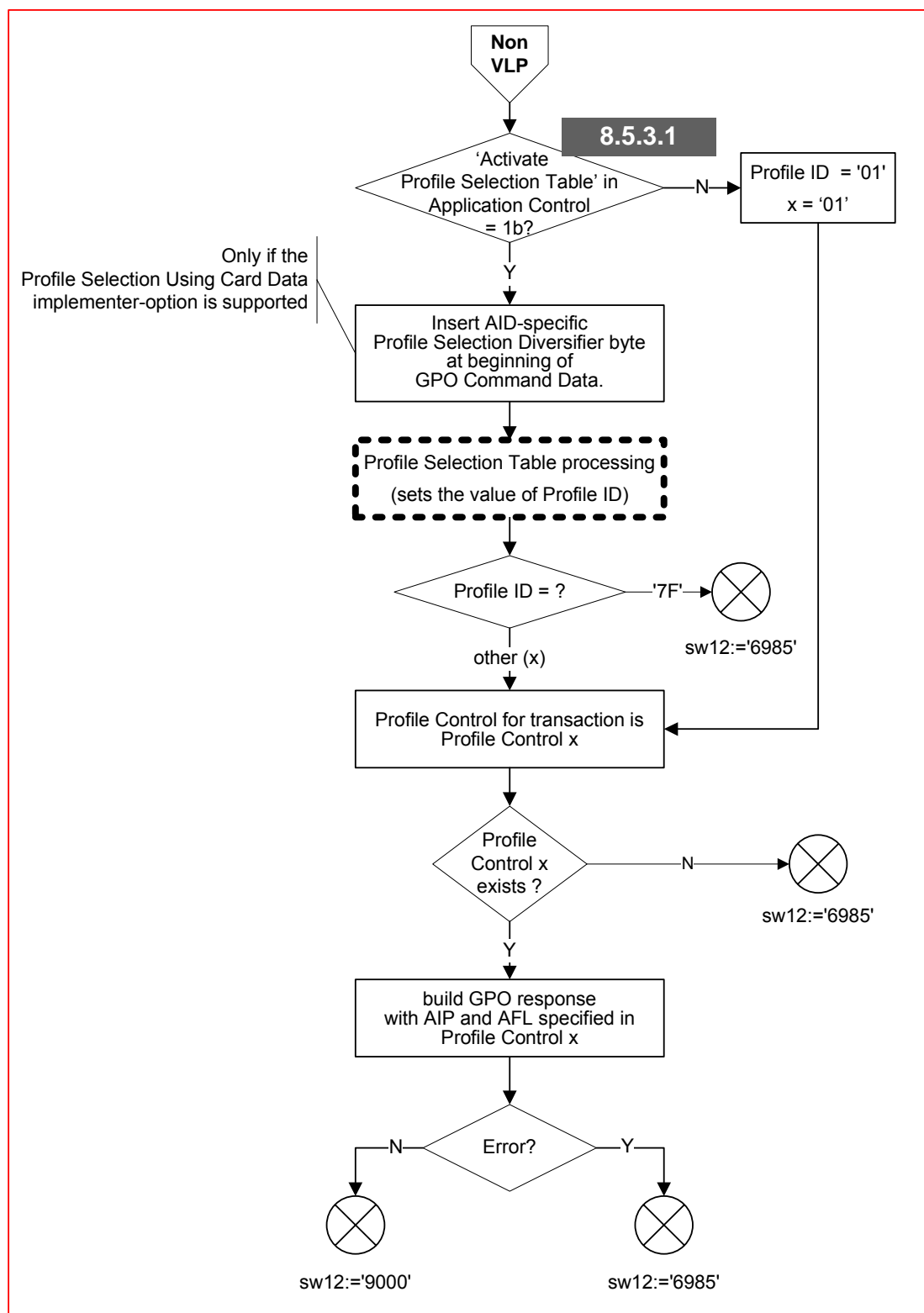
Figure 8-1: Initiate Application Processing Flow

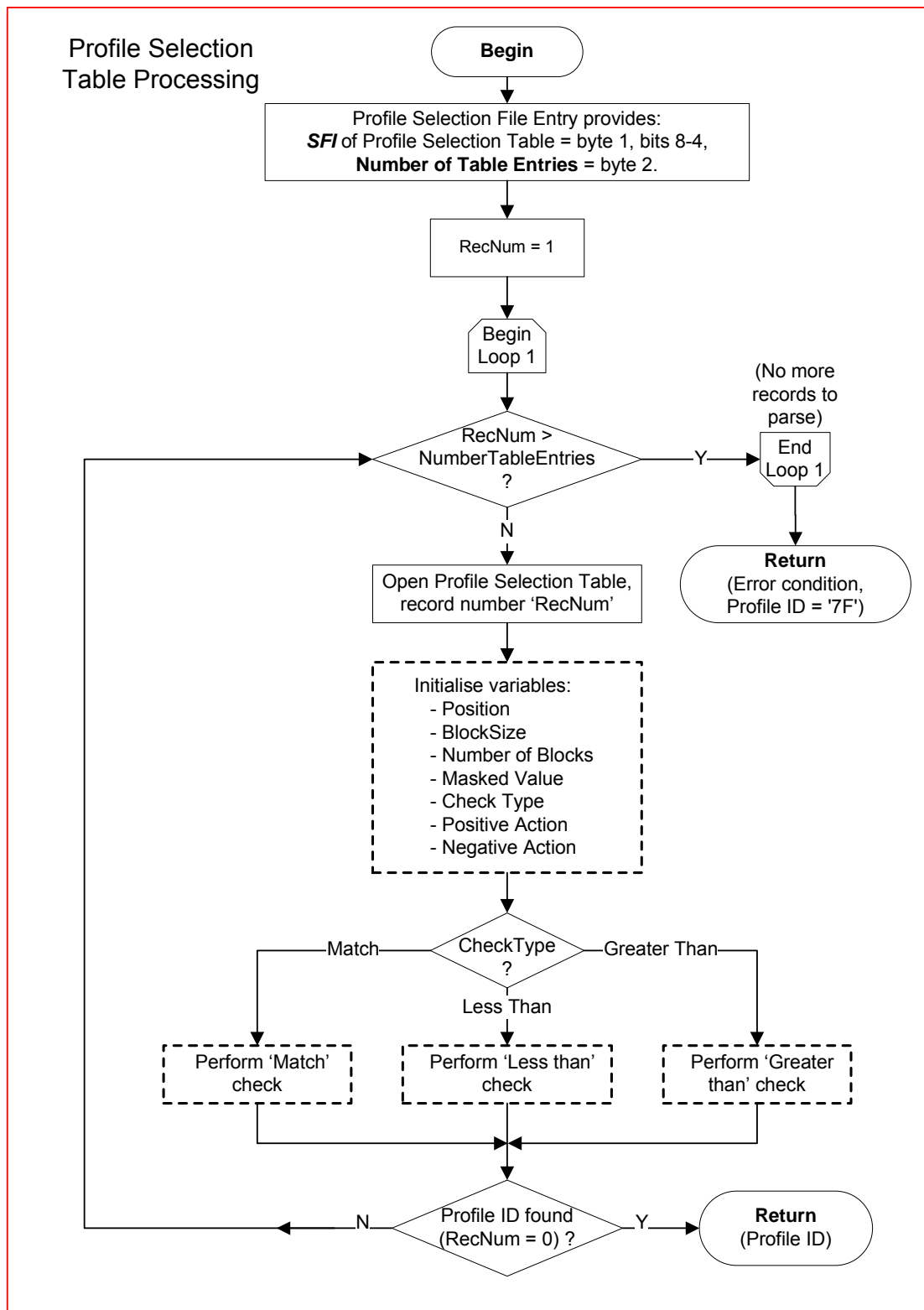


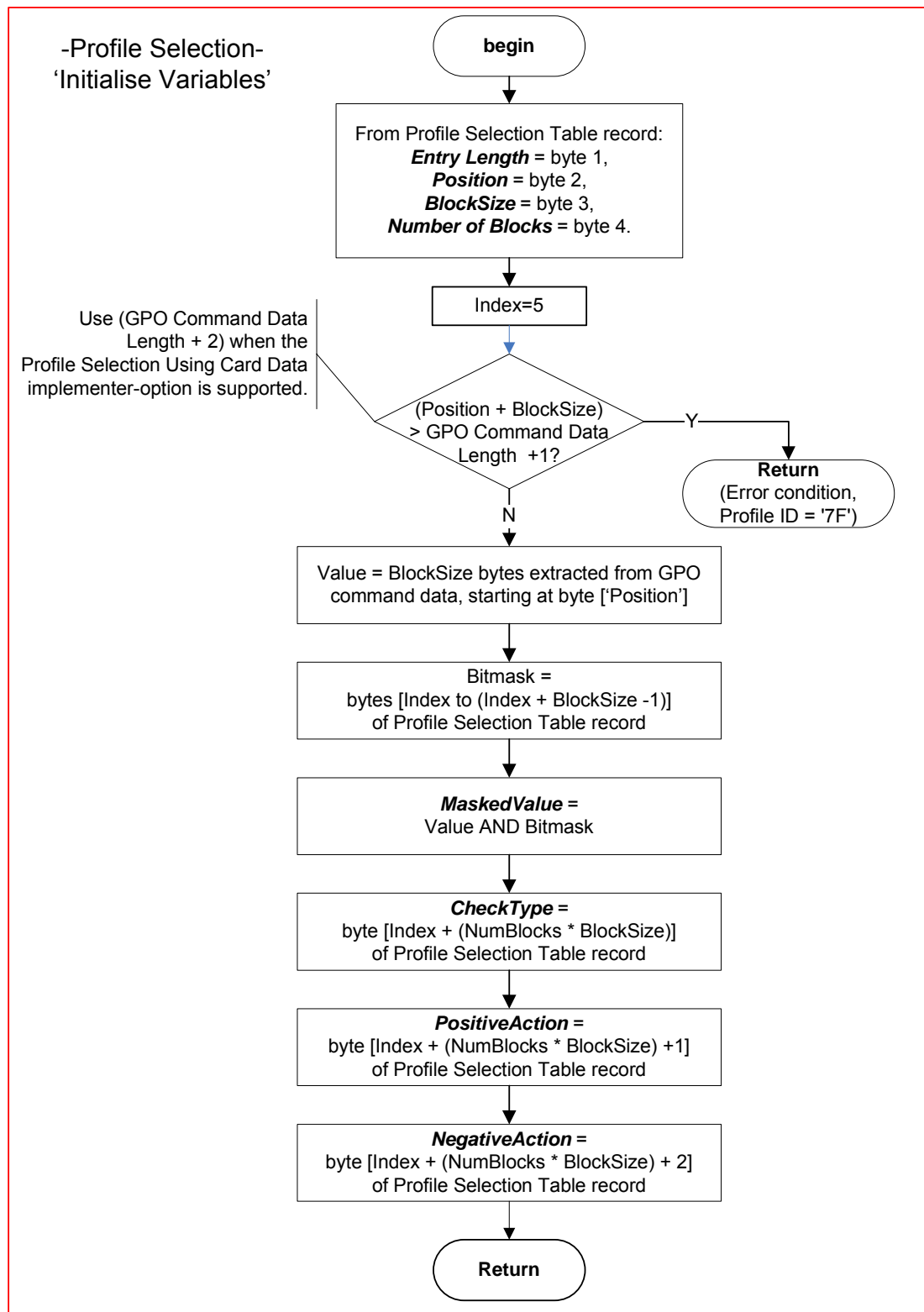


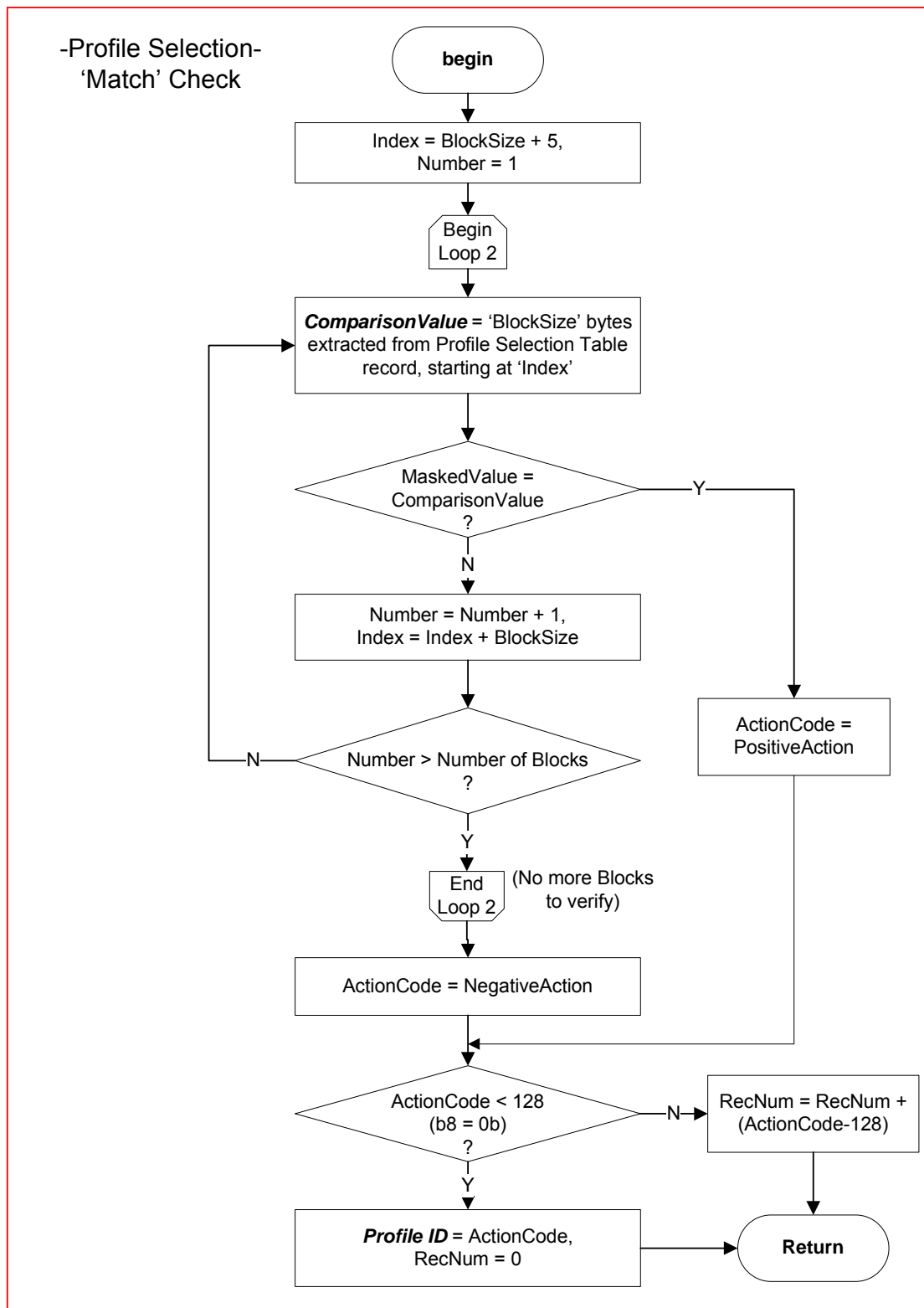




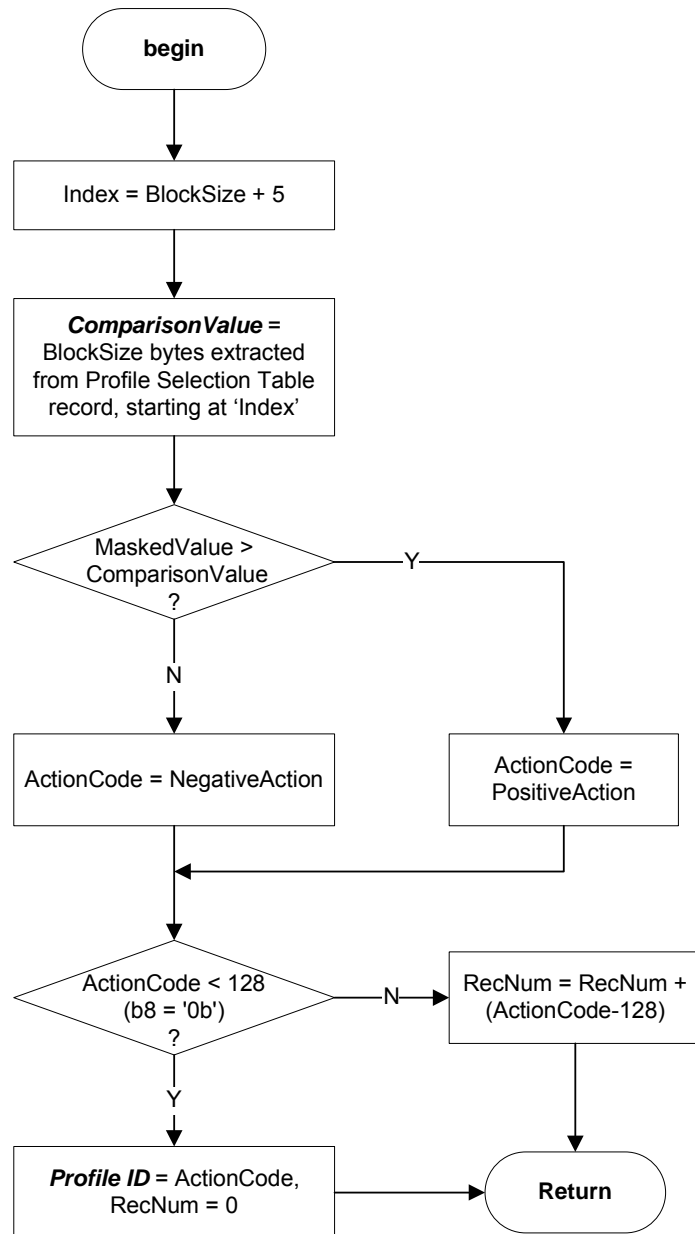


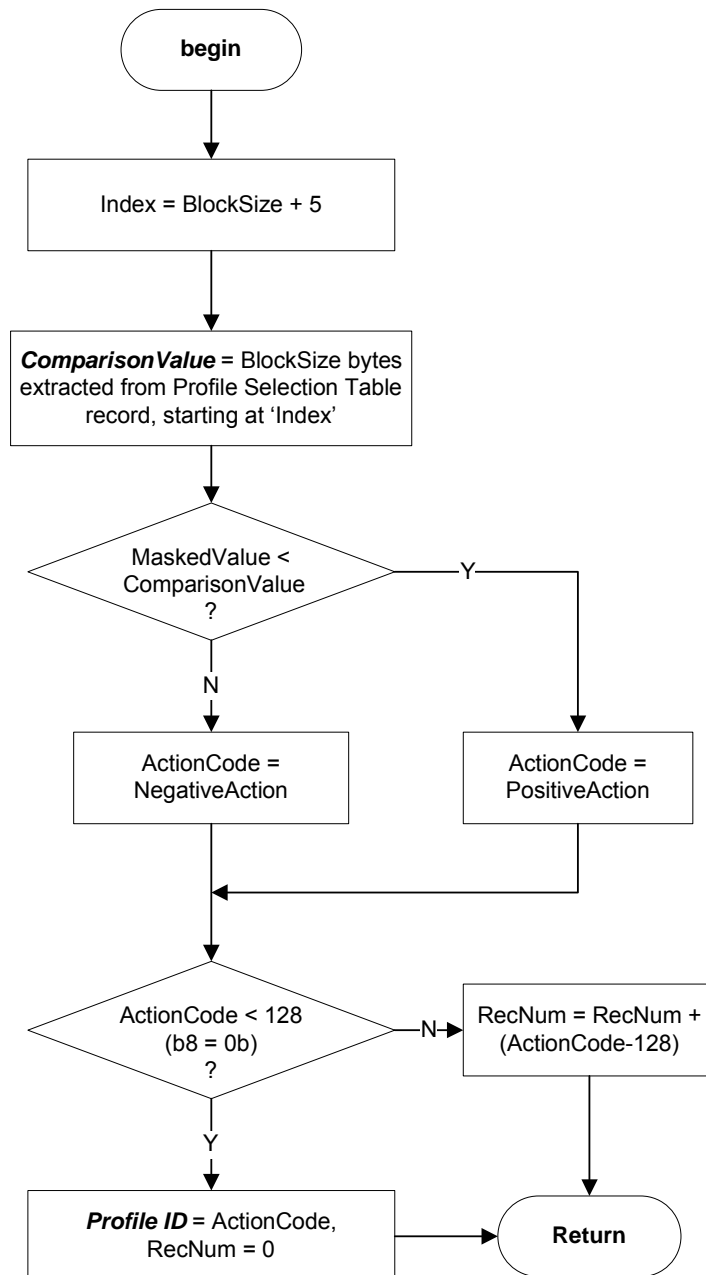






-Profile Selection-
'Greater than' Check



**-Profile Selection-
'Less than' Check**

9 Read Application Data

This section is organised as follows:

- 9.1 Purpose
- 9.2 Sequence of Execution
 - 9.2.1 Prior Related Processing
 - 9.2.2 Subsequent Related Processing
- 9.3 Card Data
- 9.4 Terminal Data
- 9.5 READ RECORD Command
 - 9.5.1 Command Coding
 - 9.5.2 Processing
 - 9.5.3 Respond to READ RECORD Command
- 9.6 Function Flow Charts
- 9.7 Additional File Requirements
 - 9.7.1 VLP Data File
 - 9.7.2 Transaction Log File
 - 9.7.3 File Containing the Profile Selection Entries

9.1 Purpose

During Read Application Data, the terminal reads the card data necessary to process the transaction and determines the data to be authenticated during Static Data Authentication (SDA), Dynamic Data Authentication (DDA), or Combined DDA/Application Cryptogram Generation (CDA).

Read Application Data is performed as described in EMV Book 3, section 10.2.

9.2 Sequence of Execution

The Read Application Data function is performed immediately following the Initiate Application Processing function.

9.2.1 Prior Related Processing

Initiate Application Processing

During Initiate Application Processing, the card sends the AFL to the terminal to designate the records the terminal should request from the card.

9.2.2 Subsequent Related Processing

Other functions use the data read during Read Application Data.

Offline Data Authentication

The terminal uses the list of static data to be authenticated that is built during Read Application Data for the validation of the Signed Static Application Data during SDA or the ICC Public Key Certificate during DDA and CDA.

9.3 Card Data

The card data used in Read Application Data are listed and described in Table 9-1. For a detailed description of these data and their usage, see Annex L: Data Dictionary.

Data	Description	Template	Tag
Application Elementary Files (AEF)	Card data files containing data used for application processing. An AEF consists of a sequence of records which are addressed by record number. Each AEF is identified by a unique SFI. The terminal reads these records using the READ RECORD command containing a designation of the SFI and record number to be read.	—	—
Application File Locator (AFL)	During the Initiate Application Processing function, the card sends the terminal the AFL which contains entries for groups of records to be read. Each entry designates: <ul style="list-style-type: none">• The SFI of the file• The record numbers of the first record and last record to read from the file• The number of records (beginning with the first record read in the file for this entry) to be used for authentication during SDA, DDA, or CDA.	—	'94'
Short File Identifier (SFI)	A number used to uniquely identify Application Elementary Files. It is listed in the AFL and used by the terminal to identify the files to be read.	—	'88'

Table 9-1: Read Application Data – Card Data

9.4 Terminal Data

The card uses no terminal data in Read Application Data.

9.5 READ RECORD Command

The READ RECORD command is performed as described in *EMV Book 3*, section 6.5.11.

9.5.1 Command Coding

The READ RECORD command received from the terminal is coded as shown in *EMV Book 3*, section 6.5.11.2, and includes the Short File Identifier (SFI) of the file to be read and the record number of the record within the file.

Code	Value
CLA	'00'
INS	'B2'
P1	Record Number
P2	Reference Control Parameter (see Table 9-3)
Lc	Not present
data	Not present
Le	'00'

Table 9-2: READ RECORD Command Message

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
x	x	x	x	x				SFI
					x	x	x	
					1	0	0	P1 is a record number

Table 9-3: Reference Control Parameter Coding for READ RECORD Command

9.5.1.1 Command Format Validation

Req 9.1 (Check P1 value for READ RECORD command):

If the P1 parameter has the value '00', the card shall discontinue processing the READ RECORD command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6A86' (Incorrect Parameters, P1-P2).

Req 9.2 (Check P2 value for READ RECORD command):

If bits 3 through 1 of the P2 parameter do not have the value 100b, the card shall discontinue processing the READ RECORD command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6A86' (Incorrect Parameters, P1-P2).

9.5.2 Processing

A READ RECORD command is received for each record designated in the AFL sent to the terminal during Initiate Application Processing. The card receives each READ RECORD command from the terminal and returns the requested record to the terminal as described in section 9.5.3.

The terminal continues to issue READ RECORD commands until all designated records within each designated file have been read.

9.5.3 Respond to READ RECORD Command

The command response returned by the card includes the requested record in the data field.

For records in files with SFI in the range from 1 to 10, the data field of the response is formatted as described in EMV Book 3, section 6.5.11.4 (that is, with template tag '70', and TLV coded).

The card is allowed to send filler bytes of value '00' in the READ RECORD response for the Profile Selection File.

The format of records in files with SFI in the range from 11 to 30 other than the VLP Data file, the Transaction Log file, and the Profile Selection File is out of scope for this specification.

Req 9.3 (SFI not found):

If the referenced SFI cannot be found, then the card shall discontinue processing the READ RECORD command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6A82' (file not found).

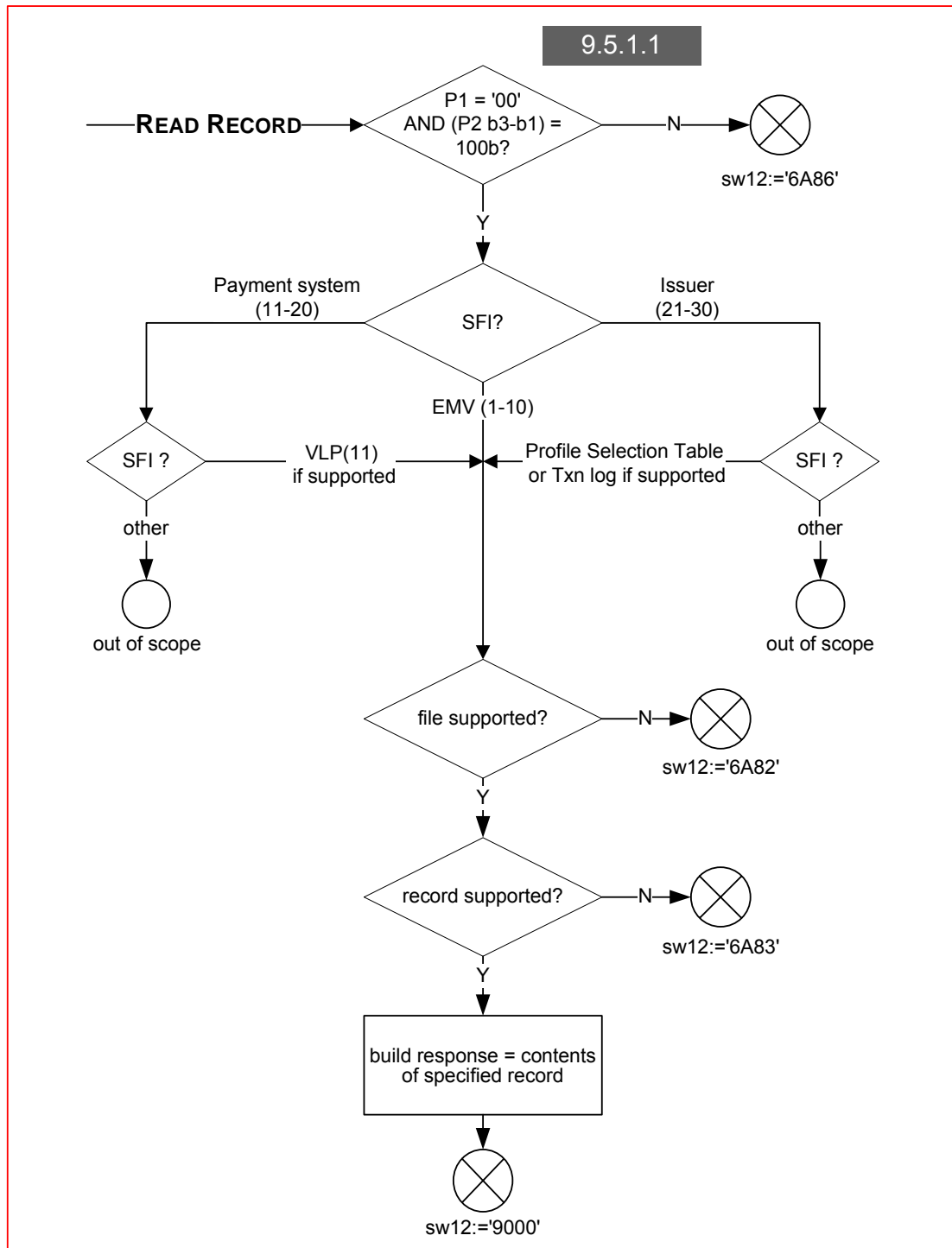
Req 9.4 (Record not found):

If the referenced record cannot be found, then the card shall discontinue processing the READ RECORD command and respond with SW1 SW2 = '6A83' (record number does not exist).

9.6 Function Flow Charts

Figure 9-1 shows the READ RECORD Command Processing flow.

Figure 9-1: READ RECORD Command Processing Flow



9.7 Additional File Requirements

The following three additional files are defined for the CPA:

- VLP Data file (SFI 11)
- Transaction Log file (an SFI in the range 21-30)
- File containing the Profile Selection Entries (an SFI in the range 21-30)

9.7.1 VLP Data File

Req 9.5 (SFI 11 used for VLP:

If VLP is supported by the implementation, the file with SFI 11 shall be reserved for use by VLP Data.

Additional requirement on the VLP Data file are specified in section 21.3.2.

9.7.2 Transaction Log File

Req 9.6 (SFI for Transaction Log file):

If transaction logging is supported by the issuer (that is, the Log Entry and Log Format data elements are personalised on the card), then the transaction log file shall have SFI in the range 21 to 30.

The transaction log file is formatted as described in *EMV Book 3*, section D4. The SFI used for the transaction log is identified in byte 1 of the Log Entry data element.

NOTE: The Log Entry data element must be personalized as a tagged data element (in addition to being included in the FCI) in order to identify for the application the SFI to be used to log transactions, and the maximum number of records to be supported in the file. See section 21.

The CPA supports flexible logging of transaction data. The content of the Transaction Log records is the concatenation of the data element values⁵ constructed as described in Annex D. The records in the Transaction Log file do not include a template tag.

Req 9.7 (Minimum number of transaction log records):

At a minimum, CPA shall support a file to contain 10 transaction log records.

The conditions whether to log a transaction, and the content of the log are defined in Annex D. Implementations may support more than ten records in the Transaction Log.

Req 9.8 (Minimum supported size for Transaction Log file):

At a minimum, the CPA shall support a transaction log file of up to 256 bytes.

If in its lifetime the CPA has not logged at least one transaction for each record in the Transaction Log, some of the entries in the Transaction Log do not represent transactions, but are empty. These empty entries are not retrievable with the READ RECORD command.

Req 9.9 (Empty Transaction Log file):

If a Transaction Log record is empty or the end of the file has been reached, the application shall discontinue processing the READ RECORD command and respond with SW1 SW2 = '6A83' (record number does not exist).

When accessing the Transaction Log the processing flow need not comply with the normal processing rules for an authorisation or financial transaction. Access to the transaction log is described in *EMV Book 3*, section D4.

⁵ Without the tag and length.

9.7.3 File Containing the Profile Selection Entries

This file contains the Profile Selection Entries; each record in the file is one Profile Selection Entry. The Profile Selection Entries do not include a template tag.

Req 9.10 (Minimum size for Profile Selection Entries):

At a minimum, the CPA shall support a file to contain the Profile Selection Entries with:

- *at a minimum, up to 15 Profile Selection Entries.*
- *at a minimum, up to 30 bytes in each Profile Selection Entry.*

Implementations may support more than 15 Profile Selection Entries, and implementations may support Profile Selection Entries that are longer than 30 bytes.

Req 9.11 (No template tag in Profile Selection Entries):

The Profile Selection Entries sent in response to READ RECORD for the file containing the Profile Selection File shall not include a template tag.

When accessing the Profile Selection File, the processing flow need not comply with the normal processing rules for an authorisation or financial transaction.

Devices that read the Profile Selection File use the Profile Selection File Entry data element to determine the location (SFI) and the number of records to read.

Table 9-4 describes the format of the Profile Selection File Entry data element (tag 'C2').

Byte	Format	Length	Value
1	b	1	SFI containing Profile Selection File
2	b	1	Number of Profile Selection Entries in the Profile Selection File

Table 9-4: Profile Selection File Entry

Req 9.12 (SFI for Profile Selection Entries):

If the Profile Selection File is supported by the Issuer (that is, the Profile Selection File Entry is personalised on the card), then the file containing the Profile Selection File shall have SFI in the range 21 to 30.

Req 9.13 (Profile Selection file supported by READ RECORD):

The Profile Selection File shall be accessible using the READ RECORD command. Each record is a variable length entry as specified in Annex L.

Req 9.14 (Profile Selection file not listed in AFL):

The Profile Selection File shall not be designated in the Application File Locator.

Req 9.15 (Profile Selection file is accessible when application is blocked):

The Profile Selection File Entry and the Profile Selection Entries shall remain accessible when the application is blocked.

To read the Profile Selection File, the special device ⁶ uses the following steps:

1. Perform Application Selection.
2. Issue a GET DATA command to retrieve the Profile Selection File Entry data element.
3. Issue READ RECORD commands to read the Profile Selection Entries in the Profile Selection File.

⁶ The Profile Selection File is typically read at a device that is designated or controlled by the issuer, and that does not process payment transactions.

10 Offline Data Authentication

This section is organised as follows:

- 10.1 Purpose
- 10.2 Sequence of Execution
 - 10.2.1 Prior Related Processing
 - 10.2.2 Subsequent Related Processing
- 10.3 Card Data
- 10.4 Terminal Data
- 10.5 Determining Whether to Perform SDA, DDA, or CDA
- 10.6 Static Data Authentication (SDA)
 - 10.6.1 Commands
 - 10.6.2 Processing
- 10.7 Offline Dynamic Data Authentication
 - 10.7.1 INTERNAL AUTHENTICATE Command
 - 10.7.2 GENERATE APPLICATION CRYPTOGRAM (AC) Command

10.1 Purpose

Offline Data Authentication is the process by which the terminal authenticates data from the card using RSA public key technology. Offline Data Authentication has three forms:

- Static Data Authentication (SDA)
- Dynamic Data Authentication (DDA)
- Combined DDA/AC Generation (CDA)

A card that supports the Dynamic-RSA implementer-option supports DDA and CDA.

Req 10.1 (Support for SDA):

All CPA cards shall accept personalisation of the data used for SDA. It is an issuer-option whether the AIP indicates that the application supports SDA.

Req 10.2 (Support for DDA/CDA):

If the application supports the Dynamic-RSA implementer-option, the cards shall support DDA and CDA. It is an issuer-option whether the AIP indicates that the application supports DDA and/or CDA.

During SDA processing the terminal authenticates static (unchanging) data from the card. See section 21.3.8 for a suggested list of static data to be authenticated. SDA ensures that issuer-selected card data elements have not been changed since the card was personalised.

During DDA and CDA processing the terminal authenticates the static card data and also authenticates a signature that the card generates using transaction-unique data. DDA and CDA ensure that issuer-selected card data elements have not been altered since the card was personalised. DDA and CDA also confirm that the card is genuine and has not been created by copying data from a valid card to a counterfeit card (skimming).

Offline Data Authentication is performed as described in EMV Book 2, sections 5 and 6.

10.2 Sequence of Execution

10.2.1 Prior Related Processing

Application Initiation Processing

During Initiate Application Processing, the card sends the AIP and AFL to the terminal to indicate the offline data authentication methods supported by the application and the records to be used for authentication.

Read Application Data

The terminal reads the application data from the card. For cards supporting SDA, this data includes the Issuer PK Certificate, other issuer key-related data, and the Signed Static Application Data (SSAD). For cards supporting DDA and CDA, this data includes the Issuer PK Certificate, other issuer key-related data, the DDOL, ICC PK Certificate, and other ICC key-related data. Static data to be authenticated is built from the records involved in offline data authentication (as specified in the AFL) and from the Static Data Authentication Tag List.

10.2.2 Subsequent Related Processing

Terminal Action Analysis

The terminal uses SDA, DDA, and CDA results as well as card and terminal parameters to determine whether the transaction should be declined offline, sent online for authorisation, or approved offline.

First Card Action Analysis (current transaction)

If the terminal requests CDA in the first GENERATE AC command and if the card is CDA-capable and requests online authorisation or approves the transaction offline, then the card will put the ARQC or TC response in an RSA envelope prior to responding to the terminal.

If the TVR received from the Terminal in the first GENERATE AC command indicates that SDA or DDA or CDA has failed, then the card sets the 'Offline Data Authentication Failed on Previous Transaction' bit in the CVR of subsequent transactions until a transaction either is approved offline or has successfully gone online, and offline data authentication has not failed.

If CDA is the offline data authentication method and the card response to the first GENERATE AC is a TC, then the terminal attempts to recover and validate the cryptogram data in the GENERATE AC response. If recovery or validation fails, then the transaction is declined offline by the terminal.

Online Processing (current transaction)

If CDA is the offline data authentication method and the card response to the first GENERATE AC is an ARQC, then the terminal attempts to recover and validate the cryptogram data in the GENERATE AC response. If recovery or validation fails, then the transaction is declined offline by the terminal.

Second Card Action Analysis (current transaction)

If the terminal requests CDA in the second GENERATE AC command and if the card is CDA-capable and approves the transaction offline, then the card will put the TC response in an RSA envelope prior to responding to the terminal.

If the TVR received from the Terminal in the first GENERATE AC command indicates that CDA failed on the first GENERATE AC command, then the card sets the 'Offline Data Authentication Failed on Previous Transaction' bit in the CVR of subsequent transactions until a transaction either is approved offline or has successfully gone online, and offline data authentication has not failed.

If CDA is the offline data authentication method and the card response to the second GENERATE AC is a TC, then the terminal attempts to recover and validate the cryptogram data in the GENERATE AC response. If recovery or validation fails, then the transaction is declined by the terminal.

First Card Action Analysis (subsequent transaction)

The 'Offline Data Authentication Failed on Previous Transaction' bit in the CVR is reset during a subsequent offline approved transaction if offline data authentication did not fail.

Second Card Action Analysis (subsequent transaction)

The 'Offline Data Authentication Failed on Previous Transaction' bit in the CVR is reset during a subsequent online transaction based upon issuer authentication conditions.

10.3 Card Data

The card data used in Offline Data Authentication are listed and described in Table 10-1. For a detailed description of these data and their usage, see Annex L: Data Dictionary.

Data	Description	Template	Tag
Application Interchange Profile (AIP)	Contains indicators for: <ul style="list-style-type: none"> • SDA is supported by card • DDA is supported by card • CDA is supported by card 	—	'82'
Card Verification Results (CVR)	Contains an indicator that is set during Card Action Analysis of subsequent transactions showing that Offline Data Authentication failed on a previous transaction. Also contains indicators set during the current transaction to indicate that Offline DDA was performed or that CDA was performed.	—	'9F52'
Certificate Authority (CA) Public Key Index	Provided by the CA with the Issuer PK Certificate. It identifies the payment system public key in the terminal to use for verifying the Issuer PK Certificate.	—	'8F'
Dynamic Data Authentication Data Object List (DDOL)	The list the card provides to the terminal that specifies the terminal data elements for the terminal to include in the INTERNAL AUTHENTICATE command.	—	'9F49'
ICC Dynamic Data	Data elements to be included in the Signed Dynamic Application Data.	—	—
ICC Dynamic Number	Part of the ICC Dynamic Data containing a time-variant number generated by the ICC.	—	—
ICC Private Key	The key used to generate the Signed Dynamic Application Data.	—	—

Table 10-1: Offline Data Authentication – Card Data
(continues)

Data	Description	Template	Tag
ICC Public Key (PK) Certificate	<p>A certificate containing all or part of the ICC Public Key and a hash of static card data elements. The ICC PK Certificate is created using the Issuer Private Key and placed on the card during card personalisation. This ICC PK Certificate is further described in <i>EMV Book 2</i>, section 6.1. The static data elements used in the ICC PK Certificate hash are the same data elements used to generate the card's SSAD used in SDA. These data elements are specified by the AFL and in the SDA Tag List during Read Application Data.</p> <p>If SDA input data can vary from one transaction to another, multiple AFLs (indicating multiple SSAD values and multiple ICC PK Certificates) must be supported. An example of when this data might not be unique is when a card uses different CVM Lists for domestic and international transactions.</p> <p>If any of the signed data elements can be changed post-issuance, the capability to change the ICC Public Key Certificate must also be supported.</p>	—	'9F46'
ICC Public Key Exponent	<p>The exponent to be used in the RSA recovery of the Signed Dynamic Application Data.</p> <p>NOTE: The ICC Public Key Exponent is 3 or $(2^{16} + 1)$.</p>	—	'9F47'
ICC Public Key Remainder	The portion, if any, of the ICC Public Key that does not fit into the ICC Public Key Certificate.	—	'9F48'

Table 10-1: Offline Data Authentication – Card Data, continued

Data	Description	Template	Tag
Issuer Public Key (PK) Certificate	The certificate containing the Issuer Public Key that has been signed with the CA Private Key. This certificate is described in <i>EMV Book 2</i> , Table 5.	—	'90'
Issuer Public Key Exponent	The exponent used in the RSA algorithm to recover the Issuer PK Certificate.	—	'9F32'
Issuer Public Key Remainder	The portion, if any, of the Issuer Public Key that does not fit into the Issuer PK Certificate.	—	'92'
Previous Transaction History (PTH)	Contains an indicator that identifies for subsequent transactions that offline data authentication failed.	—	'C7'
Registered Application Provider Identifier (RID) portion of the Application Identifier (AID)	The first five bytes of the Application Identifier (AID) that identifies the Payment System. The RID is registered with ISO. The RID and the Certificate Authority Public Key Index are used to identify the Payment System Public Key to be used to recover the Issuer Public Key Certificate.	—	—
SDA Tag List	Contains the tag of non-record EMV data that is included in the Signed Static Application Data or in the ICC Public Key Certificate.	—	'9F4A'

Table 10-1: Offline Data Authentication – Card Data, continued

Data	Description	Template	Tag
Signed Dynamic Application Data	<p>For DDA, the signature generated by the card at transaction time after receipt of the INTERNAL AUTHENTICATE command. For CDA, the signature generated by the card at transaction time after receipt of the GENERATE AC command.</p> <p>The card generates this signature using a hash of dynamic data from the terminal and card. The card signs the Signed Dynamic Application Data with the ICC Private Key. The format of the Signed Dynamic Application Data is shown in <i>EMV Book 2</i>, Table 16.</p>	—	'9F4B'
Signed Static Application Data (SSAD)	<p>For SDA, a digital signature generated from critical card data elements and used by the terminal to validate the card's static data. The SSAD is signed with the Issuer Private Key and is personalised on the card.</p>	—	'93'

Table 10-1: Offline Data Authentication – Card Data, continued

10.4 Terminal Data

The terminal data used in Offline Data Authentication are listed and described in Table 10-2. For a detailed description of these data and their usage, see Annex L: Data Dictionary.

Data	Description	Template	Tag
Certification Authority Public Key	Payment System public key used for SDA, DDA, or CDA.	—	—
Terminal Verification Results (TVR)	A series of indicators showing the results of offline processing from a terminal perspective. It is used to record the results of Offline Data Authentication.	—	'95'
Unpredictable Number	A time-variant number included in the INTERNAL AUTHENTICATE command for DDA, or in the GENERATE AC command for CDA.	—	'9F37'

Table 10-2: Offline Data Authentication – Terminal Data

10.5 Determining Whether to Perform SDA, DDA, or CDA

The terminal uses the AIP sent in the response to the GPO command from the card and the offline data authentication support provided by the terminal to determine whether to perform SDA, DDA, or CDA.

Only one method of offline data authentication is performed during a transaction.

If the terminal determines that both the card (as indicated by the AIP) and the terminal support CDA, it performs CDA. Otherwise, if both support DDA, the terminal performs DDA. Otherwise, if both the card and terminal support SDA, the terminal performs SDA.

If the card and terminal do not support a common offline data authentication method, no offline data authentication is done.

10.6 Static Data Authentication (SDA)

During SDA processing, the terminal uses RSA public key technology to validate that key data elements on the card have not been altered since card personalisation.

The format of the SSAD is shown in *EMV Book 2*, Table 6. The formats of the data elements to be hashed are in Table 3 of the same EMV document.

If more than one set of data is to be signed for the application, then multiple SSADs must be supported. An example of when more than one set of data might be required by an issuer is when a card has different CVM Lists for domestic and international transactions and the CVM List is included in the signature.

Req 10.3 (Changing signed static data elements):

If the card supports the ability to change any of the signed data elements after the card has been issued to the cardholder, the capability to change the SSAD shall also be supported.

The SDA Tag List contains the tag of the AIP if it is to be signed. It is recommended that the AIP be included in the Signed Static Data if the AIP indicates that DDA or CDA is supported. The SDA Tag List does not contain any tags other than the AIP tag.

10.6.1 Commands

No commands are used in SDA processing.

10.6.2 Processing

The card performs no processing during SDA.

During SDA, the terminal uses RSA public key technology to recover and validate the Issuer Public Key and to validate the SSAD from the card. The terminal's SDA processing steps are described in more detail in *EMV Book 2*, section 5. If all of the SDA steps are successful, SDA has passed.

10.7 Offline Dynamic Data Authentication

During offline dynamic data authentication processing, the terminal uses RSA public key technology to determine whether key data elements from the card have been altered since card personalisation and whether the card is counterfeit.

EMV supports two forms of offline dynamic data authentication: DDA and CDA. With both, the terminal validates that static card data has not been altered and also validates a dynamic cryptogram generated by the card using its unique private key. With DDA, the card generates the dynamic signature using dynamic terminal, card, and transaction data in response to an INTERNAL AUTHENTICATE command received prior to Card Action Analysis. With CDA, the card responds to the first (or second) GENERATE AC command received during First (or Second) Card Action Analysis by generating a dynamic signature that includes the Application Cryptogram and Cryptogram Information Data as well as the dynamic terminal, card, and transaction data used for DDA.

10.7.1 INTERNAL AUTHENTICATE Command

The terminal issues the INTERNAL AUTHENTICATE command during DDA processing. The command includes the terminal dynamic data specified in the DDOL.

The INTERNAL AUTHENTICATE command is performed as described in EMV Book 3, section 6.5.9. If the application supports DDA, then the card is personalised with a DDOL that contains only the Unpredictable Number generated by the terminal (tag '9F37').

When the card receives the INTERNAL AUTHENTICATE command, it computes the Signed Dynamic Application Data with the ICC Private Key. This dynamic signature is included in the INTERNAL AUTHENTICATE command response. Section 20.3 describes limitations on the number of INTERNAL AUTHENTICATE commands for each value of the ATC.

10.7.1.1 Command Coding

Code	Value
CLA	'00'
INS	'88'
P1	'00'
P2	'00'
Lc	'04'
data	Authentication-related data
Le	'00'

Table 10-3: INTERNAL AUTHENTICATE Command Message**10.7.1.1.1 Command Format Validation****Req 10.4 (Check P1/P2 values for INTERNAL AUTHENTICATE):**

If either the P1 or P2 parameter does not have the value '00', then the card shall discontinue processing the INTERNAL AUTHENTICATE command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6A86' (Incorrect Parameters, P1-P2).

Req 10.5 (Check length of INTERNAL AUTHENTICATE Command Data):

If the value of Lc does not equal '04', then the card shall discontinue processing the INTERNAL AUTHENTICATE command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6700' (Wrong Length).

10.7.1.2 Processing

During DDA processing, the terminal uses RSA public key technology to validate the Issuer PK Certificate, the ICC PK Certificate, and the Signed Dynamic Application Data (the dynamic signature) from the card.

The only functions performed by the card during DDA processing are the generation of the ICC Dynamic Number and the dynamic signature. See section 20.4 for requirements on the generation of the ICC Dynamic Number.

DDA processing is described in detail in *EMV Book 2*, section 6, *EMV Book 3*, section 10.3, and *EMV Book 4*, section 6.3.2.

The application sets the 'Offline DDA Performed' bit in the CVR to the value 1b if and only if Signed Dynamic Application Data is returned in the response to the INTERNAL AUTHENTICATE command.

If all of the DDA steps are successful, DDA has passed.

10.7.1.3 Function Flow Chart

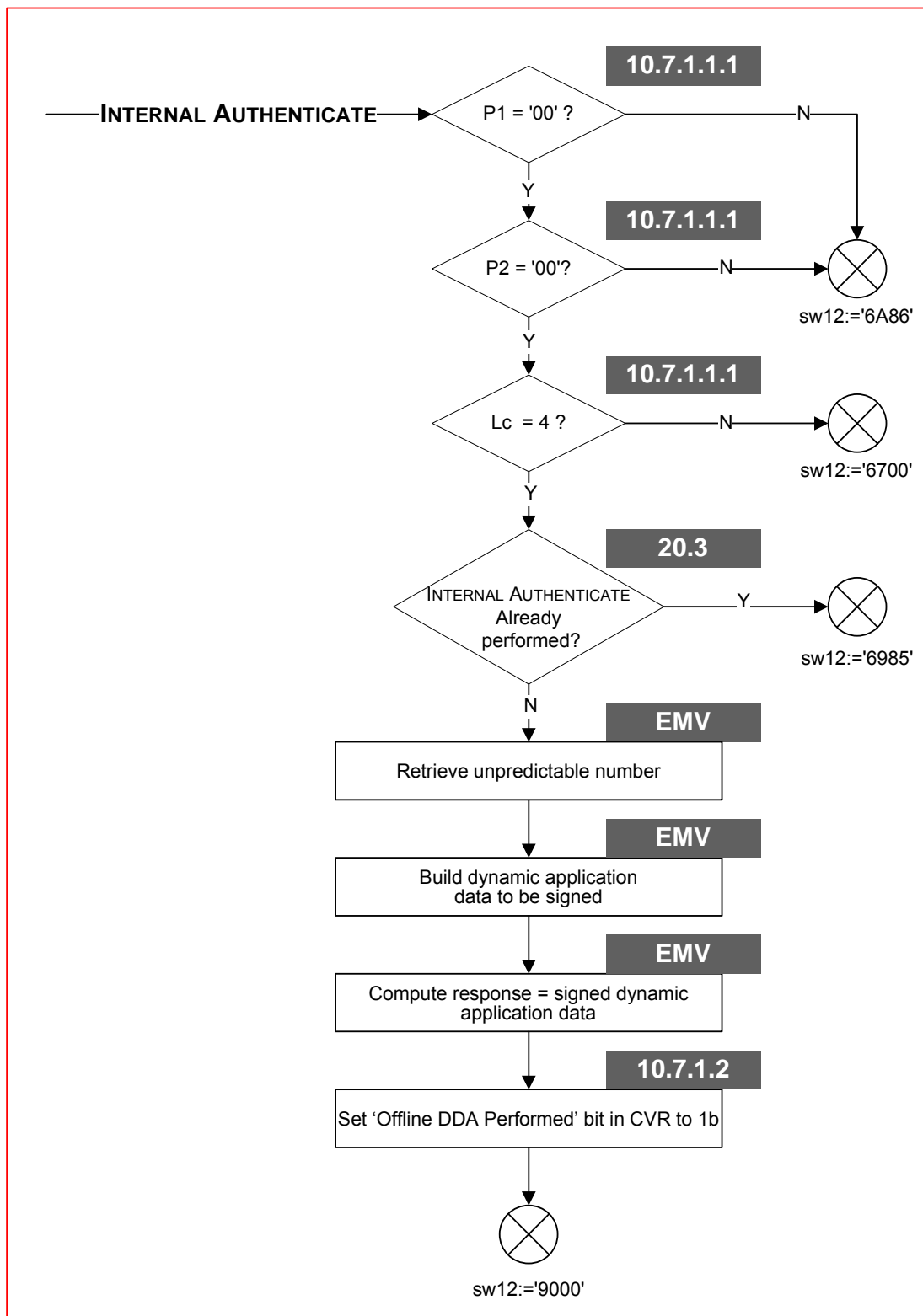


Figure 10-1: INTERNAL AUTHENTICATE Flow

10.7.2 GENERATE APPLICATION CRYPTOGRAM (AC) Command

The terminal issues the first GENERATE AC command during First Card Action Analysis processing. The terminal may also issue the second GENERATE AC command during Second Card Action Analysis processing. If the transaction is eligible for CDA, the P1 parameter of the GENERATE AC command is set to request CDA processing.

If the card receives the GENERATE AC command requesting CDA processing, a TC or ARQC returned by the card includes a public key signature as described in *EMV Book 2*, section 6.6.1. See also sections 15 and 17 of this document for additional information on this command.

10.7.2.1 CDA Processing

CDA processing is described in detail in *EMV Book 2*, section 6, *EMV Book 3*, section 10.3, and *EMV Book 4*, section 6.3.2. See section 15.5.8 of this document for further description of CDA processing in CPA, including requirements for setting the 'CDA Performed' bit in the CVR.

If all of the CDA steps are successful, CDA passes.

11 Processing Restrictions

This section is organised as follows:

- 11.1 Purpose
- 11.2 Sequence of Execution
 - 11.2.1 Prior Related Processing
 - 11.2.2 Subsequent Related Processing
- 11.3 Card Data
- 11.4 Terminal Data
- 11.5 Processing
 - 11.5.1 Application Version Number
 - 11.5.2 Application Usage Control
 - 11.5.3 Application Effective Date
 - 11.5.4 Application Expiration Date

11.1 Purpose

NOTE: In order to provide a more complete description of transaction processing with the CPA application, this section provides an overview of functionality provided by the EMV terminal. CPA does not require any change to EMV terminal functionality.

The Processing Restrictions function is performed by the terminal using data from the terminal and the card. It includes checks on application versions, effective and expiration dates, and conditions at the point of transaction.

Processing Restrictions is performed as described in *EMV Book 3*, section 10.4 and *EMV Book 4*, section 6.3.3 and Annex A.

11.2 Sequence of Execution

11.2.1 Prior Related Processing

Read Application Data

The terminal uses the READ RECORD command to obtain application data to be used for the transaction. This data includes the Issuer Country Code, Application Version Number, Application Expiration Date, and, if present, the AUC and Application Effective Date.

11.2.2 Subsequent Related Processing

Terminal Action Analysis

During Terminal Action Analysis, the terminal checks the Issuer Action Codes (IACs) and Terminal Action Codes (TACs) to determine the transaction disposition if application versions differ, the application is not yet effective or has expired, or the requested service is not allowed for the application.

11.3 Card Data

The card data used in Processing Restrictions are listed and described in Table 11-1. For a detailed description of these data and their usage, see Annex L: Data Dictionary.

Data	Description	Template	Tag
Application Effective Date	The date when the application becomes activated for use.	—	'5F25'
Application Expiration Date	The date after which the application is no longer available for use.	—	'5F24'
Application Usage Control (AUC)	Indicates any restrictions set forth by the issuer on the geographic usage and services permitted for the card application. Used in Application Usage Control checking by the terminal.	—	'9F07'
Application Version Number	This data element indicates the version of the application on the card. It is specified by the Payment System and used in Application Version Number checking by the terminal.	—	'9F08'
Issuer Country Code	The EMV-defined data element indicating the country of the card issuance. Used in Application Usage Control checking by the terminal.	—	'5F28'

Table 11-1: Processing Restrictions – Card Data

11.4 Terminal Data

The terminal data used in Processing Restrictions are listed and described in Table 11-2. For a detailed description of these data and their usage, see Annex L: Data Dictionary.

Data	Description	Template	Tag
Application Version Number	This data element indicates the version of the application in the terminal. Specified by the Payment System.	—	'9F08'
Terminal Country Code	Indicates the country where the terminal is located. Used in Application Usage Control checking by the terminal.	—	'9F1A'
Terminal Verification Results (TVR)	A series of indicators used to record the results of offline processing from a terminal perspective, including the results of terminal processing restrictions checks.	—	'95'
Transaction Date	The local date (in the terminal) when the transaction is taking place. Used by the terminal in effective date and expiration date checking.	—	'9A'
Transaction Type	Indicates the type of financial transaction. (It is represented by the first two digits of Processing Code, as defined by the payment system.) Used in Application Usage Control checking by the terminal.	—	'9C'

Table 11-2: Processing Restrictions – Terminal Data

11.5 Processing

The card does no processing during the Processing Restrictions function. The terminal uses data from the card to check that the application may be used for the transaction, and indicates possible error conditions to the issuer using the Terminal Verification Results (TVR).

The following sections describe how the terminal uses data from the card during Processing Restrictions.

11.5.1 Application Version Number

The terminal compares the Application Version Number from the card to the Application Version Number in the terminal to see if they are the same. If they differ, the terminal sets a TVR bit to indicate to the issuer that the Version Numbers did not match.

11.5.2 Application Usage Control

During Application Usage Control, the terminal checks various conditions at the point of transaction to determine whether processing should continue. If the Application Usage Control (AUC) and the Issuer Country Code were received from the card during Read Application Data, the terminal checks the following application restrictions:

Domestic and International Checking

NOTE: This check performed by the terminal is in addition to any domestic/international checking that may be performed by the application as part of Initiate Application Processing or Card Risk Management.

The terminal determines whether the transaction is domestic or international (based on the Issuer Country Code), and then checks whether the card is allowed to be used for the transaction type in the terminal's country.

ATM Checking

If the terminal is an ATM, it checks that the card is allowed to be used at an ATM. If the terminal is not an ATM, it checks that the card is allowed to be used at a terminal that is not an ATM.

If any of the above checks performed by the terminal fails, the terminal indicates that the 'Requested service is not allowed for card product' in the TVR.

The coding of the AUC is described in EMV Book 3, section C2.

11.5.3 Application Effective Date

The terminal performs Application Effective Date checking when the card application data includes the Application Effective Date. It ensures that the application is active by validating that the Application Effective Date from the card is less than or equal to the Transaction Date (local to the terminal). If the Application Effective Date is greater than the Transaction Date, the terminal indicates in the TVR that the application is not yet effective.

11.5.4 Application Expiration Date

Application Expiration Date checking is mandatory. The terminal validates that the application has not expired by ensuring that the Application Expiration Date from the card is greater than or equal to the Transaction Date (local to the terminal). If the Application Expiration Date is less than the Transaction Date, the terminal indicates in the TVR that the application has expired.

12 Cardholder Verification

This section is organised as follows:

- 12.1 Purpose
- 12.2 Sequence of Execution
 - 12.2.1 Prior Related Processing
 - 12.2.2 Subsequent Related Processing
- 12.3 Card Data
- 12.4 Terminal Data
- 12.5 GET DATA Command
 - 12.5.1 Command Coding
 - 12.5.2 Processing
 - 12.5.3 GET DATA Flow Chart
- 12.6 GET CHALLENGE Command
 - 12.6.1 Command Coding
 - 12.6.2 Processing
 - 12.6.3 GET CHALLENGE Response
 - 12.6.4 GET CHALLENGE Flow Chart
- 12.7 VERIFY Command
 - 12.7.1 Command Coding
 - 12.7.2 Processing
 - 12.7.3 VERIFY Flow Chart

12.1 Purpose

Cardholder Verification is used to ensure that the cardholder is legitimate and the card is not lost or stolen.

In Cardholder Verification, the terminal uses rules in a CVM List from the card to determine the cardholder verification method (CVM) to be used and performs the selected CVM. The results of CVM processing play a role in later processing.

With the offline plaintext PIN and offline enciphered PIN methods, the validation of the PIN is done within the card. Offline PIN verification results are included in the online authorisation message and should be considered in the issuer's authorisation decision. The card plays no role in processing of Online PIN or signature CVMs.

Cardholder Verification is performed as described in *EMV Book 3*, section 10.5.

12.2 Sequence of Execution

12.2.1 Prior Related Processing

Initiate Application Processing

The terminal receives the Application Interchange Profile (AIP) which indicates that the card supports cardholder verification in the GET PROCESSING OPTIONS response from the card.

Read Application Data

The terminal reads the CVM List and other data used in CVM processing from the card. The CVM List may be profile-dependent if the AFL sent to the terminal for different profiles identify a different CVM List for the different profiles.

12.2.2 Subsequent Related Processing

Terminal Action Analysis

The terminal uses cardholder verification results and card and terminal parameters to determine whether the transaction should be declined offline, sent online, or approved offline.

First Card Action Analysis

The card uses the Card Issuer Action Codes to determine whether the transaction should be declined offline, sent online, or approved offline if the offline PIN Try Limit was exceeded, offline PIN verification failed in the current transaction, or offline PIN verification was not performed in the current transaction.

Online Processing

If the CVM is Online PIN, then the card performs no PIN processing. The terminal enciphers the PIN and includes it in the online authorisation request.

If the CVM is Offline PIN, the PIN is not included in the online authorisation request but offline PIN verification results are included in the authorisation request and should be considered in the issuer's authorisation decision.

Second Card Action Analysis

If the terminal attempted to go online for an authorisation, but could not go online; then during Second Card Action Analysis the card will use rules personalised on the card (the CIAC-Default) to determine whether to decline the transaction (see section 17.5.5.2). These rules consider the following conditions that might occur during cardholder verification: the PIN Try Limit is exceeded, offline PIN verification failed, or offline PIN verification was not performed.

In the online response the issuer may reset or update the number of remaining PIN tries using the Card Status Update.

Issuer-to-Card Script Processing

The PIN CHANGE/UNBLOCK command can be used to change the Reference PIN or reset the PIN Try Counter.

12.3 Card Data

The card data used during Cardholder Verification are listed and described in Table 12-1. For a detailed description of these data and their usage, see Annex L: Data Dictionary.

Data	Description	Template	Tag
Application Control	Indicators used to activate or deactivate functions in the application.	—	'C1'
Application Currency Code	Used to determine whether the transaction is in the card's currency. If the CVM List is present and the value for either Amount X or Amount Y in the CVM List is not zero, the Application Currency Code shall be present in records pointed to by the AFL.	—	'9F42'
Application Interchange Profile (AIP)	Contains an indicator showing that the card supports cardholder verification.	—	'82'
Card Verification Results (CVR)	Contains a 4-bit PIN Try Counter value and indicators that the card sets for the following conditions: <ul style="list-style-type: none"> Offline PIN Verification Performed Offline PIN Verification Performed and PIN Not Successfully Verified PIN Try Limit Exceeded 	—	'9F52'
Cardholder Verification Method (CVM) List	Identifies a prioritised list of methods of cardholder verification for the card application. The terminal uses the CVM list in selecting the method(s) of cardholder verification to perform for the transaction. NOTE: The CVM List used for the transaction might vary dependent upon the profile selected for the transaction.	—	'8E'

Table 12-1: Cardholder Verification – Card Data
(continues)

Data	Description	Template	Tag
ICC PIN Encipherment Private Key or ICC Private Key	Stored securely on the card and never passed to the terminal. May be used to recover the enciphered PIN passed to the card in the VERIFY command during Offline Enciphered PIN verification.	—	—
ICC PIN Encipherment Public Key Certificate or ICC Public Key (PK) Certificate	Signed with the Issuer Private Key. Contains the public key that may be used to encipher the PIN for Offline Enciphered PIN. The format of the ICC PIN Encipherment PK Certificate is shown in <i>EMV Book 2</i> , Table 22.	—	'9F2D' or '9F46'
ICC PIN Encipherment Public Key Exponent or ICC Public Key Exponent	Contains the exponent used in the algorithm that enciphers the PIN for Offline Enciphered PIN. NOTE: The ICC PIN Encipherment Public Key Exponent or ICC Public Key Exponent is 3 or $2^{16} + 1$.	—	'9F2E' or '9F47'
ICC PIN Encipherment Public Key Remainder or ICC Public Key Remainder	Contains the portion, if necessary, of the public key that does not fit into the ICC's public key certificate.	—	'9F2F' or '9F48'
Issuer Public Key (PK) Certificate	Signed with the Payment System Private Key. Contains the public key to be used to recover the ICC PIN Encipherment or ICC Public Key from its Certificate.	—	'90'
Issuer Public Key Exponent	Contains the exponent used in the algorithm that decipheres the ICC PIN Encipherment or ICC PK Certificate.	—	'9F32'
Issuer Public Key Remainder	Contains the portion, if necessary, of the Issuer Public Key that does not fit into the Issuer PK Certificate.	—	'92'

Table 12-1: Cardholder Verification – Card Data, continued

Data	Description	Template	Tag
PIN Try Counter	Designates the number of PIN tries remaining. It is put in the VERIFY response to notify the terminal whether additional PIN entry attempts are permitted. The terminal may also obtain the value of the PIN Try Counter using the GET DATA command. The PIN Try Counter is decremented for each unsuccessful PIN verification attempt. It is reset to the PIN Try Limit each time the PIN is successfully verified and when the PIN CHANGE/UNBLOCK command changes the PIN, and may also be reset during CSU Processing in Second Card Action Analysis.	—	'9F17'
PIN Try Limit	Issuer-specified maximum number of consecutive incorrect PIN tries allowed	—	'C6'
Profile ID	Identifies the Profile Control to use in configuring the application behaviour for the transaction environment.	—	—
Reference PIN	The cardholder PIN which the card compares to the Transaction (key-entered) PIN during offline PIN processing.	—	—
Registered Application Provider Identifier (RID) portion of the Application Identifier (AID)	The first five bytes of the Application Identifier (AID) that identifies the Payment System. The RID is registered with ISO. The RID and the Certificate Authority Public Key Index are used to identify the Payment System Public Key to be used to recover the Issuer Public Key Certificate.	—	—

Table 12-1: Cardholder Verification – Card Data, continued

The issuer sets the 'Cardholder verification is supported' bit in the AIP to the value 1b to indicate that the application supports cardholder verification.

Req 12.1 (Support for CVM list):

The card shall contain a CVM List and may contain multiple CVM Lists for use in different types of transactions such as international and domestic transactions. Only a single CVM List shall be designated in an AFL to be read in a single transaction.

Req 12.2 (Format for CVM list):

Each entry in the CVM list shall be coded according to the Cardholder Verification Rule format described in EMV Book 3, section C3.

Req 12.3 (Reference PIN support):

The Reference PIN shall be present on the card if the issuer chooses to support offline PIN verification.

Req 12.4 (Reference PIN update):

The Reference PIN shall only be updated by an issuer script command using secure Messaging for integrity and confidentiality. The Reference PIN may be updated using the PIN CHANGE/UNBLOCK command described in section 18.

12.4 Terminal Data

The terminal data used in PIN Processing are listed and described in Table 12-2. For a detailed description of these data and their usage, see Annex L: Data Dictionary.

Data Element	Description	Template	Tag
CA Public Key Index	Used with the RID to identify which Payment System Private Key was used to sign the Issuer Public Key Certificate, and which corresponding Payment System Public Key is to be used to recover the Issuer Public Key Certificate.	—	'8F'
CVM Results	Indicates the results of processing the Cardholder Verification List.	—	'9F34'
Terminal Verification Results (TVR)	A series of indicators used to record the results from the terminal's perspective of card and terminal processing, including the results of cardholder verification checks.	—	'95'
Transaction PIN	Data entered by the cardholder for the purpose of PIN verification.	—	—

Table 12-2: PIN Processing – Terminal Data

12.5 GET DATA Command

The GET DATA command may be used by the terminal to obtain the PIN Try Counter from the card (for example, in order to determine whether the PIN Try Limit was exceeded on a previous transaction). The command also supports retrieval of additional application data elements, as described in section 12.5.2.

12.5.1 Command Coding

Code	Value
CLA	'80'
INS	'CA'
P1/P2	Tag
Lc	Not present
data	Not present
Le	'00'

Table 12-3: GET DATA Command Message

12.5.1.1 Command Format Validation

If P1 has the value '00', then P2 contains the single-byte Tag for the data element or template requested by the GET DATA command. Otherwise P1/P2 contains the two-byte Tag for the data element or template requested by the GET DATA command.

Req 12.5 (Check P1/P2 values for GET DATA):

If the P1 and P2 parameters are not set to the value of a Tag supported for retrieval using the GET DATA command, then the card shall discontinue processing the GET DATA command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6A88' (Referenced data object not found).

12.5.2 Processing

Req 12.6 (GET DATA support as described in EMV):

The card shall support the GET DATA command as described in EMV Book 3, section 6.5.7 for retrieval of the data elements listed in Annex J, Table J-1 that are supported by the GET DATA command.

See Annex G for additional information on the management of Profile resources using a single template tag per type of resource. The card returns all data elements in the template, in any order; and is allowed to send filler bytes of value '00' in the GET DATA response for a template tag.

Req 12.7 (GET DATA support for accumulator values and limits):

Retrieval of Values and Limits of Accumulators and Counters using the GET DATA command shall be an issuer-option. If the P1 and P2 parameters are set to the value 'BF30', then:

- *If the 'Allow Retrieval of Values and Limits of Accumulators and Counters' bit in the Application Control is set to the value 0b, then the card shall discontinue processing the GET DATA command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6985' (Conditions of use not satisfied).*
- *Otherwise, the card shall return to the terminal in the GET DATA response the template 'BF30' containing the TLV-coded concatenation of Accumulator x Value and Accumulator x Limit for all values of x for which Accumulator x is supported by the application.*

NOTE: If the GET DATA command is implemented at the application level, then the 'Allow Retrieval of Values and Limits of Accumulators and Counters' bit in the Application Control indicates whether the issuer allows the values and limits of Accumulators to be retrieved using the GET DATA command.

If GET DATA is provided by card level services (for example, the card Operating System) instead of the application, then whether the issuer allows the values and limits of Accumulators to be retrieved may be determined through other means beyond the scope of this specification, and this bit is RFU.

Req 12.8 (GET DATA support for counter values and limits):

Retrieval of Values and Limits of Accumulators and Counters using the GET DATA command shall be an issuer-option. If the P1 and P2 parameters are set to the value 'BF35', then:

- *If the 'Allow Retrieval of Values and Limits of Accumulators and Counters' bit in the Application Control is set to the value 0b, then the card shall discontinue processing the GET DATA command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6985' (Conditions of use not satisfied).*
- *Otherwise, the card shall return to the terminal in the GET DATA response the template 'BF35' containing the TLV-coded concatenation of Counter x Value and Counter x Limit for all values of x for which Counter x is supported by the application.*

NOTE: If the GET DATA command is implemented at the application level, then the 'Allow Retrieval of Values and Limits of Accumulators and Counters' bit in the Application Control indicates whether the issuer allows the values and limits of Counters to be retrieved using the GET DATA command.

If GET DATA is provided by card level services (for example, the card Operating System) instead of the application, then whether the issuer allows the values and limits of Counters to be retrieved may be determined through other means beyond the scope of this specification, and this bit is RFU.

Req 12.9 (GET DATA support for offline balance):

Retrieval of Offline Balance using the GET DATA command shall be an issuer-option. If the P1 and P2 parameters are set to the value '9F50', then:

- *If the 'Allow Retrieval of Values and Limits of Accumulators and Counters' bit in the Application Control is set to the value 0b, then the card shall discontinue processing the GET DATA command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6985' (Conditions of use not satisfied).*
- *Otherwise, the card shall return the Offline Balance to the terminal in the GET DATA response.*

NOTE: If the GET DATA command is implemented at the application level, then the 'Allow Retrieval of Values and Limits of Accumulators and Counters' bit in the Application Control indicates whether the issuer allows Offline Balance to be retrieved using the GET DATA command.

If GET DATA is provided by card level services (for example, the card Operating System) instead of the application, then whether the issuer allows retrieval of Offline Balance may be determined through other means beyond the scope of this specification, and this bit is RFU.

After the terminal determines that an offline PIN is to be entered, the terminal may transmit a GET DATA command to the card to retrieve the PIN Try Counter.

The card returns the tag, length, and value of the requested data element to the terminal in the GET DATA response as described in *EMV Book 3*, section 6.5.7.4.

12.5.3 GET DATA Flow Chart

Figure 12-1 shows the GET DATA processing flow.

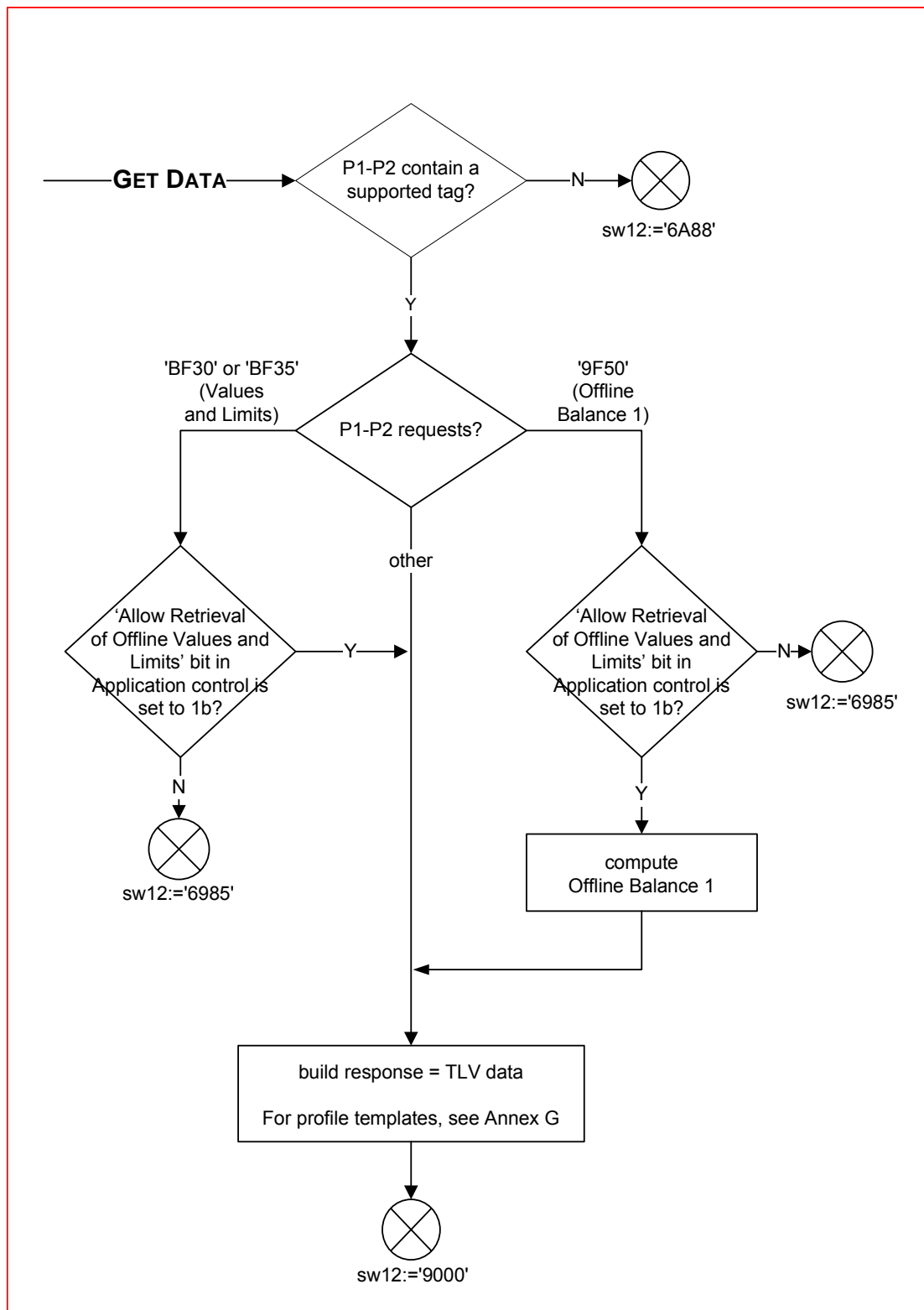


Figure 12-1: GET DATA Flow

12.6 GET CHALLENGE Command

If the CVM is Offline Enciphered PIN, then prior to issuing the VERIFY command the terminal uses the GET CHALLENGE command to request an unpredictable number from the card for use in Offline Enciphered PIN processing.

Req 12.10 (Support for GET CHALLENGE):

If the card supports the dynamic-RSA implementer-option, then the card shall support the GET CHALLENGE command as described in EMV Book 3, section 6.5.6.

12.6.1 Command Coding

Code	Value
CLA	'00'
INS	'84'
P1	'00'
P2	'00'
Lc	Not present
data	Not present
Le	'00'

Table 12-4: GET CHALLENGE Command Message

12.6.2 Processing

Command Format Validation

Req 12.11 (Check P1/P2 values for GET CHALLENGE):

If the P1 or P2 parameter is not set to the value '00', then the card shall discontinue processing the GET CHALLENGE command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6A86' (Incorrect Parameters, P1-P2).

Challenge Computation

EMV specifies that the challenge is valid only for the next issued command, so if offline enciphered PIN verification is used, the VERIFY command immediately follows the GET CHALLENGE command. That is, if a VERIFY command containing an enciphered PIN is received and the previous command was not a successful GET CHALLENGE, the VERIFY command fails PIN Verification. How this requirement is fulfilled is left to the implementation.

12.6.3 GET CHALLENGE Response

The data field of the response contains an 8-byte unpredictable number generated by the ICC.

The algorithm for computing the challenge is left to the implementation.

See section 20.4 for requirements on the generation of the ICC Unpredictable Number.

12.6.4 GET CHALLENGE Flow Chart

Figure 12-2 shows the GET CHALLENGE processing flow.

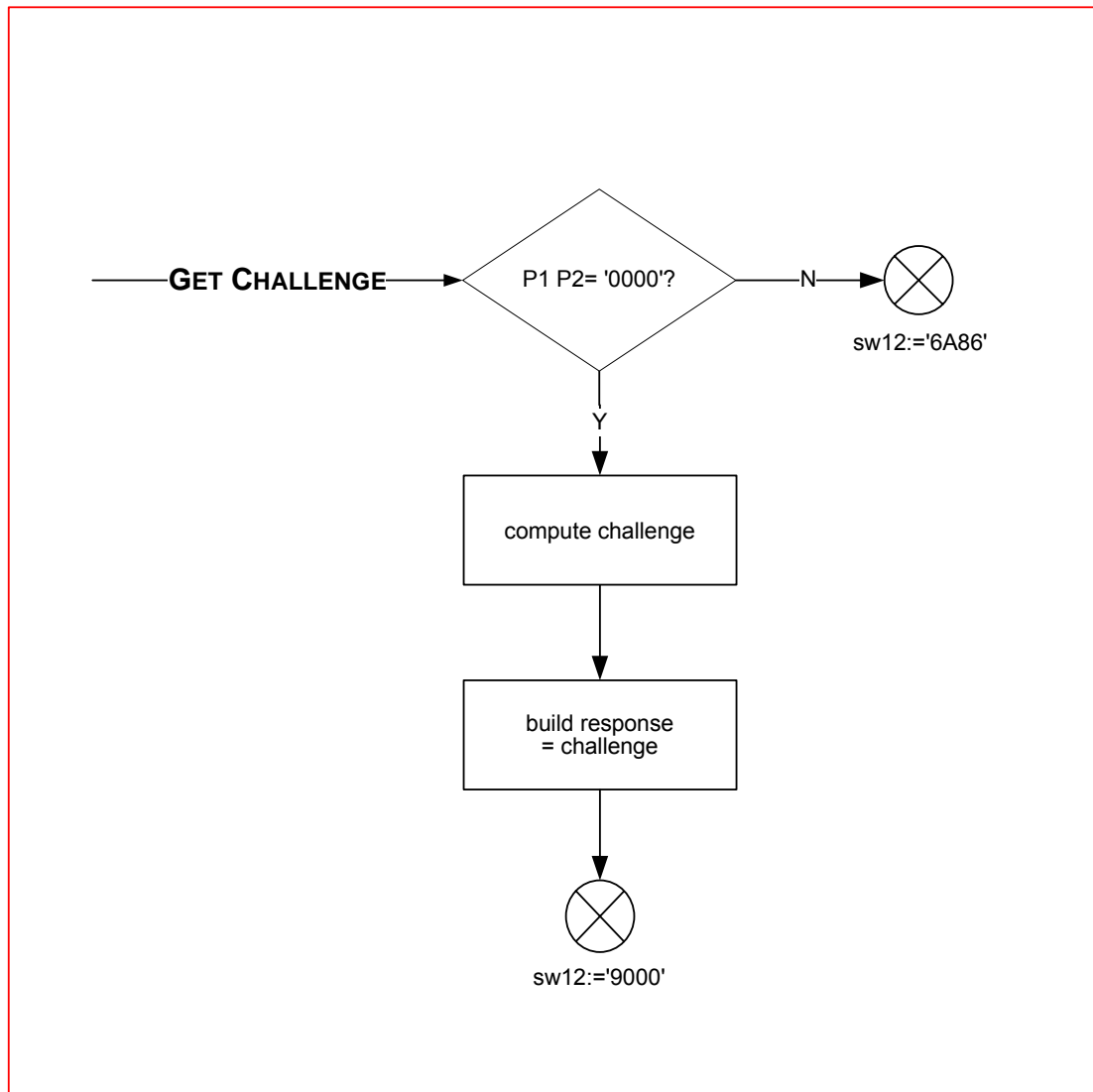


Figure 12-2: GET CHALLENGE Flow

12.7 VERIFY Command

The VERIFY command is used for Offline Enciphered PIN and Offline Plaintext PIN cardholder verification. The VERIFY command initiates the card comparison of the cardholder-entered Transaction PIN with the Reference PIN.

After each unsuccessful PIN Verification with PIN tries remaining (indicated in SW1 SW2 '63Cx' with $x \neq 0$), the terminal may request another PIN entry and send the card another VERIFY command. The cardholder may continue to enter a PIN until the PIN Try Counter is decremented to zero. At that time, the terminal is not anticipated to transmit any further VERIFY command messages to the card.

NOTE: The Status Words specified in section 12 are the EMV-specified Status Words to be used for processing the VERIFY command. They are used in the application to report any anomalies during VERIFY command processing. In the event the VERIFY command cannot be processed correctly, they allow the terminal to continue CVM processing as specified in *EMV Book 3*. Failing to use the Status Words specified in this section or using different values than those specified will cause the transaction to be terminated by the terminal, leading to an undesirable transaction outcome (from an issuer/cardholder point of view). It is considered that since such errors should not occur frequently, using the EMV-specified Status Words versus other ISO-defined Status Words would provide a better service for cardholders.

Req 12.12 (Support for VERIFY command):

The card shall support the VERIFY command as described in EMV Book 3, section 6.5.12.

12.7.1 Command Coding

Code	Value
CLA	'00'
INS	'20'
P1	'00'
P2	Qualifier of the reference data (See below)
Lc	Var.
data	Transaction PIN data
Le	Not present

Table 12-5: VERIFY Command Message

Qualifier of the reference data may take the following values:

- '80' PIN block in plaintext
- '88' enciphered PIN block

Other values are allowed, but the processing of those other values is beyond the scope of this specification.

12.7.1.1 Command Format Validation

The application sets application indicators to identify that PIN verification processing has occurred on this transaction.

NOTE: In the following requirements, CVR bits (such as Offline PIN Verification Performed) are used internal to the card to indicate the status of the offline PIN verification. The CVR is only externally available (and therefore is only testable) in the first and second GENERATE AC responses. Thus it is possible to achieve the same behaviour by using proprietary data internal to the card during processing of the VERIFY command to keep the PIN verification status, if the internal data is used to set the ADR and CVR bits in the first and second GENERATE AC.

Req 12.13 (Set offline PIN verification performed in CVR):

After the card receives the VERIFY command, and before checking the values for P1 and P2, the application shall set the 'Offline PIN Verification Performed' bit in the CVR to the value 1b.

The application validates the format of the VERIFY command.

Req 12.14 (Check P1 value for VERIFY):

If the P1 parameter is set to a value other than '00', then the card shall discontinue processing the VERIFY command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6984' (Command not allowed; referenced data invalidated).

Req 12.15 (Support for offline plaintext PIN in P2):

A CPA card shall support the value '80' (Offline Plaintext PIN) in the P2 parameter.

Req 12.16 (Support for offline enciphered PIN in P2):

A card that supports the Dynamic-RSA implementer-option shall support the value '88' (Offline Enciphered PIN) in the P2 parameter.

Req 12.17 (P2 value not supported):

If the P2 parameter is set to a value not supported by the card, then the card shall discontinue processing the VERIFY command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6984' (Command not allowed; referenced data invalidated).

12.7.2 Processing

The application checks whether further PIN tries remain.

Req 12.18 (PIN Try Limit Previously Exceeded):

If the PIN Try Counter has the value zero, then the application shall set the 'Offline PIN Verification Performed and PIN Not Successfully Verified' bit in the CVR to the value 1b, shall discontinue processing the VERIFY command, and shall respond with SW1 SW2 = '6983' (Command not allowed; authentication method blocked).

Req 12.19 (Failure of PIN verification does not block application):

The application shall not be blocked by a failure in PIN verification such as the PIN Try Limit being exceeded.

Req 12.20 (Failure of PIN verification does not block card):

The card shall not be blocked by a failure in PIN verification such as the PIN Try Limit being exceeded.

Next the application processes the VERIFY command as indicated by the P2 parameter.

Req 12.21 (Processing for offline plaintext PIN processing):

If the P2 parameter is set to the value '80' (Offline Plaintext PIN), then the card shall perform offline plaintext PIN processing as described in section 12.7.2.1.

Req 12.22 (Processing for offline enciphered PIN processing):

If the P2 parameter is set to the value '88' (Offline Enciphered PIN), then the card shall perform offline enciphered PIN processing as described in section 12.7.2.2.

12.7.2.1 Offline Plaintext PIN Verification

The card checks that the issuer allows offline plaintext PIN verification, and that the command data is the correct length.

Req 12.23 (Application Control does not support plaintext PIN):

If the 'Offline Plaintext PIN Verification Supported' bit in the Application Control has the value 0b, then the card:

- *shall set the 'Offline PIN Verification Performed and PIN Not Successfully Verified' bit in the CVR to the value 1b.*
- *shall discontinue processing the VERIFY command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6984' (Command not allowed; referenced data invalidated).*

Req 12.24 (Check command data length for plaintext PIN):

If Lc has a value other than '08', then the card:

- *shall set the 'Offline PIN Verification Performed and PIN Not Successfully Verified' bit in the CVR to the value 1b.*
- *shall discontinue processing the VERIFY command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6984' (Command not allowed; referenced data invalidated).*

The application verifies the PIN by validating that the PIN block is formatted correctly, and that the PIN received in the command matches the PIN stored internal to the application. The PIN Block format is described in *EMV Book 3*, Table 24.

Req 12.25 (Check PIN Block format for plaintext PIN):

*If the Transaction PIN Data does not meet **all** of the following conditions:*

- *the Control field of the Transaction PIN Data has the value '2'*
- *the PIN Length field has a value greater than or equal to '4' and less than or equal to 'C'.*
- *the Filler digits of the Transaction PIN Data have the value 'F'*

then the PIN Block format is invalid and the PIN cannot be verified. The application:

- *shall set the 'Offline PIN Verification Performed and PIN Not Successfully Verified' bit in the CVR to the value 1b,*
- *shall discontinue processing the VERIFY command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6984' (Command not allowed; referenced data invalidated).*

Otherwise (that is, the PIN Block format is valid), the application shall:

- *decrement the PIN Try Counter by one*
- *compare the Transaction PIN to the Reference PIN.*
 - *If the Transaction PIN does not match the Reference PIN, then PIN verification fails and the application shall:*
 - *set the 'Offline PIN Verification Performed and PIN Not Successfully Verified' bit in the CVR to the value 1b*
 - *respond to the Verify command with SW1 SW2 = '63Cx', where 'x' represents the number of PIN tries remaining.*
 - *If the Transaction PIN matches the Reference PIN, then PIN verification is successful and the application shall:*
 - *set the 'Offline PIN Verification Performed and PIN Not Successfully Verified' bit in the CVR to the value 0b.*
 - *reset the PIN Try Counter to the value of the PIN Try Limit.*
 - *indicate successful completion of the command by responding with SW1 SW2 = '9000'.*

12.7.2.2 Offline Enciphered PIN Verification

The card checks that the issuer allows offline enciphered PIN verification, and the command data is the correct length.

Req 12.26 (Application Control does not support enciphered PIN):

If the 'Offline Enciphered PIN Verification Supported' bit in the Application Control has the value 0b, then the card:

- *shall set the 'Offline PIN Verification Performed and PIN Not Successfully Verified' bit in the CVR to the value 1b.*
- *shall discontinue processing the VERIFY command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6984' (Command not allowed; referenced data invalidated).*

Req 12.27 (Check length for PIN enciphered using ICC private key):

*If the key used for Offline Enciphered PIN Verification is the ICC private key **and** Lc does not equal N_{IC} , then the card shall discontinue processing the Verify command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6984' (Command not allowed; referenced data invalidated)*

NOTE: If the VERIFY command is implemented at the application level, the 'Key Pair Used for Offline Enciphered PIN Verification' bit in the Application Control indicates whether the issuer has selected the ICC Private Key or an ICC PIN Encipherment Key for encipherment of the offline PIN. If Offline PIN Verification is provided by the platform instead of the application; the key used for Offline Enciphered PIN Verification may be determined through other means beyond the scope of this specification, and this bit is RFU.

Req 12.28 (Check length for PIN enciphered using ICC PIN encipherment key):

*If the key used for Offline Enciphered PIN Verification is the ICC PIN Encipherment Private Key **and** Lc does not equal N_{PE} , then the card shall discontinue processing the Verify command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6984' (Command not allowed; referenced data invalidated)*

In order to process the offline enciphered PIN, the terminal needs an unpredictable number from the card. The terminal issues a GET CHALLENGE command (see section 12.6) prior to processing an offline enciphered PIN. The ICC Unpredictable Number sent to the terminal in the response remains available in the card for the application to complete verification of the offline enciphered PIN. If a VERIFY command containing an enciphered PIN is received and the previous command was not a successful GET CHALLENGE, then the VERIFY command fails PIN Verification. How this requirement is fulfilled is left to the implementation.

Req 12.29 (No GET CHALLENGE before VERIFY):

If no challenge from the GET CHALLENGE command immediately prior to the VERIFY command is available, then PIN verification fails and the application:

- *shall set the 'Offline PIN Verification Performed and PIN Not Successfully Verified' bit in the CVR to the value 1b.*
- *shall discontinue processing the VERIFY command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6984' (Command not allowed; referenced data invalidated).*

The application deciphers the transaction PIN data to recover the PIN and verifies the recovered PIN as follows. EMV Book 2 provides additional detail on generating and using the RSA public/private key data elements required for Offline Enciphered PIN.

If the key used for offline enciphered PIN is the ICC PIN Encipherment key, then the application uses the ICC PIN Encipherment private key to decipher the transaction PIN data as described in *EMV Book 2*, section 7.2, steps 6 through 9.

If the key used for offline enciphered PIN is the ICC private key, then the application uses the ICC private key used for DDA to decipher the transaction PIN data as described in *EMV Book 2*, section 7.2, steps 6 through 9.

Req 12.30 (Check format of recovered data):

*After deciphering the Transaction PIN Data, if the recovered data does not meet **both** of the following conditions:*

- *the recovered ICC Unpredictable Number matches the ICC Unpredictable Number sent in the response to the GET CHALLENGE command immediately preceding the VERIFY command,*
- ***and** the recovered Data Header has the value '7F',*

then the application:

- *shall fail PIN Verification*
- *shall set the 'Offline PIN Verification Performed and PIN Not Successfully Verified' bit in the CVR to the value 1b.*
- *shall discontinue processing the VERIFY command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6984' (Command not allowed; referenced data invalidated).*

Otherwise the application shall continue with verification of the recovered PIN Block.

The application verifies the PIN by validating that the recovered PIN block is formatted correctly, and that the recovered PIN matches the PIN stored internal to the application. The PIN Block format is described in *EMV Book 3*, Table 24.

Req 12.31 (Check PIN Block format for enciphered PIN):

*If the recovered PIN Block does not meet **all** of the following conditions:*

- *the Control field of the recovered PIN block has the value '2'*
- *the PIN Length field of the recovered PIN block has a value greater than or equal to '4' and less than or equal to 'C'.*
- *the Filler digits of the recovered PIN block have the value 'F'*

then the PIN Block format is invalid and the PIN cannot be verified. The application shall set the 'Offline PIN Verification Performed and PIN Not Successfully Verified' bit in the CVR to the value 1b, shall discontinue processing the VERIFY command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6984' (Command not allowed; referenced data invalidated).

Otherwise (that is, the PIN Block format is valid), the application shall:

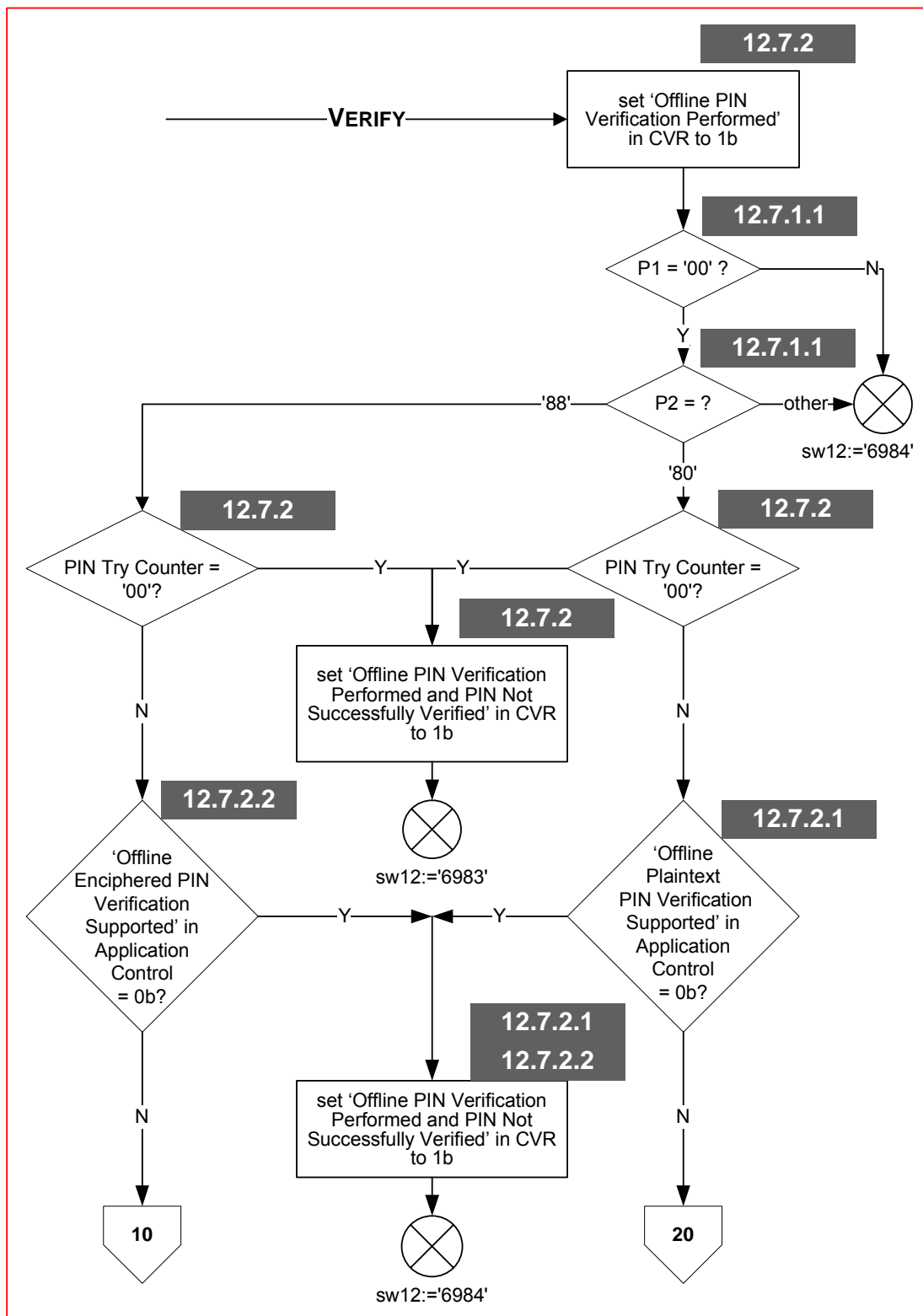
- *decrement the PIN Try Counter by one*
- *compare the recovered PIN to the Reference PIN.*
 - *If the recovered PIN does not match the Reference PIN, then PIN verification fails and the application shall:*
 - *set the 'Offline PIN Verification Performed and PIN Not Successfully Verified' bit in the CVR to the value 1b*
 - *respond to the Verify command with SW1 SW2 = '63Cx', where 'x' represents the number of PIN tries remaining.*
 - *If the recovered PIN matches the Reference PIN, then PIN verification is successful and the application shall:*
 - *set the 'Offline PIN Verification Performed and PIN Not Successfully Verified' bit in the CVR to the value 0b.*
 - *reset the PIN Try Counter to the value of the PIN Try Limit.*
 - *indicate successful completion of the command by responding with SW1 SW2 = '9000'.*

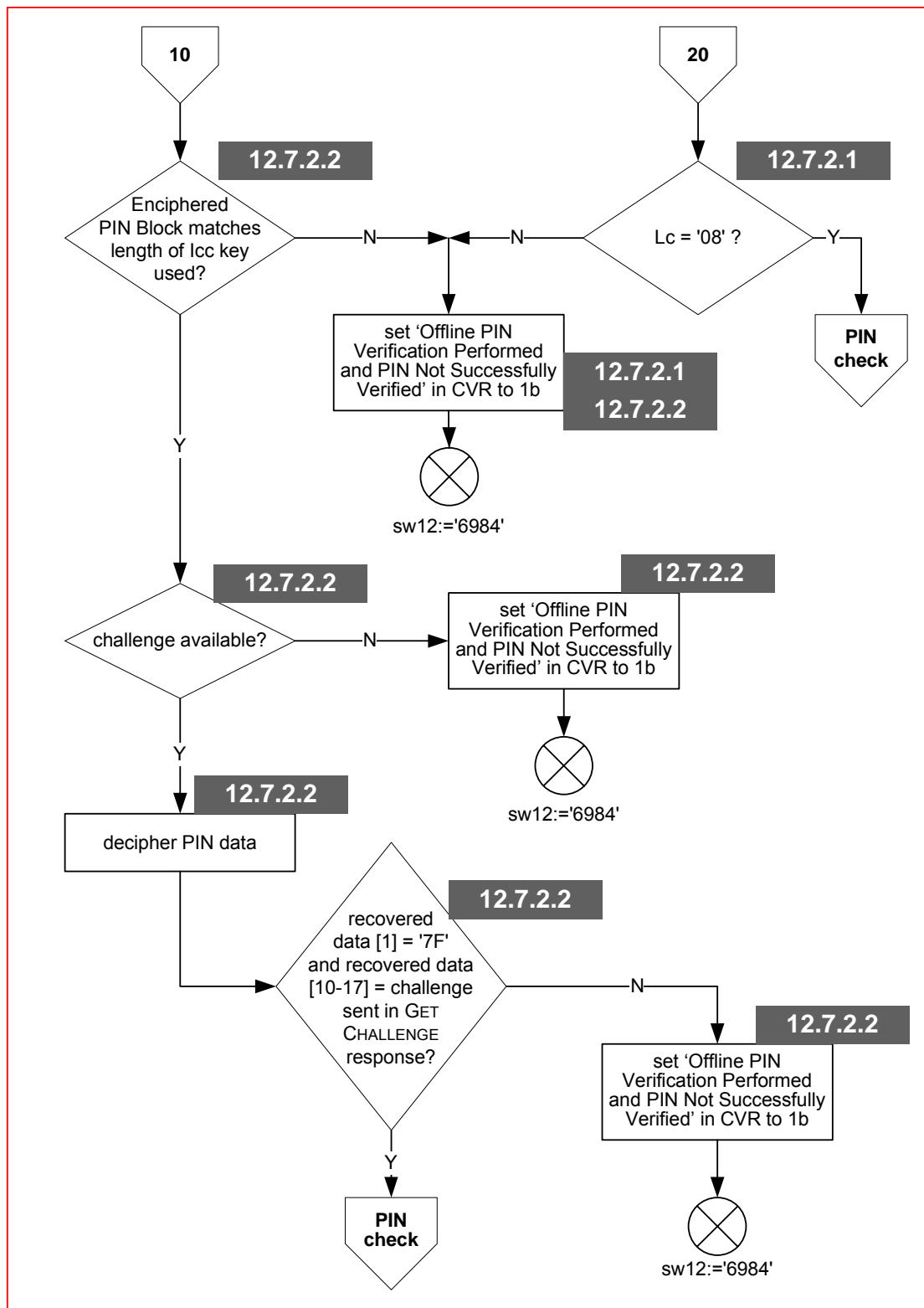
NOTE: Additional security requirements for protection of the Reference PIN and RSA private keys are detailed in sections 20.1 and 20.2.

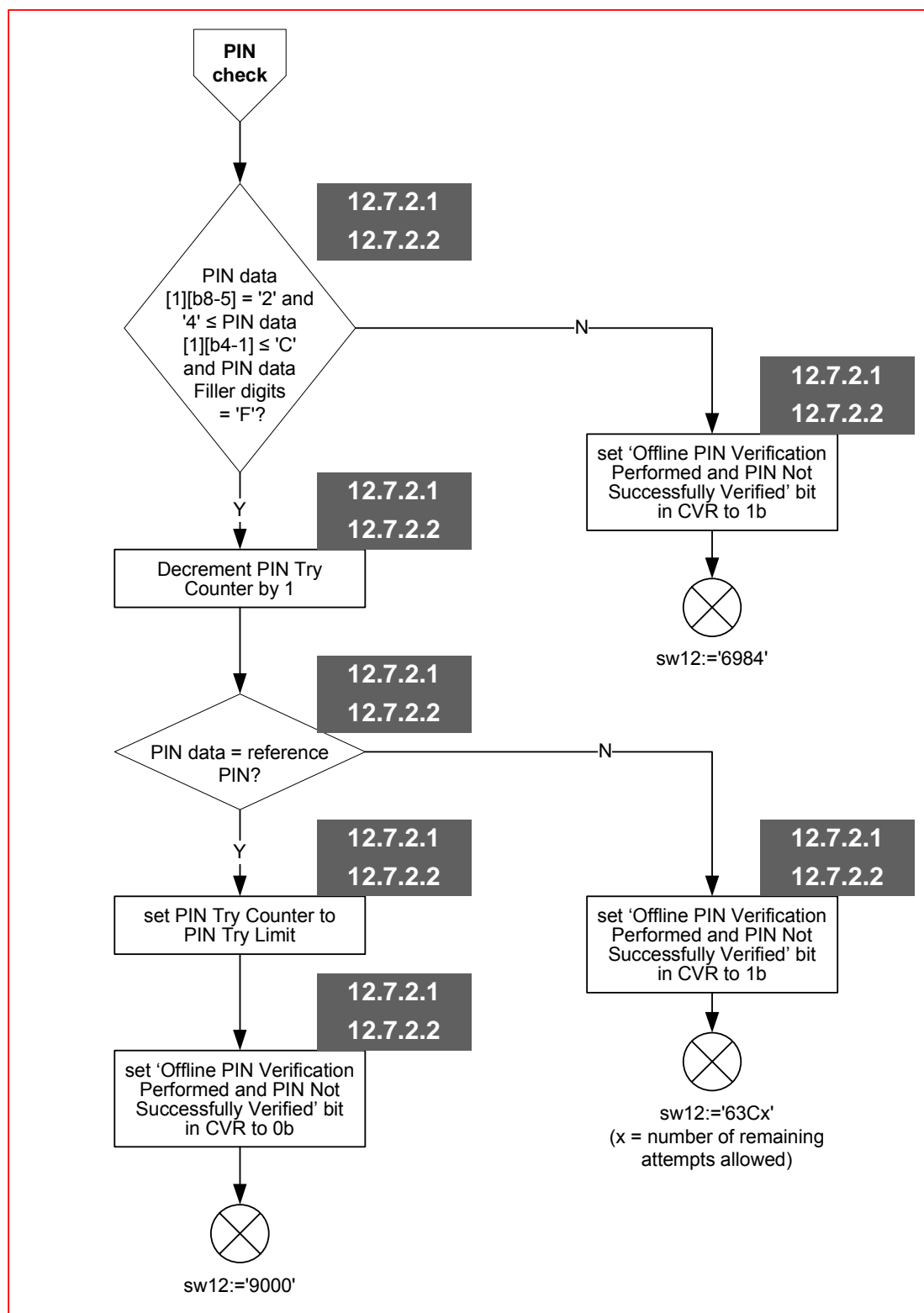
12.7.3 VERIFY Flow Chart

Figure 12-3 shows how an application could process a VERIFY command.

Figure 12-3: VERIFY Flow







13 Terminal Risk Management

This section is organised as follows:

- 13.1 Purpose
- 13.2 Sequence of Execution
 - 13.2.1 Prior Related Processing
 - 13.2.2 Subsequent Related Processing
- 13.3 Card Data
- 13.4 Terminal Data
- 13.5 Processing
 - 13.5.1 Terminal Exception File
 - 13.5.2 Merchant Forced Transaction Online
 - 13.5.3 Floor Limits
 - 13.5.4 Random Transaction Selection
 - 13.5.5 Terminal Velocity Checking

13.1 Purpose

NOTE: In order to provide a more complete description of transaction processing with the CPA application, this section provides an overview of functionality provided by the EMV terminal. CPA does not require any change to EMV terminal functionality.

Terminal Risk Management ensures higher-value transactions go online for issuer authorisation, and ensures that transactions go online periodically to protect against credit and fraud risks that might be undetectable in an offline environment.

Issuers may require the terminal to perform Terminal Risk Management by setting the 'Terminal Risk Management is to be performed' bit in the AIP to the value 1b. Terminals may perform Terminal Risk Management whether or not it is required by the card.

Terminal Risk Management is performed as described in *EMV Book 3*, section 10.6 and *EMV Book 4*, section 6.3.5.

13.2 Sequence of Execution

13.2.1 Prior Related Processing

Read Application Data

The following data is read from the card:

- Application Primary Account Number (PAN) and Application PAN Sequence Number used in checking the terminal exception file and Floor Limit.

13.2.2 Subsequent Related Processing

Terminal Action Analysis

Based on card and terminal settings, the terminal determines what action to take if:

- Card was on terminal exception file
- Merchant forced transaction online
- Terminal Floor Limit was exceeded
- Transaction was randomly selected for online processing

13.3 Card Data

The card data used in Terminal Risk Management are listed and described in Table 13-1. For a detailed description of these data and their usage, see Annex L: Data Dictionary.

Data	Description	Template	Tag
Application PAN Sequence Number	An identifier used to differentiate cards with the same Application PAN.	—	'5F34'
Application Primary Account Number (PAN)	The cardholder account number for the application	—	'5A'

Table 13-1: Terminal Risk Management – Card Data

13.4 Terminal Data

The terminal data used in Terminal Risk Management are described in *EMV Book 3*, section 10.6 and annex A1.

13.5 Processing

The CPA does no processing during Terminal Risk Management.

The following describes how the terminal uses data from the card during the Terminal Risk Management processes:

13.5.1 Terminal Exception File

If a terminal exception file is present, the terminal may check whether the Application Primary Account Number (PAN) and Application PAN Sequence Number from the card is listed on the exception file.

13.5.2 Merchant Forced Transaction Online

At online-capable terminals, the merchant may indicate to the terminal that the transaction should be processed online. No card data is used in this process.

13.5.3 Floor Limits

Floor limit checking is performed so that transactions with amounts above the Terminal Floor Limit are sent online for authorisation. See *EMV Book 3*, section 10.6.1 for further information. No card data is used in this process.

13.5.4 Random Transaction Selection

Terminals capable of supporting both offline and online transactions may randomly select transactions for online processing. See *EMV Book 3*, section 10.6.2 for further information. No card data is used in this process.

13.5.5 Terminal Velocity Checking

The CPA performs velocity-checking risk management on the card as part of card risk management, so the CPA does not request the terminal to perform velocity-checking risk management.

Because the Lower Cumulative Offline Limit and Upper Cumulative Offline Limit are not present in the application, the terminal will not perform the Velocity Checking and New Card Check described in *EMV Book 3*, section 10.6.3.

14 Terminal Action Analysis

This section is organised as follows:

- 14.1 Purpose
- 14.2 Sequence of Execution
 - 14.2.1 Prior Related Processing
 - 14.2.2 Subsequent Related Processing
- 14.3 Card Data
- 14.4 Terminal Data
- 14.5 Processing
 - 14.5.1 Review Offline Processing Results
 - 14.5.2 Request Cryptogram Processing

14.1 Purpose

NOTE: In order to provide a more complete description of transaction processing with the CPA application, this section provides an overview of functionality provided by the EMV terminal. CPA does not require any change to EMV terminal functionality.

In Terminal Action Analysis, the terminal applies rules set by the issuer in the card and by the payment system in the terminal to the results of offline processing to determine whether the transaction should be approved offline, declined offline, or sent online for an authorisation.

Terminal Action Analysis is performed as described in *EMV Book 3*, section 10.7.

14.2 Sequence of Execution

14.2.1 Prior Related Processing

Read Application Data

During Read Application Data, the card sends application data records to the terminal. These data records include the IACs and CDOL1 that are used during Terminal Action Analysis.

14.2.2 Subsequent Related Processing

First Card Action Analysis

During First Card Action Analysis, the card performs additional risk management to determine whether it overrules the terminal's Terminal Action Analysis decision to approve offline or send online. A terminal decision to decline offline cannot be overridden by the card.

14.3 Card Data

The card data used in Terminal Action Analysis are listed and described in Table 14-1. For a detailed description of these data and their usage, see Annex L: Data Dictionary.

Data	Description	Template	Tag
Card Risk Management Data Object List 1 (CDOL1)	The CDOL1 contains the tags and lengths for the terminal data objects which are needed by the card to generate an application cryptogram or for other card processing. Refer to Annex L: Data Dictionary for CDOL1 requirements. Section 15, First Card Action Analysis, shows the CDOL1 requirements for Card Action Analysis.	—	'8C'
Issuer Action Codes (IACs)	The IACs are three data elements, each consisting of a series of bits corresponding to the bits in the Terminal Verification Results (TVR). During personalisation, the issuer should set an IAC bit to 1b if the corresponding TVR condition is to result in the action designated by the IAC. The three IACs are:		
	<ul style="list-style-type: none"> IAC—Denial The issuer sets to the value 1b the IAC bits that correspond to the TVR bits for conditions which the issuer wishes to result in an offline decline. 	—	'9F0E'
	<ul style="list-style-type: none"> IAC—Online The issuer sets to the value 1b the IAC bits that correspond to the TVR bits for conditions which the issuer would like to result in an online authorisation. 	—	'9F0F'
	<ul style="list-style-type: none"> IAC—Default The issuer sets to the value 1b the IAC bits that correspond to the TVR bits for conditions which the issuer would like to default to an offline decline if online processing is requested but not available. 	—	'9F0D'

Table 14-1: Terminal Action Analysis – Card Data

The IACs are included in the data elements recommended for validation by Offline Data Authentication. If the IACs are included in the validation data for the Signed Static Application Data (SSAD), the ICC PK Certificate, or the ICC PIN Encipherment PK Certificate, then the IACs should not be changed without also updating the SSAD and certificates as necessary. Otherwise SDA, DDA, CDA, and (if the ICC Private Key is used for offline enciphered PIN) offline enciphered PIN will fail.

14.4 Terminal Data

The terminal data used in Terminal Action Analysis are listed and described in Table 14-2. For a detailed description of these data and their usage, see Annex L: Data Dictionary and *EMV Book 3*, section 10.7.

Data	Description	Template	Tag
Terminal Action Codes (TACs)	The TACs are three data elements each consisting of a series of bits corresponding to the bits in the Terminal Verification Results (TVR). Similar to card's IACs, the TAC bits are set to 1b if the corresponding TVR bit is to result in the action specified by the TAC. These actions are decline offline, go online for an authorisation, and decline offline if the online authorisation is unable to complete.	—	—
Terminal Verification Results (TVR)	A series of indicators used to record the results of offline processing by the terminal, including the results of all terminal risk management checks.	—	'95'

Table 14-2: Review Offline Processing Results—Terminal Data

14.5 Processing

Terminal Action Analysis consists of two steps:

- Review Offline Processing Results
- Request Cryptogram

14.5.1 Review Offline Processing Results

The Review Offline Processing Results step of Terminal Action Analysis is performed entirely within the terminal. The terminal reviews the results of offline processing to determine whether the transaction should go online, be approved offline, or be declined offline. This process considers issuer criteria from the card called Issuer Action Codes (IACs), and acquirer criteria in the terminal called Terminal Action Codes (TACs).

The card performs no processing during the Review Offline Processing step. During processing the terminal compares bits in the IACs and TACs to the corresponding bits in the TVR.

If any TVR bit has the value 1b, and the corresponding bit in either the IAC or TAC also has the value 1b, then the terminal requests the action specified by the IAC/TAC type:

- decline the transaction offline by requesting an AAC type Application Cryptogram
- send the transaction online by requesting an ARQC type Application Cryptogram

Otherwise, the terminal will request that the card approve the transaction offline by requesting a TC type Application Cryptogram.

For a detailed description of this processing, see *EMV Book 3*, section 10.7.

14.5.2 Request Cryptogram Processing

In the Request Cryptogram Processing step of Terminal Action Analysis, the terminal formats the first GENERATE APPLICATION CRYPTOGRAM (GENERATE AC) command and issues it to the card requesting generation of a Triple DES Application Cryptogram. The command includes the value of the data elements whose tags and lengths were listed in the CDOL1. The terminal indicates in a command parameter whether Combined DDA/AC Generation (CDA) is to be performed. CDA is not requested when the terminal requests an AAC type Application Cryptogram (offline decline). For details regarding the coding and processing of the first GENERATE AC command see section 15.5, First GENERATE AC Command.

Terminal Action Analysis is complete when the terminal issues the First GENERATE AC command to the card. When the card receives the First GENERATE AC command, it proceeds to First Card Action Analysis (section 15).

15 First Card Action Analysis

This section is organised as follows:

- 15.1 Purpose
- 15.2 Sequence of Execution
 - 15.2.1 Prior Related Processing
 - 15.2.2 Subsequent Related Processing
- 15.3 Card Data
- 15.4 Terminal Data
- 15.5 First GENERATE AC Command
 - 15.5.1 Command Coding
 - 15.5.2 Profile Behaviour
 - 15.5.3 Card Risk Management Processing
 - 15.5.4 Determine Response Application Cryptogram Type
 - 15.5.5 Application Approves Transaction Offline
 - 15.5.6 Application Requests Online Processing
 - 15.5.7 Application Declines Transaction Offline
 - 15.5.8 Respond to GENERATE AC Command
- 15.6 Function Flow Charts

15.1 Purpose

First Card Action Analysis allows issuers to perform velocity checking and other risk management checks that are internal to the card. Card Risk Management features described in this section include checking:

- Activity on the current and previous transactions
- Offline transaction counters and amount accumulators

After completing Card Risk Management, the card returns an Application Cryptogram to the terminal. This cryptogram is an AAC for an offline decline, an ARQC for a request for an online authorisation, and a TC for an offline approval. If supported by both the card and terminal, the terminal will request Combined DDA/AC Generation (CDA) if it requests an ARQC or TC as part of a dynamic signature.

Card Action Analysis is performed as described in *EMV Book 3*, section 10.8 and section 10.11.

15.2 Sequence of Execution

15.2.1 Prior Related Processing

Initiate Application Processing

The card determines the Profile ID of the Profile to be used for processing the transaction. The Profile Control associated with the Profile ID specifies how Card Risk Management is configured for processing the transaction.

The terminal receives the Application Interchange Profile (AIP) in the GET PROCESSING OPTIONS response from the card which indicates whether the card supports CDA.

Read Application Data

The terminal reads the CDOL1 from the card.

Terminal Action Analysis

The terminal issues the first GENERATE AC command to the card to request a cryptogram. The command contains the data requested in the CDOL1 including the data required for Card Risk Management and generation of the application cryptogram.

15.2.2 Subsequent Related Processing

Online Processing

The terminal uses the cryptogram type specified in the Cryptogram Information Data (CID) of the first GENERATE AC response to determine whether to perform an online authorisation.

Second Card Action Analysis

If online processing was requested but the terminal was unable to go online, additional Card Risk Management checks are performed.

If the transaction went online, indicators and counters used in Card Action Analysis are updated based upon Issuer Authentication status and card options regarding Issuer Authentication.

15.3 Card Data

The card data used in Card Action Analysis are listed and described in Table 15-1. For a detailed description of these data and their usage, see Annex L: Data Dictionary.

NOTE: Several data have multiple instances in the application. This is indicated with the notation Data Name x (for example, Accumulator 1 and Accumulator 2 may be referred to as Accumulator x).

Data	Description	Template	Tag
Accumulator Currency Code	A code indicating the currency in which an accumulator is managed.	—	—
Accumulator Profile Control for Accumulator x	Defines the profile-specific behaviour for Accumulator x in the Profile selected for the transaction; including whether to send the value in the Issuer Application Data, and which limit set to use.	—	—
Accumulator Profile Control x	Defines profile-specific behaviour for an accumulator; including whether to send the value in the Issuer Application Data, and which limit set to use. The Profile Control identifies which Accumulator Profile Control x is used by each accumulator that is active in the Profile.	'BF31'	'DF0x'
Accumulator x	Represents a cumulative amount of transactions. May include offline approved transactions, and may also include online approved transactions. Transactions can be accumulated if they are in the accumulator currency, or (if currency conversion is allowed for the accumulator) in a currency that can be converted to the accumulator currency using the Currency Conversion Table for Accumulator x.	'BF30'	'DF0x'
Accumulator x Balance	Represents the amount of offline spending available, calculated as: Accumulator x Upper Limit minus the value of Accumulator x	—	—

Table 15-1: First Card Action Analysis – Card Data

Data	Description	Template	Tag
Accumulator x Control	Identifies the behaviour specific to Accumulator x independent of the Profile selected for the transaction; including which types of transactions are accumulated, and the currency code in which transactions are accumulated.	'BF32'	'DF0x'
Accumulator x Limit Set	Each Accumulator x has two Limit Sets, with one each of the following in each Limit Set:	'BF30'	'DF1x'
	Accumulator x Lower Limit	The lower of two limits for the maximum value of Accumulator x. A bit is set in the CVR and ADR when the Accumulator x value exceeds this limit.	
	Accumulator x Upper Limit	The upper of two limits for the maximum value of Accumulator x. A bit is set in the CVR and ADR when the Accumulator x value exceeds this limit.	
Additional Check Table x	Table containing values that are compared to values returned by the terminal in first GENERATE AC command data. Allows the issuer to specify additional Card Risk Management checks at personalisation.	'BF33'	'DF0x'
Application Control	Indicators used to activate or deactivate functions in the application.	—	'C1'
Application Cryptogram	<p>A cryptogram value that is returned by the card in the response to the first GENERATE AC command. The cryptogram type will be one of the following:</p> <ul style="list-style-type: none"> • an Application Authentication Cryptogram (AAC) for offline declines • a Transaction Certificate (TC) for offline approvals • an Authorisation Request Cryptogram (ARQC) when online processing is requested 	—	'9F26'

Table 15-1: First Card Action Analysis – Card Data, continued

Data	Description	Template	Tag
Application Decisional Results (ADR)	Internal indicators used to identify exceptional conditions that occur during the current and previous transactions. The Card Issuer Action Codes (CIACs) are compared to the Application Decisional Results to decide whether transactions should be declined offline or go online.	—	—
Application Transaction Counter (ATC)	A counter of the number of transactions initiated since the application was put on the card.	—	'9F36'
Card Risk Management Data Object List 1 (CDOL1)	List of tags and lengths of the data elements that the terminal must pass to the card application with the first GENERATE AC command.	—	'8C'
Card Verification Results (CVR)	Indicates the results of offline processing from current and previous transactions from an application perspective. This data is transmitted online as part of the Issuer Application Data.	—	'9F52'
CIACs Entry x	Identifies the profile-specific options selected by the issuer for processing transactions with CIACs ID = 'x'.	'BF34'	'DF0x'
	Card Issuer Action Code – Decline (CIAC – Decline)	Compared to the Application Decisional Results to indicate situations when the issuer specifies a transaction is to be declined offline.	
	Card Issuer Action Code – Default (CIAC – Default)	Compared to the Application Decisional Results to indicate situations when the issuer specifies a transaction is to be declined if the terminal is unable to go online.	
	Card Issuer Action Code – Online (CIAC – Online)	Compared to the Application Decisional Results to indicate situations when the issuer specifies a transaction is to go online if the terminal is online-capable.	

Table 15-1: First Card Action Analysis – Card Data, continued

Data	Description		Template	Tag
Counter Profile Control for Counter x	Defines the profile-specific behaviour for Counter x in the Profile selected for the transaction; including whether to send the value in the Issuer Application Data, and which limit set to use.		—	—
Counter Profile Control x	Defines profile-specific behaviour for a counter; including whether to send the value in the Issuer Application Data, and which limit set to use. The Profile Control identifies which Counter Profile Control x is used by each counter that is active in the Profile.		'BF36'	'DF0x'
Counter x	Represents a count of applicable transactions since Counter x was last reset. May include offline approved transactions, international country transactions, non-accumulated transactions, online approved transactions, or declined transactions.		'BF35'	'DF0x'
Counter x Control	Identifies the behaviour specific to Counter x independent of the Profile used for the transaction, such as which transactions are counted.		'BF37'	'DF0x'
Counter x Limit Set	Each Counter x has two Limit Sets, with one each of the following in each Limit Set:		'BF35'	'DF1x'
	Counter x Lower Limit	The lower of two limits for the maximum value of Counter x. A bit is set in the CVR and ADR when the Counter x value exceeds this limit.		
	Counter x Upper Limit	The upper of two limits for the maximum value of Counter x. A bit is set in the CVR and ADR when the Counter x value exceeds this limit.		

Table 15-1: First Card Action Analysis – Card Data, continued

Data	Description	Template	Tag
Cryptogram Information Data (CID)	Returned to the terminal in the GENERATE AC response. The CID designates the type of cryptogram that is being returned, and indicates that no advice is required and that no information is given for the reason/advice/referral code.	—	'9F27'
Currency Conversion Table x	Contains the Target Currency Code and one or more Currency Conversion Parameters that may each be used to convert a transaction in a recognised currency into an approximate value for the transaction in the accumulator currency or in the currency associated with the Maximum Transaction Amount.	'BF38'	'DF0x'
Cyclic Accumulator Profile Control for Cyclic Accumulator x	Defines the profile-specific behaviour for Cyclic Accumulator x in the Profile selected for the transaction, including whether to accumulate in the Profile, and which Limit Entry to use.	—	—
Cyclic Accumulator Profile Control x	Defines profile-specific behaviour for a cyclic accumulator; including whether to accumulate in the Profile, and which Limit Entry to use. The Profile Control identifies which Cyclic Accumulator Profile Control x is used by each cyclic accumulator that is active in the Profile.	'BF39'	'DF0x'

Table 15-1: First Card Action Analysis – Card Data, continued

Data	Description	Template	Tag
Cyclic Accumulator x	Represents a cumulative amount of approved transactions processed within a cycle. Includes at least offline approved transactions, but may also include online approved transactions. The cycle may be a day, week, or month. Transactions can be accumulated if they are in the accumulator currency, or in a currency that can be converted to the accumulator currency using the Currency Conversion Table for Cyclic Accumulator x.	'BF42'	'DF0x'
Cyclic Accumulator x Control	Identifies the behaviour specific to Cyclic Accumulator x independent of the Profile used for the transaction, including the Accumulator Currency Code, which transactions are accumulated, and the length of the counter cycle.	'BF3A'	'DF0x'
Cyclic Accumulator x Limit	The limit for the value of Cyclic Accumulator x. A bit is set in the ADR when the Cyclic Accumulator x value exceeds this limit.	—	—
Cyclic Accumulator x Reference Date	For a daily or monthly cyclic accumulator, represents the Transaction Date (YYMMDD) of the last transaction that reset Cyclic Accumulator x.	'BF42'	'DF1x'
Cyclic Accumulator x Reference Day	The Reference Day is the representation in Days (see Annex E) for the day at the beginning of the week in which Cyclic Accumulator x was last reset.	'BF42'	'DF2x'

Table 15-1: First Card Action Analysis – Card Data, continued

Data	Description	Template	Tag
First GEN AC Log Data Table	Provides information necessary to support optional extension of the data to be logged for a transaction. Identifies data that might change during processing of the transaction. In order to log the value of this data in the second GENERATE AC command processing, these data elements must be extracted from Second GENERATE AC Command Data using Second GEN AC Log Data Table.	'BF40'	'DF01'
First GEN AC Unchanging Log Data Table	Provides information necessary to support optional extension of the data to be logged for a transaction. Identifies data that will not change during a transaction and thus does not need to be requested again from the terminal in the Second GENERATE AC Command Data. This data is saved in the application from the First GENERATE AC Command Data in case it is to be logged during processing of the second GENERATE AC command.	'BF40'	'DF03'
Issuer Application Data (IAD)	Informs the issuer about the application. Used to send the CVR and other application information to the issuer.	—	'9F10'
Issuer Country Code	Indicates the country of the issuer.	—	'5F28'
Issuer Options Profile Control x	Defines the profile-specific behaviour selected by the issuer for processing transactions with Issuer Options Profile Control ID = 'x'.	'BF3B'	'DF0x'
	First GENERATE AC Command Data Length	Expected value for the length of the First GENERATE AC Command Data.	
Issuer Script Command Counter	An internal application counter that indicates the number of Issuer Script commands successfully processed.	—	—

Table 15-1: First Card Action Analysis – Card Data, continued

Data	Description	Template	Tag
Last Online Transaction Date	The Transaction Date of the last transaction that successfully went online.	—	—
Last Online Transaction Date in Days	The Transaction Date in Days of the last transaction that successfully went online.	—	—
Limits Entry x	An accumulator limit used as the limit for either the Maximum Transaction Amount or for one of the Cyclic Accumulators in the application.	'BF3C'	'DF0x'
Maximum Transaction Amount (MTA)	A limit on the value of Amount, Authorised for the transaction.	—	—
MTA Profile Control	Defines the profile-specific behaviour for the MTA card risk management check when processing a transaction in the Profile selected for the transaction, including which currency conversion table and limit entry to use.	'BF3D'	'DF0x'
Number of Days Offline Limit	Limits the number of days the card may perform transactions offline before attempting to go online.	—	'C3'
PIN Try Counter	Number of PIN tries remaining.	—	'9F17'
Previous Transaction History (PTH)	Indicators used to store information about previous transactions that is used in Card Risk Management for subsequent transactions.	—	'C7'

Table 15-1: First Card Action Analysis – Card Data, continued

Data	Description	Template	Tag
Profile Control x	Identifies which application resources are to be used when processing a transaction with Profile ID x. The following contents of Profile Control x are used in Card Action Analysis:	'BF3F'	'DFxx'
	Accumulator Profile Control ID for Accumulator x		
	Identifies the Accumulator Profile Control that defines the behaviour for Accumulator x when processing a transaction in the Profile.		
	CIACs ID		
	Identifies the CIACs Entry x containing the CIAC – Decline, CIAC – Online, and CIAC – Default to be used when processing the transaction.		
	Counter Profile Control ID for Counter x		
	Identifies the Counter Profile Control that defines the behaviour for Counter x when processing a transaction in the Profile.		
	Cyclic Accumulator Profile Control ID for Cyclic Accumulator X		
	Identifies the Cyclic Accumulator Profile Control that defines the behaviour for Cyclic Accumulator x when processing a transaction in the Profile.		
	Issuer Options Profile Control ID		
	Identifies the Issuer Options Profile Control to be used when processing the transaction.		
	MTA Profile Control ID		
	Identifies the MTA Profile Control that defines the behaviour for the Card Risk Management Maximum Transaction Amount Check when processing a transaction in the Profile.		
	VLP Profile Control ID		
	Identifies the Accumulator Profile Control to be used as the VLP Profile Control that defines the behaviour for the VLP Available Funds when processing a transaction in the Profile.		

Table 15-1: First Card Action Analysis – Card Data, continued

Data	Description	Template	Tag
Profile ID	Identifies the Profile Control to use in configuring the application behaviour for the transaction environment.	—	—
Signed Dynamic Application Data	The signature generated by the card at transaction time as part of the response to a GENERATE AC command with CDA processing. The card generates this signature using a hash of dynamic data from the terminal and card. The card signs the Signed Dynamic Application Data with the ICC Private Key. The format of the Signed Dynamic Application Data is shown in <i>EMV Book 2</i> , Table 16.	—	'9F4B'
Source Currency Code	The currency code of the currency from which amounts may be converted using a single Currency Conversion Parameter in the Currency Conversion Table. A Currency Conversion Parameter may only be used if the Transaction Currency Code matches the Source Currency Code in the Currency Conversion Parameter.	—	—
Target Currency Code	The currency code of the currency to which amounts may be converted using a Currency Conversion Table. A Currency Conversion Table may only be used to convert an amount if the Target Currency Code matches the associated Accumulator Currency Code or MTA Currency Code.	—	—
Transaction Date in Days	Represents the Transaction Date as the number of days since a base date (see Annex E).	—	—
VLP Available Funds	The amount available to spend offline for the VLP Accumulator.	—	'9F79'

Table 15-1: First Card Action Analysis – Card Data, continued

15.4 Terminal Data

The terminal data listed in Table 15-2 are used in First Card Action Analysis for Card Risk Management and generation of the response to the GENERATE AC command. They are passed to the card in the first GENERATE AC command data if their tag and length were included in the CDOL1 from the card.

For a detailed description of these data and their usage, see Annex L: Data Dictionary.

Data	Description	Template	Tag
Amount, Authorised	The amount of the transaction.	—	'9F02'
Amount, Other	Cashback amount for the transaction.	—	'9F03'
CVM Results	Indicates the results of the last CVM performed.	—	'9F34'
Terminal Country Code	Terminal data indicating the country of the terminal.	—	'9F1A'
Terminal Type	Indicates the type of the terminal (used to determine whether the terminal is online-capable).	—	'9F35'
Terminal Verification Results (TVR)	A series of indicators used to record the results of offline processing from a terminal perspective including the results of all terminal risk management checks.	—	'95'
Transaction Currency Code	A code that indicates the currency of the transaction.	—	'5F2A'
Transaction Date	Local date that the transaction was authorised.	—	'9A'
Transaction Type	The type of financial transaction requested.	—	'9C'
Unpredictable Number	A number used to provide variability and uniqueness to data that is sent by the application.	—	'9F37'

Table 15-2: First Card Action Analysis – Terminal Data

15.5 First GENERATE AC Command

The GENERATE AC (GENERATE APPLICATION CRYPTOGRAM) command is used by the terminal to request that the card provide an application cryptogram and the card's decision on further action.

Figure 15-1 shows an overview of First Card Action Analysis processing and indicates the section that describes each step of processing.

(Profile IDs '7D' and '7E', mentioned in Figure 15-1, are discussed in section 15.5.2.)

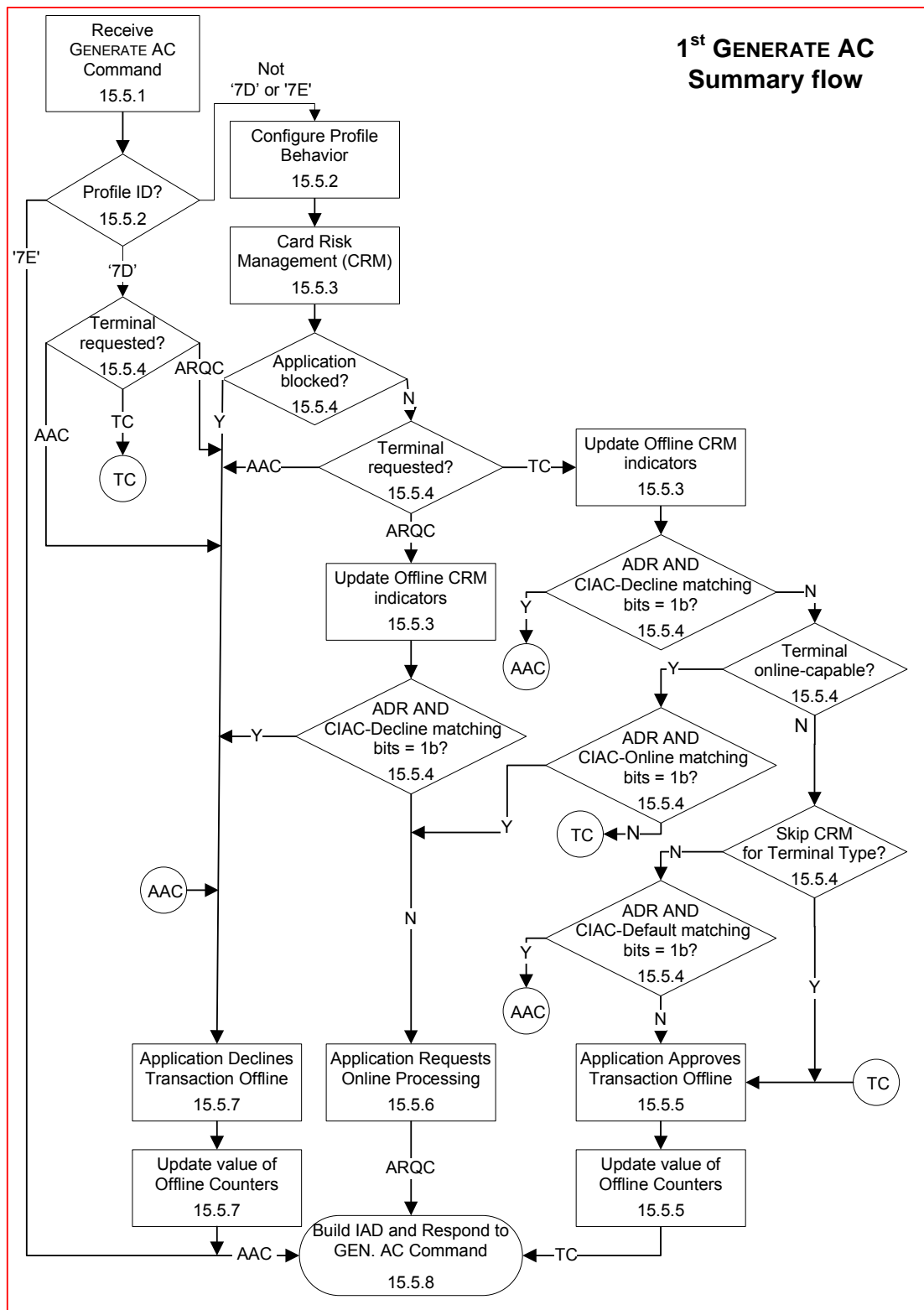


Figure 15-1: First Card Action Analysis Processing Flow

15.5.1 Command Coding

The P1 parameter of the GENERATE AC command indicates the type of cryptogram the terminal is requesting and whether the transaction is eligible for Combined DDA/AC Generation (CDA). Table 15-4 shows the format of P1. The data portion of the command contains the data requested in the CDOL1.

Code	Value
CLA	'80'
INS	'AE'
P1	Reference Control Parameter
P2	'00'
Lc	Var.
Data	First GENERATE AC Command Data
Le	'00'

Table 15-3: First GENERATE AC Command Message

NOTE: First GENERATE AC Command Data is called Transaction-related data in EMV Book 3.

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
x	x							Cryptogram Type
0	0							AAC
0	1							TC
1	0							ARQC
1	1							RFU
		x						RFU
			x					CDA Requested
			0					CDA Not Requested
			1					CDA Requested
				x	x	x	x	RFU

Table 15-4: Reference Control Parameter Coding for First GENERATE AC

NOTE: CDA Requested and CDA Not Requested are respectively called CDA signature requested and CDA signature not requested in EMV Book 3.

Combined DDA/AC Generation is only supported by the Dynamic-RSA implementer-option.

Req 15.1 (Interpretation of first GENERATE AC command data):

First GENERATE AC Command Data shall be interpreted by the application as consisting of the data elements listed in Table 15-5, in the order shown.

Data Element	Length
<i>Amount, Authorised</i>	<i>6</i>
<i>Amount, Other</i>	<i>6</i>
<i>Terminal Country Code</i>	<i>2</i>
<i>TVR</i>	<i>5</i>
<i>Transaction Currency Code</i>	<i>2</i>
<i>Transaction Date</i>	<i>3</i>
<i>Transaction Type</i>	<i>1</i>
<i>Unpredictable Number</i>	<i>4</i>
<i>Terminal Type</i>	<i>1</i>
<i>CVM Results</i>	<i>3</i>
<i>First GENERATE AC Extension Data</i>	<i>var.</i>

Table 15-5: First GENERATE AC Command Data

The application does not use the First GENERATE AC Extension Data as an individual element, but always as part of the First GENERATE AC Command Data. The First GENERATE AC Extension Data might be used for transaction logging or for the Additional Check Table x Check.

Req 15.2 (Length of First GENERATE AC Extension Data supported):

At a minimum, the application shall support First GENERATE AC Extension Data length of up to 32 bytes.

15.5.1.1 Command Format Validation**Req 15.3 (Check P1 value for first GENERATE AC command):**

If the 'Cryptogram Type' bits in the P1 parameter have the value 11b, then the card shall discontinue processing the command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6A86' (Incorrect Parameters, P1-P2).

Req 15.4 (Check P2 value for first GENERATE AC command):

If the P2 parameter is not set to the value '00', then the card shall discontinue processing the command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6A86' (Incorrect Parameters, P1-P2).

Req 15.5 (Select Issuer Options Profile Control for the transaction):

The Issuer Options Profile Control used in processing the transaction shall be Issuer Options Profile Control x, where x is the Issuer Options Profile Control ID in the Profile Control for the transaction.

Req 15.6 (Check first GENERATE AC command data length):

If the value of Lc does not equal the value of First GENERATE AC Command Data Length parameter in the Issuer Options Profile Control for the transaction, then the card shall discontinue processing the command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6700' (Wrong Length).

Req 15.7 (Check command data length at least meets the minimum length allowed):

If the value of Lc is less than 33 (the minimum length of CDOL related data), then the card shall discontinue processing the command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6700' (Wrong Length).

15.5.2 Profile Behaviour

The Profile ID selected during Initiate Application Processing identifies the Profile Control to be used to configure the application behaviour for Card Action Analysis. For further explanation, see Annex H.

NOTE: Profile ID '7E' identifies a special non-payment transaction which requires a special format for the Issuer Application Data sent in the response to the GENERATE AC command. The card always declines transactions processed with this profile ID, and does not perform Card Risk Management.

NOTE: Profile ID '7D' identifies the VLP profile, a special profile that if supported in the card and requested by the terminal, may be chosen to process the transaction. The purpose of the VLP profile is to have a fast offline transaction for very low value transactions. No Card Risk Management is performed in this profile. If the terminal requests an offline approval, the card will approve the transaction. If the terminal requests an offline decline or online request, the card will decline the transaction.

Req 15.8 (Configure Profile behaviour for special profiles):

If the Profile ID has the value '7E', then the application shall:

- *perform no card risk management checks*
- *set the Cryptogram Information Data (CID) to the value '00' to indicate that an AAC is requested and no advice is required.*
- *continue processing by responding to the GENERATE AC Command as described in section 15.5.8.*

If the Profile ID has the value '7D', then the application shall:

- *perform no card risk management checks*
- *configure the Accumulator Profile Control for each Accumulator x as described in Req 15.10*
- *configure the Counter Profile Control for each Counter x as described in Req 15.13*
- *configure the VLP Profile Control as described in Req 15.21*
- *continue processing with determining the response Application Cryptogram type, as described in section 15.5.4*

Otherwise, the application shall perform Card Risk Management as configured in the Profile Control selected for the transaction.

Req 15.9 (Select CIACs Entry for the transaction):

The CIACs Entry used in processing the transaction shall be CIACs Entry y, where y is the CIACs ID in the Profile Control for the transaction.

Req 15.10 (Configure accumulator behaviour controls for the transaction):

For each Accumulator x; let y represent the value of Accumulator Profile Control ID for Accumulator x in the Profile Control for the transaction:

- *if y has the value 'F', then the accumulator is not active for the transaction.*
- *otherwise:*
 - *Accumulator x shall be active for the transaction,*
 - ***and** the Accumulator Profile Control for Accumulator x used in processing the transaction shall be Accumulator Profile Control y.*

Req 15.11 (Configure accumulator limits for the transaction):

For each Accumulator x that is active for the transaction; the Accumulator x Limit Set used in processing the transaction shall be the Accumulator x Limit Set z (Lower Limit z and Upper Limit z), where z is determined by the Limit Set ID in the Accumulator Profile Control for Accumulator x for the transaction.

Req 15.12 (Configure currency conversion for accumulators):

*For each Accumulator x that is active for the transaction, if **either** of the following is true:*

- *the Currency Conversion Table ID in the Accumulator Profile Control for Accumulator x has the value 'F',*
- ***or** the Accumulator Currency Code in Accumulator x Control does not match the Target Currency Code in Currency Conversion Table w, where w is the value of the Currency Conversion Table ID in the Accumulator Profile Control for Accumulator x for the transaction:*

then no Currency Conversion Table shall be active for Accumulator x.

Otherwise the Currency Conversion Table used in processing the transaction for Accumulator x shall be Currency Conversion Table w, where w is the value of the Currency Conversion Table ID in the Accumulator Profile Control for Accumulator x for the transaction.

Req 15.13 (Configure counter behaviour controls for the transaction):

For each Counter x; let y represent the value of Counter Profile Control ID for Counter x in the Profile Control for the transaction:

- *if y has the value 'F', then Counter x is not active for the transaction.*
- *otherwise:*
 - *Counter x shall be active for the transaction*
 - ***and** the Counter Profile Control for Counter x used in processing the transaction shall be Counter Profile Control y.*

Req 15.14 (Configure counter limits for the transaction):

For each Counter x that is active for the transaction; the Counter x Limit Set used in processing the transaction shall be the Counter x Limit Set z (Lower Limit z and Upper Limit z), where z is determined by the Limit Set ID in the Counter Profile Control for Counter x for the transaction.

Req 15.15 (Configure cyclic accumulator behaviour controls for the transaction):

For each Cyclic Accumulator x ; let y represent the value of Cyclic Accumulator Profile Control ID for Cyclic Accumulator x in the Profile Control for the transaction:

- if y has the value 'F', then the accumulator is not active for the transaction.
- otherwise:
 - the Cyclic Accumulator x shall be active for the transaction
 - **and** the Cyclic Accumulator Profile Control for Cyclic Accumulator x used in processing the transaction shall be Cyclic Accumulator Profile Control y .

Req 15.16 (Configure cyclic accumulator limits for the transaction):

For each Cyclic Accumulator x that is active for the transaction; the Cyclic Accumulator x Limit used in processing the transaction shall be Limits Entry z , where z is the value of the Limit Entry ID in the Cyclic Accumulator Profile Control for Cyclic Accumulator x for the transaction.

Req 15.17 (Configure currency conversion for cyclic accumulators):

For each Cyclic Accumulator x that is active for the transaction, if **either** of the following is true:

- the Currency Conversion Table ID in the Cyclic Accumulator Profile Control for Cyclic Accumulator x has the value 'F',
- **or** the Accumulator Currency Code in Cyclic Accumulator x Control does not match the Target Currency Code in Currency Conversion Table w , where w is the value of the Currency Conversion Table ID in the Cyclic Accumulator Profile Control for Cyclic Accumulator x for the transaction,

then no Currency Conversion Table shall be active for Cyclic Accumulator x .

Otherwise the Currency Conversion Table used in processing the transaction for Cyclic Accumulator x shall be Currency Conversion Table w , where w is the value of the Currency Conversion Table ID in the Cyclic Accumulator Profile Control for Cyclic Accumulator x for the transaction.

Req 15.18 (Configure MTA controls for the transaction):

Let x represent the value of the Maximum Transaction Amount (MTA) Profile Control ID in the Profile Control for the transaction:

- if x has the value 'F', then the MTA Check is not active for the transaction.
- otherwise:
 - the MTA Check shall be active for the transaction
 - **and** the MTA Profile Control used in processing the transaction shall be MTA Profile Control x .

Req 15.19 (Configure MTA for the transaction):

If the MTA Check is active for the transaction, then the MTA used in processing the transaction shall be Limits Entry y , where y is the value of the Limits Entry ID in the MTA Profile Control for the transaction.

Req 15.20 (Configure currency conversion for MTA):

*If the MTA Check is active for the transaction, then if **either** of the following is true:*

- *the Currency Conversion Table ID in the MTA Profile Control has the value 'F',*
- ***or** the MTA Currency Code in the MTA Profile Control does not match the Target Currency Code in Currency Conversion Table y, where y is the value of the Currency Conversion Table ID in the MTA Profile Control for the transaction*

then no Currency Conversion Table shall be active for the MTA Check.

Otherwise the MTA Currency Conversion Table used in processing the transaction shall be Currency Conversion Table y, where y is the value of the Currency Conversion Table ID in the MTA Profile Control for the transaction.

Req 15.21 (Configure VLP Profile Control for the transaction):

If VLP is supported and the VLP Profile Control ID in the Profile Control for the transaction has the value 'F', then VLP Available Funds shall not be active for the transaction.

If VLP is supported and the VLP Profile Control ID in the Profile Control for the transaction has a value other than 'F', then:

- *VLP Available Funds shall be active for the transaction,*
- ***and** the VLP Profile Control used in processing the transaction shall be Accumulator Profile Control x, where x is the value of the VLP Profile Control ID in the Profile Control for the transaction.*

NOTE: Only the 'Reset Accumulator with Online Response' and 'Send Accumulator in IAD' bits are used to control the VLP Available Funds. All other bits in the VLP Profile Control are RFU.

15.5.3 Card Risk Management Processing

The application does not perform any card risk management for Profile '7D' (VLP Profile) because the card will approve or decline the transaction offline based only on whether the terminal requests an offline approval. The application does not perform any card risk management for Profile '7E' (Authentication Token Profile) because the card will decline the transaction offline. For all other Profiles, the card performs each mandatory Card Risk Management check for every transaction. The application must support the optional checks. It is an issuer-option whether each check is performed for transactions processed using the Profile.

The application performs each of the following Card Risk Management checks that is active (either mandatory or activated for the Profile) to see whether the condition has occurred, then proceeds to the next active check. The application may perform these Card Risk Management checks in any order.

Req 15.22 (Determine which card risk management checks to perform):

If the Profile ID has a value other than '7D' or '7E', then all Card Risk Management checks that are either:

- *mandatory, or*
- *optional and active for the transaction*

shall be completed before the application determines the type of Application Cryptogram to be sent in the GENERATE AC response.

Table 15-6 summarises the Card Risk Management checks provided and describes the result if the condition being checked for occurs. The section that describes the check is also indicated.

Risk Management Check	Result (if condition occurs)	See section:
Accumulator x Lower Limit Exceeded Check	<p>If Accumulator x is active in the Profile:</p> <ul style="list-style-type: none"> • Sets a CVR bit to indicate that a Lower Cumulative Offline Amount Limit either has already been exceeded, or (optionally) will be exceeded if this transaction is approved offline. • Sets an ADR bit to indicate that the Accumulator x Lower Limit either has already been exceeded, or (optionally) will be exceeded if this transaction is approved offline. 	15.5.3.15
Accumulator x Upper Limit Exceeded Check	<p>If Accumulator x is active in the Profile:</p> <ul style="list-style-type: none"> • Sets a CVR bit to indicate that an Upper Cumulative Offline Amount Limit either has already been exceeded, or (optionally) will be exceeded if this transaction is approved offline. • Sets an ADR bit to indicate that the Accumulator x Upper Limit either has already been exceeded, or (optionally) will be exceeded if this transaction is approved offline. 	15.5.3.16
Additional Check Table x Check	<p>If an Additional Check Table x is active in the Profile:</p> <ul style="list-style-type: none"> • Sets a CVR bit to indicate that a match was found with data sent by the terminal in the GENERATE AC command. • Sets one of two ADR bits, one if a match was found with data sent by the terminal in the GENERATE AC command, the other one if no match was found. The use of two different bits allows action to be taken either if a match was found or if a match was not found. 	0

Table 15-6: Card Risk Management Checks

Risk Management Check	Result (if condition occurs)	See section:
Counter x Lower Limit Exceeded Check	<p>If Counter x is active in the Profile:</p> <ul style="list-style-type: none"> • Sets a CVR bit to indicate that a Lower Offline Transactions Count Limit either has already been exceeded, or (optionally) will be exceeded if this transaction is approved offline. • Sets an ADR bit to indicate that the Counter x Lower Limit either has already been exceeded, or (optionally) will be exceeded if this transaction is approved offline. 	15.5.3.17
Counter x Upper Limit Exceeded Check	<p>If Counter x is active in the Profile:</p> <ul style="list-style-type: none"> • Sets a CVR bit to indicate that an Upper Offline Transactions Count Limit either has already been exceeded, or (optionally) will be exceeded if this transaction is approved offline. • Sets an ADR bit to indicate that the Counter x Upper Limit either has already been exceeded, or (optionally) will be exceeded if this transaction is approved offline. 	15.5.3.18
Cyclic Accumulator x Limit Exceeded Check	<p>If Cyclic Accumulator x is active in the Profile, sets an ADR bit to indicate that a limit for the cumulative amount of approved transactions within a specified cycle (such as a day, week, or month) has been exceeded.</p> <p>May set an ADR bit and a CVR bit to indicate an error in the Transaction Date.</p>	15.5.3.22
Go Online on Next Transaction Check	Sets both a CVR bit and an ADR bit to indicate that the issuer wants this transaction to go online.	15.5.3.9
Issuer Authentication Failed Check	Sets both a CVR bit and an ADR bit to indicate that Issuer Authentication failed on a previous online transaction.	15.5.3.10

Table 15-6: Card Risk Management Checks, continued

Risk Management Check	Result (if condition occurs)	See section:
Issuer Authentication Not Performed Check	Sets a CVR bit to indicate that Issuer Authentication was not performed on a previous online transaction. Sets an ADR bit to indicate that Issuer Authentication Data was not received on a previous online transaction. Sets an ADR bit to indicate that the transaction was unable to go online on a previous transaction that attempted to go online.	15.5.3.14
Issuer Script Processing Failed Check	Sets both a CVR bit and an ADR bit to indicate that script command processing failed on a previous online transaction.	15.5.3.12
Issuer Script Received Check	Sets an ADR bit to indicate that an Issuer Script Command was received on a previous online transaction.	15.5.3.11
Last Online Transaction Not Completed Check	Sets both a CVR bit and an ADR bit to indicate that a previous transaction that attempted to go online did not complete successfully.	15.5.3.13
Maximum Number of Days Offline Check	If the Maximum Number of Days Offline Check is active in the Profile, sets an ADR bit to indicate that the card has exceeded the limit on the number of days since the last transaction that successfully went online. May set an ADR bit and a CVR bit to indicate an error in the Transaction Date.	15.5.3.20
Maximum Transaction Amount (MTA) Check	If the MTA is active in the Profile, sets an ADR bit to indicate that the value of the transaction exceeds the limit.	15.5.3.21
Number of Issuer Script Commands Check	Sets CVR bits to indicate the value of Issuer Script Command Counter.	15.5.3.8
Offline Data Authentication Failed on Previous Transaction Check	Sets both a CVR bit and an ADR bit to indicate that SDA, DDA, or CDA failed during processing of a previous transaction.	15.5.3.7

Table 15-6: Card Risk Management Checks, continued

Risk Management Check	Result (if condition occurs)	See section:
Offline PIN Verification Performed and PIN Not Successfully Verified Check	Sets both a CVR bit and an ADR bit to indicate that offline PIN verification has been performed on this transaction and the PIN was not successfully verified.	15.5.3.3
Offline PIN Verification Performed Check	Sets both a CVR bit and an ADR bit to indicate that offline PIN verification has been performed on this transaction.	15.5.3.2
PIN Try Counter Check	Sets CVR bits to the current value of the PIN Try Counter.	15.5.3.6
PIN Try Limit Exceeded Check	Sets both a CVR bit and an ADR bit to indicate that the PIN Try limit has been exceeded on this or a previous transaction.	15.5.3.5
Terminal Erroneously Considers Offline PIN OK Check	Sets an ADR bit if the CVM Results indicates that the PIN was verified and the card did not check the PIN or did not successfully verify the PIN.	15.5.3.4

Table 15-6: Card Risk Management Checks, continued

15.5.3.1 Additional Check Table x Check

The Additional Check Table x provides the issuer with a flexible mechanism to add a check to the specified Card Risk Management checks. It is possible to apply the check to the value of any data element that can be requested from the terminal. This issuer-optional check notifies the issuer whether there were any matches in the Additional Check Table x. This enables the application to consider this information when deciding whether to approve or decline the transaction offline, or to send the transaction online.

Req 15.23 (Number of Additional Check Tables supported):

At a minimum, the application shall provide support for two Additional Check Tables: Additional Check Table 1 and Additional Check Table 2.

Req 15.24 (Size supported for Additional Check Tables):

At a minimum, the application shall provide support for up to 18 bytes for each Additional Check Table.

A bit in Issuer Options Profile Control is used to activate this check for each Additional Check Table x.

Req 15.25 (Check whether to perform Additional Check Table check):

For each value of x for which the 'Activate Additional Check Table x' bit in the Issuer Options Profile Control has the value 1b, the application shall perform this check.

Processing of Additional Check Table x is described in Annex B.

Req 15.26 (Set Additional Check Table bits):

If the masked value derived from First GENERATE AC Command Data using Additional Check Table x matches any of the Values in Additional Check Table x, then the application shall:

- *set the 'Match Found in Additional Check Table x' bit in the ADR to 1b.*
- *set the 'Match Found in Additional Check Table' bit in the CVR to 1b.*

If the masked value derived from First GENERATE AC Command Data using Additional Check Table x does not match any of the Values in Additional Check Table x, then the application shall:

- *set the 'No Match Found in Additional Check Table x' bit in the ADR to 1b.*

15.5.3.2 Offline PIN Verification Performed Check

This mandatory check provides the issuer notification of whether Offline PIN verification has been performed on this transaction. It also enables the application to consider this information when deciding whether to approve or decline the transaction offline, or to send the transaction online.

Req 15.27 (Set PIN verification performed bits):

If the 'Offline PIN Verification Performed' bit in the CVR has the value 0b, then the application shall set the 'Offline PIN Verification not Performed' bit in the ADR to 1b.

NOTE: Indication of Offline PIN Verification status may be implemented using internal PIN Verification Status instead of the CVR bit. The external application behaviour shall be the same as if the CVR bit were used as the indicator.

15.5.3.3 Offline PIN Verification Performed and PIN Not Successfully Verified Check

This mandatory check provides the issuer notification of whether Offline PIN verification has been performed and failed on this transaction. It also enables the application to consider this information when deciding whether to approve or decline the transaction offline, or to send the transaction online.

Req 15.28 (Set PIN verification failed bits):

If the 'Offline PIN Verification Performed and PIN Not Successfully Verified' bit in the CVR has the value 1b, then the application shall set the 'Offline PIN Verification Failed' bit in the ADR to 1b.

NOTE: Indication of Offline PIN Verification status may be implemented using internal PIN Verification Status instead of the CVR bit. The external application behaviour shall be the same as if the CVR bit were used as the indicator.

15.5.3.4 Terminal Erroneously Considers Offline PIN OK Check⁷

This mandatory check determines whether the terminal considers (in CVM Results) that Offline PIN processing passed, when the card reported Offline PIN processing as having failed. This check enables the application to consider this information when deciding whether to approve or decline the transaction offline, or to send the transaction online.

Req 15.29 (Set terminal erroneously considers PIN OK bit):

*If **all** of the following are true:*

- *bits b6-b1 of the CV Rule byte in the CVM Results has **any** of the following values:*
 - *Plaintext PIN verification performed by ICC (000001b)*
 - *Plaintext PIN verification performed by ICC and signature (000011b)*
 - *Enciphered PIN verification performed by ICC (000100b)*
 - *Enciphered PIN verification performed by ICC and signature (000101b)*
- ***and** the CVM Result in the CVM Results sent to the card in the GENERATE AC command data indicates Successful CVM ('02'),*
- ***and either** of the following is true:*
 - *the 'Offline PIN Verification Performed' bit in the CVR has the value 0b*
 - ***or** the 'Offline PIN Verification Performed and PIN Not Successfully Verified' bit in the CVR has the value 1b*

then the application shall:

- *set the 'Terminal Erroneously Considers Offline PIN OK' bit in the ADR to the value 1b.*

15.5.3.5 PIN Try Limit Exceeded Check

This mandatory check provides the issuer notification that the PIN Try Limit has been exceeded (in either the current or a preceding transaction). It also enables the application to consider this information when deciding whether to approve or decline the transaction offline, or to send the transaction online.

Req 15.30 (Set PIN Try Limit exceeded bits):

If the value of the PIN Try Counter is zero, then the application shall:

- *set the 'PIN Try Limit Exceeded' bit in the ADR to 1b.*
- *set the 'PIN Try Limit Exceeded' bit in the CVR to 1b.*

⁷ This is to validate that the card and terminal agree on the CVM results.

15.5.3.6 PIN Try Counter Check

This mandatory check provides the issuer notification of the number of PIN tries remaining in the PIN Try Counter.

The application sets the 'Low Order Nibble of PIN Try Counter' bits in the CVR to the value of the low-order nibble of the PIN Try Counter, using identical bit settings.

15.5.3.7 Offline Data Authentication Failed on Previous Transaction Check

This mandatory Card Risk Management check provides the issuer notification that offline data authentication failed during a previous transaction. It also enables the application to consider this information when deciding whether to approve or decline the transaction offline, or to send the transaction online.

Failure of offline data authentication in the previous transaction is shown by the 'Offline Data Authentication Failed on Previous Transaction' bit that the application sets to 1b in the Previous Transaction History (PTH) data element when the TVR returned during the previous transaction contained any of the following bits set to the value 1b:

- SDA Failed
- DDA Failed
- CDA Failed

NOTE: This indicator in PTH is set prior to issuing the GENERATE AC response. Once set, this indicator in PTH remains set until it is reset during either:

- First Card Action Analysis if the transaction is approved offline and the 'SDA Failed', 'DDA Failed', and 'CDA Failed' bits in the TVR are all set to 0b.
- Second Card Action Analysis if Issuer Authentication status and application parameter conditions are met and the 'SDA Failed', 'DDA Failed', and 'CDA Failed' bits in the TVR are all set to 0b.

Req 15.31 (Set offline data authentication failed bits):

If the 'Offline Data Authentication Failed on Previous Transaction' bit in the PTH data element has the value 1b, then the application shall:

- *set the 'Offline Data Authentication Failed on Previous Transaction' bit in the ADR to 1b.*
- *set the 'Offline Data Authentication Failed on Previous Transaction' bit in the CVR to 1b.*

15.5.3.8 Number of Issuer Script Commands Check

This mandatory check provides the issuer with the value of bits b4 through b1 in the Issuer Script Command Counter, which counts the Issuer Script commands successfully processed in previous online transactions.

Req 15.32 (Set CVR bits for number of script commands processed):

The application shall set the 'Number of Issuer Script Commands Containing Secure Messaging Processed' bits in the CVR to the value of bits b4 through b1 in the Issuer Script Command Counter using identical bit settings.

15.5.3.9 Go Online on Next Transaction Check

This mandatory check provides the issuer notification that either the card is a new card (one that has never been approved online) that is personalised to go online when new, or the last successfully-recovered CSU indicated that the next transaction should attempt to go online. It also enables the application to consider this information when deciding whether to approve or decline the transaction offline, or to send the transaction online.

An issuer can cause a new card to go online by setting the 'Go Online on Next Transaction' bit in the PTH data element to the value 1b. The condition that the last successfully-recovered CSU indicated the next transaction should attempt to go online is shown by the application setting the 'Go Online on Next Transaction' bit in the PTH to the value 1b.

NOTE: Once set, this indicator in PTH remains set until it is reset during Second Card Action Analysis based on Issuer Authentication status and application parameters.

Req 15.33 (Set go online on next transaction bits):

If the 'Go Online on Next Transaction' bit in the PTH has the value 1b, then the application shall:

- *set the 'Go Online on Next Transaction was Set' bit in the ADR to the value 1b.*
- *set the 'Go Online on Next Transaction was Set' bit in the CVR to the value 1b.*

15.5.3.10 Issuer Authentication Failed Check

This mandatory check provides the issuer notification that Issuer Authentication failed on the last online transaction. It also enables the application to consider this information when deciding whether to approve or decline the transaction offline, or to send the transaction online.

Issuer Authentication failure on the last online transaction is shown by the 'Issuer Authentication Failed' bit that the application sets to 1b in the PTH data element when Issuer Authentication fails during transaction processing.

NOTE: Once set, this indicator in PTH remains set until it is reset during Second Card Action Analysis based on Issuer Authentication status and application parameters.

Req 15.34 (Set Issuer Authentication failed bits):

If the 'Issuer Authentication Failed' bit in the PTH has the value 1b, then the application shall:

- *set the 'Issuer Authentication Failed' bit in the ADR to 1b.*
- *set the 'Issuer Authentication Failed' bit in the CVR to 1b.*

15.5.3.11 Issuer Script Received Check

This mandatory check determines whether an issuer script was received in a previous online transaction. This enables the application to consider this information when deciding whether to approve or decline the transaction offline, or to send the transaction online.

Receipt of an issuer script is shown by the 'Script Received' bit that the application sets to 1b in the PTH data element during processing of issuer script commands.

NOTE: Once set, this indicator in PTH remains set until it is reset during Second Card Action Analysis based on Issuer Authentication status and application parameters.

Req 15.35 (Set script received bit):

If the 'Script Received' bit in the PTH has the value 1b, then the application shall set the 'Script Received' bit in the ADR to 1b.

15.5.3.12 Issuer Script Processing Failed Check

This mandatory check provides the issuer with notification that issuer script processing failed in a previous online transaction. It also enables the application to consider this information when deciding whether to approve or decline the transaction offline, or to send the transaction online.

A failure in script processing is shown by the 'Script Failed' bit that the application sets to 1b in the PTH data element when a failure occurs during processing of issuer script commands.

NOTE: Once set, this indicator in PTH remains set until it is reset during Second Card Action Analysis based on Issuer Authentication status and application parameters.

Req 15.36 (Set script failed bits):

If the 'Script Failed' bit in the PTH has the value 1b, then the application shall:

- *set the 'Issuer Script Processing Failed' bit in the ADR to 1b.*
- *set the 'Issuer Script Processing Failed' bit in the CVR to 1b.*

15.5.3.13 Last Online Transaction Not Completed Check

This mandatory check determines whether during a previous transaction, the card was removed from the terminal after the card requested an online authorisation and prior to receipt of an online response or terminal processing for unable to go online. It provides the issuer notification that the last online transaction failed. It also enables the application to consider this information when deciding whether to approve or decline the transaction offline, or to send the transaction online.

Failure to complete the last online transaction is shown by the 'Last Online Transaction Not Completed' bit that the application sets to 1b in the PTH data element when an online authorisation is requested.

NOTE: Once set, this indicator in PTH remains set until it is reset during Second Card Action Analysis based on Issuer Authentication status and application parameters.

Req 15.37 (Set last online transaction not completed bits):

If the 'Last Online Transaction Not Completed' bit in the PTH data element has the value 1b, then the application shall:

- *set the 'Last Online Transaction Not Completed' bit in the ADR to 1b.*
- *set the 'Last Online Transaction Not Completed' bit in the CVR to 1b.*

15.5.3.14 Issuer Authentication Not Performed Check

This mandatory check provides the issuer notification that Issuer Authentication was not performed on the last online transaction; that is, either the transaction was unable to go online or Issuer Authentication Data was not received in the response message.

The 'Unable to Go Online' bit in the ADR enables the application to consider this information when deciding whether to approve or decline the transaction offline, or to send the transaction online.

The condition that transaction was unable to go online on the last online transaction is shown by the 'Unable to Go Online' bit that the application sets to 1b in the PTH data element when the application receives a Second Card Action Analysis indicating that the terminal was unable to go online on a previous transaction that attempted to go online.

The 'Issuer Authentication Data not Received in Previous Online Transaction' bit in the ADR enables the application to consider this information when deciding whether to approve or decline the transaction offline, or to send the transaction online.

The condition that Issuer Authentication Data was not received on the last online transaction is shown by the 'Issuer Authentication Data not Received in Online Response' bit that the application sets to 1b in the PTH data element when Issuer Authentication Data is not received during Second Card Action Analysis of a previous transaction but the transaction went online.

NOTE: Once set, these indicators in PTH remain set until they are reset during Second Card Action Analysis based on Issuer Authentication status and application parameters.

Req 15.38 (Set Issuer Authentication data not received bits):

If the 'Issuer Authentication Data Not Received in Online Response' bit in the PTH has the value 1b, then the application shall:

- *set the 'Issuer Authentication Data not Received in Previous Online Transaction' bit in the ADR to 1b.*
- *set the 'Issuer Authentication Not Performed' bit in the CVR to 1b.*

Req 15.39 (Set transaction unable to go online bits):

If the 'Unable to Go Online' bit in the PTH has the value 1b, then the application shall:

- *set the 'Unable to Go Online' bit in the ADR to 1b.*
- *set the 'Issuer Authentication Not Performed' bit in the CVR to 1b.*

15.5.3.15 Accumulator x Lower Limit Exceeded Check

This issuer-optional check provides the issuer notification that the lower limit for a cumulative amount of offline transactions has been exceeded. It also enables the application to consider this information when deciding whether to approve or decline the transaction offline, or to send the transaction online.

This check is performed for each Accumulator x that is active for the transaction. If active, Accumulator x may accumulate transactions approved offline where the transaction currency is **either**:

- the accumulator currency
- any currency whose transaction value can be approximated in the accumulator currency using the Currency Conversion Table for Accumulator x if a Currency Conversion Table is active for Accumulator x in the Profile.

NOTE: If the Currency Conversion function is used, the amount converted to the accumulator currency is an approximation based on the conversion rate in the Currency Conversion Parameter.

NOTE: If the active Currency Conversion Table for Accumulator x in the Profile does not contain conversion rates for any additional currencies, then the accumulator contains the amount of offline transactions conducted in the accumulator currency.

The 'Include offline approvals' bit in the Accumulator x Control controls whether the application may add the value of an offline approved transaction to Accumulator x.

If the terminal requests an AAC (offline decline), then the transaction will be declined regardless of the setting of any ADR bits. Thus, the application only needs to perform the card risk management checks that might set CVR bits. The processing that sets the CVR bit when the Accumulator x Lower Limit has been exceeded (for a terminal requesting an AAC) is described in section 15.5.7.

If the terminal requests a TC (offline approval) or an ARQC (go online for authorisation) and offline approved transactions are not accumulated in Accumulator x in the Profile; then if the transaction is to be approved offline, the value of the current transaction is not included when performing this check. The check tests whether the amount from previous transactions has exceeded the Accumulator x Lower Limit.

The current transaction could be accumulated if all of the following are true:

- the terminal requests an ARQC or a TC
- accumulation is allowed
- offline approvals are included in Accumulator x for the Profile
- the transaction either is in the accumulator currency or can be converted into the accumulator currency

If the terminal requests an ARQC (go online for authorisation) and the current transaction could be accumulated, then it is an issuer option (using the 'Include ARQC in CRM Test' bit ⁸ in Accumulator x Control) whether to include the value of the current transaction when determining whether the Accumulator x Lower Limit would be exceeded.

If the terminal requests a TC (offline approval) and the current transaction could be accumulated, then the value of the current transaction is included when determining whether the Accumulator x Lower Limit would be exceeded.

Req 15.40 (Check if accumulator lower limit was previously exceeded):

For each Accumulator x that is active for the transaction:

- *if **both** of the following are true:*
 - *the terminal requested a TC or an ARQC in the first GENERATE AC command*
 - ***and** the value of Accumulator x is greater than the Accumulator x Lower Limit*
- then the application shall:*
- *set the 'Accumulator x Lower Limit Exceeded' bit in the ADR to 1b*
 - *set the 'Lower Cumulative Offline Amount Limit Exceeded' bit in the CVR to 1b*

⁸ If the terminal requests an ARQC, some issuer systems expect the value of the current transaction to be included when performing this card risk management check, while others do not. This option allows issuers to configure the Accumulator x Lower Limit Exceeded Check to function in a manner consistent with previous implementations.

Req 15.41 (Check if accumulator lower limit is exceeded by this transaction):

For each Accumulator x that is active for the transaction:

- *if **all** of the following are true:*
 - *the 'Allow Accumulation' bit in the Accumulator Profile Control for Accumulator x in this profile has the value 1b*
 - *the 'Include offline approvals' bit in the Accumulator x Control has the value 1b,*
 - ***and either** of the following is true:*
 - *the terminal requested a TC in the first GENERATE AC command*
 - ***or** the terminal requested an ARQC in the first GENERATE AC command, **and** the 'Include ARQC Transaction in CRM Test' bit in the Accumulator x Control has the value 1b*
 - ***and either** of the following is true:*
 - *the Transaction Currency Code matches the Accumulator Currency Code **and** the sum⁹ of the value of Accumulator x and Amount, Authorised is greater than the Accumulator x Lower Limit*
 - ***or** the Transaction Currency Code does not match the Accumulator Currency Code **and** the Transaction Currency Code matches the Source Currency Code in one of the Currency Conversion Parameters in the Currency Conversion Table active for Accumulator x¹⁰ **and** the sum¹¹ of the value of Accumulator x and the amount converted to the accumulator currency (using Amount, Authorised and the Currency Conversion Table active for Accumulator x) is greater than the Accumulator x Lower Limit*

then the application shall:

- *set the 'Accumulator x Lower Limit Exceeded' bit in the ADR to 1b*
- *set the 'Lower Cumulative Offline Amount Limit Exceeded' bit in the CVR to 1b*

⁹ If adding the Amount, Authorised to the value of Accumulator x would cause the accumulator to overflow, then instead compare the value '99 99 99 99 99 99' (rather than the sum of Amount, Authorised and the value of Accumulator x) to the Accumulator x Lower Limit.

¹⁰ If the Currency Conversion Table ID in the Accumulator Profile Control for Accumulator x for the transaction has the value 'F', then no Currency Conversion Table is active for Accumulator x.

¹¹ If adding the converted Amount, Authorised to the value of Accumulator x would cause the accumulator to overflow, then instead compare the value '99 99 99 99 99 99' (rather than the sum of the converted Amount, Authorised and the value of Accumulator x) to the Accumulator x Lower Limit.

15.5.3.16 Accumulator x Upper Limit Exceeded Check

This issuer-optional check provides the issuer notification that the upper limit for a cumulative amount of offline transactions has been exceeded. It also enables the application to consider this information when deciding whether to approve or decline the transaction offline, or to send the transaction online.

This check is performed for each Accumulator x that is active for the transaction. If active, Accumulator x accumulates transactions approved offline where the transaction currency is either:

- the accumulator currency
- any currency whose transaction value can be approximated in the accumulator currency using the Currency Conversion Table for Accumulator x if a Currency Conversion Table is active for Accumulator x in the Profile.

NOTE: If the Currency Conversion function is used, the amount converted to the accumulator currency is an approximation based on the conversion rate in the Currency Conversion Parameter.

NOTE: If the active Currency Conversion Table for Accumulator x in the Profile does not contain conversion rates for any additional currencies, then the accumulator contains the amount of offline transactions conducted in the accumulator currency.

The 'Include Offline Approvals' bit in the Accumulator x Control controls whether the application may add the value of an offline approved transaction to Accumulator x.

If the terminal requests an AAC (offline decline), then the transaction will be declined regardless of the setting of any ADR bits. Thus, the application only needs to perform the card risk management checks that might set CVR bits. The processing that sets the CVR bit when the Accumulator x Upper Limit has been exceeded (for a terminal requesting an AAC) is described in section 15.5.7.

If the terminal requests a TC (offline approval) or an ARQC (go online for authorisation), and offline approved transactions are not accumulated in Accumulator x in the Profile; then if the transaction is to be approved offline, the value of the current transaction is not included when performing this check. The check tests whether the amount from previous transactions has exceeded the Accumulator x Upper Limit.

The current transaction could be accumulated if all of the following are true:

- if the terminal requests an ARQC or a TC
- accumulation is allowed
- offline approvals are included in Accumulator x for the Profile
- the transaction either is in the accumulator currency or can be converted into the accumulator currency

If the terminal requests an ARQC (go online for authorisation), and the current transaction could be accumulated, then it is an issuer option (using the 'Include ARQC in CRM Test' bit ¹² in Accumulator x Control) whether to include the value of the current transaction when determining whether the Accumulator x Upper Limit would be exceeded.

If the terminal requests a TC (offline approval) and the current transaction could be accumulated, then the value of the current transaction is included when determining whether the Accumulator x Upper Limit would be exceeded.

Req 15.42 (Check if accumulator upper limit was previously exceeded):

For each Accumulator x that is active for the transaction:

- If **both** of the following are true:
 - the terminal requested a TC or an ARQC in the first GENERATE AC command
 - **and** the value of Accumulator x is greater than the Accumulator x Upper Limitthen the application shall:
 - set the 'Accumulator x Upper Limit Exceeded' bit in the ADR to 1b
 - set the 'Upper Cumulative Offline Amount Limit Exceeded' bit in the CVR to 1b

¹² If the terminal requests an ARQC, some issuer systems expect the value of the current transaction to be included when performing this card risk management check, while others do not. This option allows issuers to configure the Accumulator x Upper Limit Exceeded Check to function in a manner consistent with previous implementations.

Req 15.43 (Check if accumulator upper limit is exceeded by this transaction):

For each Accumulator x that is active for the transaction:

- *if **all** of the following are true:*
 - *the 'Allow Accumulation' bit in the Accumulator Profile Control for Accumulator x in this profile has the value 1b*
 - *the 'Include Offline Approvals' bit in the Accumulator x Control has the value 1b,*
 - ***and either** of the following is true:*
 - *the terminal requested a TC in the first GENERATE AC command*
 - ***or** the terminal requested an ARQC in the first Generate AC command **and** the 'Include ARQC Transaction in CRM Test' bit in the Accumulator x Control has the value 1b*
 - ***and either** of the following is true:*
 - *the Transaction Currency Code matches the Accumulator Currency Code **and** the sum¹³ of the value of Accumulator x and Amount, Authorised is greater than the Accumulator x Upper Limit*
 - ***or** the Transaction Currency Code does not match the Accumulator Currency Code **and** the Transaction Currency Code matches the Source Currency Code in one of the Currency Conversion Parameters in the Currency Conversion Table active for Accumulator x **and** the sum¹⁴ of the value of Accumulator x and the amount converted to the accumulator currency (using Amount, Authorised and the Currency Conversion Table active for Accumulator x) is greater than the Accumulator x Upper Limit*

then the application shall:

- *set the 'Accumulator x Upper Limit Exceeded' bit in the ADR to 1b*
- *set the 'Upper Cumulative Offline Amount Limit Exceeded' bit in the CVR to 1b*

¹³ If adding the Amount, Authorised to the value of Accumulator x would cause the accumulator to overflow, then instead compare the value '99 99 99 99 99 99' (rather than the sum of Amount, Authorised and the value of Accumulator x) to the Accumulator x Upper Limit.

¹⁴ If adding the converted Amount, Authorised to the value of Accumulator x would cause the accumulator to overflow, then instead compare the value '99 99 99 99 99 99' (rather than the sum of the converted Amount, Authorised and the value of Accumulator x) to the Accumulator x Upper Limit.

15.5.3.17 Counter x Lower Limit Exceeded Check

This issuer-optional check provides the issuer notification that the lower limit for a count of transactions has been exceeded. It also enables the application to consider this information when deciding whether to approve or decline the transaction offline, or to send the transaction online.

This check is performed for each Counter x that is active for the transaction. Counter x Control is personalised to indicate which transactions are counted in Counter x. If active, Counter x may count transactions that are:

- declined offline
- approved offline:
 - all offline approvals,
 - or only international (based on country) offline approvals,
 - or only non-cumulated offline approvals,
 - or only non-cumulated, international-country offline approvals

If the terminal requests an AAC (offline decline), then the transaction will be declined regardless of the setting of any ADR bits. Thus, the application only needs to perform the card risk management checks that might set CVR bits. The processing that sets the CVR bit when the Counter x Lower Limit has been exceeded (for a terminal requesting an AAC) is described in section 15.5.7.

If the terminal requests a TC (offline approval) or an ARQC (go online for authorisation) and offline approved transactions would not be counted ¹⁵ in Counter x in the Profile, then if the transaction is to be approved offline, the current transaction is not included in the count when performing this check. The check tests whether the count from previous transactions has exceeded the Counter x Lower Limit.

¹⁵ That is, only declines are counted, it is not an international transaction if only international offline-approved transactions are counted, or it can be accumulated in an active accumulator if only non-cumulated offline-approved transactions are counted.

If the terminal requests an ARQC (go online for authorisation) and the current transaction could be counted,¹⁶ then it is an issuer option (using the 'Include ARQC Transaction in CRM Test' bit in the Counter x Control) whether to include the current transaction when determining whether the Counter x Lower Limit would be exceeded.¹⁷

If the terminal requests a TC (offline approval) and the current transaction could be counted,¹⁶ then the current transaction is included when determining whether the Counter x Lower Limit would be exceeded.

Req 15.44 (Check if counter lower limit was previously exceeded):

For each Counter x that is active for the transaction:

- *If **both** of the following are true:*
 - *the terminal requested a TC or an ARQC in the first GENERATE AC command*
 - ***and** the value of Counter x is greater than the Counter x Lower Limit*
- then the application shall:*
- *set the 'Counter x Lower Limit Exceeded' bit in the ADR to 1b*
 - *set the 'Lower Offline Transaction Count Limit Exceeded' bit in the CVR to 1b*

¹⁶ That is, it is an international transaction if only international offline-approved transactions are counted, and it cannot be accumulated in any active accumulator if only non-cumulated offline-approved transactions are counted.

¹⁷ If the terminal requests an ARQC, some issuer systems expect the current transaction to be included in the transaction count when performing this card risk management check, while others do not. This option allows issuers to configure the Counter x Lower Limit Exceeded Check to function in a manner consistent with previous implementations.

Req 15.45 (Check if counter lower limit is exceeded by this transaction):

For each Counter x that is active for the transaction:

- *if **all** of the following are true:*
 - *the 'Allow Counting' bit in the Counter Profile Control for Counter x has the value 1b,*
 - ***and** the 'Include Offline Approvals' bit in the Counter x Control has the value 1b,*
 - ***and either** of the following is true:*
 - *the terminal requested a TC in the first GENERATE AC command*
 - ***or** the terminal requested an ARQC in the first Generate AC command **and** the 'Include ARQC Transaction in CRM Test' bit in the Counter x Control has the value 1b*
 - ***and either** of the following is true:*
 - *the 'Include Only If International' bit in the Counter x Control has the value 0b*
 - ***or** the Terminal Country Code does not match the Issuer Country Code*

- **and either** of the following is true:
 - the 'Include Only If Not Accumulated' bit in the Counter x Control has the value 0b
 - **or** for **all** values of y, the transaction could not be accumulated in Accumulator y because one of the conditions listed in Table 15-7 is not true

Accumulator y is active for the Profile.
The 'Allow Accumulation' bit in the Accumulator Profile Control for Accumulator y in this profile has the value 1b.
The 'Include Offline Approvals' bit in the Accumulator y Control has the value 1b.
Either of the following is true: <ul style="list-style-type: none"> • the terminal requested a TC in the first GENERATE AC command • or the terminal requested an ARQC in the first GENERATE AC command and the 'Include ARQC Transaction in CRM Test' bit in the Accumulator y Control has the value 1b
Either of the following is true: <ul style="list-style-type: none"> • the Transaction Currency Code matches the Accumulator Currency Code • or the Transaction Currency Code does not match the Accumulator Currency Code and the Transaction Currency Code matches the Source Currency Code in one of the Currency Conversion Parameters in the Currency Conversion Table active for Accumulator y¹⁸

Table 15-7: Conditions for Accumulating Transaction in Accumulator y

- **and** the value of Counter x +1 is greater than the Counter x Lower Limit¹⁹ then the application shall:
 - set the 'Counter x Lower Limit Exceeded' bit in the ADR to 1b
 - set the 'Lower Offline Transaction Count Limit Exceeded' bit in the CVR to 1b

¹⁸ If the Currency Conversion Table ID in the Accumulator Profile Control for Accumulator y for the transaction has the value 'F', then no Currency Conversion Table is active for Accumulator y.

¹⁹ If incrementing Counter x would cause the counter to overflow, then instead compare the value 'FF' (rather than the value of Counter x plus 1) to the Counter x Lower Limit.

15.5.3.18 Counter x Upper Limit Exceeded Check

This issuer-optional check provides the issuer notification that the upper limit for a count of transactions has been exceeded. It also enables the application to consider this information when deciding whether to approve or decline the transaction offline, or to send the transaction online.

This check is performed for each Counter x that is active for the transaction. Counter x Control is personalised to indicate which transactions are counted in Counter x. If active, Counter x may count transactions that are:

- declined offline
- approved offline:
 - all offline approvals,
 - or only international (based on country) offline approvals,
 - or only non-cumulated offline approvals,
 - or only non-cumulated, international-country offline approvals

If the terminal requests an AAC (offline decline), then the transaction will be declined regardless of the setting of any ADR bits. Thus, the application only needs to perform the card risk management checks that might set CVR bits. The processing that sets the CVR bit when the Counter x Upper Limit has been exceeded (for a terminal requesting an AAC) is described in section 15.5.7.

If the terminal requests a TC (offline approval) or an ARQC (go online for authorisation) and offline approved transactions would not be counted²⁰ for an offline approval in Counter x in the Profile, then if the transaction is to be approved offline, the current transaction is not included in the count when performing this check. The check tests whether the count from previous transactions has exceeded the Counter x Upper Limit.

²⁰ That is, only declines are counted, it is not an international transaction if only international offline-approved transactions are counted, or it can be accumulated in an active accumulator if only non-cumulated offline-approved transactions are counted.

If the terminal requests an ARQC (go online for authorisation) and the current transaction could be counted,²¹ then it is an issuer option (using the 'Include ARQC Transaction in CRM Test' bit in the Accumulator x Control) whether to include the current transaction when determining whether the Counter x Upper Limit would be exceeded.²²

If the terminal requests a TC (offline approval) and the current transaction could be counted,²¹ then the current transaction is included when determining whether the Counter x Upper Limit would be exceeded.

Req 15.46 (Check if counter upper limit was previously exceeded:

For each Counter x that is active for the transaction:

- *If **both** of the following are true:*
 - *the terminal requested a TC or an ARQC in the first GENERATE AC command*
 - ***and** the value of Counter x is greater than the Counter x Upper Limit*
- then the application shall:*
- *set the 'Counter x Upper Limit Exceeded' bit in the ADR to 1b*
 - *set the 'Upper Offline Transaction Count Limit Exceeded' bit in the CVR to 1b*

²¹ That is, it is an international transaction if only international offline-approved transactions are counted, and it cannot be accumulated in any active accumulator if only non-cumulated offline-approved transactions are counted.

²² If the terminal requests an ARQC, some issuer systems expect the current transaction to be included in the transaction count when performing this card risk management check, while others do not. This option allows issuers to configure the Counter x Upper Limit Exceeded Check to function in a manner consistent with previous implementations.

Req 15.47 (Check if counter upper limit is exceeded by this transaction):

For each Counter x that is active for the transaction:

- *if **all** of the following are true:*
 - *the 'Allow Counting' bit in the Counter Profile Control for Counter x has the value 1b,*
 - ***and** the 'Include Offline Approvals' bit in the Counter x Control has the value 1b,*
 - ***and either** of the following is true:*
 - *the terminal requested a TC in the first GENERATE AC command*
 - ***or** the terminal requested an ARQC in the first GENERATE AC command **and** the 'Include ARQC Transaction in CRM Test' bit in the Counter x Control has the value 1b*
 - ***and either** of the following is true:*
 - *the 'Include Only If International' bit in the Counter x Control has the value 0b*
 - ***or** the Terminal Country Code does not match the Issuer Country Code*
 - ***and either** of the following is true:*
 - *the 'Include Only If Not Accumulated' bit in the Counter x Control for Counter x has the value 0b*
 - ***or** for **all** values of y, the transaction could not be accumulated in Accumulator y because one of the conditions listed in Table 15-7 is not true*
 - ***and** the value of Counter x + 1 is greater than the Counter x Upper Limit²³*
- then the application shall:*
 - *set the 'Counter x Upper Limit Exceeded' bit in the ADR to 1b*
 - *set the 'Upper Offline Transaction Count Limit Exceeded' bit in the CVR to 1b*

²³ If incrementing Counter x would cause the counter to overflow, then instead compare the value 'FF' (rather than the value of Counter x plus 1) to the Counter x Upper Limit.

15.5.3.19 Issuer-discretionary bit 1 and Issuer-discretionary bit 2 Checks

The setting of Issuer-discretionary bit 1 and Issuer-discretionary bit 2 in the CVR is beyond the scope of this specification.

NOTE: If these bits are used to indicate conditions on which an issuer would want to take action using the ADR and CIACs, the issuer-discretionary portion of the ADR and CIACs may be used. See section 19.

15.5.3.20 Maximum Number of Days Offline Check

This issuer-optional check determines whether the limit for the number of days since the application last sent a transaction online has been exceeded. The Number of Days Offline is measured from the date of the previous transaction that went online and the terminal did not indicate it was unable to go online. This enables the application to consider this information when deciding whether to approve or decline the transaction offline, or to send the transaction online.

This check uses the management of a Transaction Date as a Transaction Date in Days, as described in Annex E.

The 'Activate Maximum Number of Days Offline Check' bit in the Issuer Options Profile Control is used to activate this check.

Req 15.48 (Check whether to perform Maximum Number of Days Offline check):

If the 'Activate Maximum Number of Days Offline Check' bit in the Issuer Options Profile Control has the value 1b, then the application shall perform this check.

The application first checks whether:

- the Transaction Date is correctly formatted – the year is between 0 and 99, the month is between 1 and 12, and the day is between 1 and 31
- and the Transaction Date is not earlier than the Last Online Transaction Date in Days

Req 15.49 (Verify that date is valid for number of days offline check):

If any of the following is true:

- Transaction Date byte 1 (YY) is not in the range '00' to '99'
- **or** Transaction Date byte 2 (MM) is not in the range '01' to '12'
- **or** Transaction Date byte 3 (DD) is not in the range '01' to '31'
- **or** the current Transaction Date in Days is less than the Last Online Transaction Date in Days

then the application shall:

- set the 'Check Failed' bit in the ADR to the value 1b
- set the 'Check Failed' bit in the CVR to the value 1b
- discontinue processing this check

Req 15.50 (Check whether too much time has lapsed since card last went online):

If the difference between the current Transaction Date in Days and the Last Online Transaction Date in Days is greater than the Number of Days Offline Limit, then the application shall set the 'Number of Days Offline Limit Exceeded' bit in the ADR to the value 1b.

15.5.3.21 Maximum Transaction Amount Check

This issuer-optional check determines whether the limit for the Maximum Transaction Amount has been exceeded. This enables the application to consider this information when deciding whether to approve or decline the transaction offline, or to send the transaction online.

The MTA Profile Control ID in the Profile Control selected for the transaction is used to activate this check.

Req 15.51 (Check whether to perform Maximum Transaction Amount check):

If the MTA Check is active (that is, the MTA Profile Control ID in the Profile Control has a value other than 'F'), then the application shall perform this check.

Req 15.52 (Check whether Maximum Transaction Amount is exceeded):

*If **both** of the following are true:*

- *the terminal requested a TC or an ARQC in the first GENERATE AC command*
- ***and either** of the following is true:*
 - *the Transaction Currency Code matches the MTA Currency Code **and** Amount, Authorised is greater than the MTA*
 - ***or** the Transaction Currency Code does not match the MTA Currency Code **and** the Currency Conversion Table ID in the MTA Profile Control does not have the value 'F' **and** the Transaction Currency Code matches the Source Currency Code in one of the Currency Conversion Parameters in the Currency Conversion Table for the Maximum Transaction Amount **and** the amount converted to the accumulator currency²⁴ (using Amount, Authorised and the Currency Conversion Table for the Maximum Transaction Amount) is greater than the MTA*

then the application shall set the 'MTA exceeded' bit in the ADR to the value 1b.

²⁴ If the converted Amount, Authorised would overflow the maximum value for an amount ('99 99 99 99 99 99'), then instead compare the value '99 99 99 99 99 99' (rather than the converted Amount, Authorised) to the MTA.

15.5.3.22 Cyclic Accumulator x Limit Exceeded Check

This issuer-optional check determines whether a limit for the cumulative amount of approved transactions within a single cycle (such as a Day, Week, or Month) has been exceeded. This enables the application to consider this information when deciding whether to approve or decline the transaction offline, or to send the transaction online.

A Weekly Cyclic Accumulator uses the management of a Transaction Date as a Transaction Date in Days, as described in Annex E.

This check is performed for each Cyclic Accumulator x that is active for the transaction.

Req 15.53 (Check whether to perform Cyclic Accumulator check):

For each value of x, if Cyclic Accumulator x is active for the transaction then the application shall perform this check.

The Cyclic Accumulator accumulates all transactions approved offline within a single cycle where the transaction currency is **either**:

- the accumulator currency
- any currency whose transaction value can be approximated in the accumulator currency using the Currency Conversion Table for Cyclic Accumulator x if a Currency Conversion Table is active for Cyclic Accumulator x in the Profile.

NOTE: If the Currency Conversion function is used, the amount converted to the accumulator currency is an approximation based on the conversion rate in the Currency Conversion Parameter.

NOTE: If the active Currency Conversion Table for Cyclic Accumulator x does not contain conversion rates for any additional currencies, the accumulator contains the amount of offline transactions conducted in the accumulator currency.

If the terminal requests an AAC (offline decline), then the transaction will be declined regardless of the setting of any ADR bits. The application only needs to perform the processing that would reset the Cyclic Accumulator if the transaction date is in a new cycle.

If the terminal requests a TC (offline approval) or an ARQC (go online for authorisation) and the current transaction could be accumulated, then the current transaction is included when testing whether the limit is exceeded. The current transaction could be accumulated if the transaction either is in the accumulator currency, or can be converted into the accumulator currency; and accumulation is allowed in Cyclic Accumulator x for the Profile.

The application first checks whether:

- the transaction date is correctly formatted – the year is between 0 and 99, the month is between 1 and 12, and the day is between 1 and 31
- and the transaction Date is not earlier than the date for the beginning of the current cycle – for a daily or monthly check, the Transaction Date on which the accumulator was last reset; or for a weekly check the day at the beginning of the week in which the accumulator was last reset.

Req 15.54 (Verify that date is valid for Cyclic Accumulator check):

If **any** of the following is true:

- Transaction Date byte 1 (YY) is not in the range '00' to '99',
- **or** Transaction Date byte 2 (MM) is not in the range '01' to '12',
- **or** Transaction Date byte 3 (DD) is not in the range '01' to '31',
- **or both** of the following are true:
 - the 'Cycle Type' in the Cyclic Accumulator x Control has the value 01b (daily) or 11b (monthly)
 - **and** the Transaction Date is less than the Cyclic Accumulator x Reference Date,
- **or both** of the following are true:
 - the 'Cycle Type' in the Cyclic Accumulator x Control has the value 10b (weekly)
 - **and** the Transaction Date in Days (see annex E) is less than the Cyclic Accumulator x Reference Day,

then the transaction date is not valid for this check, so the application shall:

- set the 'Check Failed' bit in the ADR to the value 1b
- set the 'Check Failed' bit in the CVR to the value 1b
- discontinue processing this check.

The application then checks whether the transaction is within the current cycle.

- The transaction is within the current day if the Transaction Date (YYMMDD) matches the Reference Date (YYMMDD).
- The transaction is within the current week if the Transaction Date is before the start of the week following the current weekly cycle.
- The transaction is within the current month if the Transaction Date (YYMM) matches the Reference Date (YYMM).

Req 15.55 (Check whether transaction date is in current cycle):

*If **any** of the following is true:*

- **both** of the following are true:
 - the 'Cycle Type' in the Cyclic Accumulator x Control has the value 01b (daily)
 - **and** the Transaction Date is equal to the Cyclic Accumulator x Reference Date
- **or both** of the following are true:
 - the 'Cycle Type' in the Cyclic Accumulator x Control has the value 10b (weekly)
 - **and** the Transaction Date in Days is less than the Cyclic Accumulator x Reference Day plus 7.
- **or both** of the following are true:
 - the 'Cycle Type' in the Cyclic Accumulator x Control has the value 11b (monthly)
 - **and** the year and month of the Transaction Date is equal to the year and month of the Cyclic Accumulator x Reference Date

then the transaction is within the current cycle.

If the transaction is within a new cycle (not within the current cycle), then before checking whether the limit is exceeded, the application resets the reference date for the cycle and resets the value in the cyclic accumulator to zero.

Req 15.56 (Reset daily or monthly cyclic accumulator if date is in a new cycle):

If **either** of the following is true:

- **both** of the following are true:
 - the 'Cycle Type' in the Cyclic Accumulator x Control has the value 01b (daily)
 - **and** the Transaction Date is not within the current cycle (that is, the Transaction Date is greater than the Cyclic Accumulator x Reference Date)
- **or both** of the following are true:
 - the 'Cycle Type' in the Cyclic Accumulator x Control has the value 11b (monthly)
 - **and** the Transaction Date is not within the current cycle (that is, the year and month of the Transaction Date is greater than the year and month of the Cyclic Accumulator x Reference Date)

then the application shall:

- reset the Cyclic Accumulator x Reference Date to the Transaction Date
- reset Cyclic Accumulator x to zero.

Req 15.57 (Reset weekly cyclic accumulator if date is in a new cycle):

If **both** of the following are true:

- the 'Cycle Type' in the Cyclic Accumulator x Control has the value 10b (weekly)
- **and** the Transaction Date is not within the current cycle (that is, the Transaction Date in Days is greater than the Cyclic Accumulator x Reference Day plus 7)

then the application shall:

- reset the Cyclic Accumulator x Reference Day to the Transaction Date Reference Day
- reset Cyclic Accumulator x to zero.

If the transaction is within the current cycle, and the transaction could be approved (that is, the terminal requested an offline approval or to go online), and the limit for the cycle has been exceeded, then the application will set an ADR bit.

Req 15.58 (Determine whether limit was previously exceeded for current cycle):

*If **all** of the following are true:*

- *the transaction is within the current cycle,*
- ***and** the terminal requested a TC or an ARQC in the first GENERATE AC command,*
- ***and** the value of Cyclic Accumulator x is greater than the Cyclic Accumulator x Limit*

then the application shall:

- *set the 'Cyclic Accumulator x Limit Exceeded' bit in the ADR to the value 1b*
- *discontinue processing this check.*

If the current transaction could be accumulated in Cyclic Accumulator x, then the application checks whether accumulating the transaction would result in exceeding the limit for the cycle. If the limit would be exceeded, then the application will set an ADR bit.

Req 15.59 (Determine whether limit is exceeded if accumulating for current cycle):

*After performing any resets required for the beginning of a new cycle, if **all** of the following are true:*

- *the terminal requested a TC or an ARQC in the first GENERATE AC command,*
- ***and** the 'Allow Accumulation' bit in the Cyclic Accumulator Profile Control for Cyclic Accumulator x in this profile has the value 1b,*
- ***and either** of the following is true:*
 - *the Transaction Currency Code matches the Accumulator Currency Code, **and** the sum²⁵ of the value of Cyclic Accumulator x and Amount, Authorised is greater than the Cyclic Accumulator x Limit*
 - ***or** the Transaction Currency Code does not match the Accumulator Currency Code **and** the Transaction Currency Code matches the Source Currency Code in one of the Currency Conversion Parameters in the Currency Conversion Table active for Cyclic Accumulator x²⁶ **and** the sum²⁷ of the value of Cyclic Accumulator x and Amount, Authorised converted to the accumulator currency (using the Currency Conversion Table) is greater than the Cyclic Accumulator x Limit*

then the application shall:

- *set the 'Cyclic Accumulator x Limit Exceeded' bit in the ADR to the value 1b,*
- *discontinue processing this check.*

²⁵ If adding the Amount, Authorised to the value of Cyclic Accumulator x would cause the cyclic accumulator to overflow, then instead compare the value '99 99 99 99 99 99' (rather than the sum of Amount, Authorised and the value of Cyclic Accumulator x) to the Cyclic Accumulator x Limit.

²⁶ If the Currency Conversion Table ID in the Cyclic Accumulator Profile Control for Cyclic Accumulator x for the transaction has the value 'F', then no Currency Conversion Table is active for Cyclic Accumulator x.

²⁷ If adding the converted Amount, Authorised to the value of Cyclic Accumulator x would cause the cyclic accumulator to overflow, then instead compare the value '99 99 99 99 99 99' (rather than the sum of the converted Amount, Authorised and the value of Cyclic Accumulator x) to the Cyclic Accumulator x Limit.

15.5.4 Determine Response Application Cryptogram Type

The transaction will be declined offline if the terminal requests an offline decline, the application is blocked, or the terminal requests to send a VLP transaction online (VLP transactions are offline-only). Otherwise, the Application Decisional Results (ADR), Application Control, Issuer Options Profile Control, and Card Issuer Action Codes (CIACs) data elements shall be used to determine the Application Cryptogram Type for the first GENERATE AC response.

NOTE: Checking the Application Blocked bit in the PTH is one method to determine whether the application is blocked. It is at the discretion of the implementer to provide the functionality to determine whether the application is blocked, but the behaviour of the application shall be the same as if the bit in the PTH were used.

Each CIAC is structured the same as the ADR (bit strings, with the bits referring to the same conditions). The ADR indicates that the condition or event has occurred in the transaction. The CIACs control the action to be taken by the application when the corresponding bit in the ADR is set.

CIAC - Decline: A bit shall be set to the value 1b in the CIAC - Decline if the application is to decline the transaction at the first GENERATE AC (that is, offline) when the corresponding bit has the value 1b in the ADR.

CIAC - Default: A bit shall be set to the value 1b in the CIAC - Default if the application is to decline the transaction when the corresponding bit has the value 1b in the ADR and either the terminal is not capable of going online or no response is received from the issuer.

CIAC - Online: A bit shall be set to the value 1b in the CIAC - Online if the application is to go online if the terminal is capable and the corresponding bit has the value 1b in the ADR.

Based on the results of this Card Risk Management, the application determines the Application Cryptogram type to be sent in the response to the GENERATE AC command issued by the terminal. The card's response may override the cryptogram type designated by the terminal in the P1 parameter of the first GENERATE AC command according to the following rules:

- The card may override the terminal's decision to approve offline by deciding to either send online or decline offline.
- The card may override the terminal's decision to go online by deciding to decline offline.

Table 15-8 illustrates these decision rules:

		Card Responds		
		AAC	ARQC	TC
Terminal Requests	AAC	Decline	—	—
	ARQC	Decline	Go Online	—
	TC	Decline	Go Online	Approve
NOTE: In CPA, the card shall never respond with an AAR (referral).				

Table 15-8: Card's Response to First GENERATE AC Command

Req 15.60 (Automatically decline transaction offline):

*If **any** of the following is true:*

- *the terminal requests an AAC*
- *or the 'Application Blocked' bit in the PTH has the value 1b*
- *or all of the following are true:*
 - *VLP is supported in the application,*
 - *and the Profile ID is '7D' (VLP)*
 - *and the terminal requests an ARQC*

then the application shall generate an AAC type Application Cryptogram response to the first GENERATE AC

Req 15.61 (Terminal requested to go online, decide card response):

If the Terminal requests an ARQC, then the application compares the CIAC - Decline and the ADR.

- *If both a CIAC - Decline bit and the corresponding ADR bit are set, then the application shall generate an AAC type Application Cryptogram.*
- *Otherwise, the application shall generate an ARQC type Application Cryptogram.*

Req 15.62 (Terminal requested offline approval, decide card response):

*If the Terminal requests a TC, **and** VLP is supported in the application, **and** the Profile ID is '7D' (VLP), then the application shall generate a TC type Application Cryptogram.*

Otherwise, if the Terminal requests a TC, then the application compares the CIAC - Decline and the ADR.

- *If both a CIAC - Decline bit and the corresponding ADR bit are set, then the application shall generate an AAC type Application Cryptogram.*
- *Otherwise, the application shall determine whether the terminal is online-capable (that is, that the Terminal Type is not 13, 16, 23, 26, or 36).*
 - *If the terminal is online-capable, then the application compares the CIAC - Online and the ADR:*
 - *If both a CIAC - Online bit and the corresponding ADR bit are set, then the application shall generate an ARQC type Application Cryptogram.*
 - *Otherwise, the application shall generate a TC type Application Cryptogram.*
 - *Otherwise (that is, the terminal is not online-capable) if **both** of the following are true:*
 - *the Terminal Type is 26,*
 - ***and** the 'Allow Override of CIAC-Default for Transactions at Terminal Type 26' bit in Issuer Options Profile Control has the value 1b.*
- 1. *then the application shall generate a TC type Application Cryptogram.*
- *Otherwise (that is, Offline-only terminal of type other than 26, or terminal type 26 and conditions to override the CIAC - Default check are not met), the application compares the CIAC - Default and the ADR:*
 - *If both a CIAC - Default bit and the corresponding ADR bit are set, then the application shall generate an AAC type Application Cryptogram.*
 - *Otherwise, the application shall generate a TC type Application Cryptogram.*

Additional card processing for each response decision is outlined in the following sections.

15.5.5 Application Approves Transaction Offline

When the transaction is to be approved offline, the application returns a Transaction Certificate (TC) type Application Cryptogram in the response to the first GENERATE AC command. The application sets bits in the CVR to indicate that a TC (offline approval) is returned in response to the first GENERATE AC, and updates offline counts and amounts used for velocity-checking so that the updated amounts can be included in the Counters portion of Issuer Application Data.

Req 15.63 (Update accumulators for offline approval):

*If **either** VLP is not supported by the application, **or** the Profile ID is not '7D', then prior to building the Issuer Application Data data element and generating the response cryptogram, the application shall:*

- *For each Accumulator x that is active for the transaction **and** the 'Include Offline Approvals' bit in the Accumulator x Control has the value 1b **and** the 'Allow Accumulation' bit in the Accumulator Profile Control for Accumulator x for this profile has the value 1b:*
 - *If the Transaction Currency Code matches the Accumulator Currency Code, then the application shall add Amount, Authorised to Accumulator x .*
 - *If adding the Amount, Authorised to the value of Accumulator x would exceed the value '99 99 99 99 99 99', then Accumulator x shall be set to the value '99 99 99 99 99 99'*
 - *If the Transaction Currency Code does not match the Accumulator Currency Code and the Transaction Currency Code matches the Source Currency Code in one of the Currency Conversion Parameters in the Currency Conversion Table active for Accumulator x ,²⁸ then the application shall add Amount, Authorised converted to the accumulator currency (using the Currency Conversion Table) to Accumulator x .*
 - *If adding the Amount, Authorised converted to the accumulator currency to the value of Accumulator x would exceed the value '99 99 99 99 99 99', then Accumulator x shall be set to the value '99 99 99 99 99 99'*

²⁸ If the Currency Conversion Table ID in the Accumulator Profile Control for Accumulator x for the transaction has the value 'F', then no Currency Conversion Table is active for Accumulator x .

Req 15.64 (Update counters for offline approval):

If **either** VLP is not supported by the application, **or** the Profile ID is not '7D', then prior to building the Issuer Application Data data element and generating the response cryptogram, the application shall:

- For each Counter *x* that is active for the transaction **and** the 'Allow Counting' bit in the Counter Profile Control for Counter *x* has the value 1b:
 - If **all** of the following are true:
 - if the 'Include Offline Approvals' bit in the Counter *x* Control has the value 1b,
 - **and either** of the following is true:
 - the 'Include Only If International' bit in the Counter *x* Control has the value 0b
 - **or** the Terminal Country Code does not match the Issuer Country Code
 - **and either** of the following is true:
 - the 'Include Only If Not Accumulated' bit in the Counter *x* Control has the value 0b
 - **or** for all values of *y* for which Accumulator *y* is active for the transaction, allows accumulation, and includes offline approvals for the Profile; the Transaction Currency Code does not match **either** of the following (that is, the transaction would be accumulated in at least one accumulator in the Profile):
 - the Accumulator Currency Code
 - **or** the Source Currency Code in any of the Currency Conversion Parameters in the Currency Conversion Table active for Accumulator *y*²⁹
 - **and** the value of Counter *x* is less than 'FF',
- then the application shall increment Counter *x* by one.

²⁹ If the Currency Conversion Table ID in the Accumulator Profile Control for Accumulator *y* for the transaction has the value 'F', then no Currency Conversion Table is active for Accumulator *y*.

Req 15.65 (Update CVR for offline approval):

Prior to building the Issuer Application Data data element and generating the response cryptogram, the application shall:

- *Set the 'Application Cryptogram Type Returned in First GENERATE AC' bits in the CVR to the value 01b to indicate a TC.*
- *Set the 'Application Cryptogram Type Returned in Second GENERATE AC' bits in the CVR to the value 10b to indicate Second GENERATE AC Not Requested.*
- *If the 'CDA Requested' bit of the P1 parameter in the GENERATE AC command has the value 1b (the terminal is requesting Combined DDA/AC Generation), then the application shall set the 'CDA Performed' bit in the CVR to the value 1b.*

The application updates the PTH to indicate conditions in the current transaction that are used in processing subsequent transactions. The application also sets the Cryptogram Information Data (CID) to indicate to the terminal the type of Application Cryptogram in the GENERATE AC response.

Req 15.66 (Update CID for offline approval):

Prior to responding to the GENERATE AC command, the application shall set the Cryptogram Information Data (CID) to the value '40' to indicate that a TC is being returned and no advice is required.

Req 15.67 (Update offline data authentication indicators for offline approval):

Prior to responding to the GENERATE AC command, if any of the following bits in the TVR provided by the terminal in the Generate AC command data has the value 1b:

- *SDA Failed*
- *DDA Failed*
- *CDA Failed*

then the application shall set the 'Offline Data Authentication Failed on Previous Transaction' bit in PTH to the value 1b.

Otherwise, the application shall set the 'Offline Data Authentication Failed on Previous Transaction' bit in PTH to the value 0b.

Req 15.68 (Update cyclic accumulators for offline approval):

If **either** VLP is not supported by the application, **or** the Profile ID is not '7D', then prior to responding to the Generate AC command, for each value of x for which **all** of the following are true:

- Cyclic Accumulator x is active for the transaction
- **and** the 'Allow Accumulation' bit in the Cyclic Accumulator Profile Control for Cyclic Accumulator x for this profile has the value 1b
- **and** the Transaction Date is valid for this check (see Req 15.54)

then:

- If the Transaction Currency Code matches the Accumulator Currency Code, then the application shall add Amount, Authorised to Cyclic Accumulator x.
 - If adding the Amount, Authorised to the value of Cyclic Accumulator x would exceed the value '99 99 99 99 99 99', then Cyclic Accumulator x shall be set to the value '99 99 99 99 99 99'
- If the Transaction Currency Code does not match the Accumulator Currency Code **and** the Transaction Currency Code matches the Source Currency Code in one of the Currency Conversion Parameters in the Currency Conversion Table active for Cyclic Accumulator x³⁰, then the application shall add Amount, Authorised converted to the accumulator currency (using the Currency Conversion Table) to Cyclic Accumulator x.
 - If adding the Amount, Authorised converted to the accumulator currency to the value of Cyclic Accumulator x would exceed the value '99 99 99 99 99 99', then Cyclic Accumulator x shall be set to the value '99 99 99 99 99 99'

Continue with section 15.5.8.

³⁰ If the Currency Conversion Table ID in the Cyclic Accumulator Profile Control for Cyclic Accumulator x for the transaction has the value 'F', then no Currency Conversion Table is active for Cyclic Accumulator x.

15.5.6 Application Requests Online Processing

When the transaction is to go online for an authorisation, the application returns an ARQC type Application Cryptogram in the response to the first GENERATE AC command. The application sets bits in the CVR to indicate that an ARQC (online request) is returned in response to the first GENERATE AC.

Req 15.69 (Update CVR for going online):

Prior to building the Issuer Application Data data element and generating the response cryptogram, the application shall:

- *set the 'Application Cryptogram Type Returned in First GENERATE AC' bits in the CVR to the value 10b to indicate an ARQC.*
- *set the 'Application Cryptogram Type Returned in Second GENERATE AC' bits in the CVR to the value 10b to indicate Second GENERATE AC Not Requested.*
- *set the 'CDA Performed' bit in the CVR to the value 1b if the terminal requested CDA (that is, if the 'CDA Requested' bit of the P1 parameter in the GENERATE AC command has the value 1b).*

The application updates the PTH to indicate conditions in the current transaction that are used in processing subsequent transactions. The application also sets the Cryptogram Information Data (CID) to indicate to the terminal the type of Application Cryptogram in the GENERATE AC response.

Req 15.70 (Update CID and Last Online Transaction indicators for going online):

Prior to responding to the GENERATE AC command, the application shall:

- *set the Cryptogram Information Data (CID) to the value '80' to indicate that an ARQC is being returned and no advice is required*
- *set the 'Last Online Transaction not Completed' bit in the PTH to the value 1b.*

Req 15.71 (Update offline data authentication indicator for going online):

*Prior to responding to the GENERATE AC command, if **any** of the following bits in the TVR provided by the terminal in the GENERATE AC command data has the value 1b:*

- *SDA Failed*
- *DDA Failed*
- *CDA Failed*

then the application shall set the 'Offline Data Authentication Failed on Previous Transaction' bit in PTH to the value 1b.

Accumulator x, Counter x, and Cyclic Accumulator x counters are not incremented at this time.

Continue with section 15.5.8.

15.5.7 Application Declines Transaction Offline

When the transaction is to be declined offline, the application returns an AAC type Application Cryptogram in the first GENERATE AC response. The application sets bits in the CVR to indicate that an AAC (offline decline) is to be returned in response to the first GENERATE AC.

If a VLP transaction is declined offline, the transaction amount that was deducted from the VLP Available Funds during processing of the GET PROCESSING OPTIONS command is refunded (that is, added back) to the VLP Available Funds during GENERATE AC command processing.

Req 15.72 (Update VLP Available Funds for offline decline):

Prior to building the Issuer Application Data data element and generating the response cryptogram:

- *If the application supports VLP and the Profile ID is '7D' (VLP), then the application shall add the value of Amount, Authorised to the VLP Available Funds Amount.*

Req 15.73 (Update counters for offline decline):

If the Profile ID is not '7D', then prior to building the Issuer Application Data data element and generating the response cryptogram, for each Counter x that is active for the transaction:

- *If the 'Include Offline Declines' bit in the Counter x Control has the value 1b and the 'Allow Counting' bit in Counter Profile Control x has the value 1b and the value of Counter x is less than 'FF', then the application shall increment Counter x by one.*
- *If the value in Counter x is greater than the Counter x Lower Limit, then the application shall set the 'Lower Offline Transaction Count Limit Exceeded' bit in the CVR to the value 1b.*
- *If the value in Counter x is greater than the Counter x Upper Limit, then the application shall set the 'Upper Offline Transaction Count Limit Exceeded' bit in the CVR to the value 1b.*

Req 15.74 (Reset counter-related bits in CVR for offline decline):

If the Profile ID is not '7D', then prior to building the Issuer Application Data data element and generating the response cryptogram, for each Counter x that is active for the transaction:

- *If none of the Counter x values has exceeded the associated Counter x Lower Limit, then the application shall reset the 'Lower Offline Transaction Count Limit Exceeded' bit in the CVR to the value 0b.*
- *If none of the Counter x values has exceeded the associated Counter x Upper Limit, then the application shall reset the 'Upper Offline Transaction Count Limit Exceeded' bit in the CVR to the value 0b.*

Req 15.75 (Update accumulator-related bits in CVR for offline decline):

If the Profile ID is not '7D', then prior to building the Issuer Application Data data element and generating the response cryptogram, for each Accumulator x that is active for the transaction:

- *If the value in Accumulator x is greater than the Accumulator x Lower Limit, then the application shall set the 'Lower Cumulative Offline Amount Limit Exceeded' bit in the CVR to the value 1b.*
- *If the value in Accumulator x is greater than the Accumulator x Upper Limit, then the application shall set the 'Upper Cumulative Offline Amount Limit Exceeded' bit in the CVR to the value 1b.*

Req 15.76 (Reset accumulator-related bits in CVR for offline decline):

If the Profile ID is not '7D', then:

- *If none of the Accumulator x values has exceeded the associated Accumulator x Lower Limit, then the application shall reset the 'Lower Cumulative Offline Amount Limit Exceeded' bit in the CVR to the value 0b.*
- *If none of the Accumulator x values has exceeded the associated Accumulator x Upper Limit, then the application shall reset the 'Upper Cumulative Offline Amount Limit Exceeded' bit in the CVR to the value 0b.*

Req 15.77 (Update CVR for offline decline):

Prior to building the Issuer Application Data data element and generating the response cryptogram, the application shall:

- *set the 'Application Cryptogram Type Returned in First GENERATE AC' bits in the CVR to the value 00b to indicate an AAC.*
- *set the 'Application Cryptogram Type Returned in Second GENERATE AC' bits in the CVR to the value 10b to indicate Second GENERATE AC Not Requested.*

The application updates the PTH to indicate conditions in the current transaction that are used in processing subsequent transactions. The application also sets the Cryptogram Information Data (CID) to indicate to the terminal the type of Application Cryptogram in the GENERATE AC response.

Req 15.78 (Update CID for offline decline):

Prior to responding to the GENERATE AC command, the application shall set the Cryptogram Information Data (CID) to the value '00' to indicate that an AAC is requested and no advice is required.

Req 15.79 (Update offline data authentication indicators for offline decline):

*Prior to responding to the GENERATE AC command, if **any** of the following bits in the TVR provided by the terminal in the Generate AC command data has the value 1b:*

- *SDA Failed*
- *DDA Failed*
- *CDA Failed*

then the application shall set the 'Offline Data Authentication Failed on Previous Transaction' bit in PTH to the value 1b.

Continue with section 15.5.8.

15.5.8 Respond to GENERATE AC Command

15.5.8.1 Build Issuer Application Data

Req 15.80 (Build Issuer Application Data for Token Authentication profile):

If the Profile ID has the value '7E' (Token Authentication Profile – see Annex H10.2), then the application shall build the Issuer Application Data (IAD) to be sent in the response, coded as specified in the CCD Part of EMV Book 3, Annex C.7, for a CCD-compliant application with a Format Code of 'A' with Cryptogram Version of '5'; with the profile-specific requirements shown in Table 15-9.

IAD Byte	Description	Value
1	Length	'0F'
2	CCI	'A5'
3	DKI	issuer-discretionary
4-8	CVR	set to zero except for the following bits used to indicate the results of offline PIN verification: <ul style="list-style-type: none"> 'Offline PIN Verification Performed' 'Offline PIN Verification Performed and PIN Not Successfully Verified'
9-16	Counters	zero
17	Length	'0F'
18	Profile ID	'7E'
19-32	issuer-discretionary	zero

Table 15-9: Issuer Application Data for Profile '7E' (Authentication Token)

Req 15.81 (Build Issuer Application Data for other profiles):

*If the Profile ID is **not** '7E'; then the application shall build the Issuer Application Data (IAD) to be sent in the response, coded as specified in the CCD Part of EMV Book 3, Annex C.7, for a CCD-compliant application with a Format Code of 'A', with:*

- *For each Accumulator x that is active for the transaction **and** for which the 'Send Accumulator in IAD' bit in the Accumulator Profile Control for Accumulator x has the value 1b:*
 - *If the 'Send Accumulator Balance' bit in the Accumulator Profile Control for Accumulator x has the value 1b, then the value (Accumulator x Upper Limit minus Accumulator x) shall be sent.*
 - *Otherwise ('Send Accumulator Balance' = 0b) the value of Accumulator x shall be sent.*
- *the profile-specific requirements shown in Table 15-10*
- *If the 'Encipher Counters Portion of IAD' bit in the Issuer Options Profile Control has the value 1b, then the Counters portion (bytes 9-16) of Issuer Application Data shall be enciphered (see section 20) before generating the Application Cryptogram.*

IAD Byte	Description	Value
1	Length	'0F'
2	CCI	Set to the value of the Profile CCI in the Issuer Options Profile Control for the transaction ('A5' for CCD-compliant profiles)
3	DKI	Set to the value of the Profile DKI in the Issuer Options Profile Control for the transaction (issuer-discretionary)
4-8	CVR	Set by application processing
9-16	Counters	<p>Begins with the following:</p> <ul style="list-style-type: none"> • If Accumulator 1 is active for the transaction and the 'Send Accumulator in IAD' bit in the Accumulator Profile Control for Accumulator 1 has the value 1b, then Accumulator 1 (Value or Balance) is sent in Counters bytes 1-6. • Otherwise, if Accumulator 2 is active for the transaction and the 'Send Accumulator in IAD' bit in the Accumulator Profile Control for Accumulator 2 has the value 1b, then Accumulator 2 (Value or Balance) is sent in Counters bytes 1-6. • Otherwise, if VLP Available Funds is active for the transaction and the 'Send Accumulator in IAD' bit in the VLP Profile Control has the value 1b, then the VLP Available Funds is sent in Counters bytes 1-6. <p>The remaining bytes shall contain the values of each Counter x that is active for the transaction and for which the 'Send Counter in IAD' bit in the Counter Profile Control for Counter x has the value 1b; in priority order based upon the counter number (that is, the value of x for Counter x), with the lowest numbered counter having the highest priority.</p> <p>The default value for these bytes is personalised in bytes 9-16 of the Default Issuer Application Data. Any portion of these bytes not filled with an accumulator or counter should use the default value.</p>
17	Length	'0F'

Table 15-10: Issuer Application Data for Profile Not '7E'

IAD Byte	Description	Value
18	Profile ID	Profile ID used for the transaction
19-32	issuer-discretionary	<p>If more than one accumulator is to be sent in the IAD, these bytes contain the remaining accumulators that were not sent in bytes 9-16, in the order shown:</p> <ul style="list-style-type: none"> • Accumulator 2 if Accumulator 2 is active for the transaction and the 'Send Accumulator in IAD' bit in the Accumulator Profile Control for Accumulator 2 has the value 1b. • VLP Available Funds if VLP is supported and VLP Available Funds is active for the transaction and the 'Send Accumulator in IAD' bit in the VLP Profile Control has the value 1b. <p>The remaining bytes shall contain the values of each Counter x not included in bytes 9-16 that is active for the transaction and for which the 'Send Counter in IAD' bit in the Counter Profile Control for Counter x has the value 1b; in priority order based upon the counter number (that is, the value of x for Counter x), with the lowest numbered counter having the highest priority.</p> <p>The default value for these bytes is personalised in bytes 19-32 of the Default Issuer Application Data. Any portion of these bytes not filled with an accumulator or counter should use the default value.</p> <p>NOTE: Issuers may request specific data in this field, but this functionality is outside the scope of this specification.</p>

Table 15-10: Issuer Application Data for Profile Not '7E', continued

The examples in Figure 15-2 illustrate the result of building the IAD with accumulators and counters.

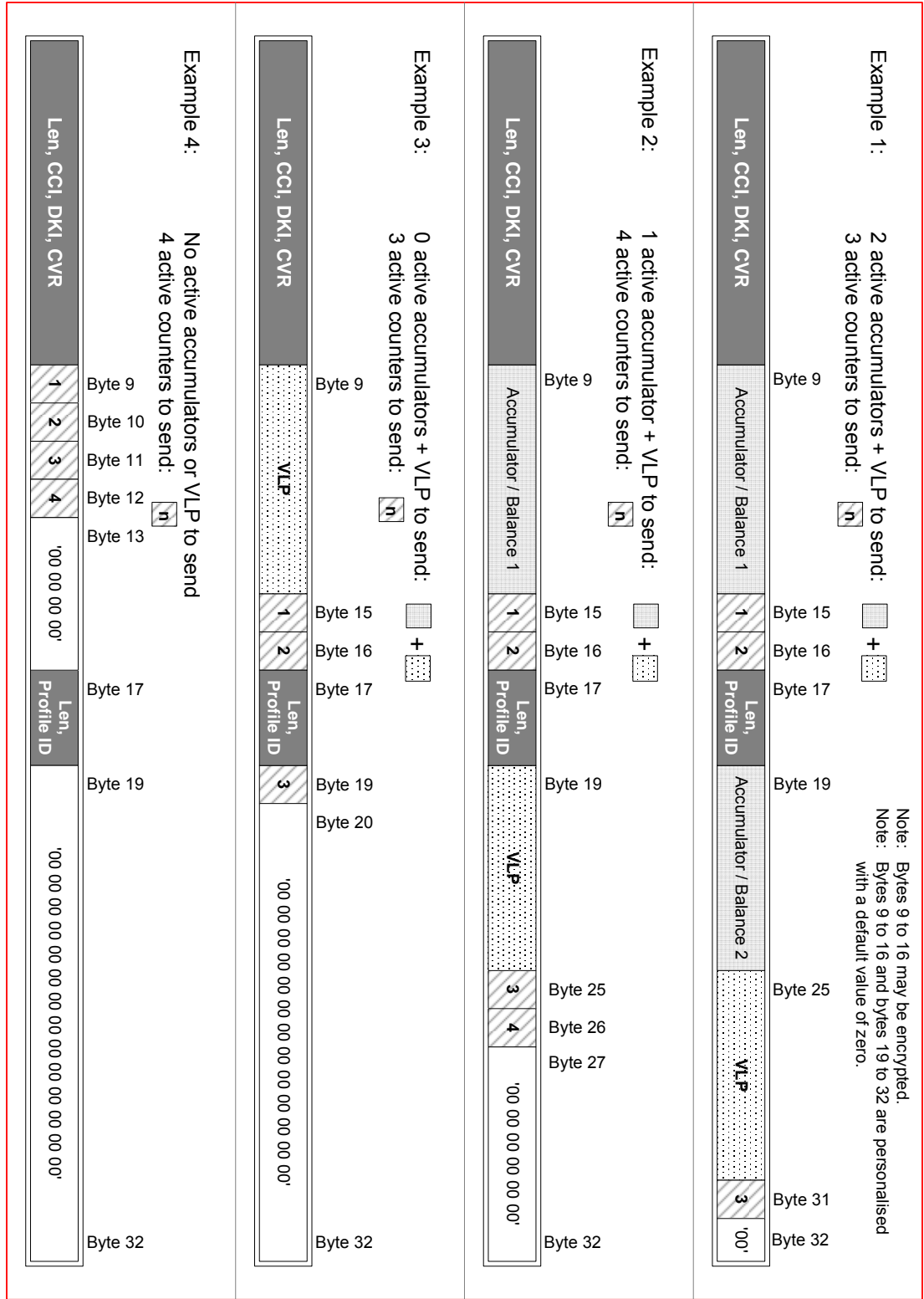


Figure 15-2: Examples – Building the IAD

15.5.8.2 Generate Application Cryptogram

The application generates an Application Cryptogram using the data provided by the terminal and data from the card.

Data requirements, key requirements, and the algorithms used in the cryptogram generation process are as detailed in Table CCD-3 and section 8 of the CCD part of *EMV Book 2*, for a CCD-compliant application with Cryptogram Version of '5'.

NOTE: Support for Cryptogram Versions with a value in the range '0' to '3' is beyond the scope of this specification. A Profile that uses a Cryptogram Version in this range is not CCD-compliant.

15.5.8.3 Log Transaction

If the issuer chooses to log transactions and the response to the First GENERATE AC is a TC or an AAC, then the application appends the information to the Transaction Log. If the issuer chooses to log transactions and the response to the First GENERATE AC is an ARQC, then the application will need to save data to be logged during the Second GENERATE AC (see Annex D4.1).

Req 15.82 (Log transaction at first GENERATE AC):

*Prior to responding to the GENERATE AC command, if **both** of the following are true:*

- *the 'Log Transactions' bit in the Issuer Options Profile Control has the value 1b,*
- ***and either** of the following is true:*
 - ***both** of the following are true:*
 - *the response is a TC type Application Cryptogram*
 - ***and** the 'Log Approved Transactions' bit in the Application Control has the value 1b,*
 - ***or both** of the following are true:*
 - *the response is an AAC type Application Cryptogram*
 - ***and** the 'Log Declined Transactions' bit in the Application Control has the value 1b*

then the application shall append to the Transaction Log the value only (omitting the tag and length) for the data elements listed in Table 15-11, in the order shown.

Data to Log	Condition
<i>Amount, Authorised</i>	<i>always</i>
<i>Transaction Currency Code</i>	<i>always</i>
<i>Transaction Date</i>	<i>always</i>
<i>CVR</i>	<i>if 'Log the CVR' in Application Control=1b</i>
<i>ATC</i>	<i>if 'Log the ATC' in Application Control=1b</i>
<i>CID</i>	<i>if 'Log the CID' in Application Control=1b</i>
<i>Data extracted from the First GENERATE AC Command Data using the First GEN AC Unchanging Log Data Table</i>	<i>if any</i>
<i>Data extracted from the First GENERATE AC Command Data using the First GEN AC Log Data Table</i>	<i>if any</i>

Table 15-11: Data Appended to Transaction Log

15.5.8.4 Return GENERATE AC Response

If **both** of the following are true:

- CDA processing is requested by the terminal
- **and** the application is responding with either an ARQC or a TC type Application Cryptogram

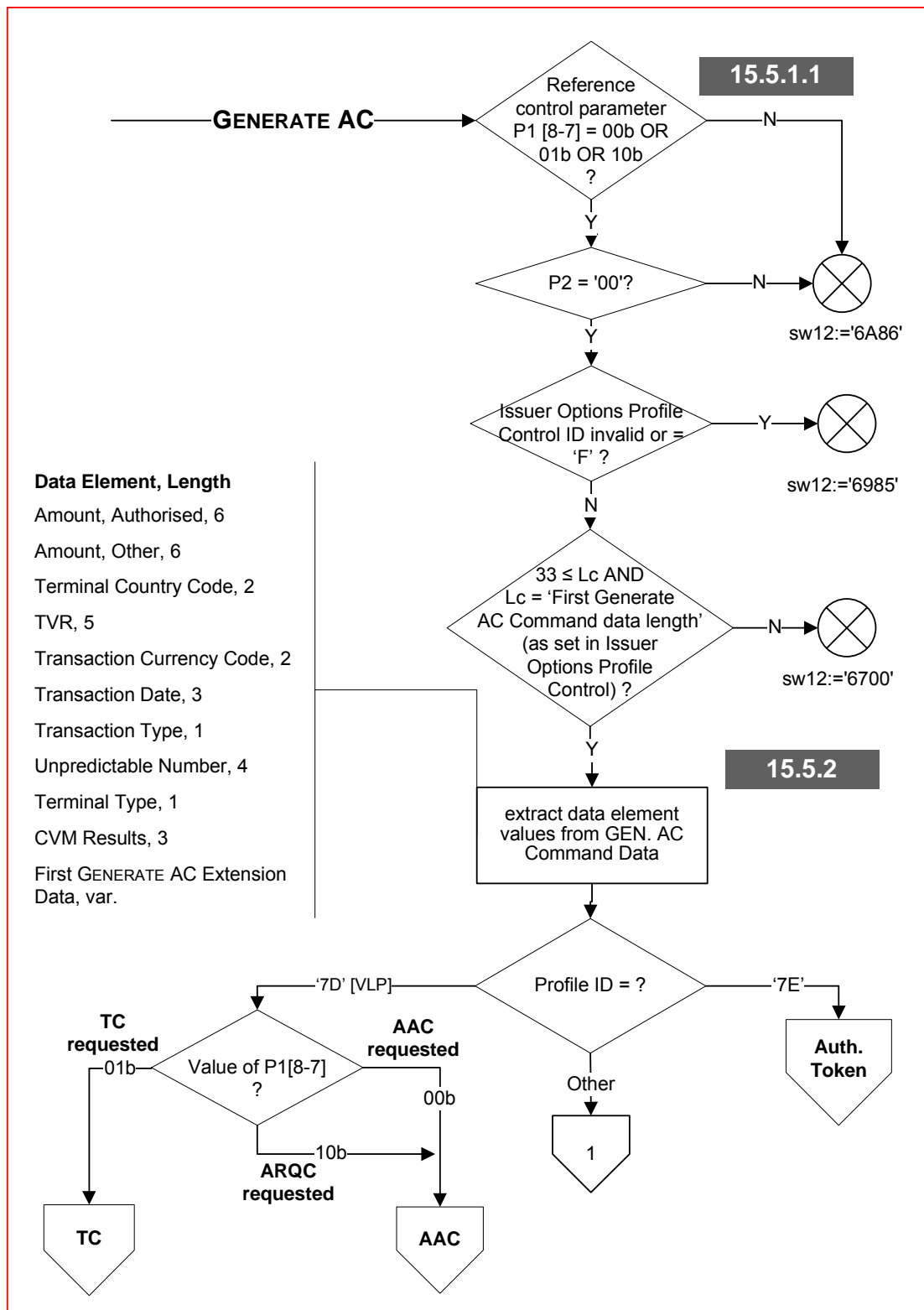
then the application:

- generates a dynamic signature from the Application Cryptogram as described in *EMV Book 2*, section 6.6.1
- returns the first GENERATE AC response as described in the CCD part of *EMV Book 2*, section 6.6.1 and Table CCD-1.

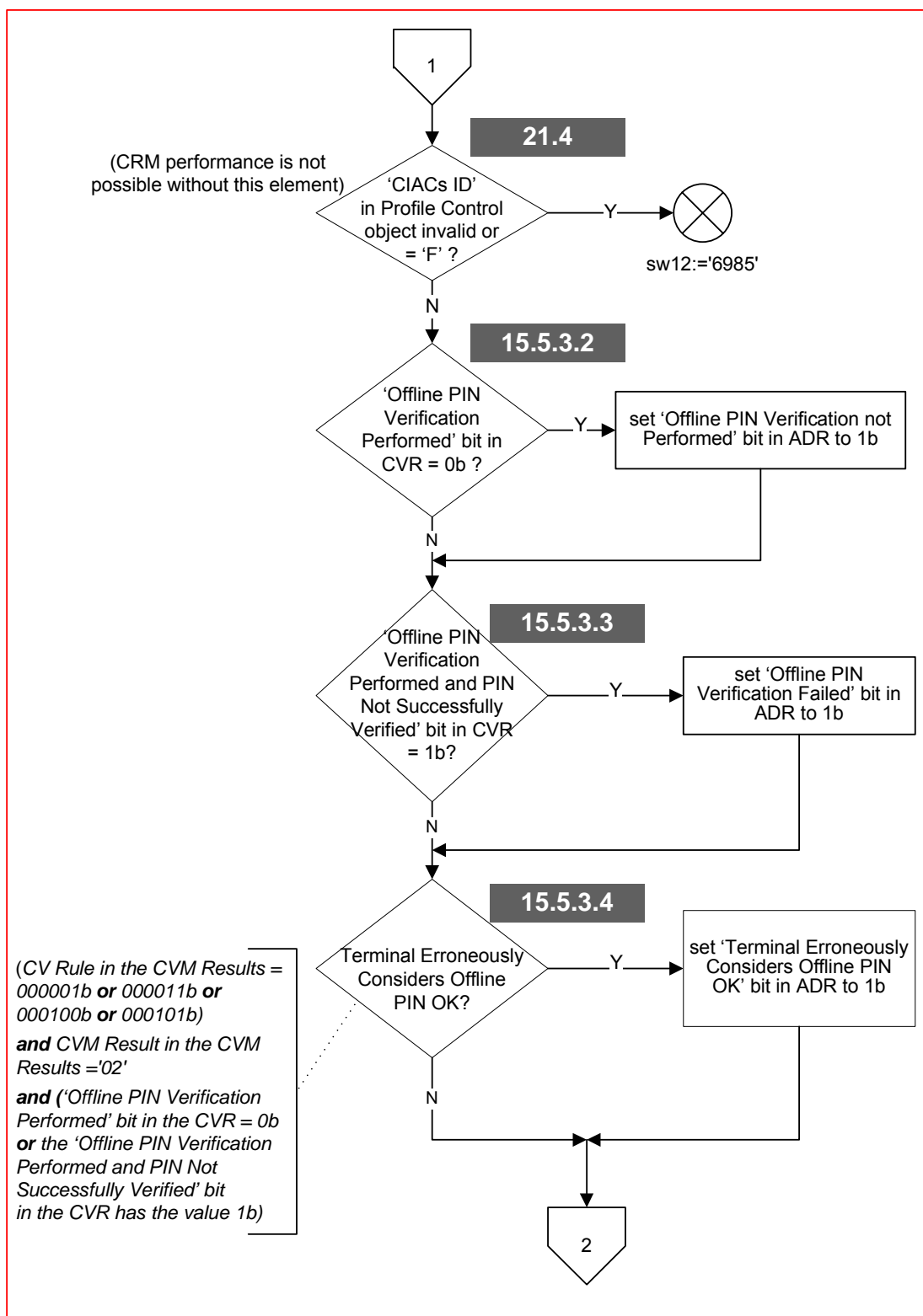
Otherwise, the application returns the first GENERATE AC response as described in the CCD part of *EMV Book 3*, section 6.5.5.4 and Table CCD-2.

15.6 Function Flow Charts

The following flow shows how an application could perform First Card Action Analysis processing.

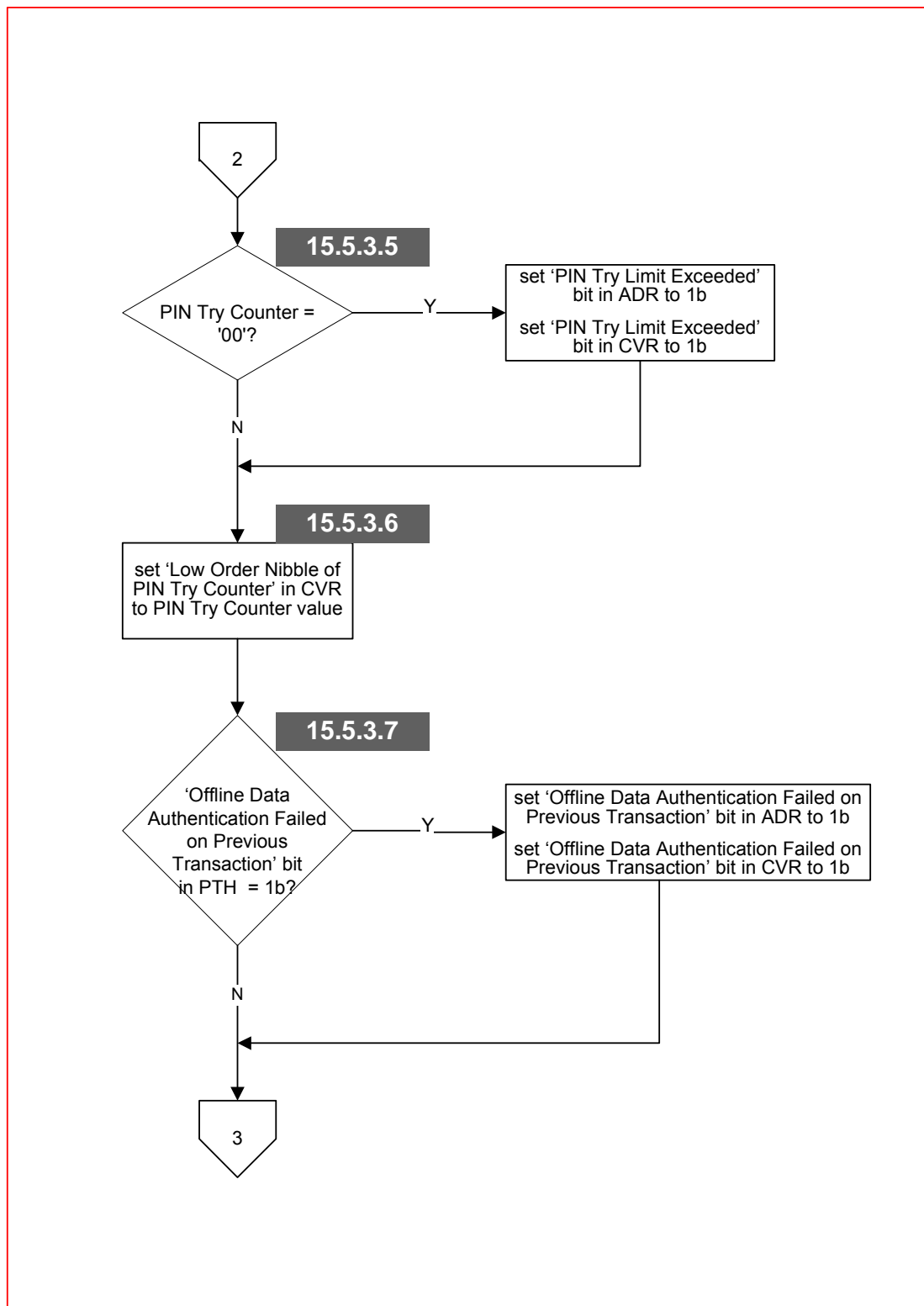


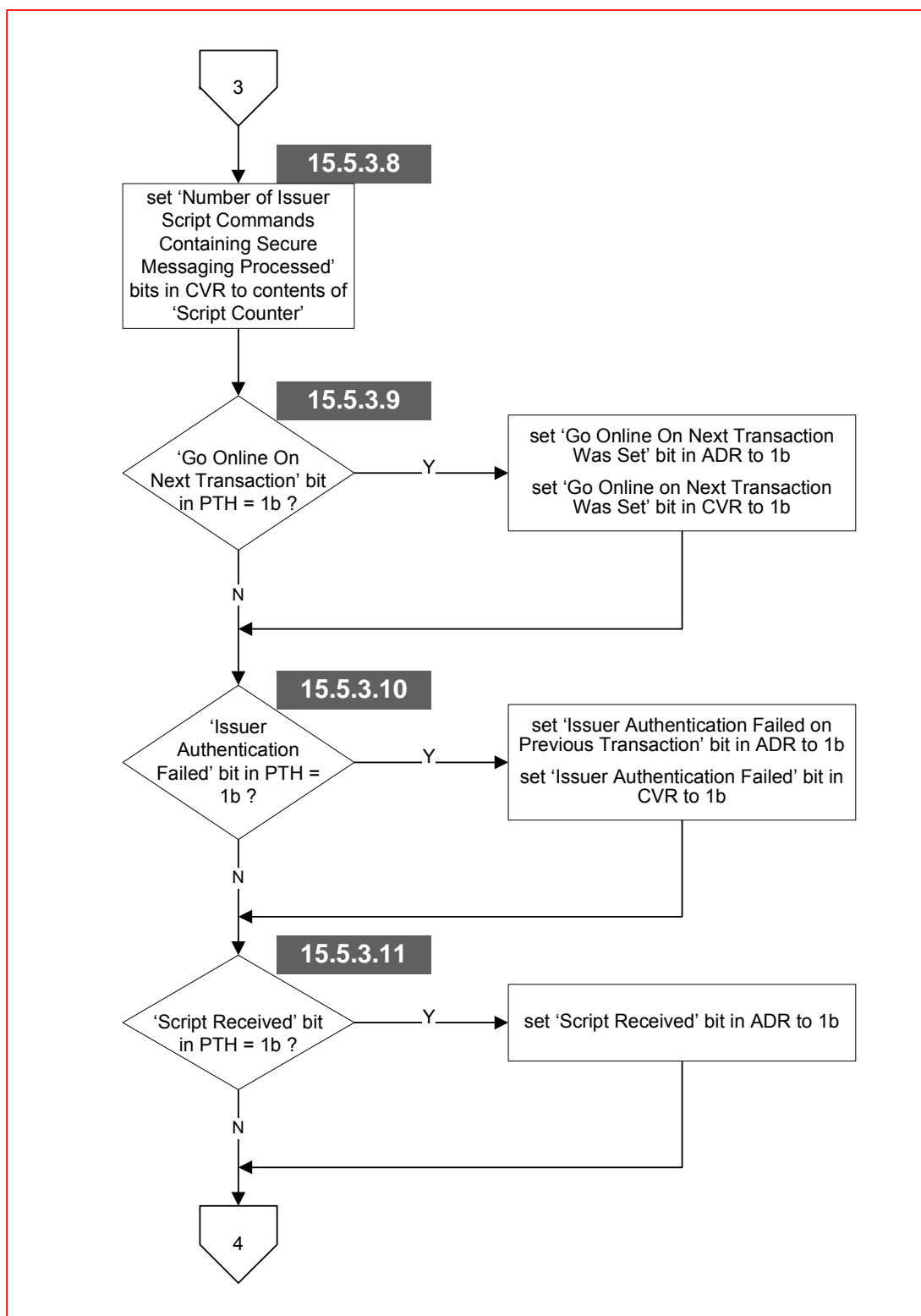
Flow 15-1 First GENERATE AC Initial Flow



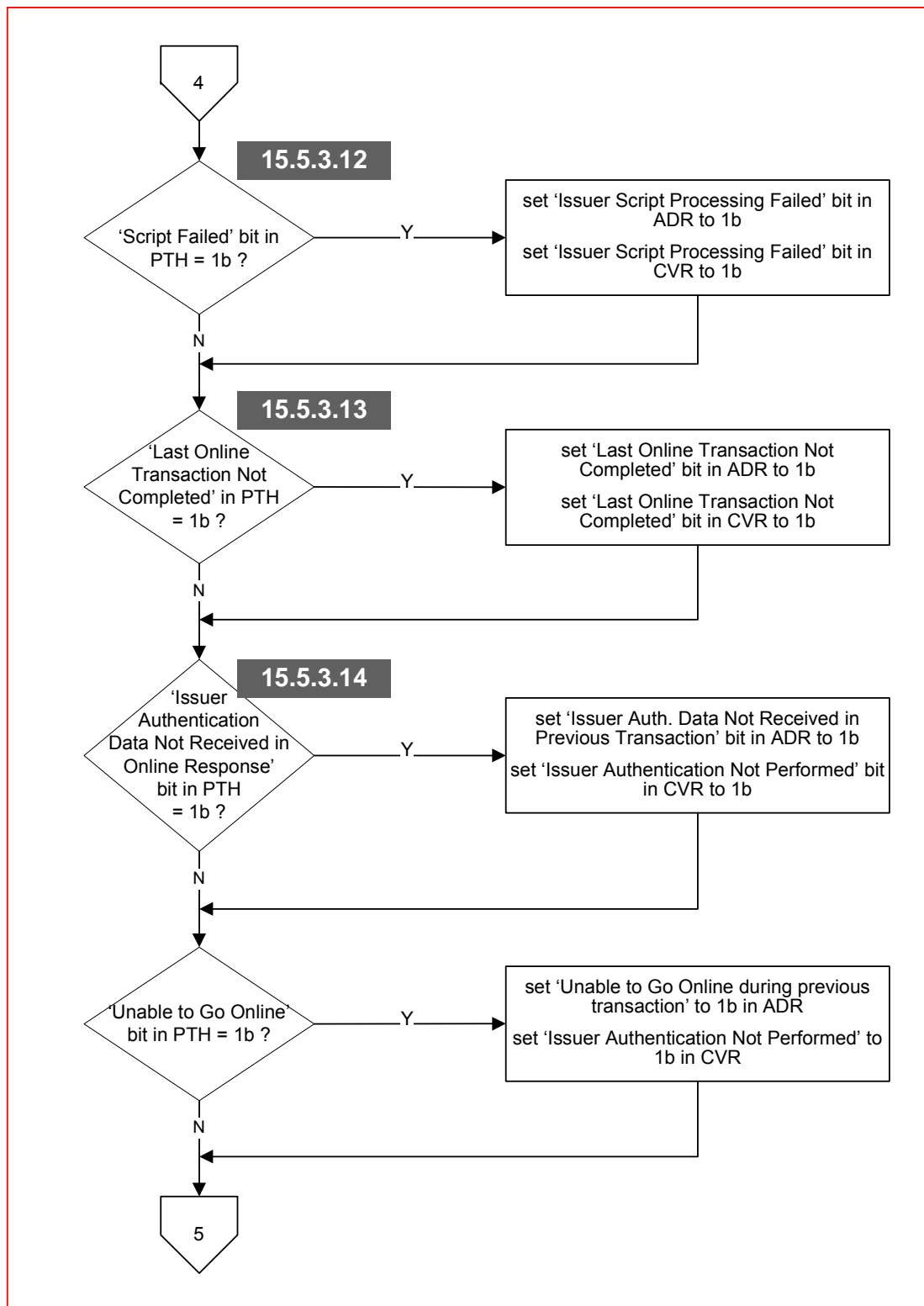
Flow 15-1.1

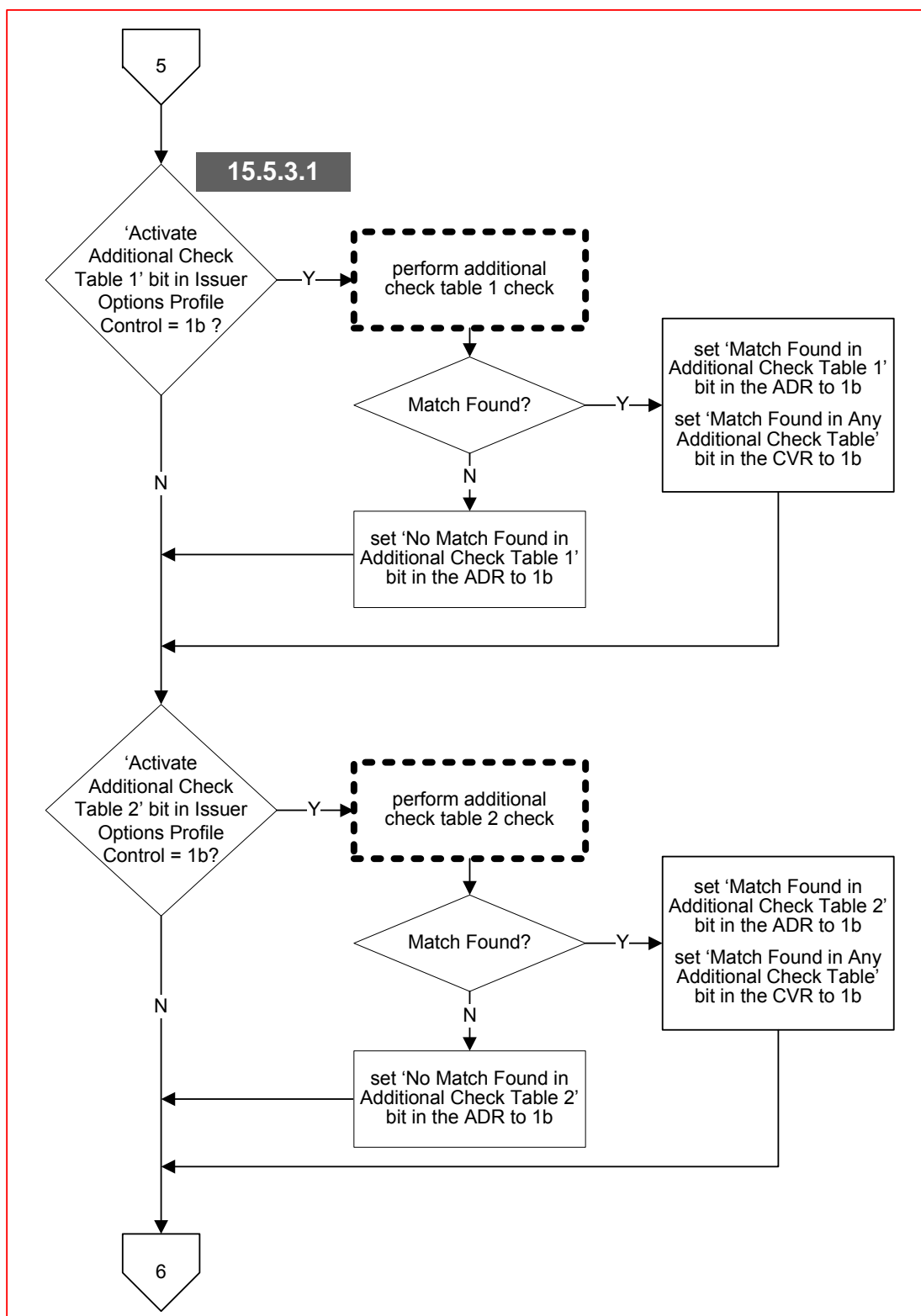
First GENERATE AC Initial Flow, continued

**Flow 15-1.2 First GENERATE AC Initial Flow, continued**

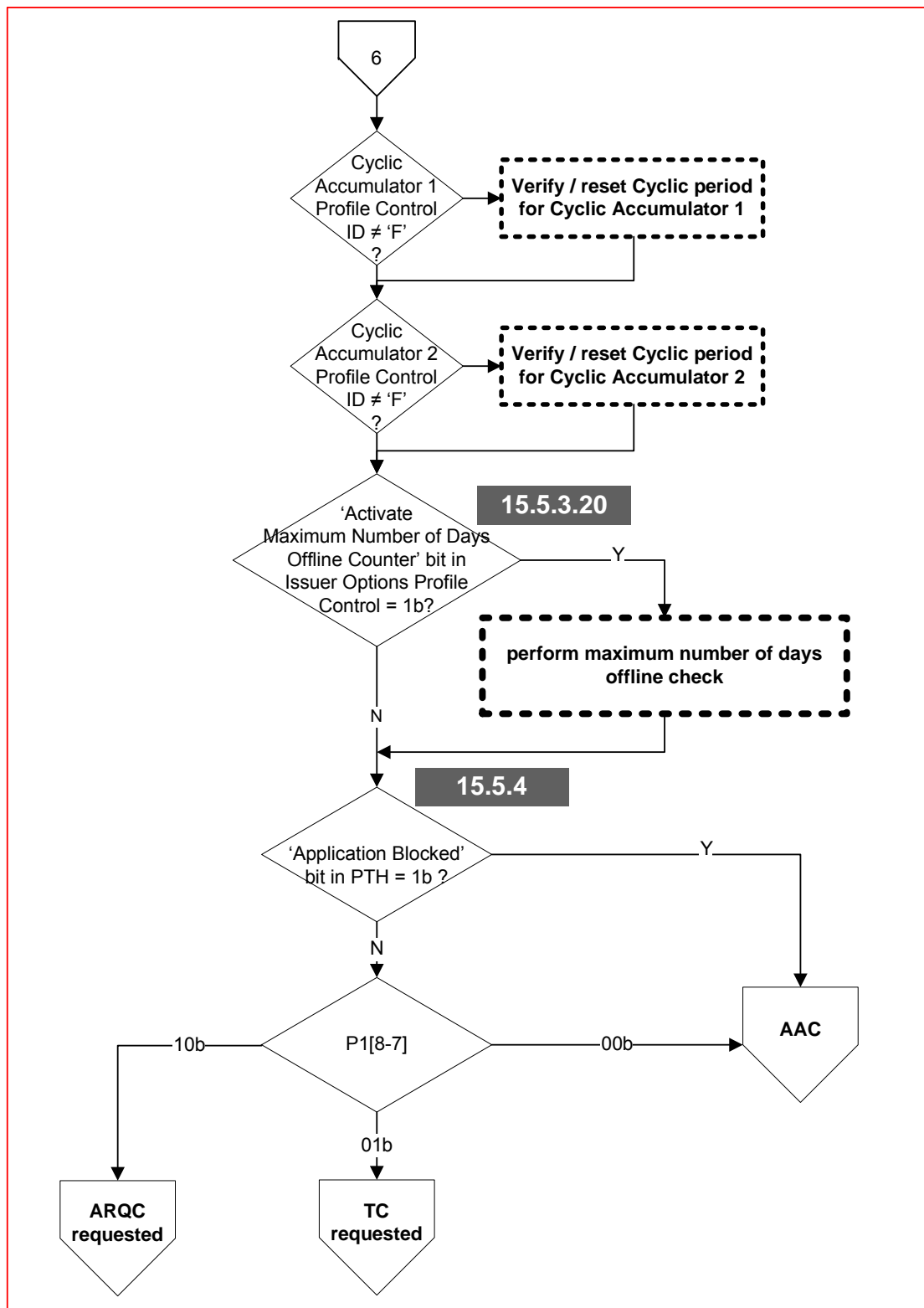


Flow 15-1.3 First GENERATE AC Initial Flow, continued

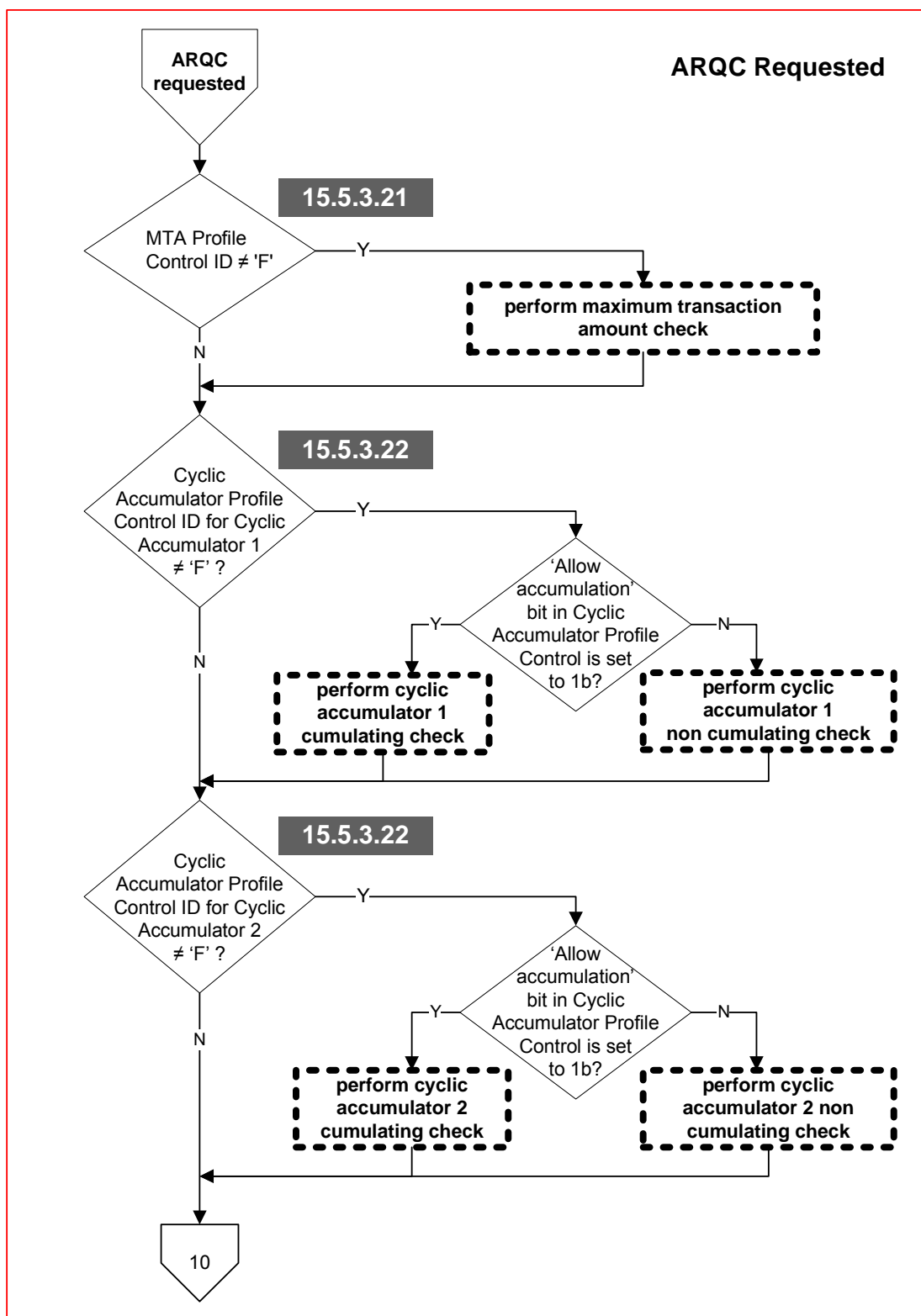
**Flow 15-1.4 First GENERATE AC Initial Flow, continued**



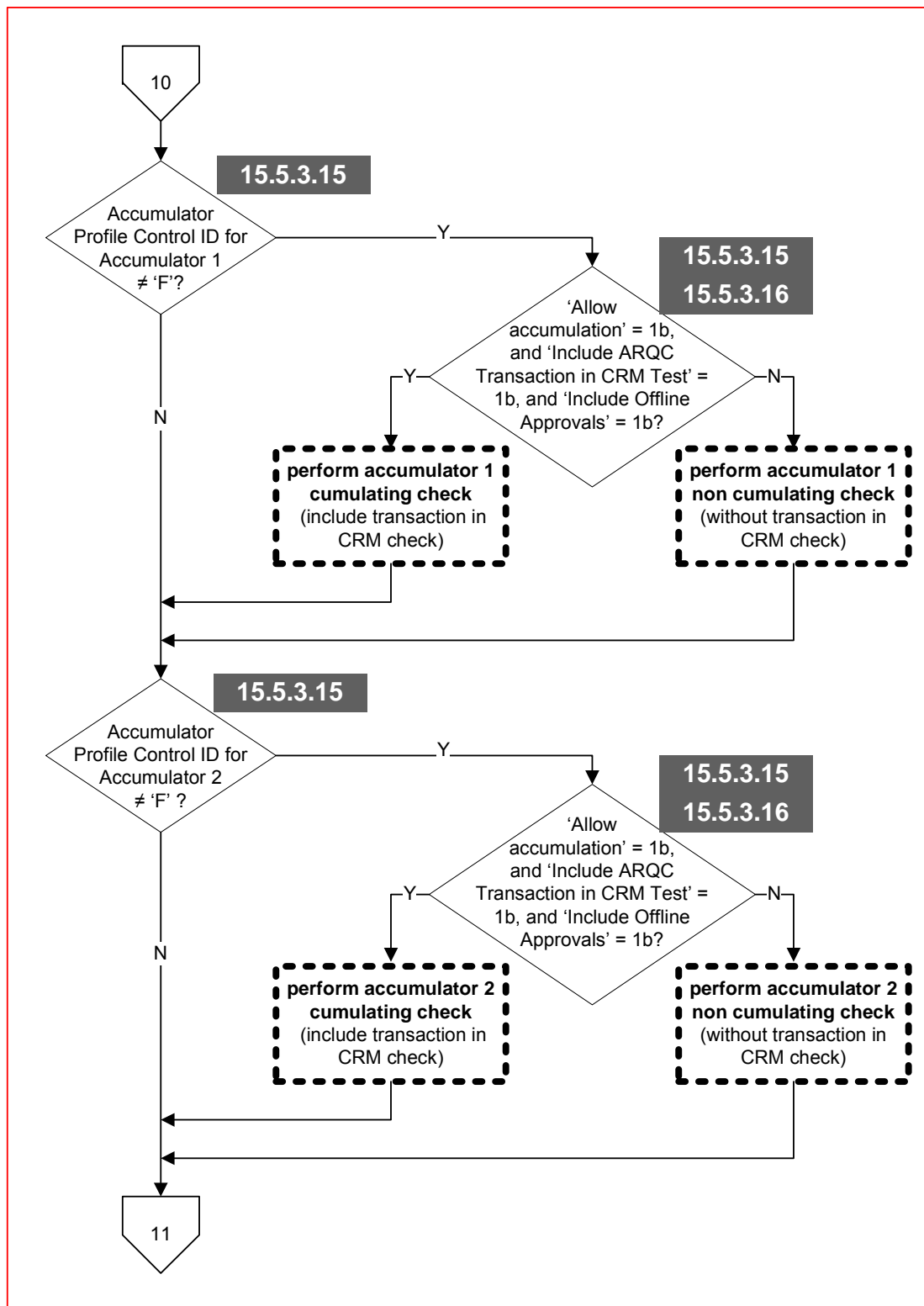
Flow 15-1.5 First GENERATE AC Initial Flow, continued



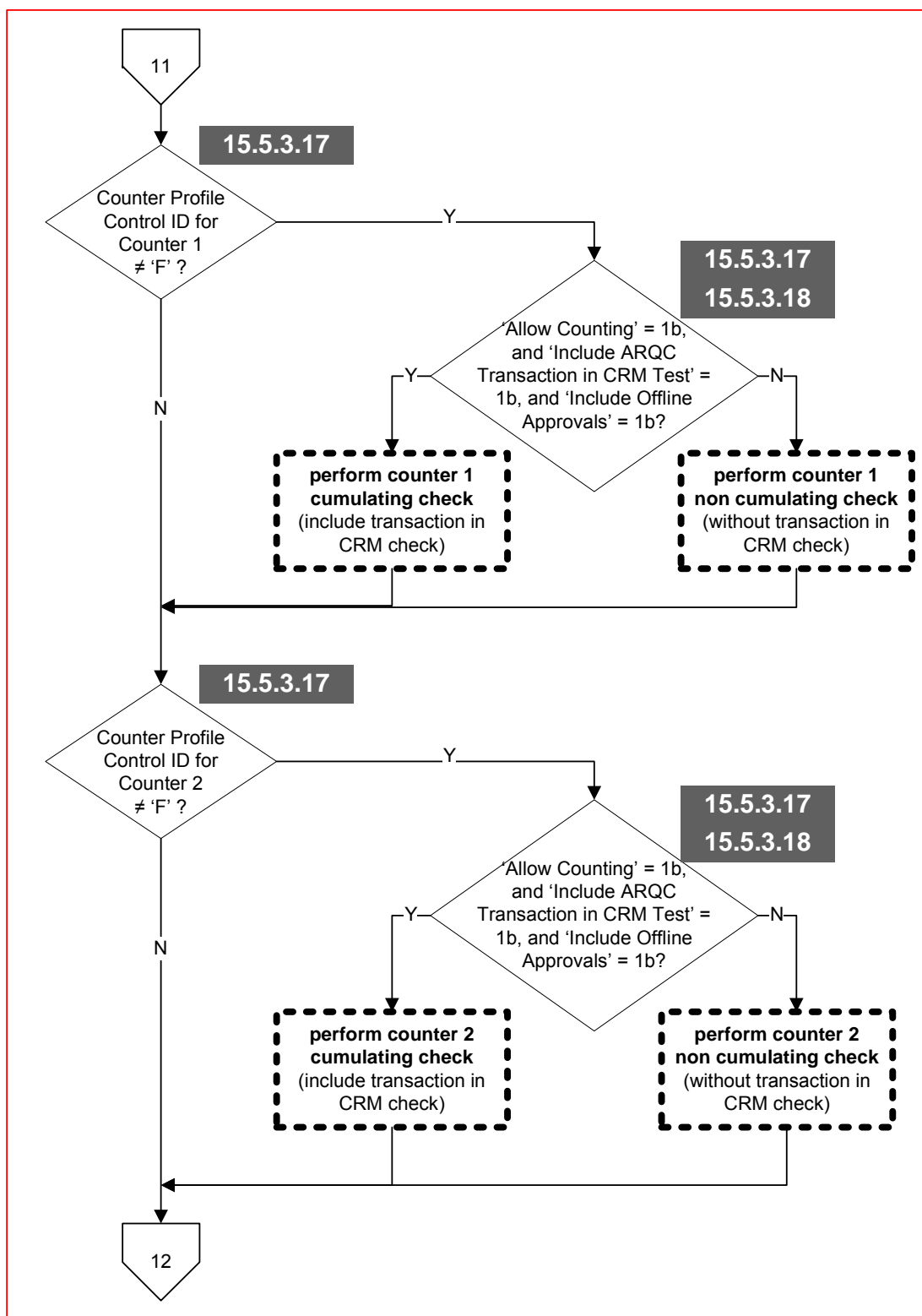
Flow 15-1.6 First GENERATE AC Initial Flow, continued



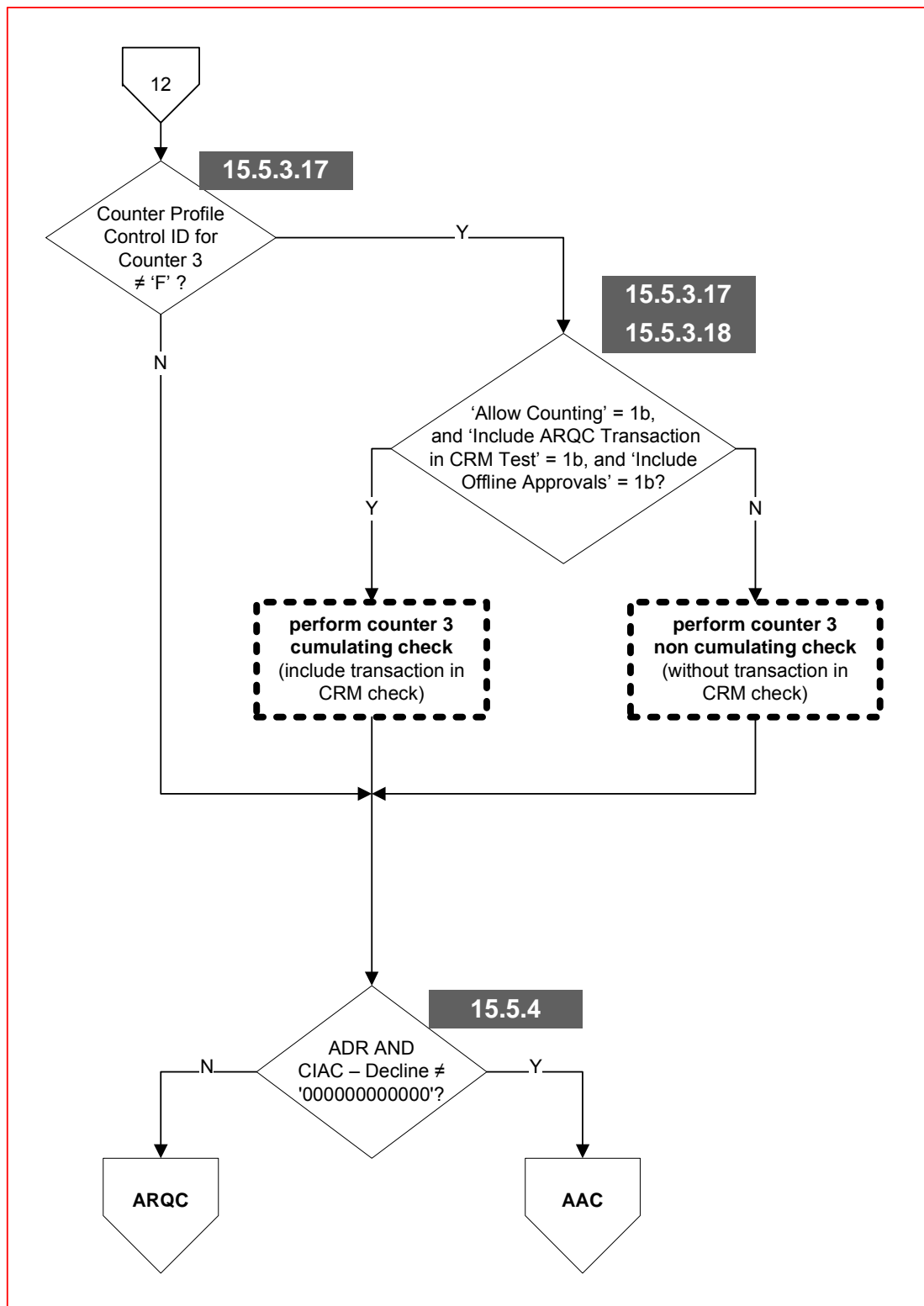
Flow 15-2 ARQC Requested



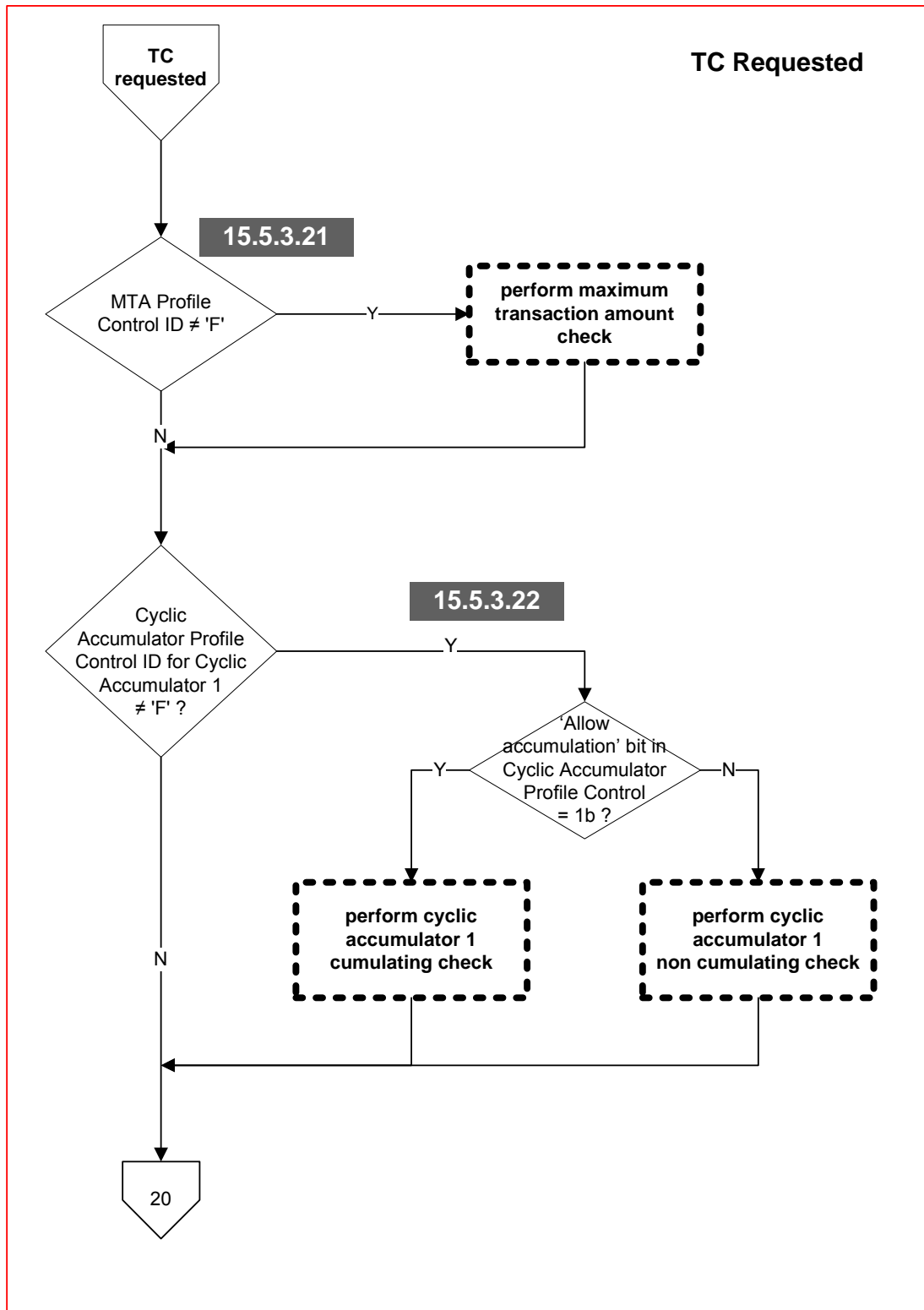
Flow 15-2.1 ARQC Requested, continued



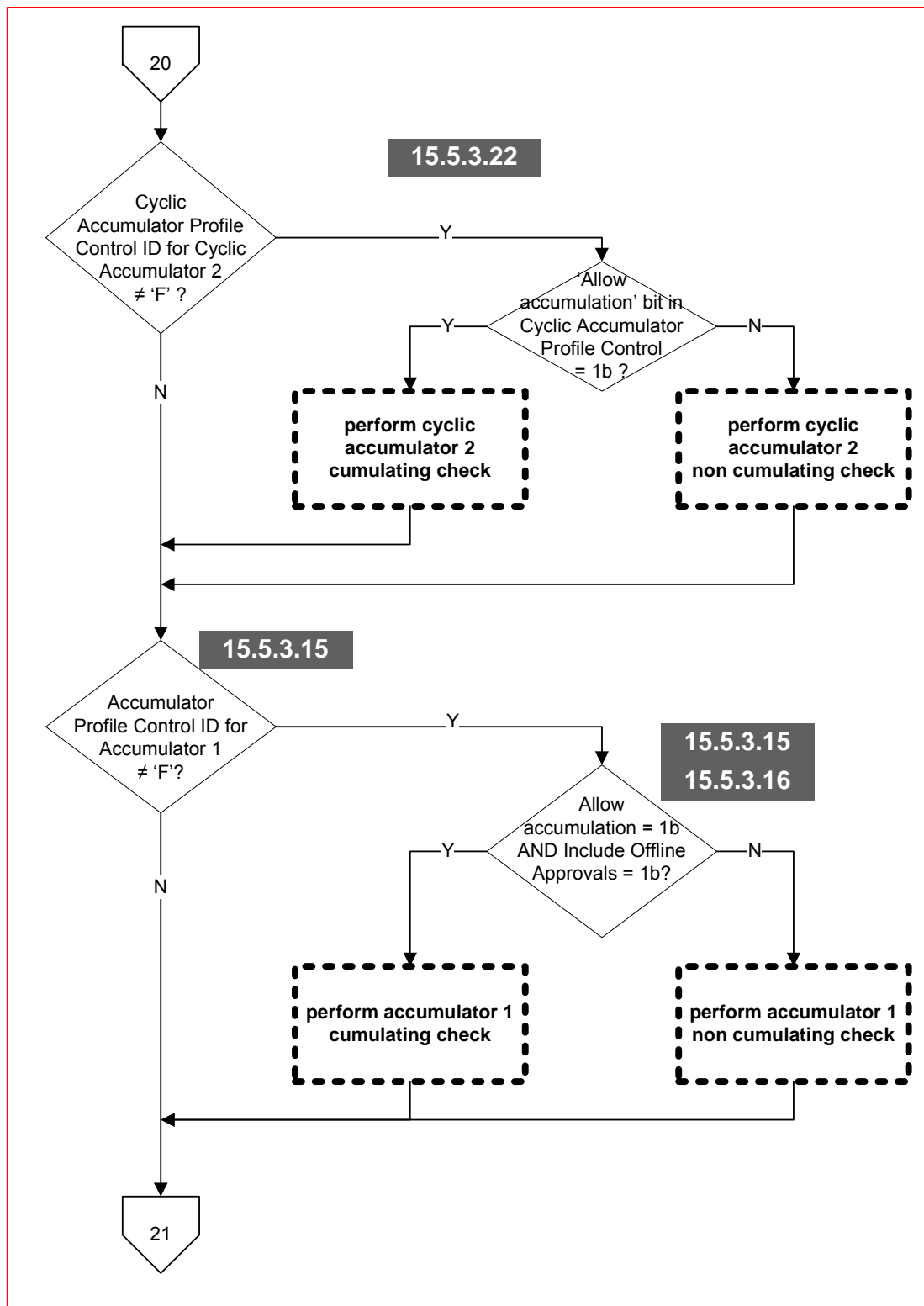
Flow 15-2.2 ARQC Requested, continued



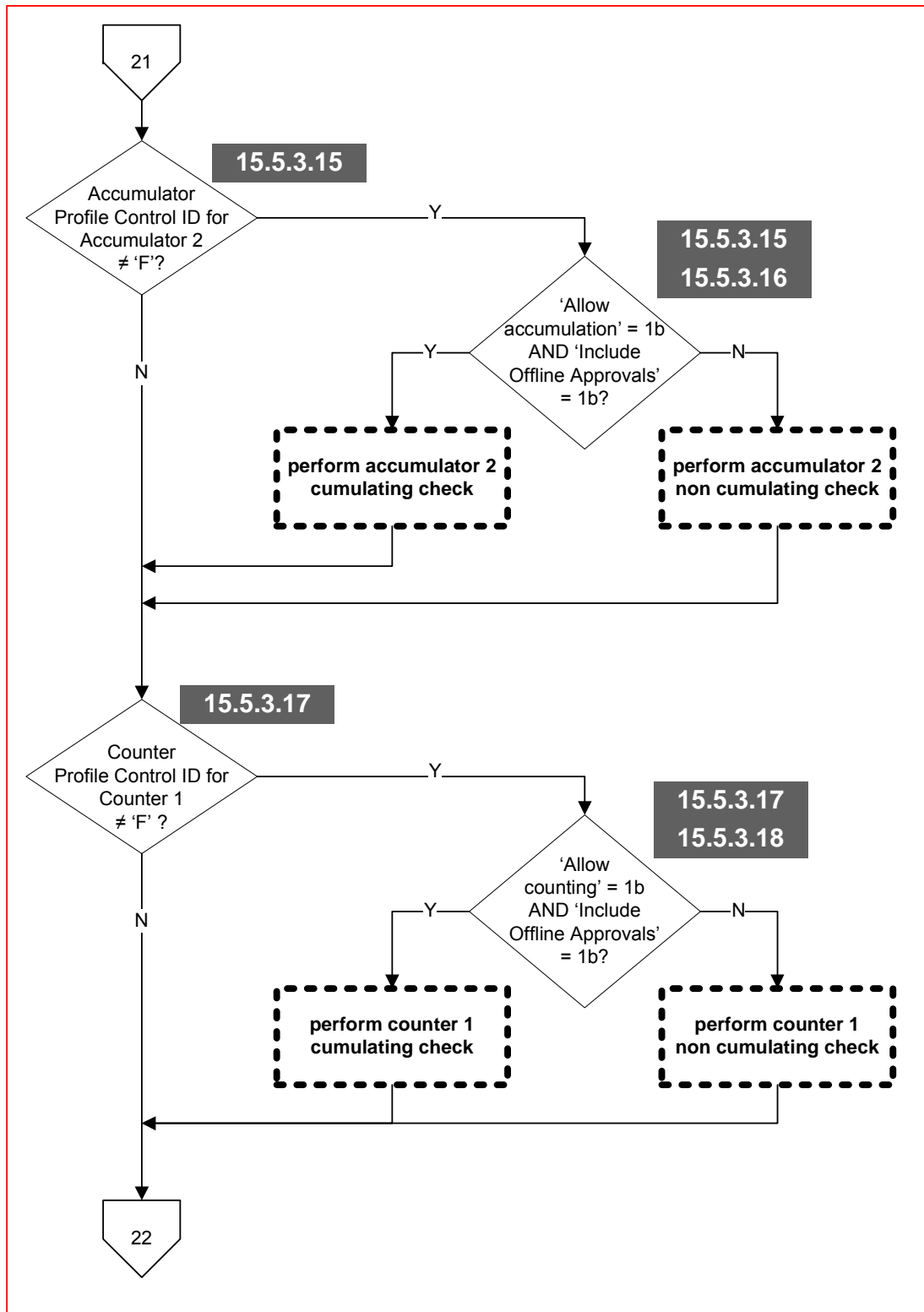
Flow 15-2.3 ARQC Requested, continued



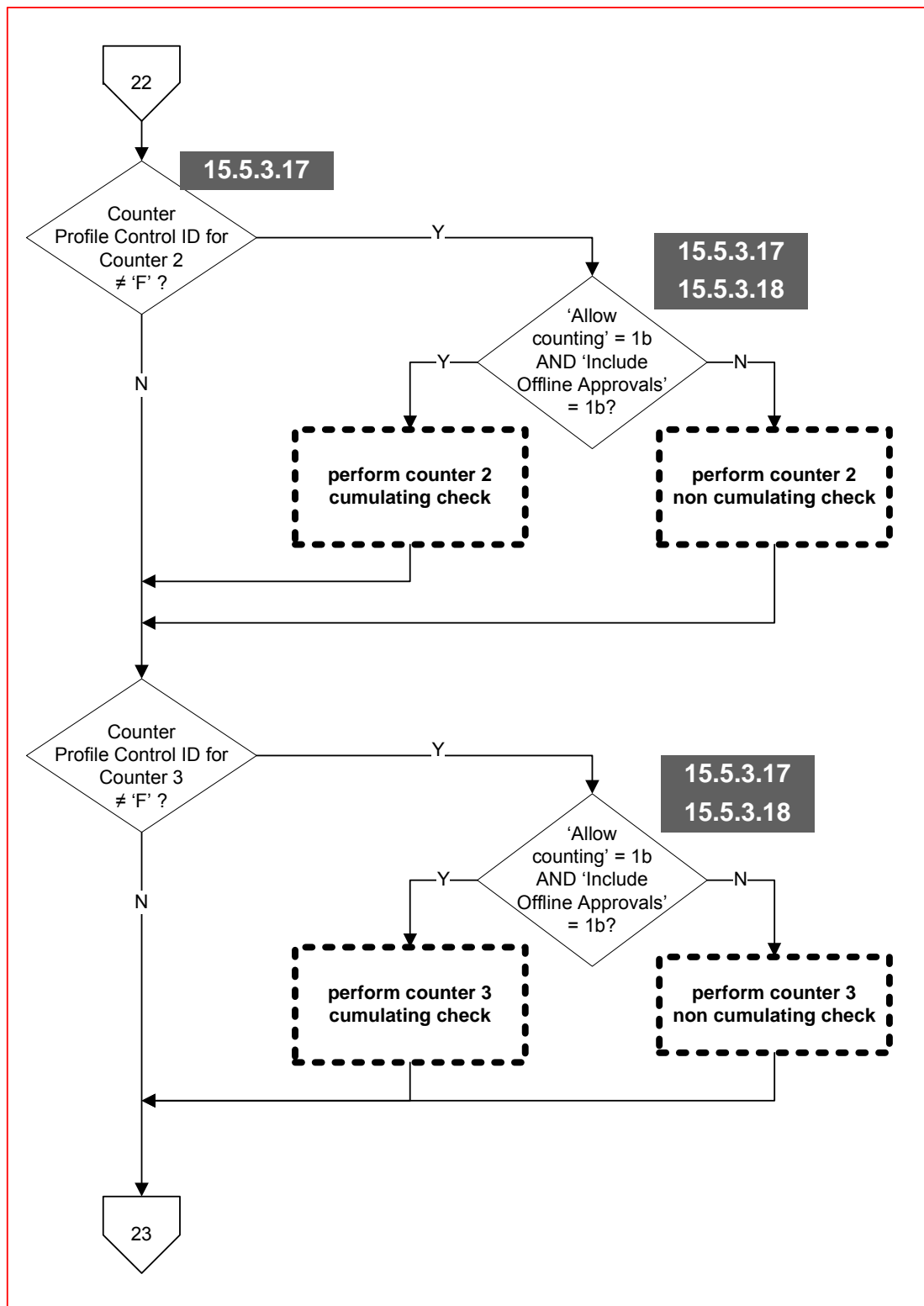
Flow 15-3 TC Requested



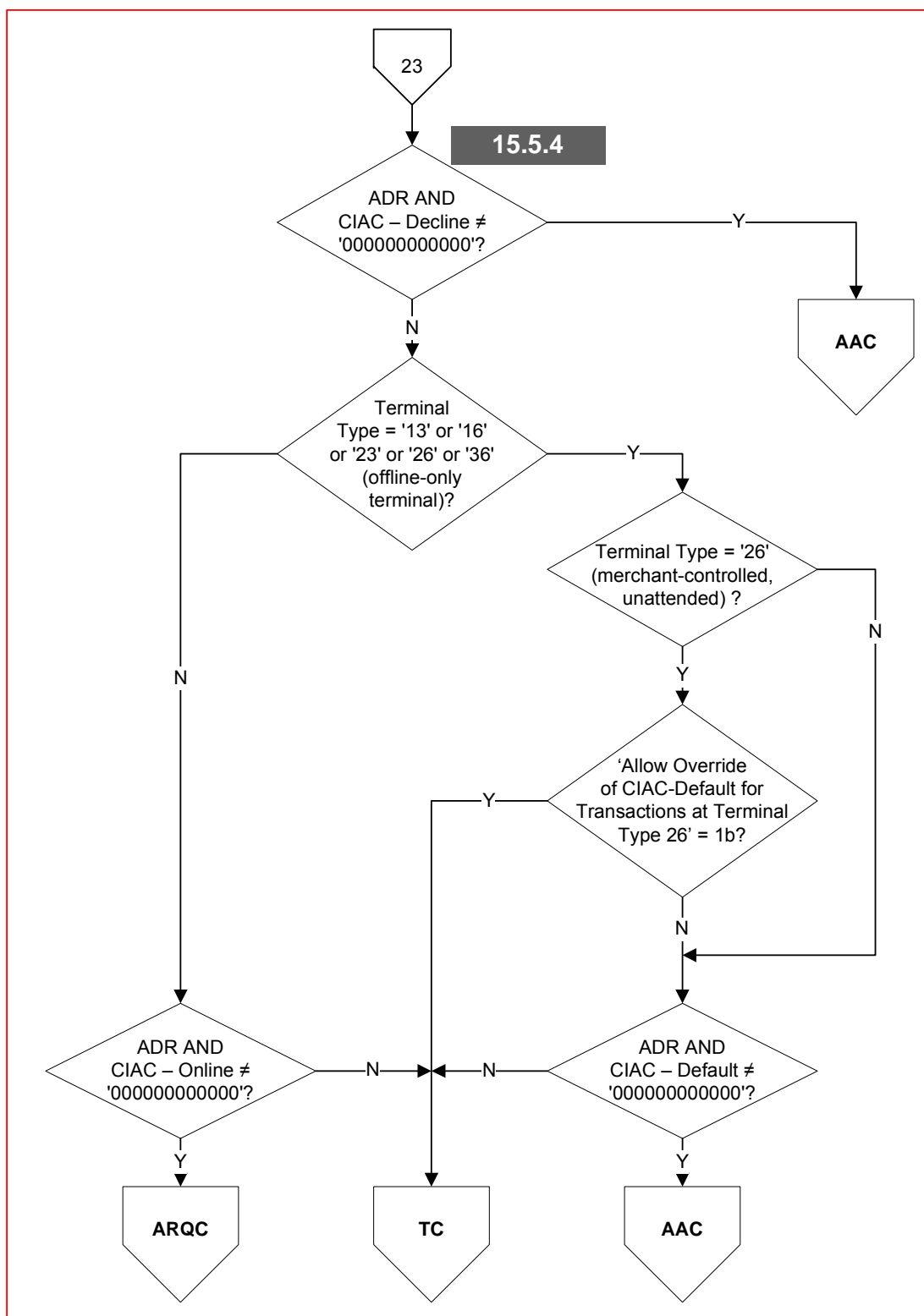
Flow 15-3.1 TC Requested, continued



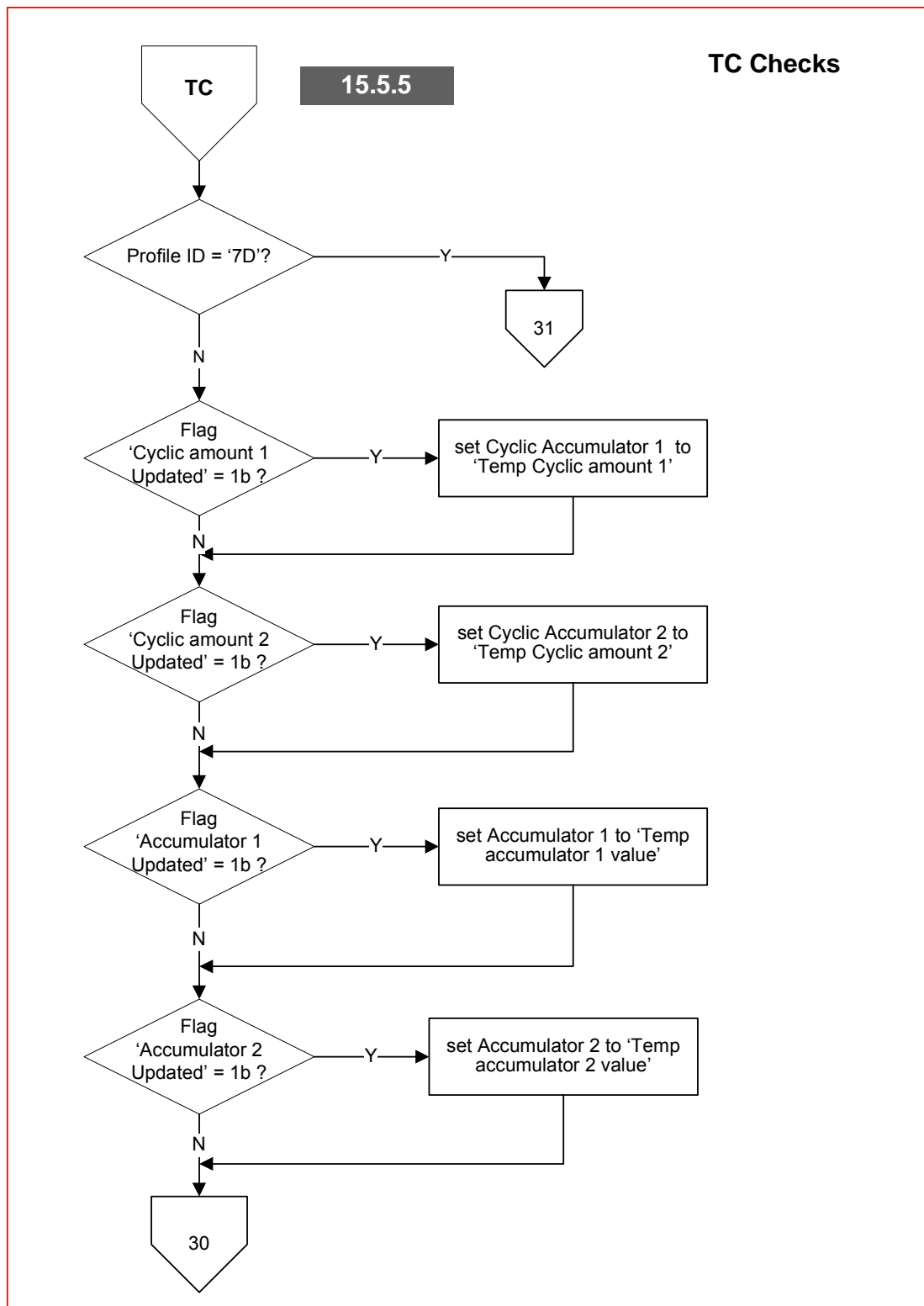
Flow 15-3.2 TC Requested, continued

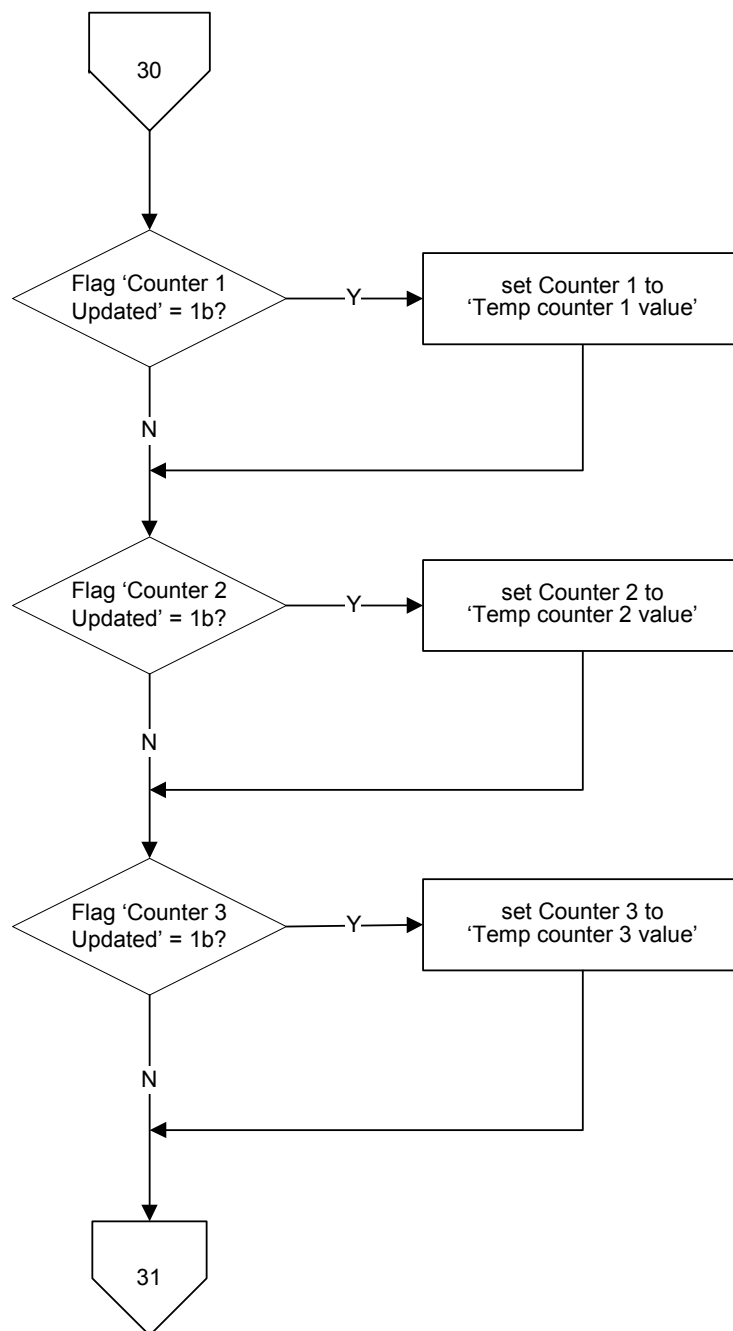


Flow 15-3.3 TC Requested, continued

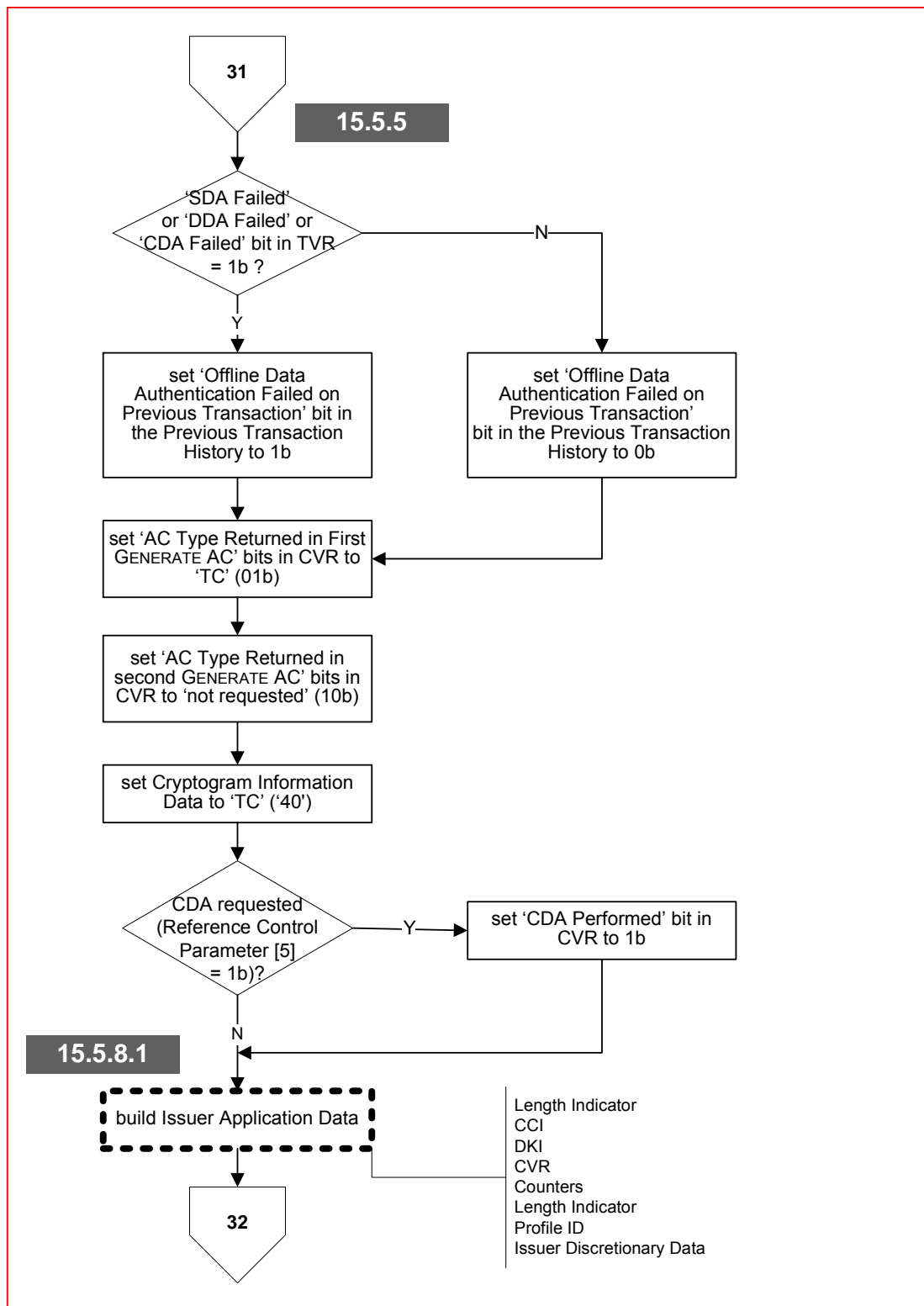


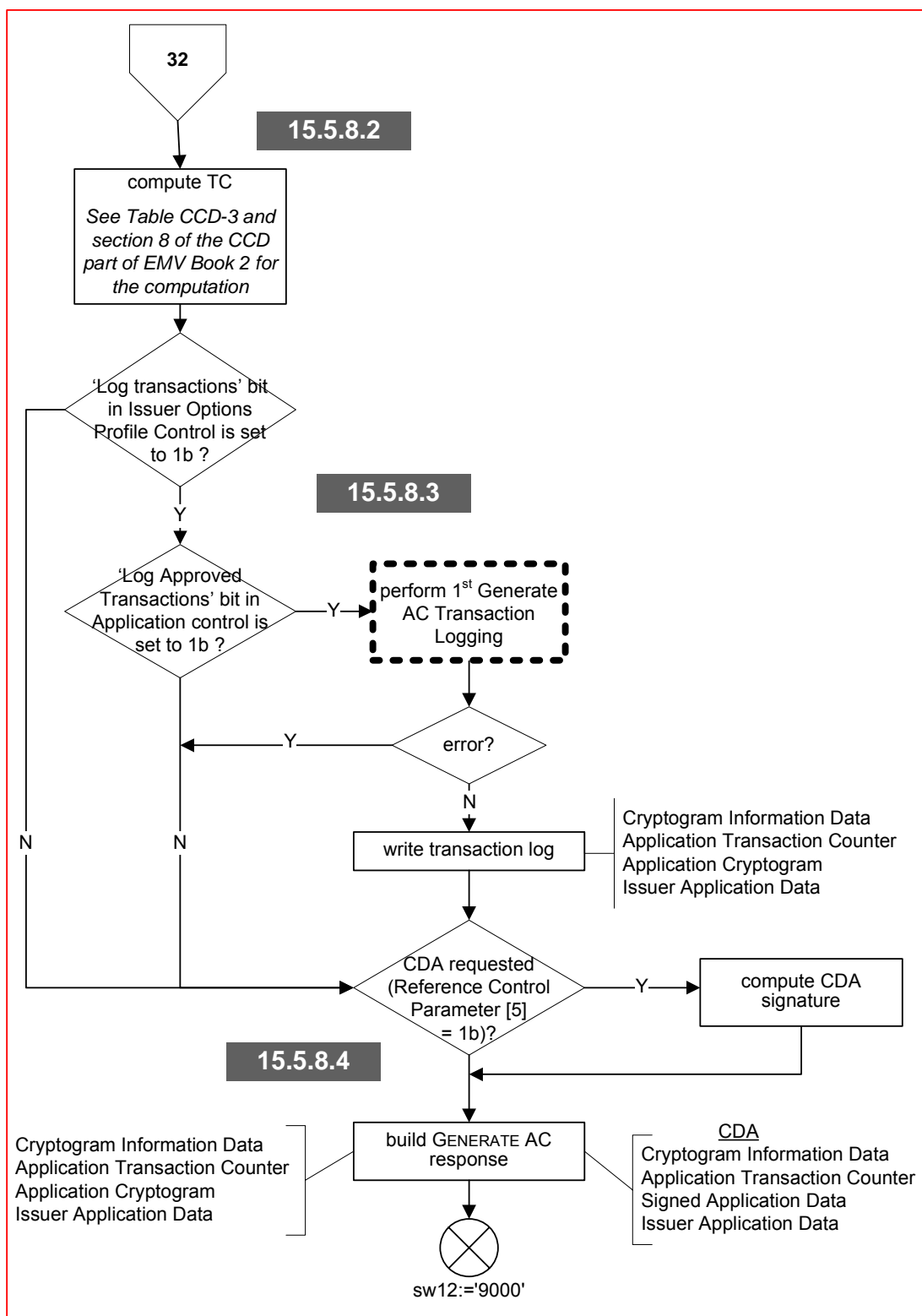
Flow 15-3.4 TC Requested, continued

**Flow 15-4 TC Checks**

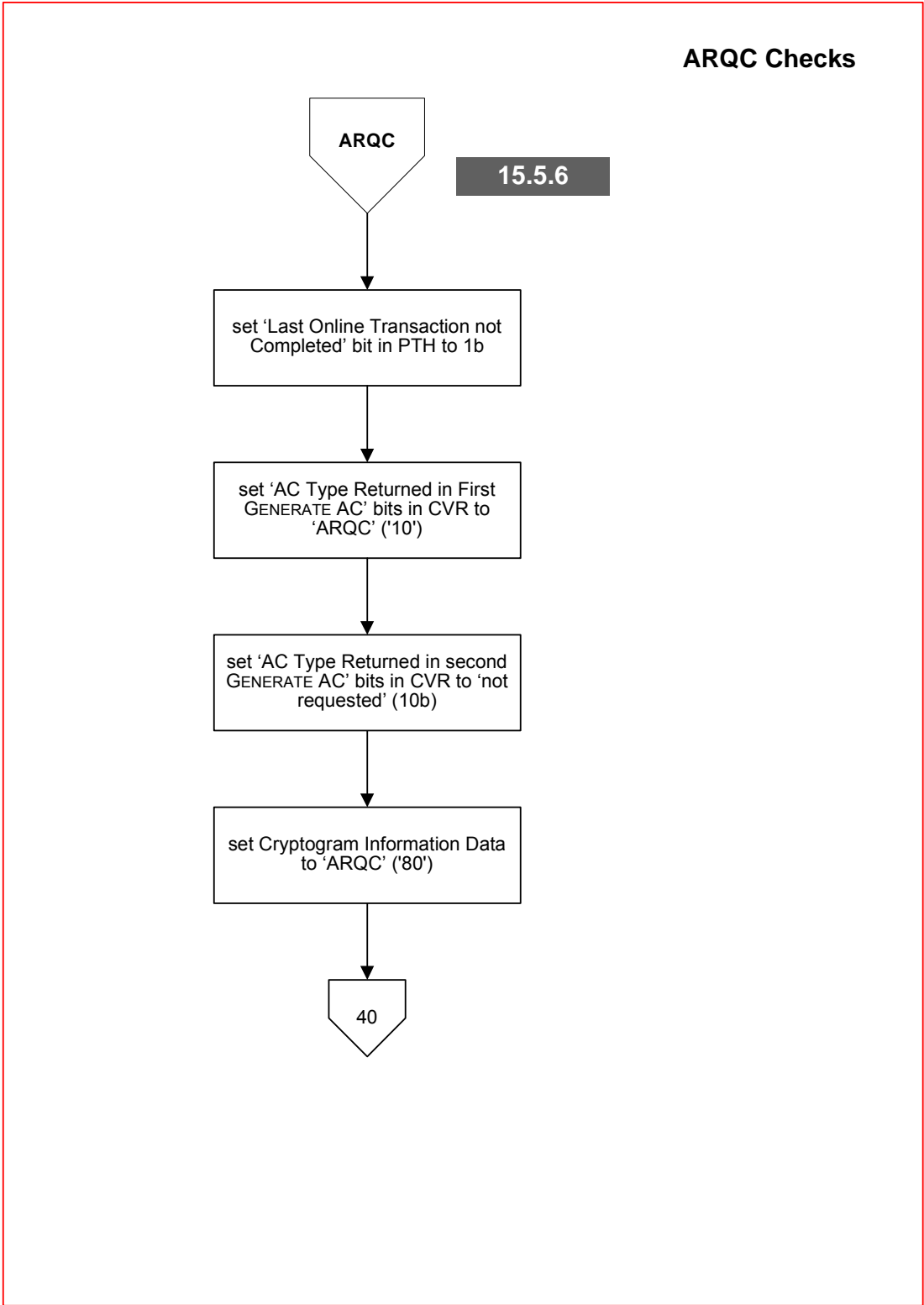


Flow 15-4.1 TC Checks, continued

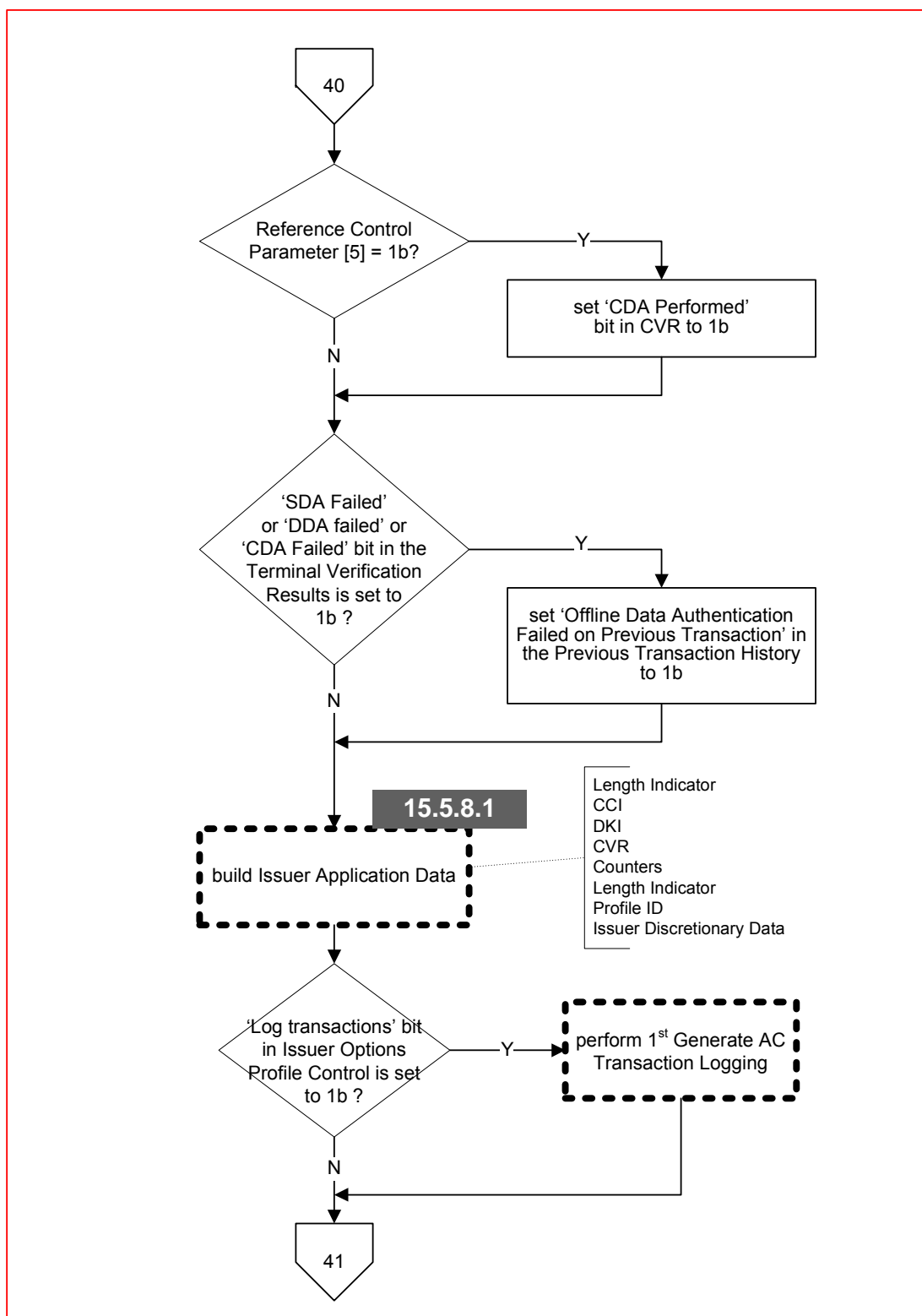
**Flow 15-4.2 TC Checks, continued**



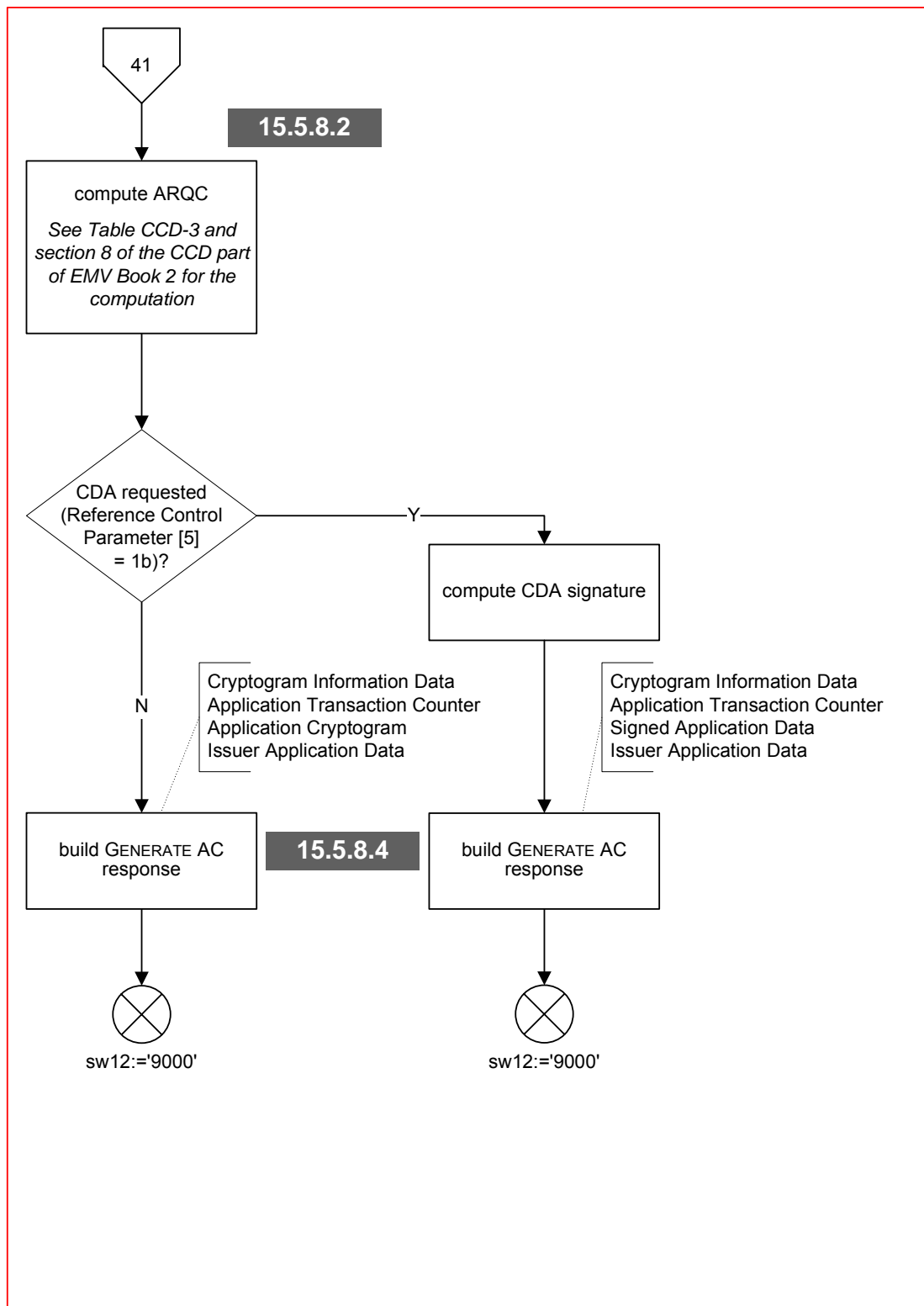
Flow 15-4.3 TC Checks, continued

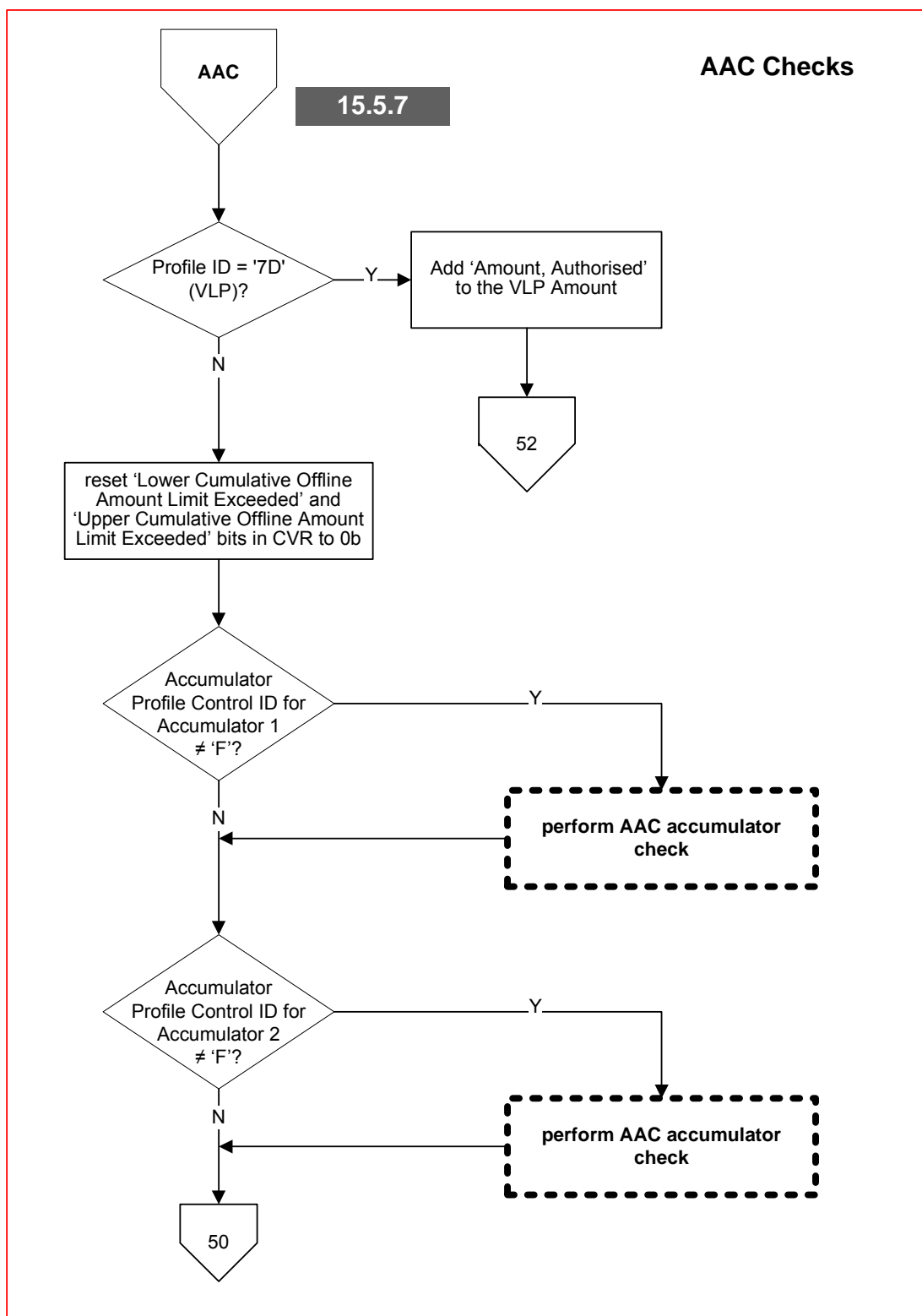


Flow 15-5 ARQC Checks

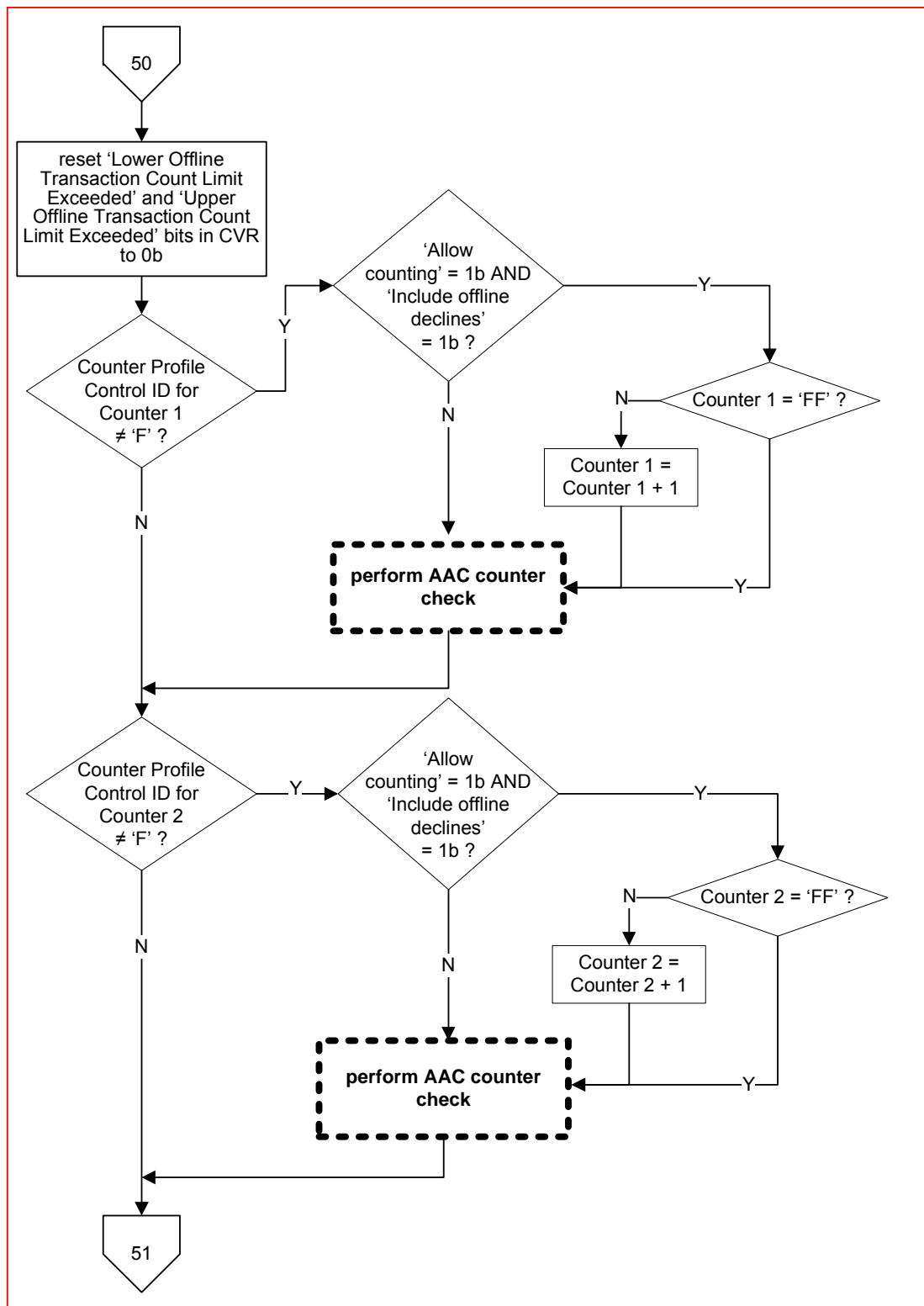


Flow 15-5.1 ARQC Checks, continued

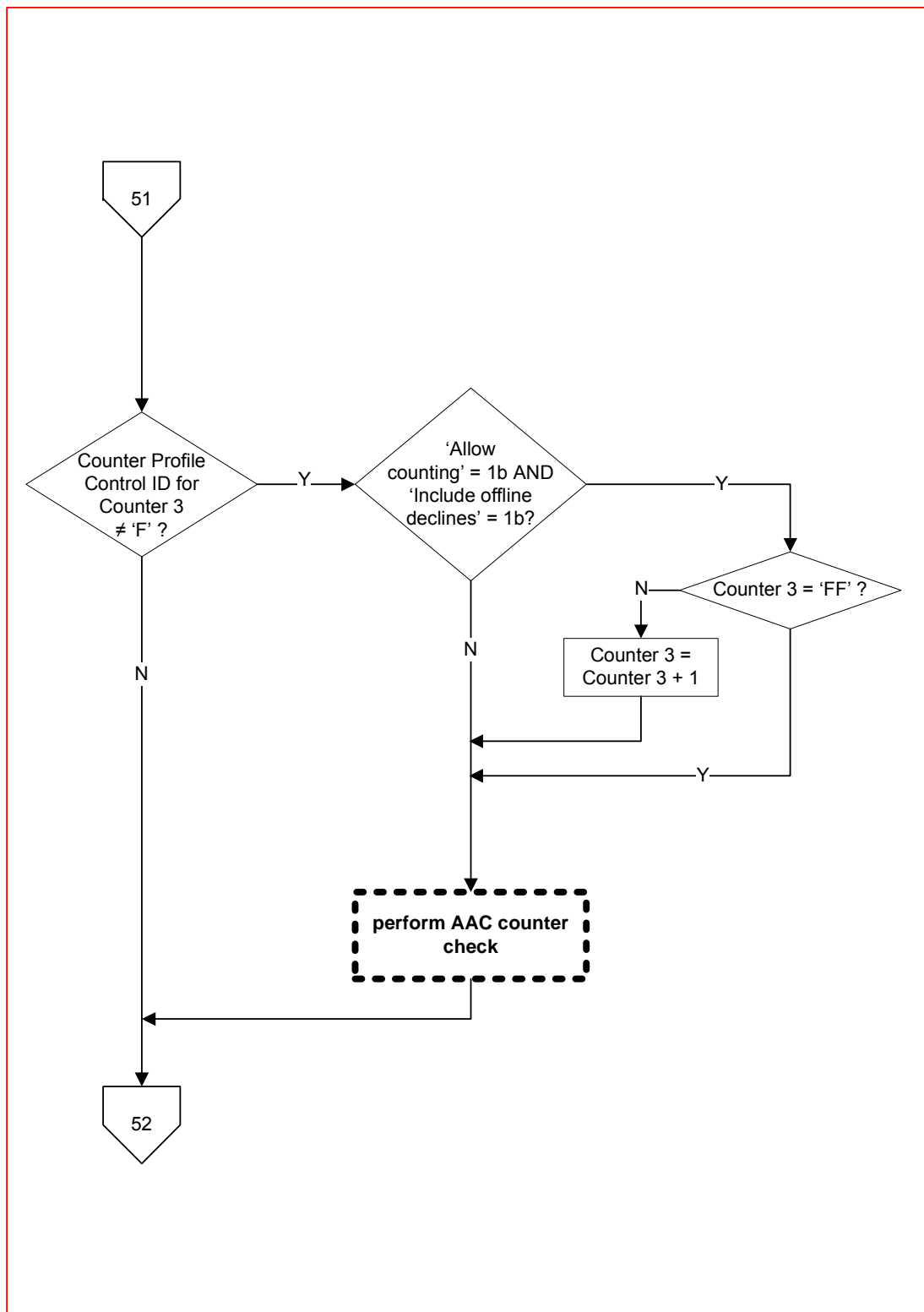
**Flow 15-5.2 ARQC Checks, continued**



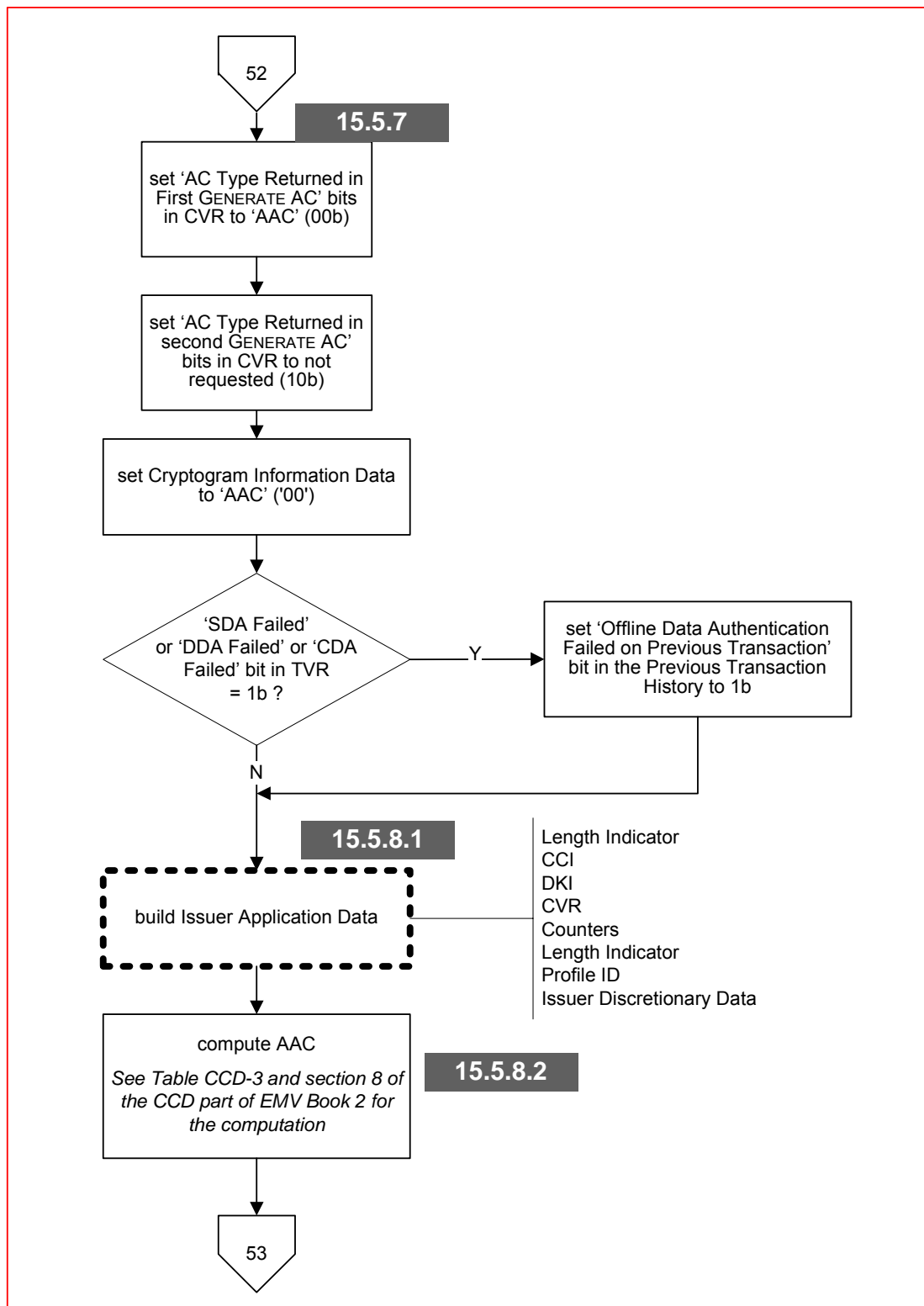
Flow 15-6 AAC Checks



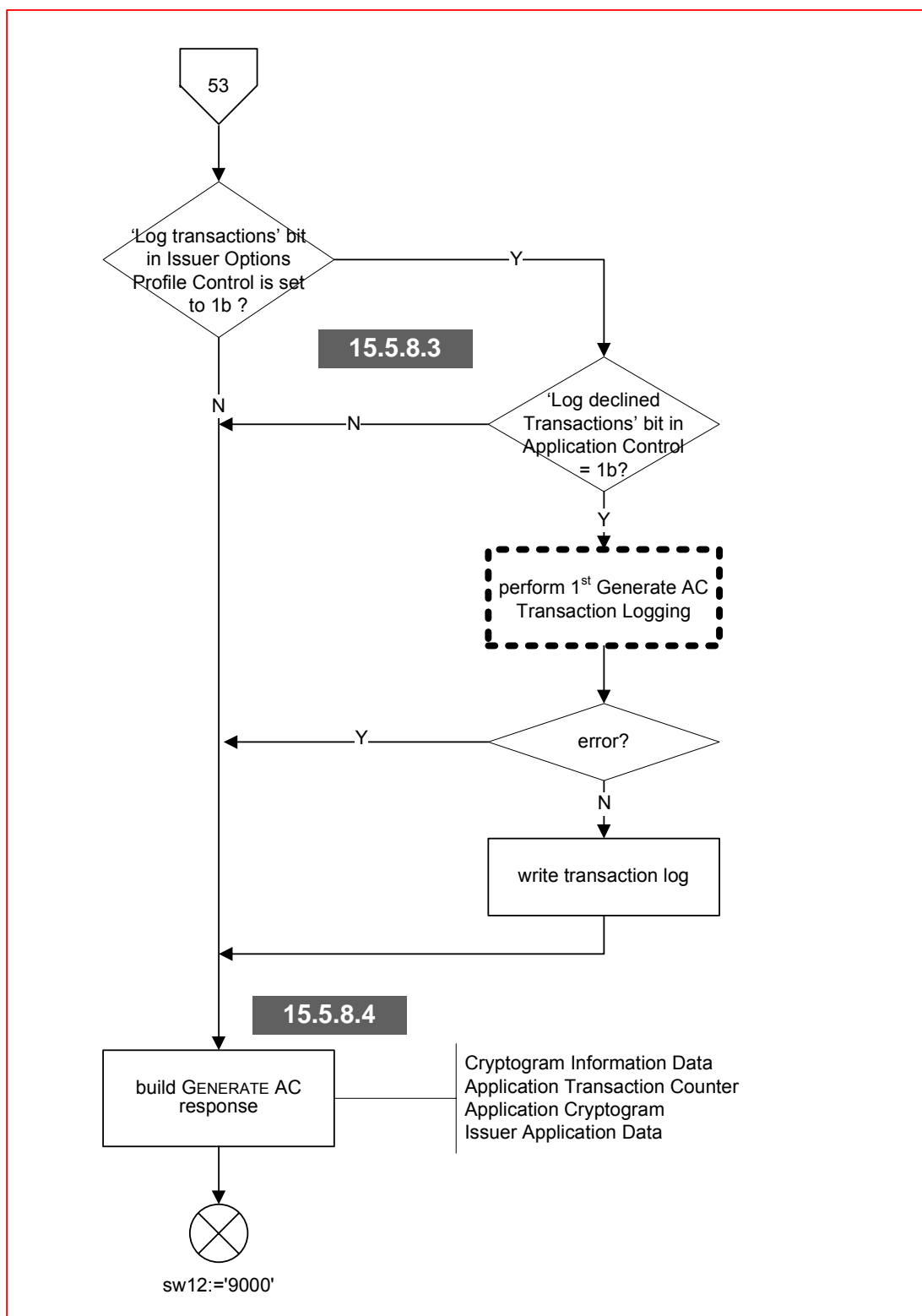
Flow 15-6.1 AAC Checks, continued



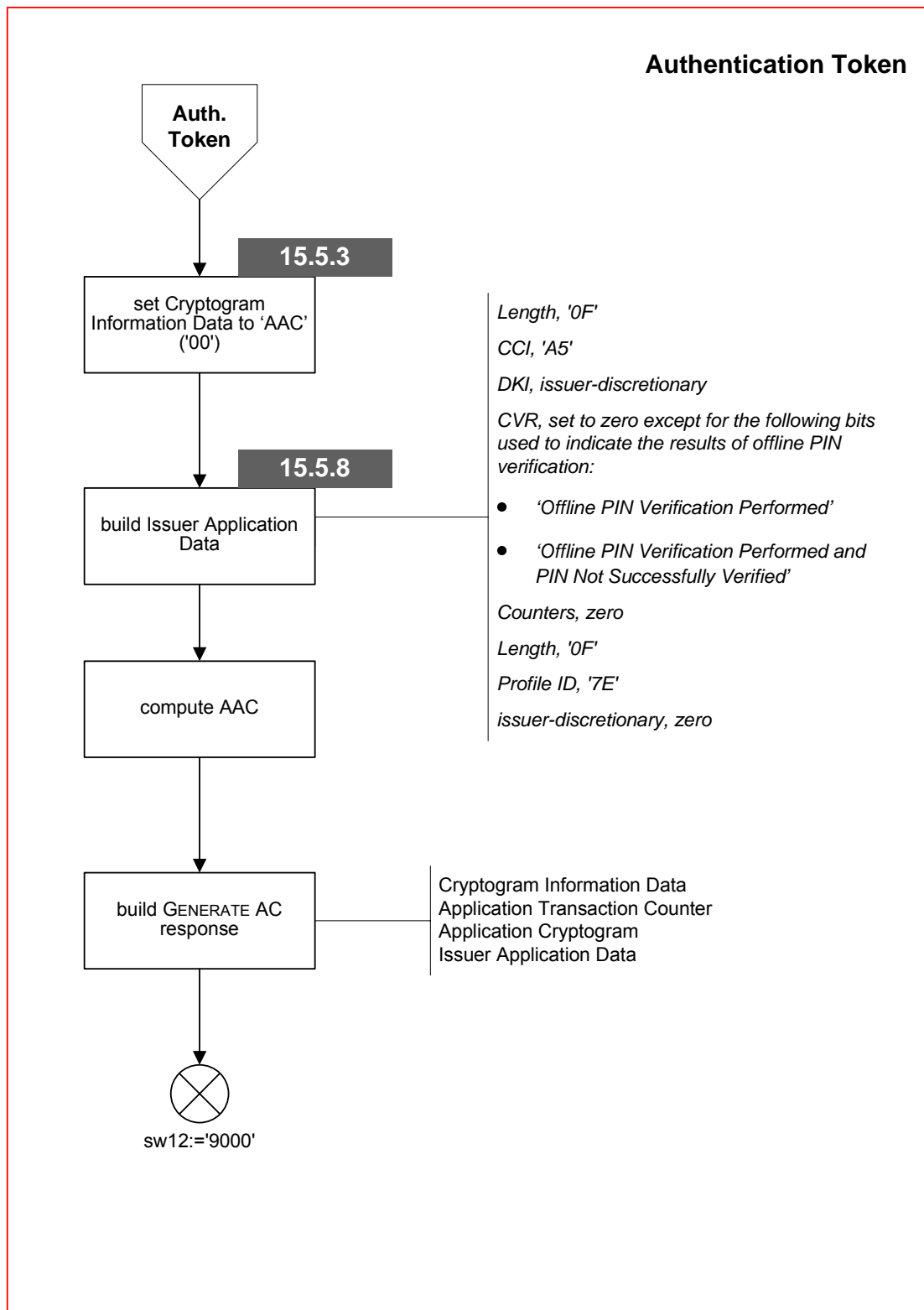
Flow 15-6.2 AAC Checks, continued



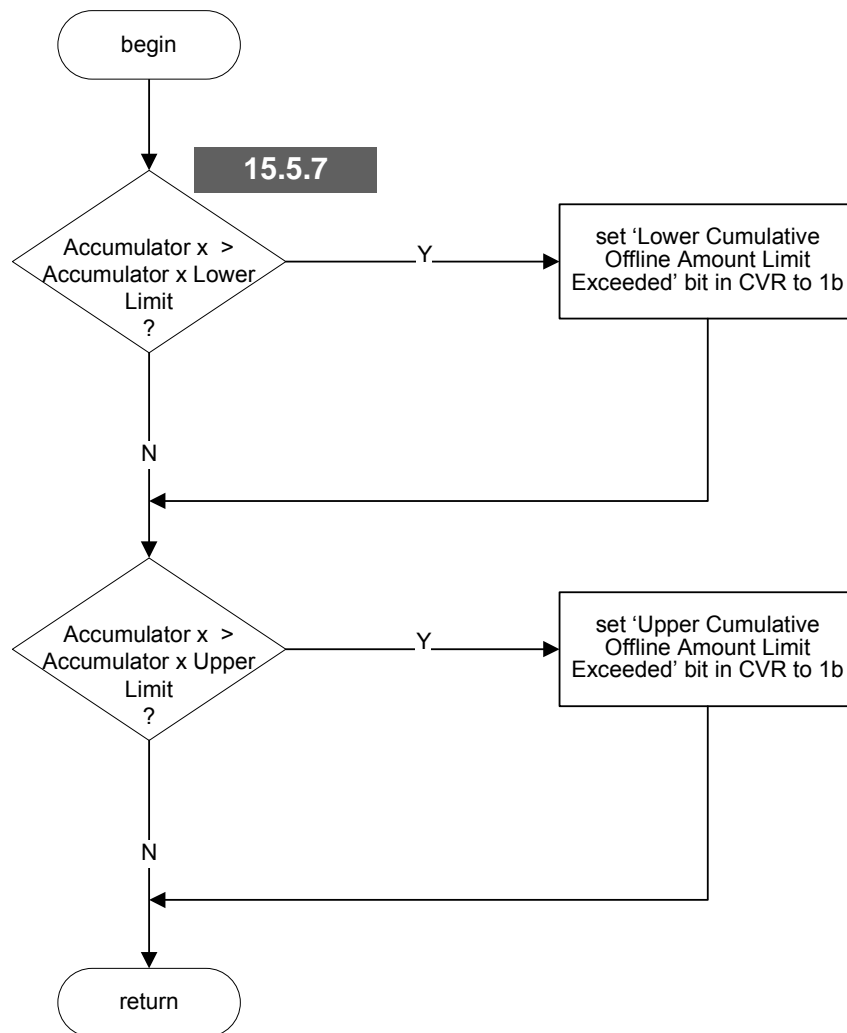
Flow 15-6.3 AAC Checks, continued



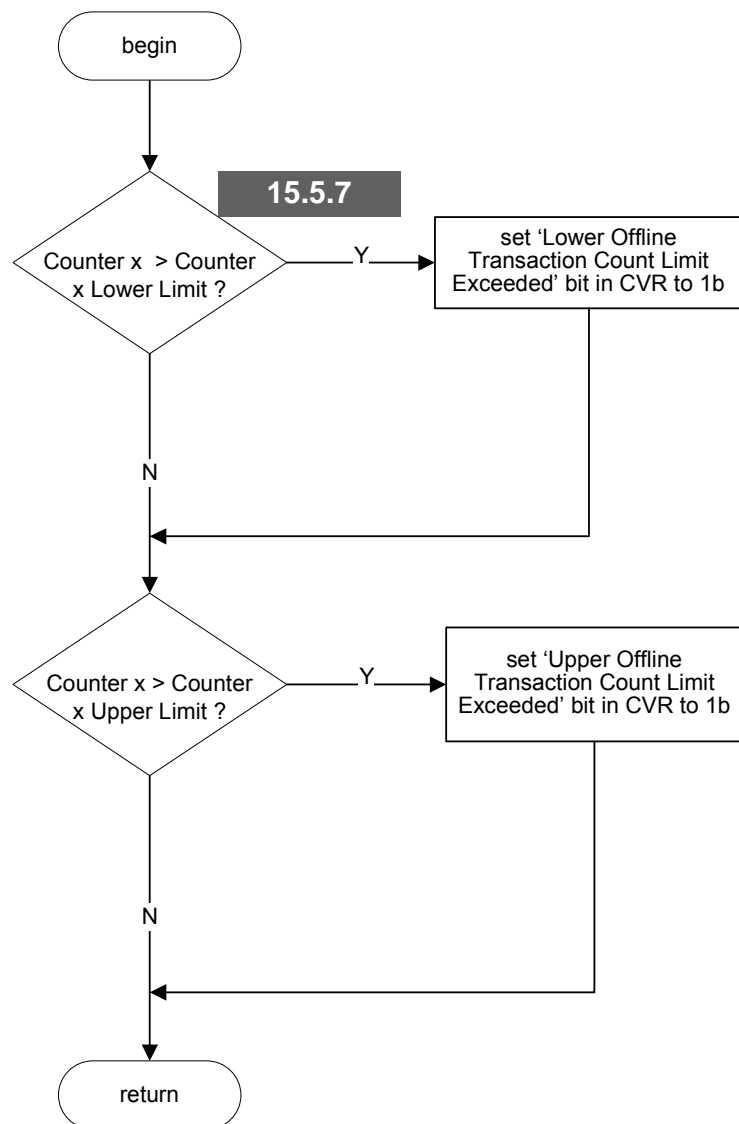
Flow 15-6.4 AAC Checks, continued

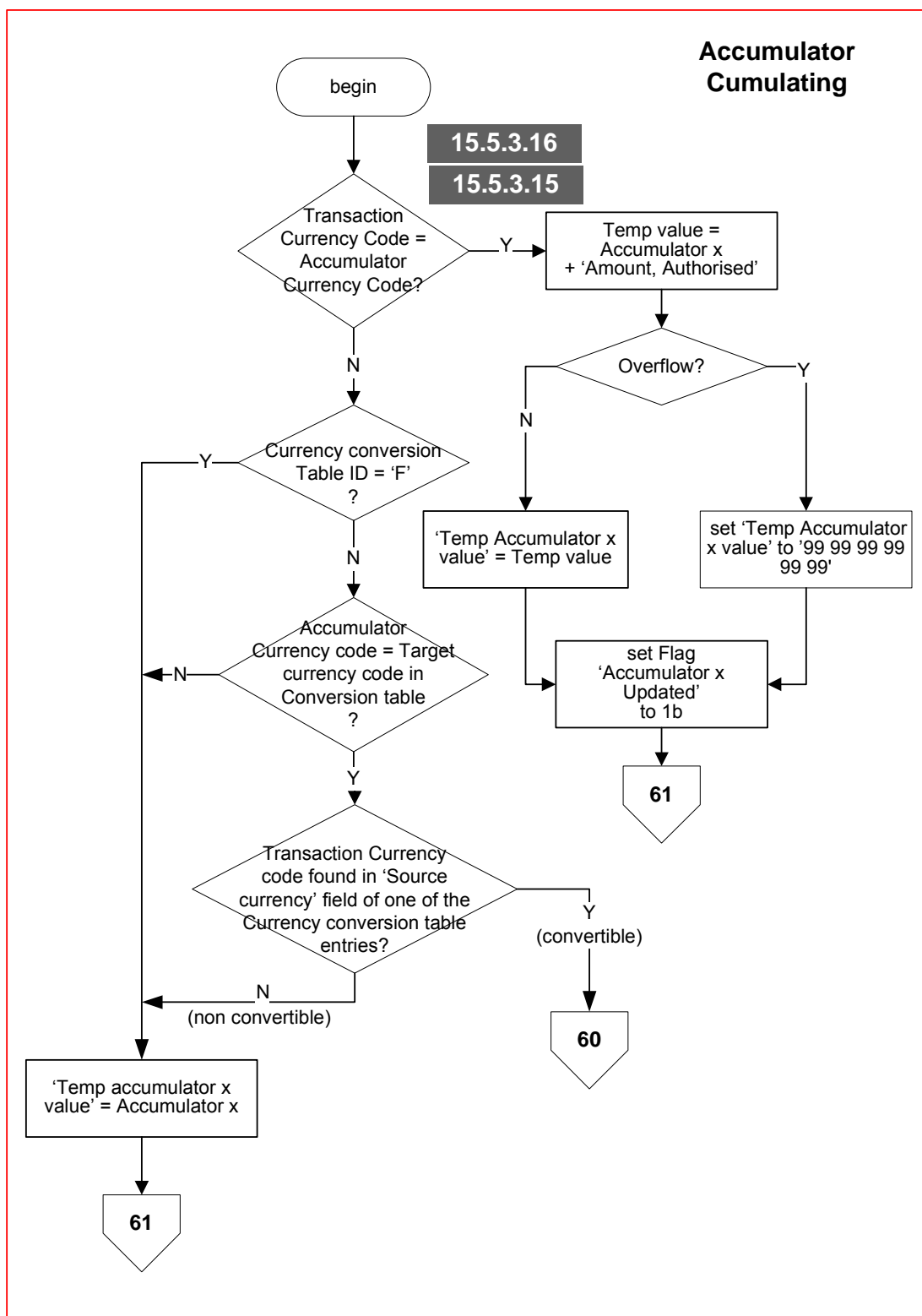
**Flow 15-7 Authentication Token**

AAC Accumulator Check

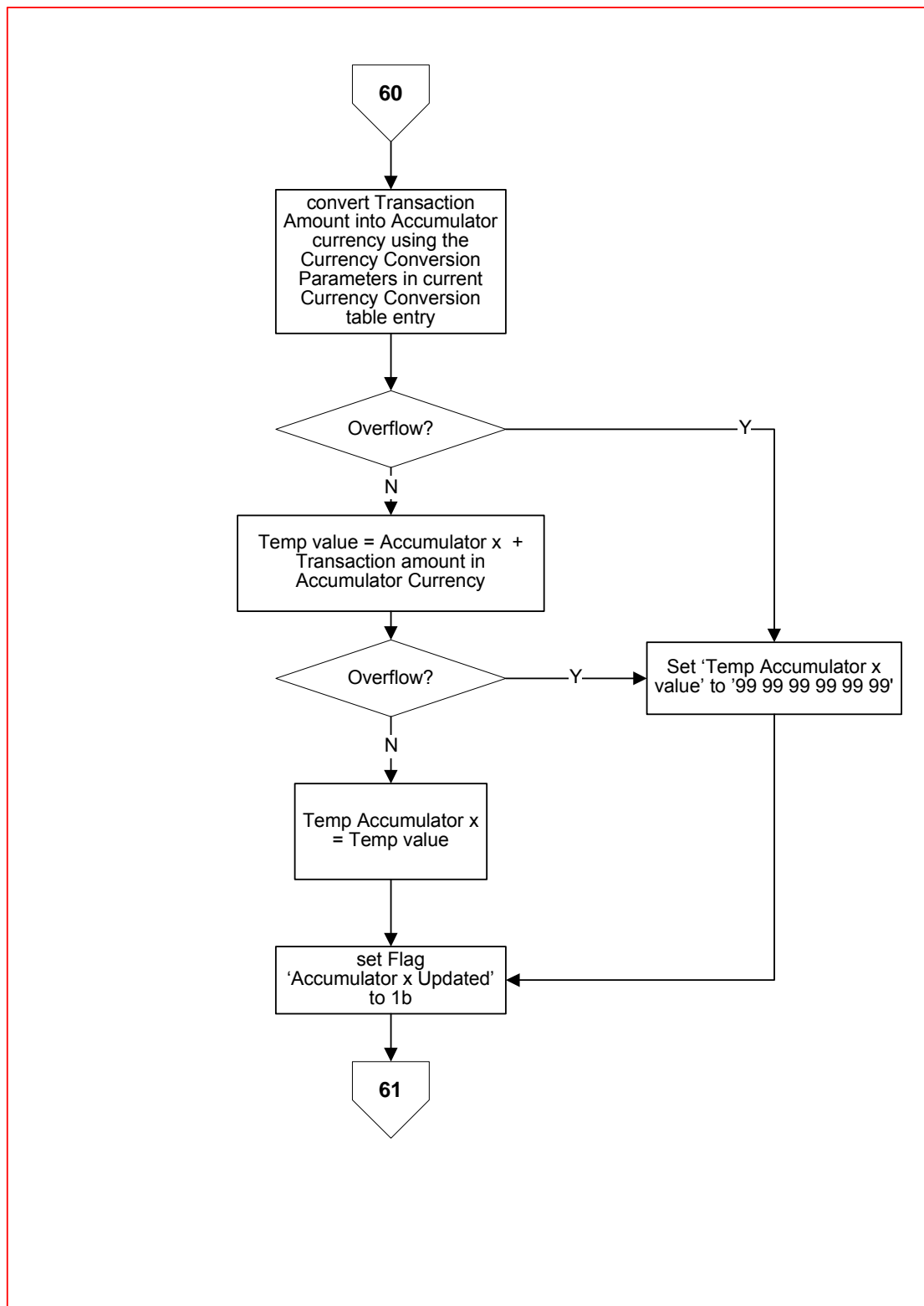


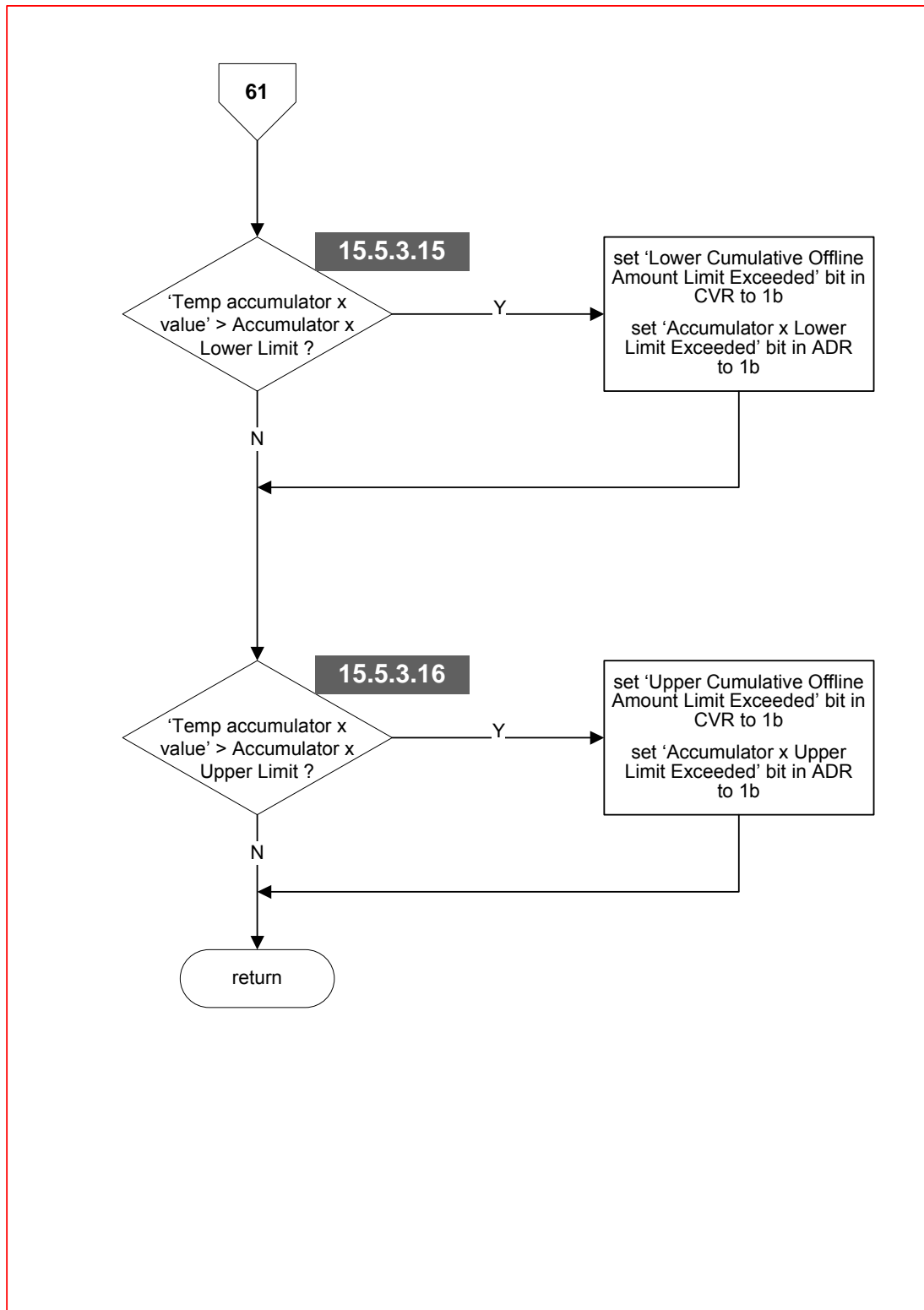
Flow 15-8 AAC Accumulator Check

AAC Counter Check**Flow 15-9 AAC Counter Check**

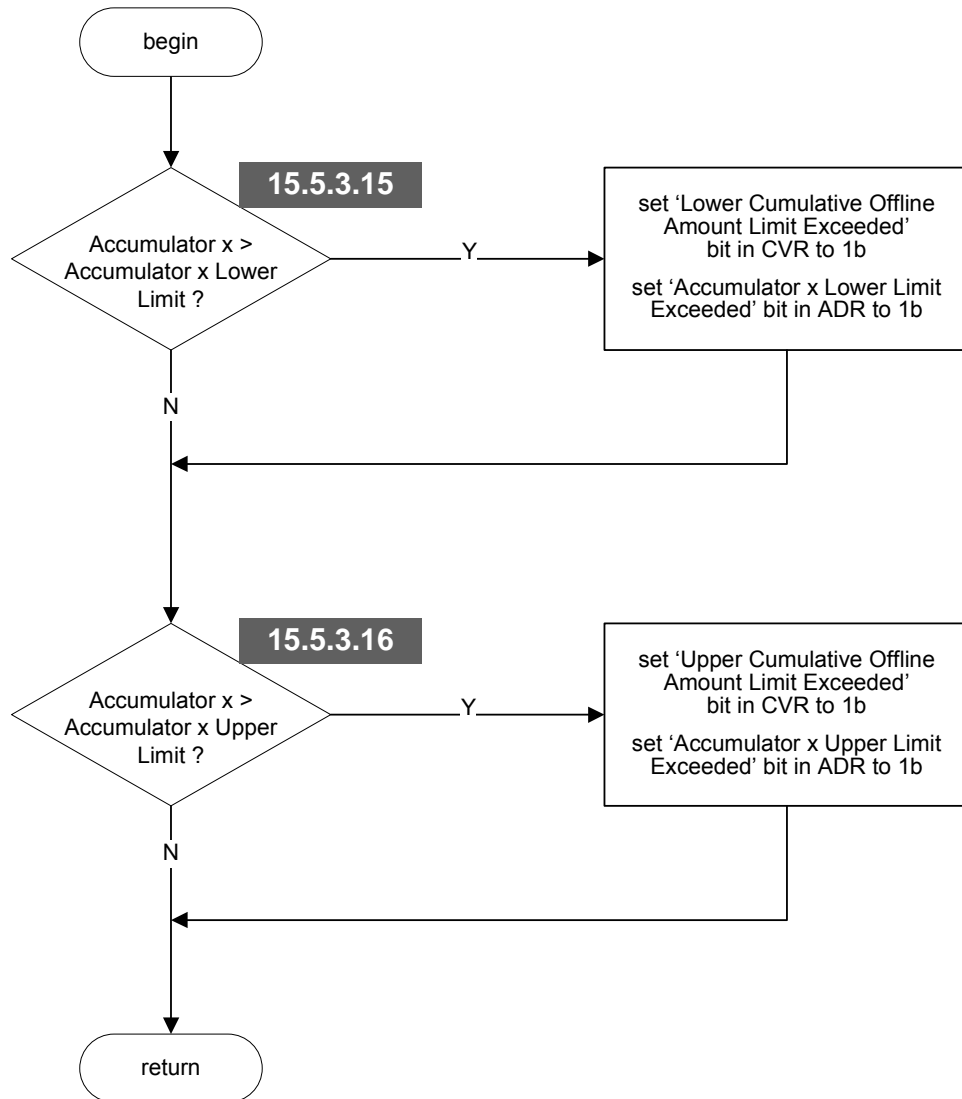


Flow 15-10 Accumulator Cumulating

**Flow 15-10.1 Accumulator Cumulating, continued**

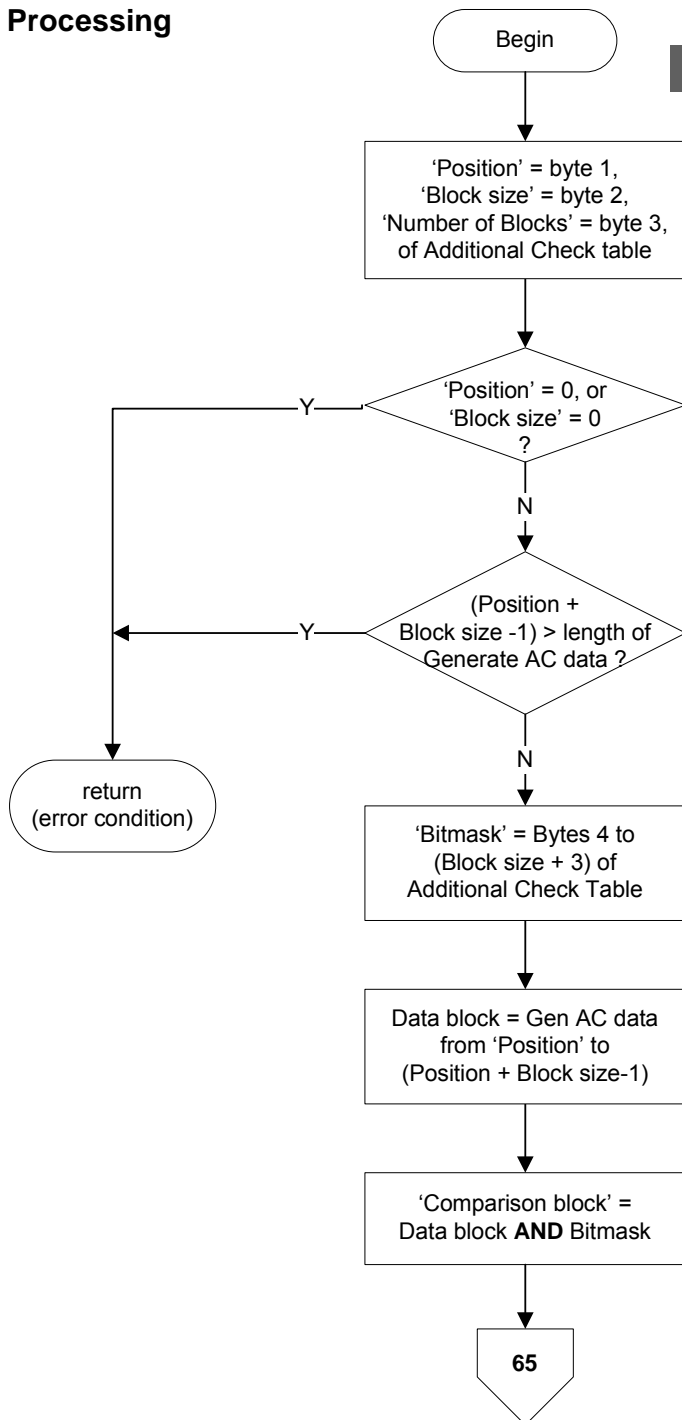


Flow 15-10.2 Accumulator Cumulating, continued

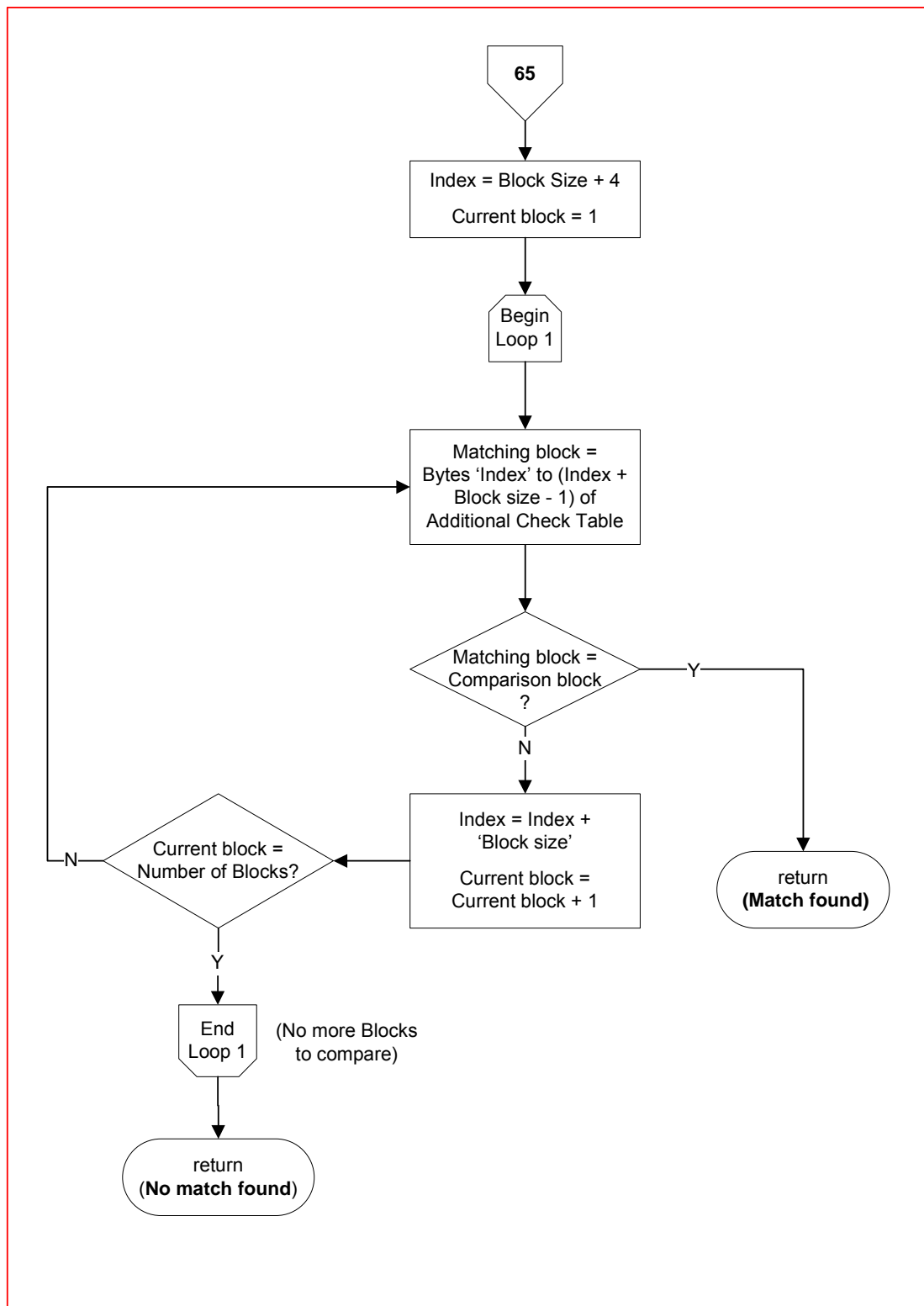
**Accumulator x
Non-cumulating Check
(Without Transaction in
CRM Check)****Flow 15-11 Accumulator x Non-cumulating Check**

Additional Check Table Processing

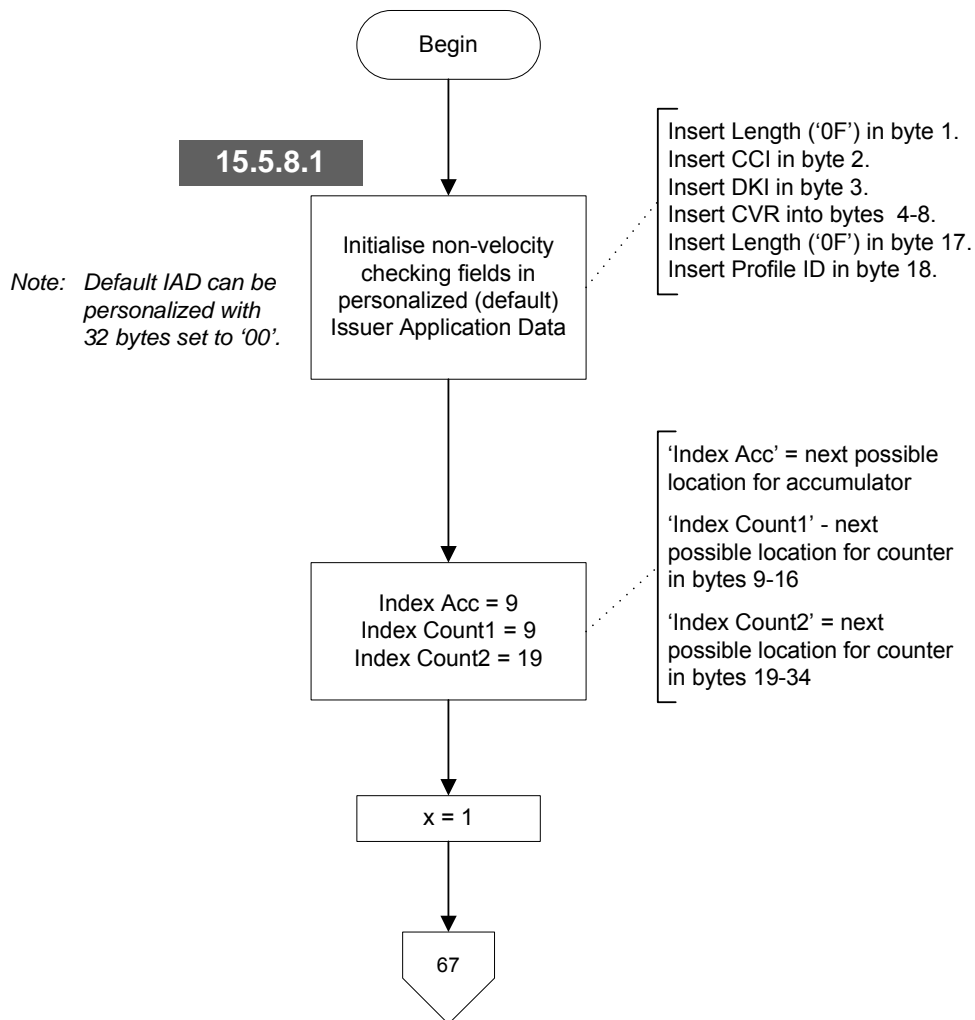
15.5.3.1



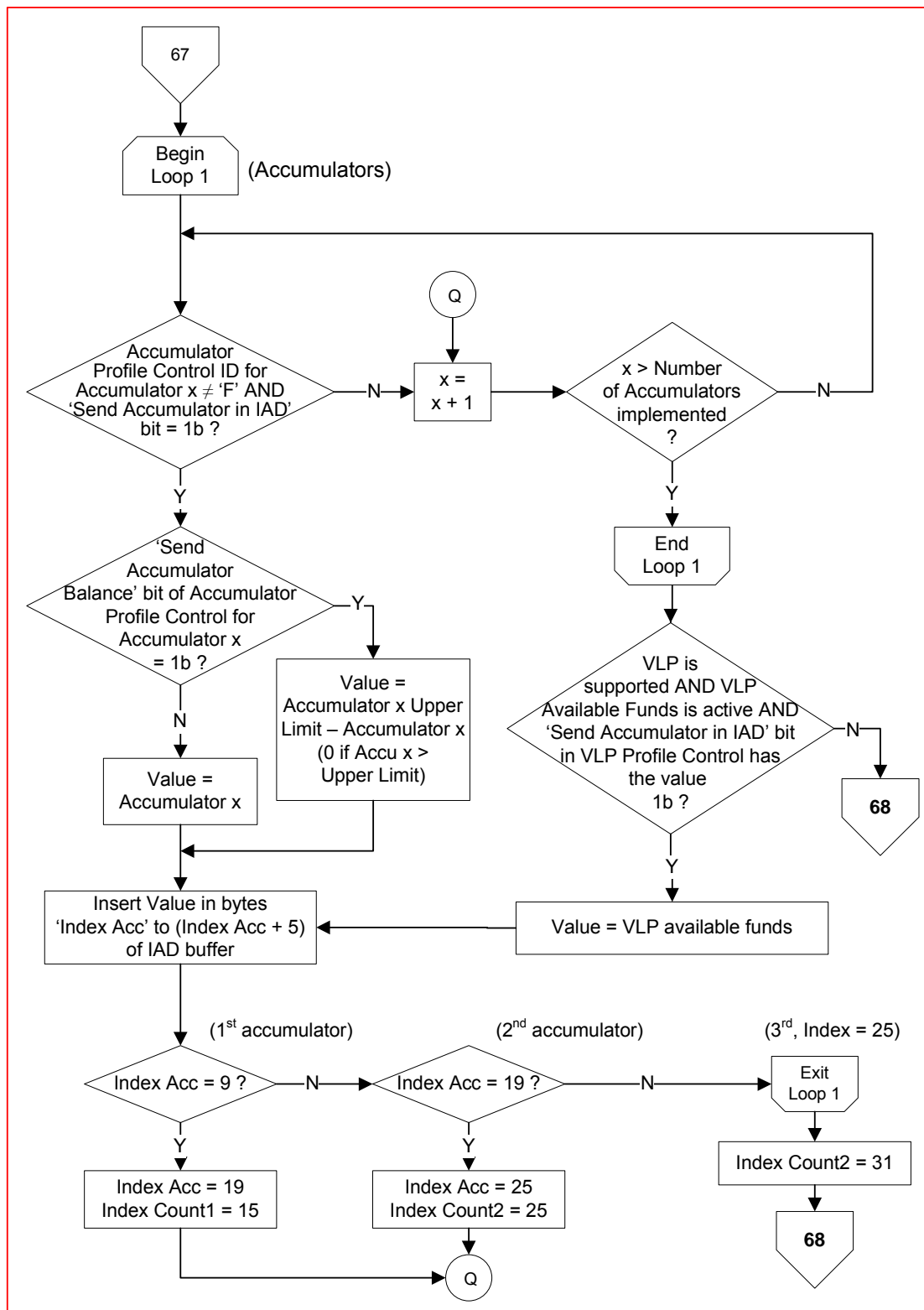
Flow 15-12 Additional Check Table Processing

**Flow 15-12.1 Additional Check Table Processing, continued**

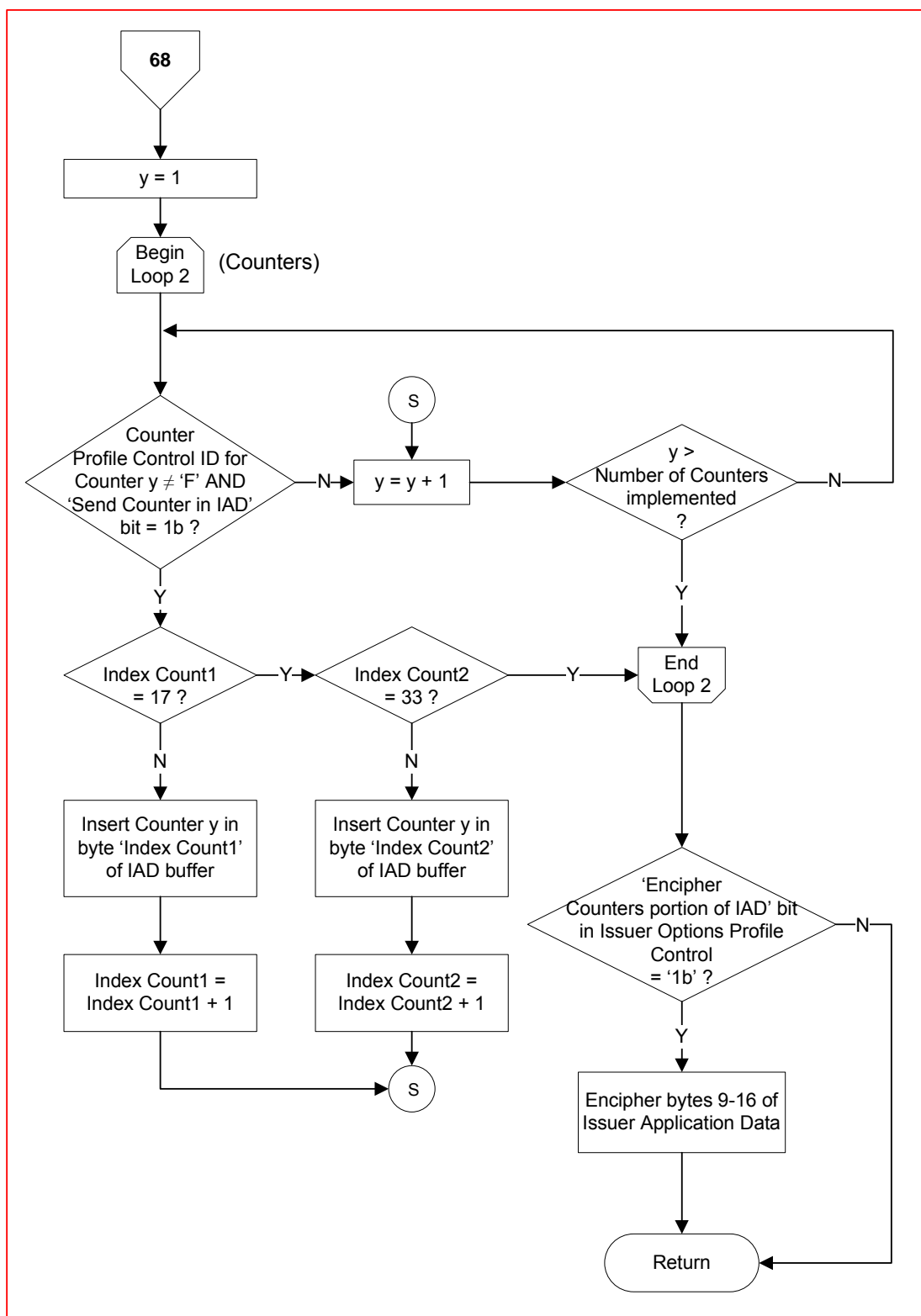
Build Issuer Application Data



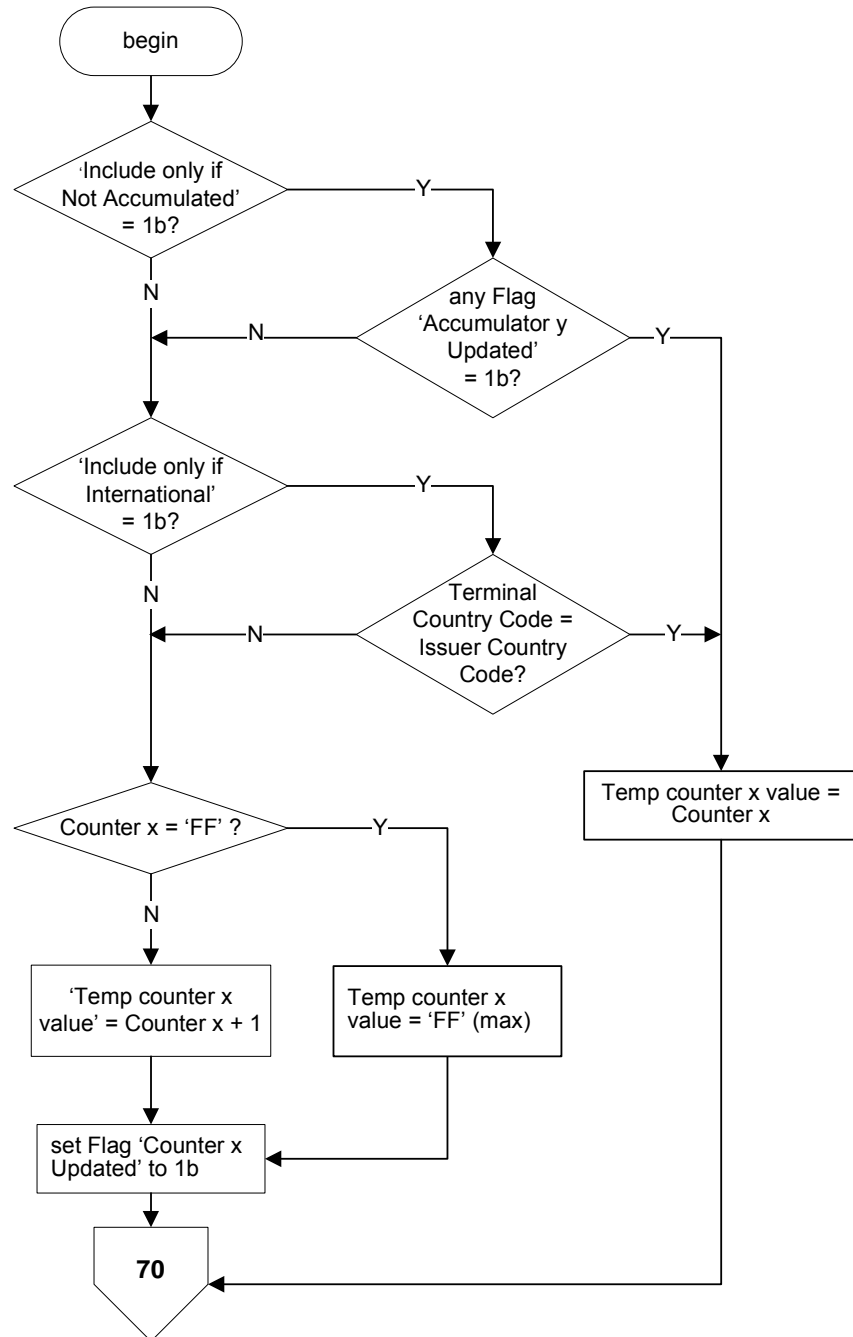
Flow 15-13 Build Issuer Application Data

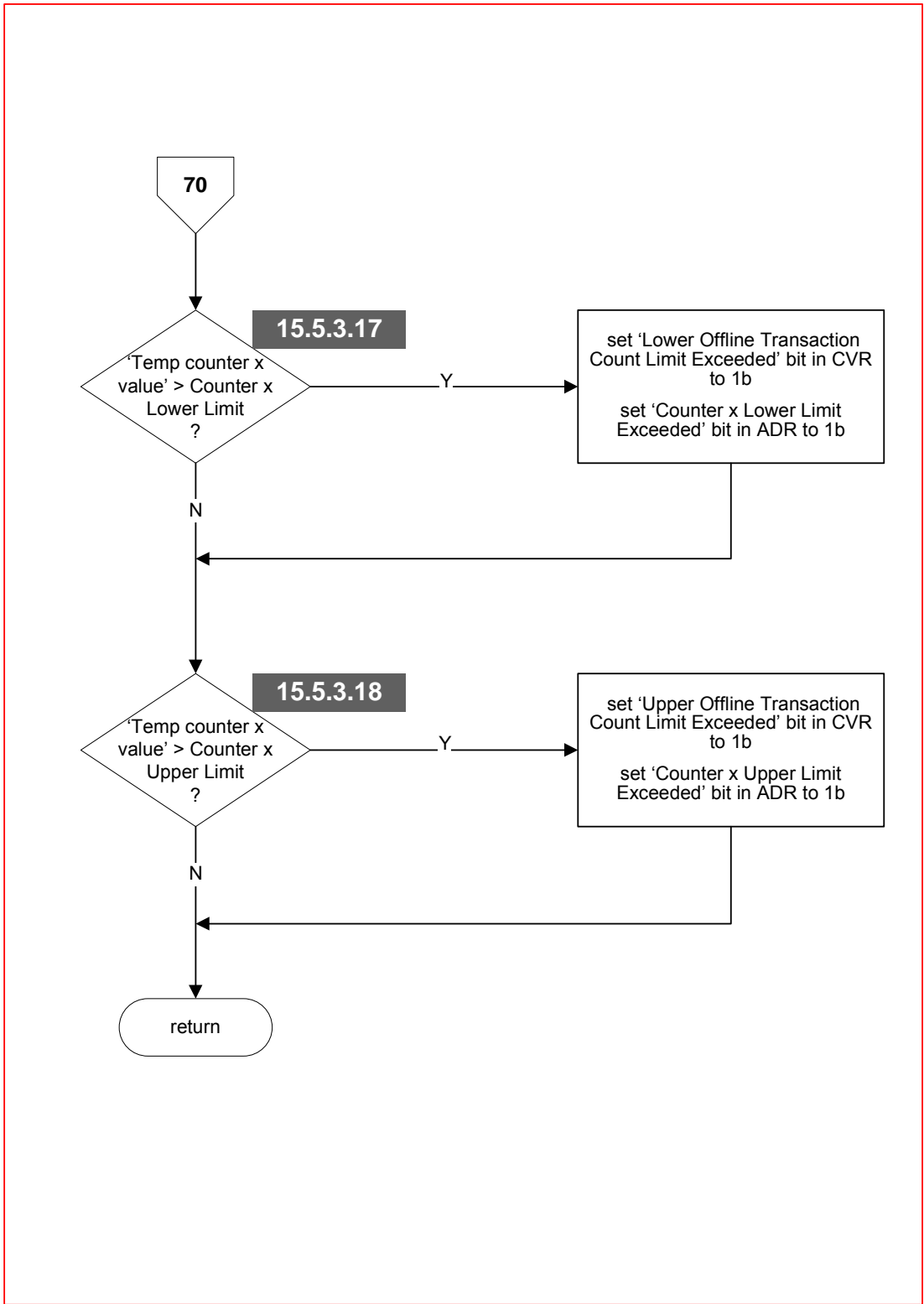


Flow 15-13.1 Build Issuer Application Data, continued

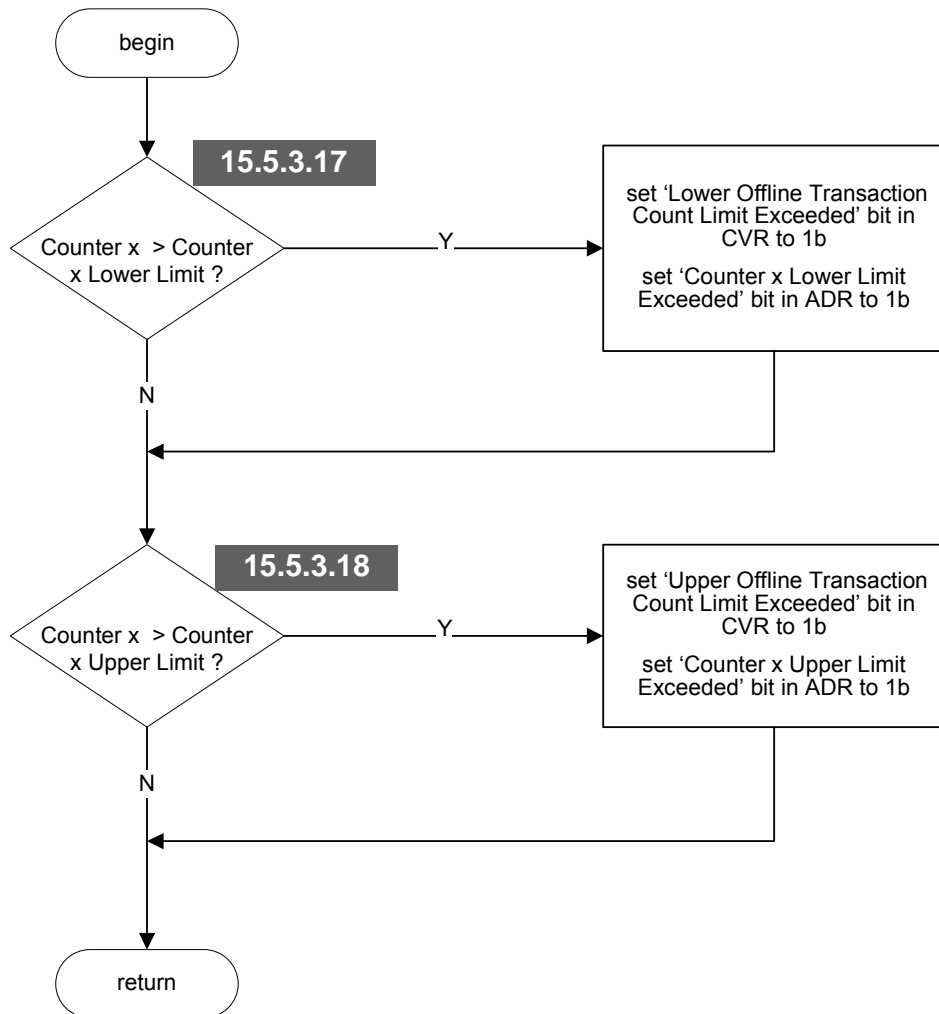


Flow 15-13.2 Build Issuer Application Data, continued

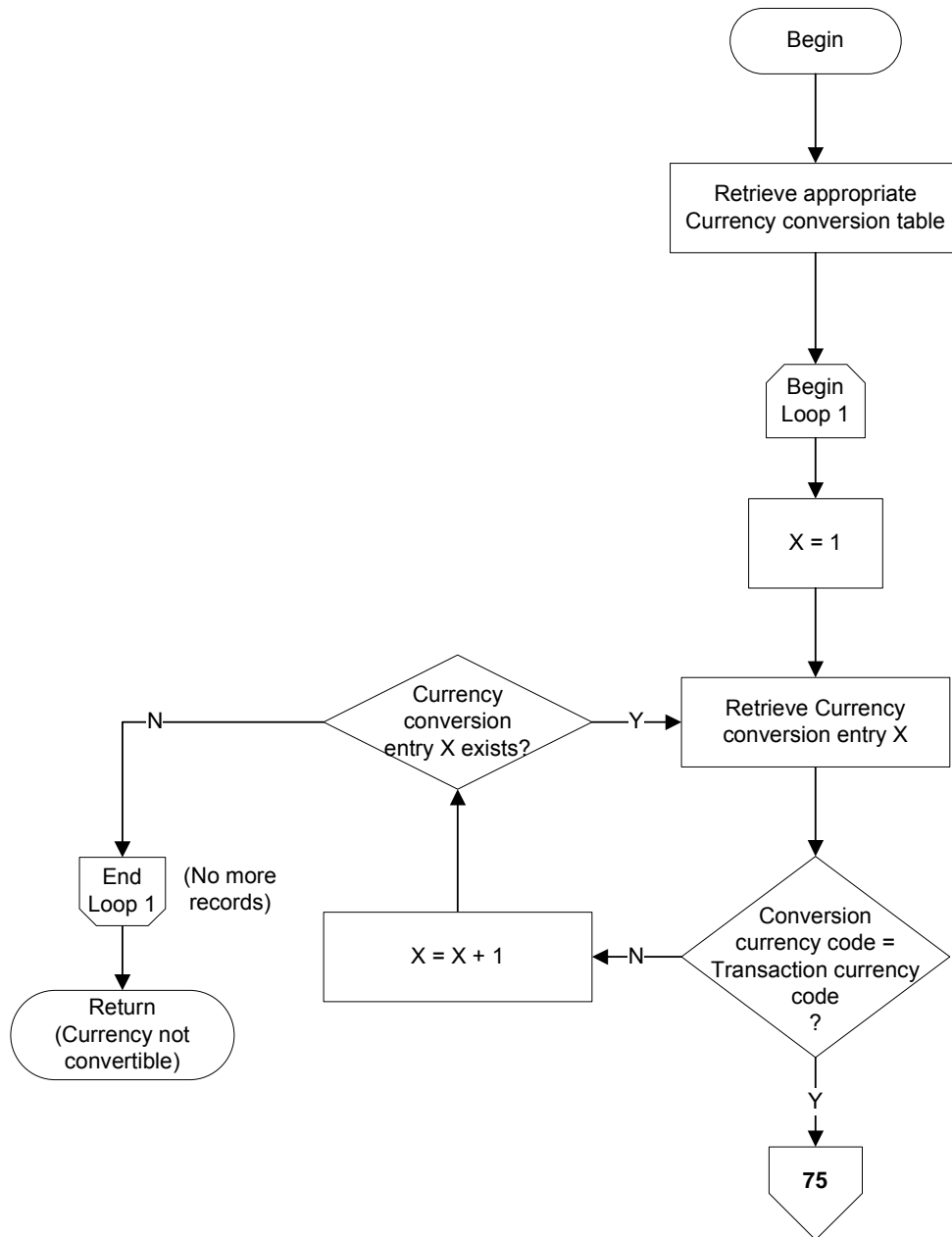
**Counter x Cumulating Check
(Include ARQC Transaction
in CRM Check)****Flow 15-14 Counter x Cumulating Check**



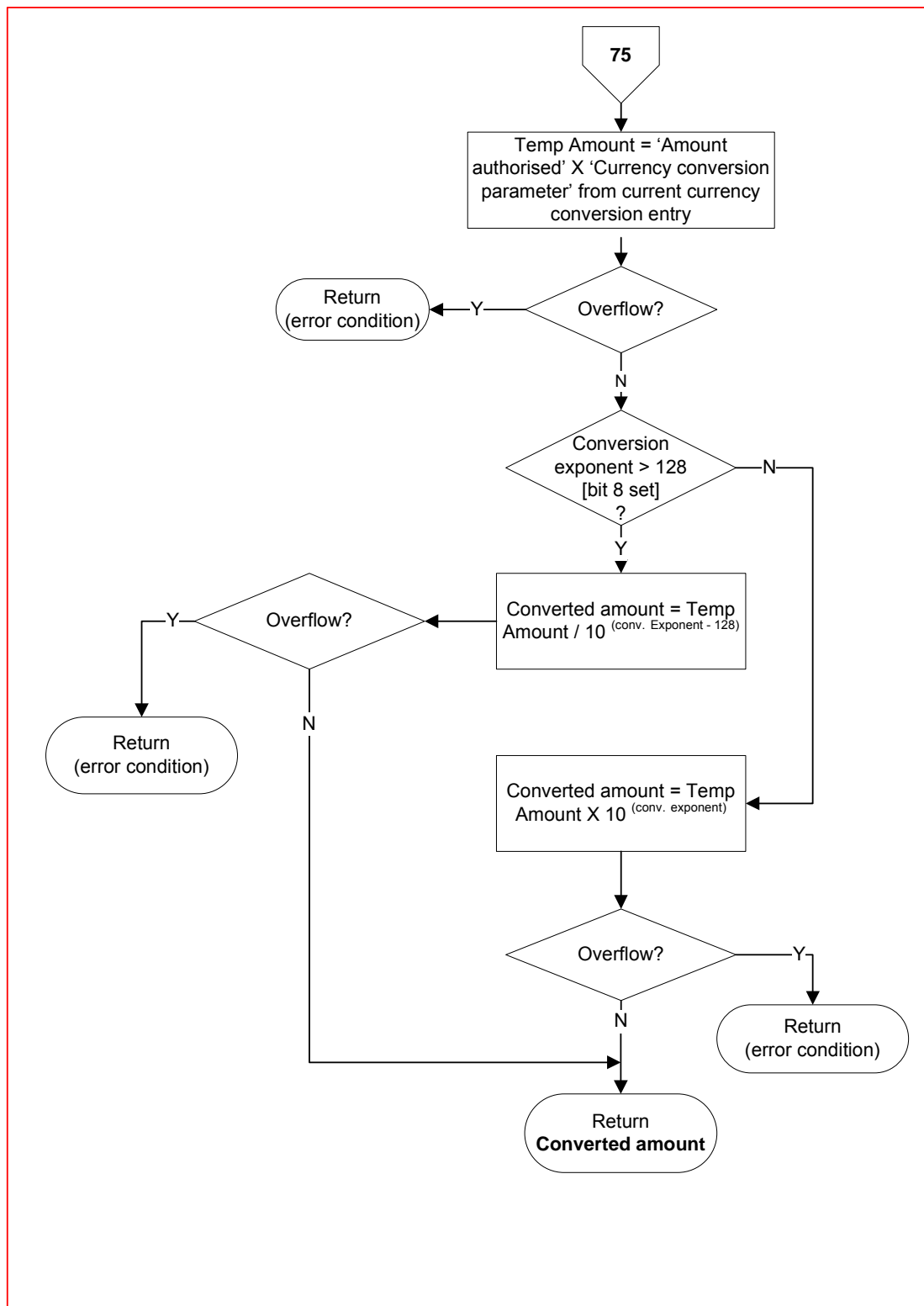
Flow 15-14.1 Counter x Cumulating Check, continued

**Counter
Non-cumulating Check****Flow 15-15 Counter Non-cumulating Check**

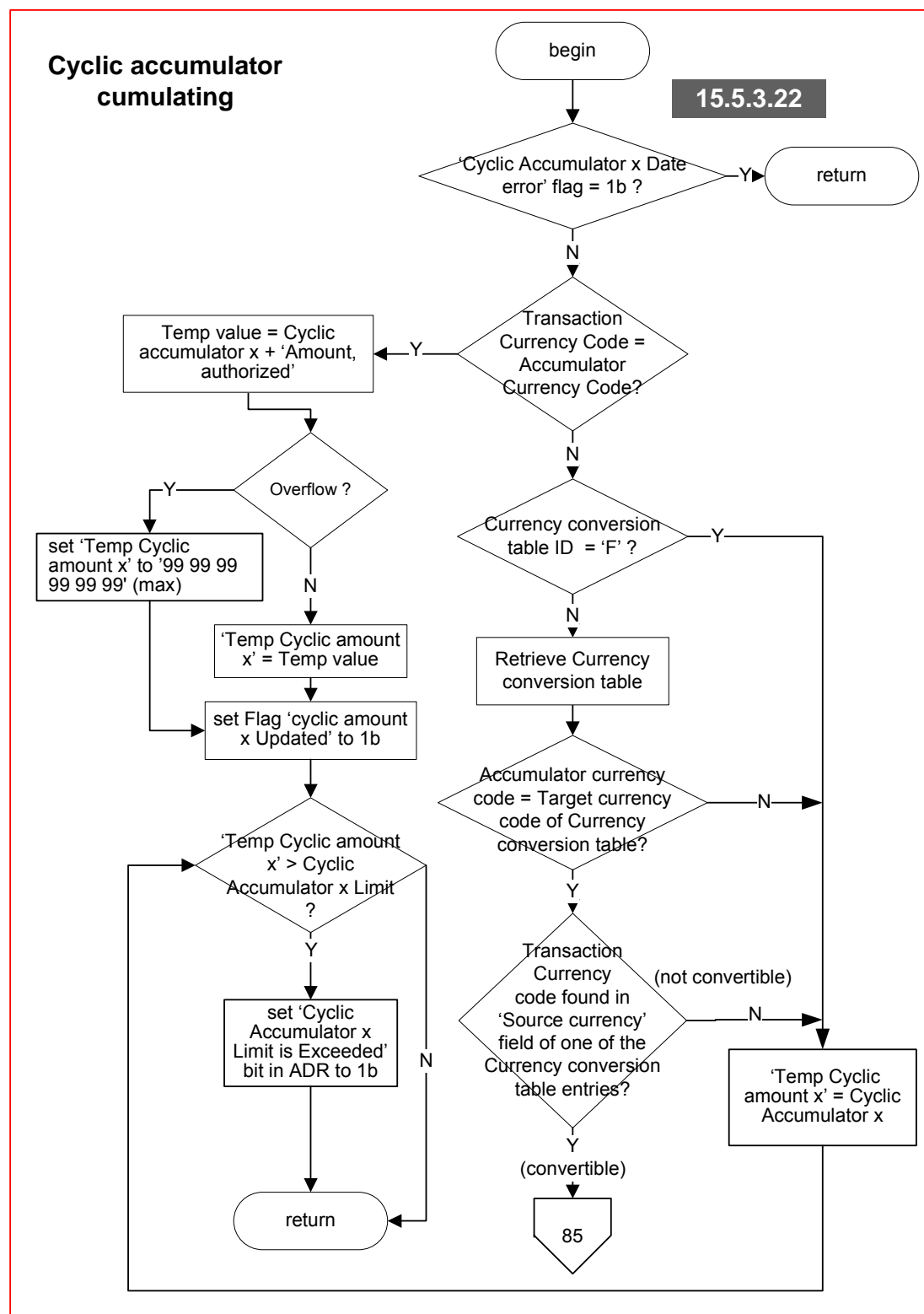
Currency Conversion



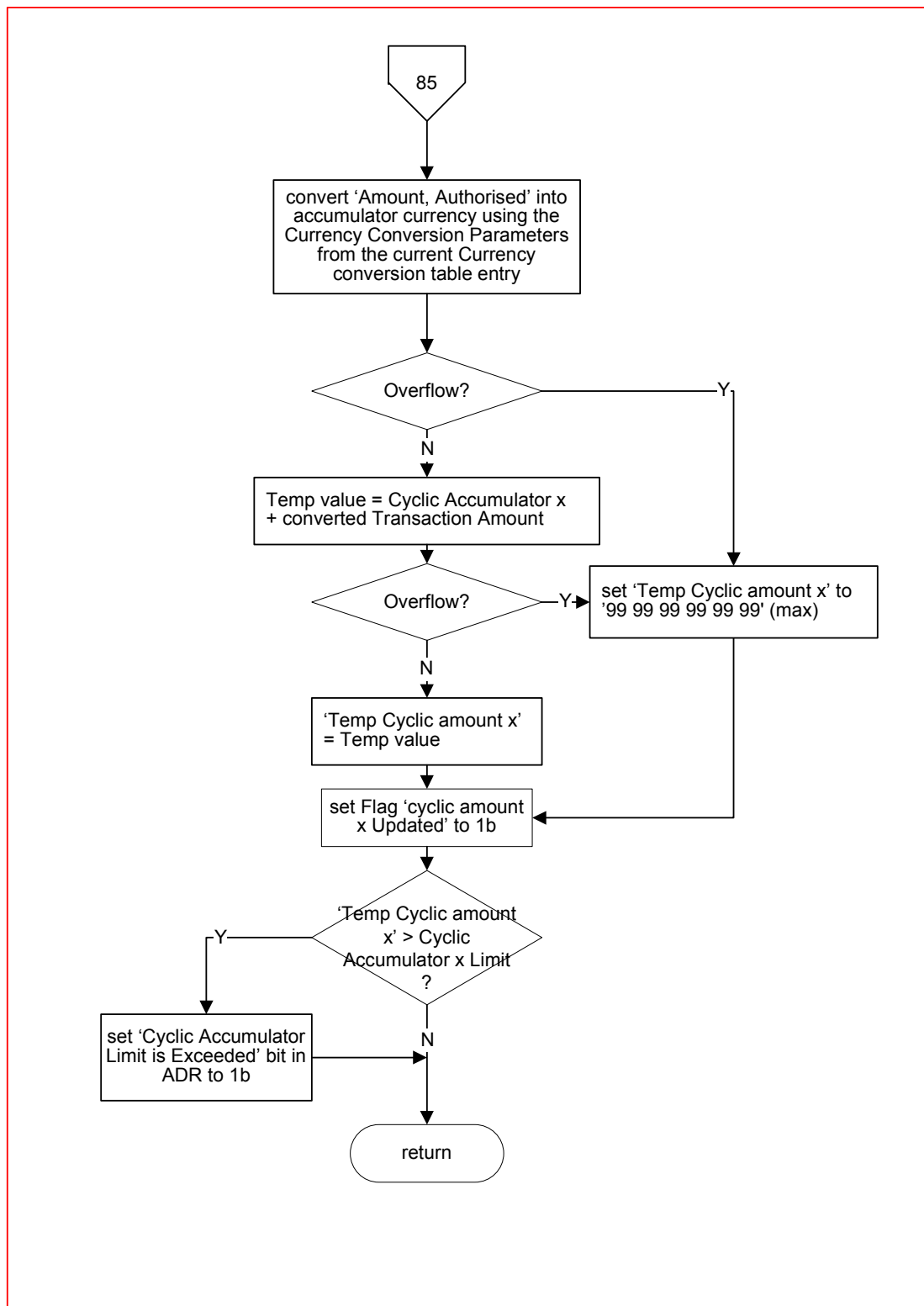
Flow 15-16 Currency Conversion



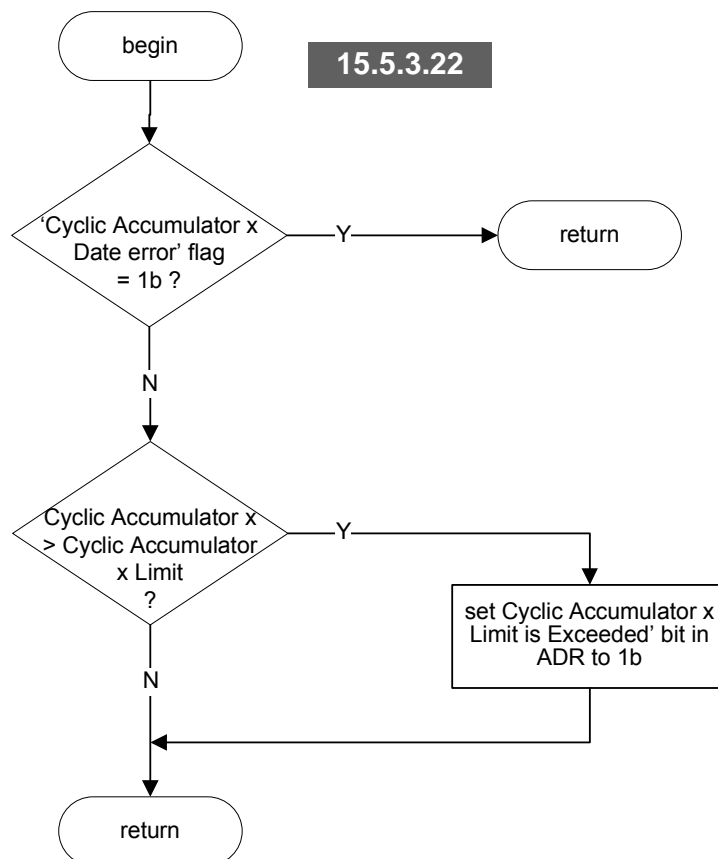
Flow 15-16.1 Currency Conversion, continued



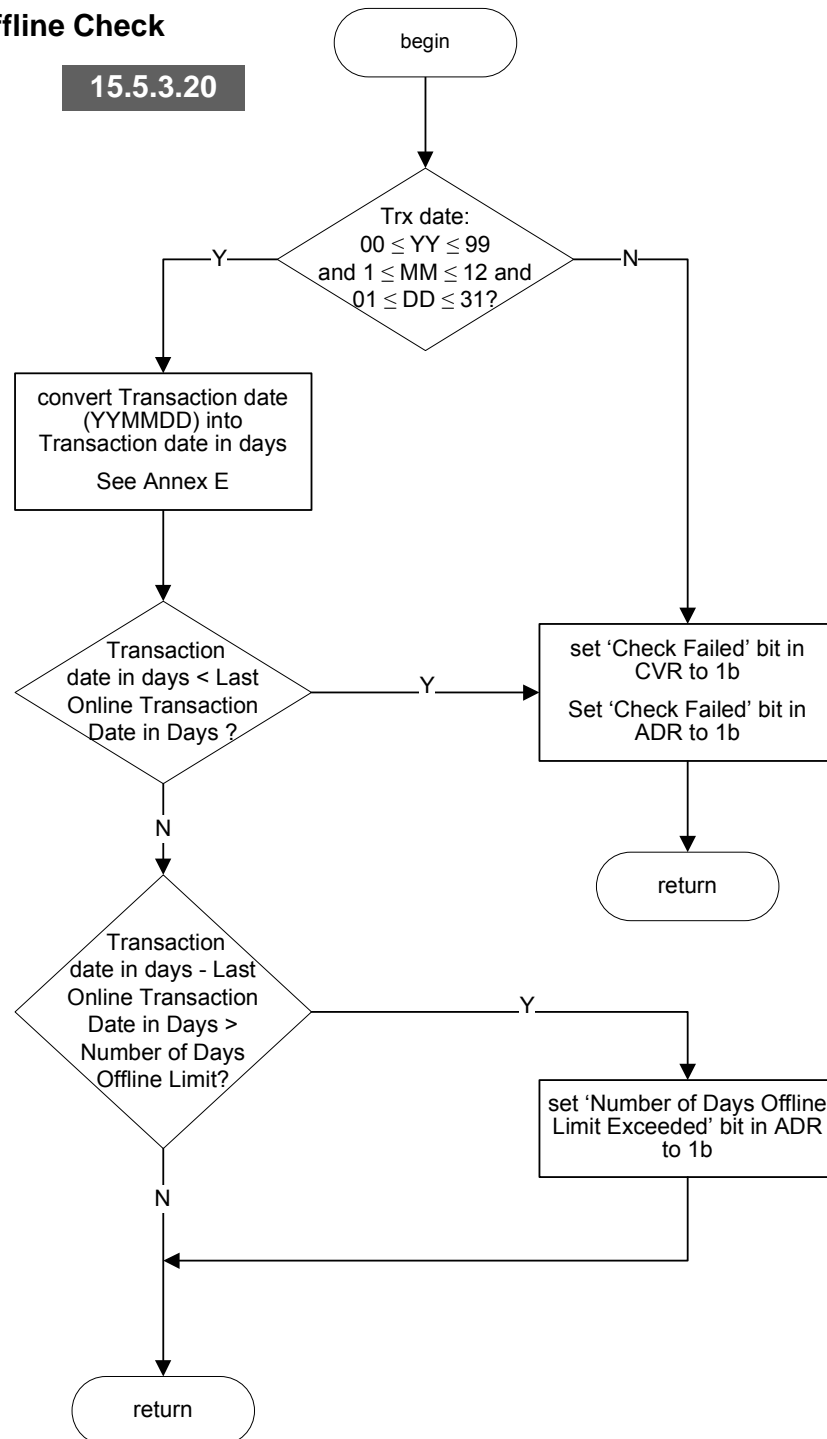
Flow 15-17 Cyclic Accumulator Cumulating

**Flow 15-17.1 Cyclic Accumulator Cumulating, continued**

Cyclic accumulator non-cumulating check

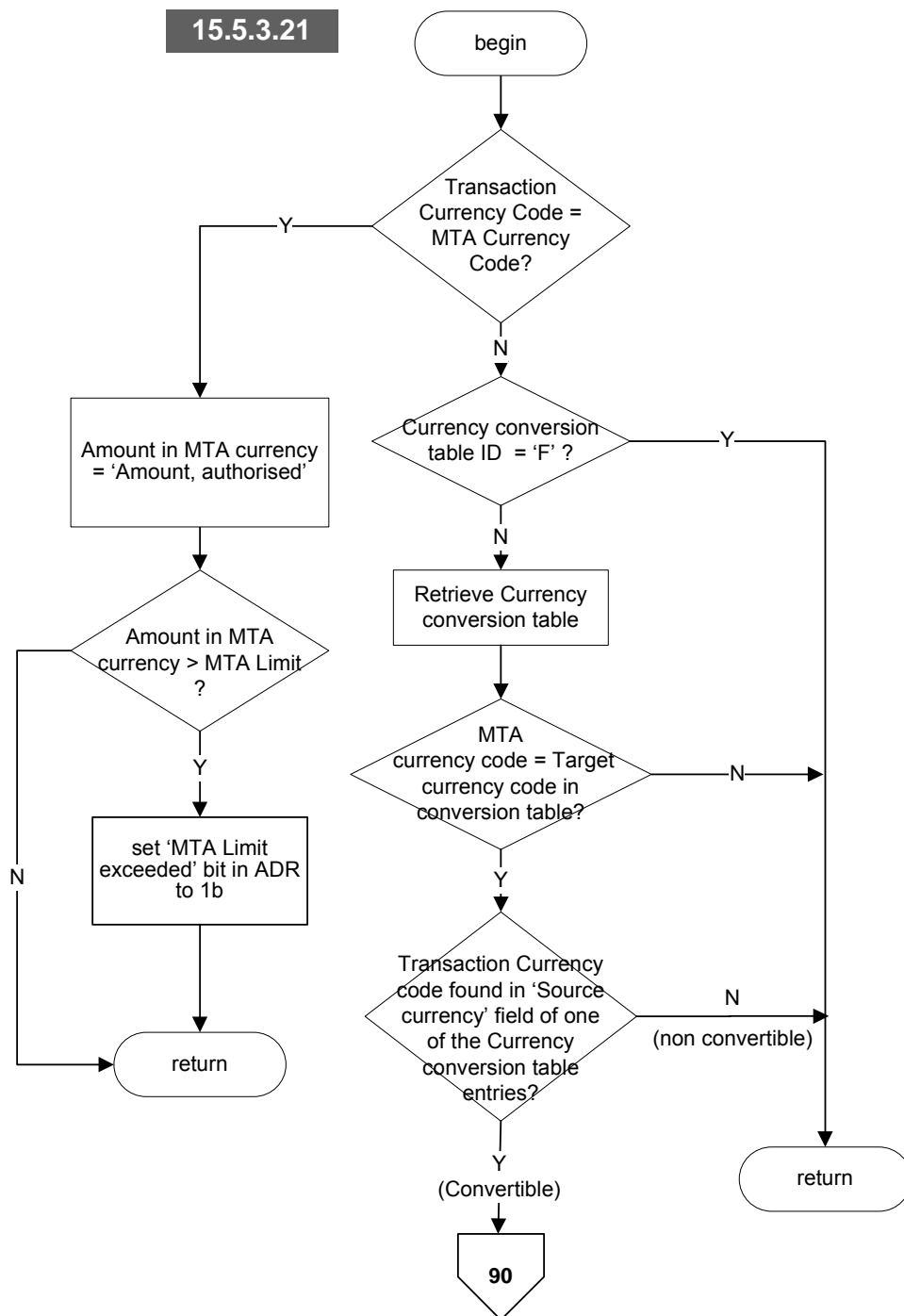


Flow 15-18 Cyclic Accumulator Non-cumulating Check

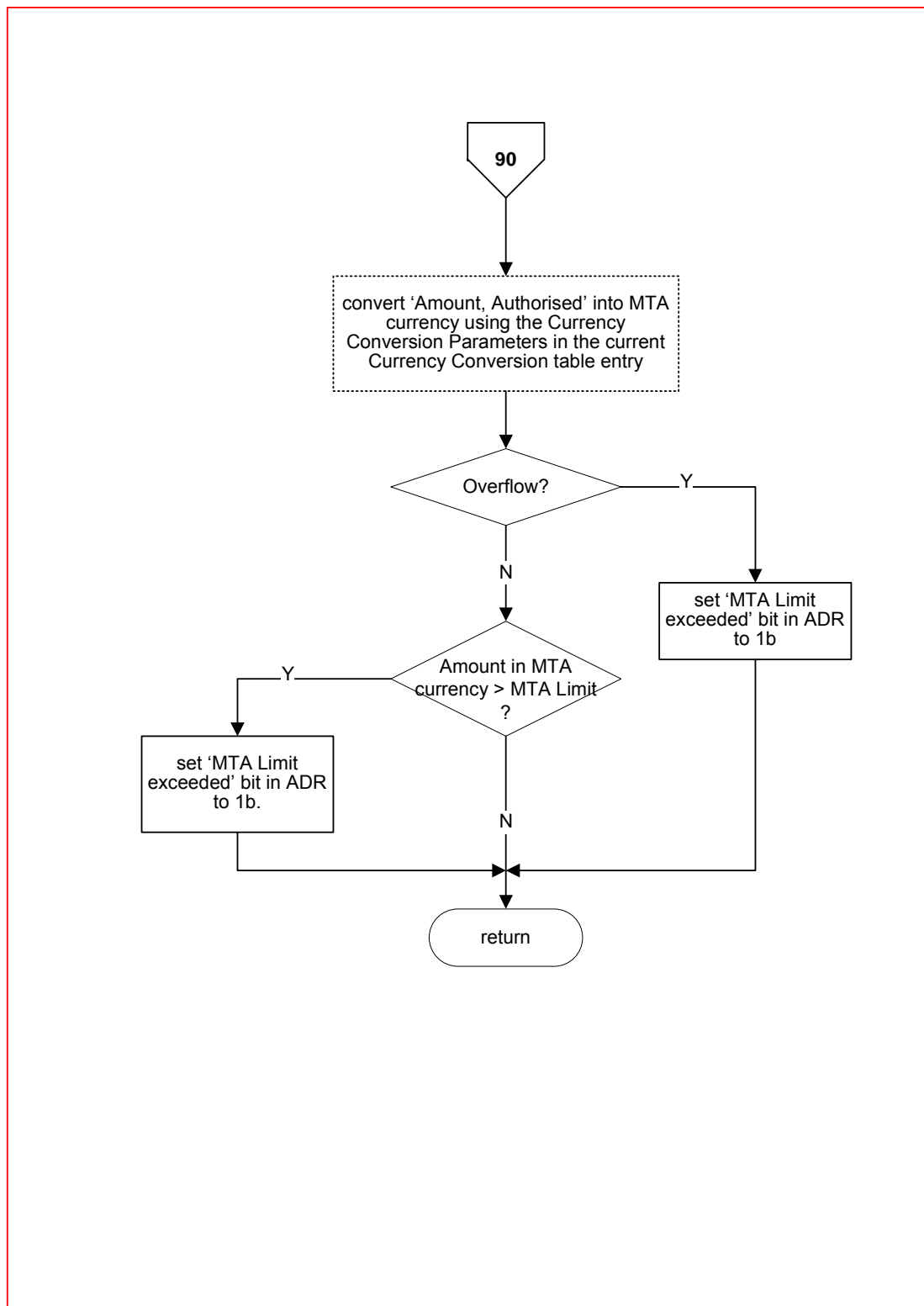
**Maximum Number of
Days Offline Check****15.5.3.20****Flow 15-19 Maximum Number of Days Offline Check**

Maximum Transaction Amount Check

15.5.3.21

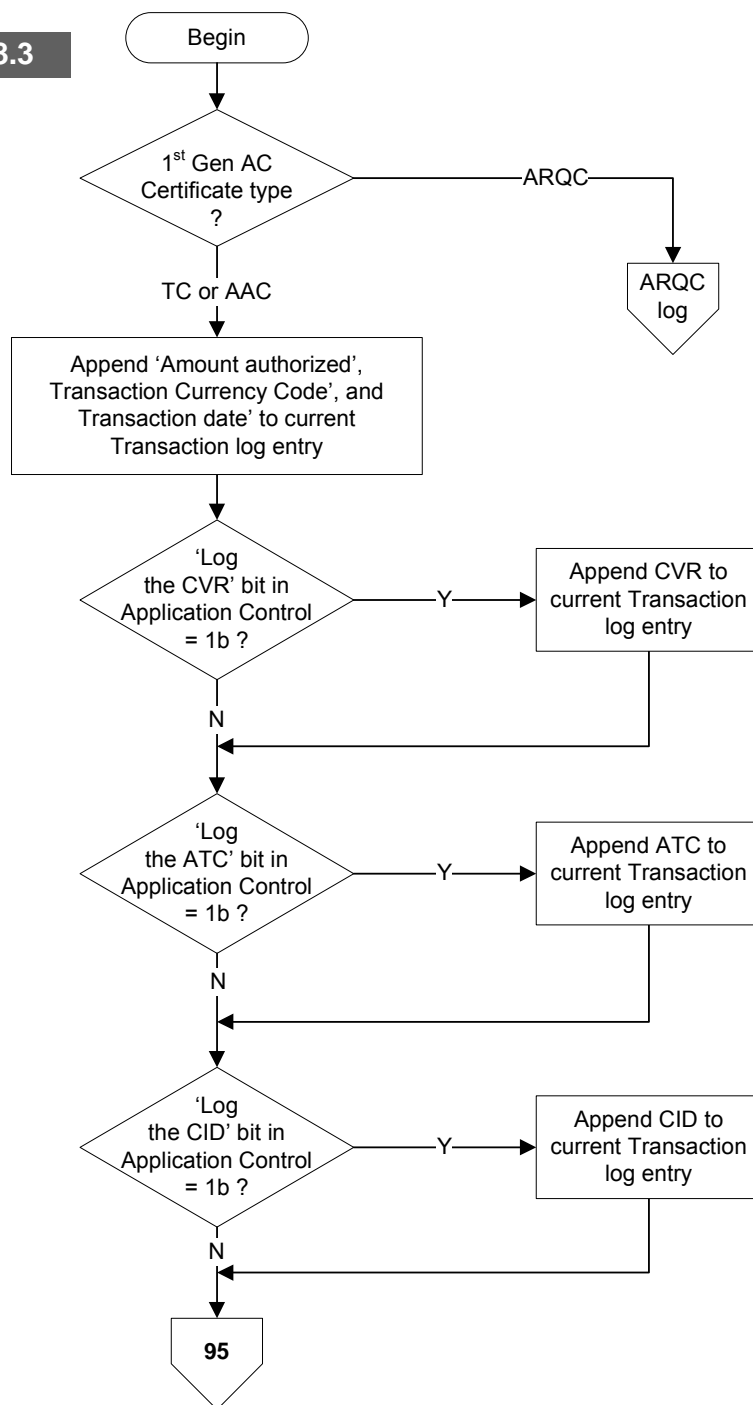


Flow 15-20 Maximum Transaction Amount Check

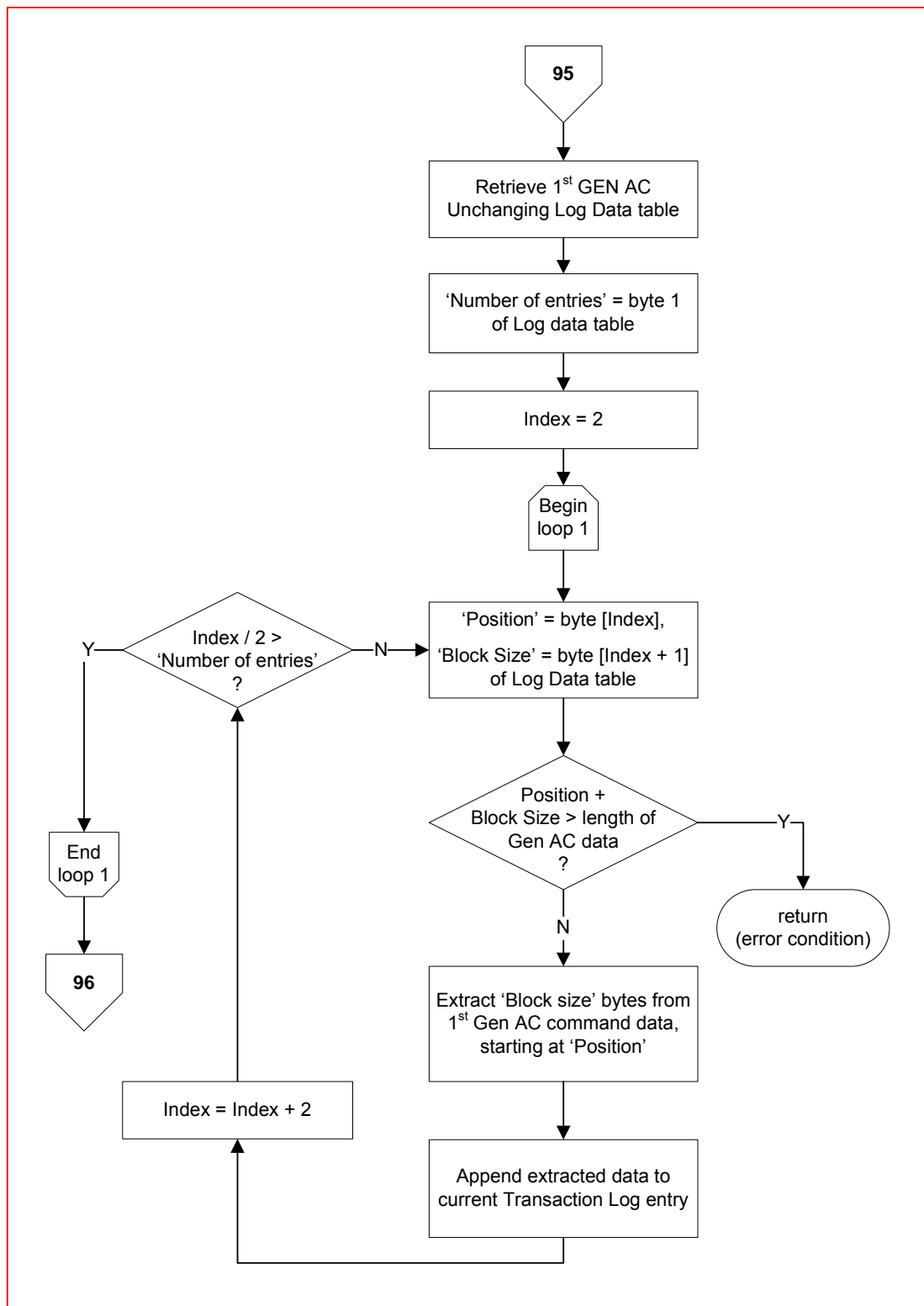
**Flow 15-20.1 Maximum Transaction Amount Check, continued**

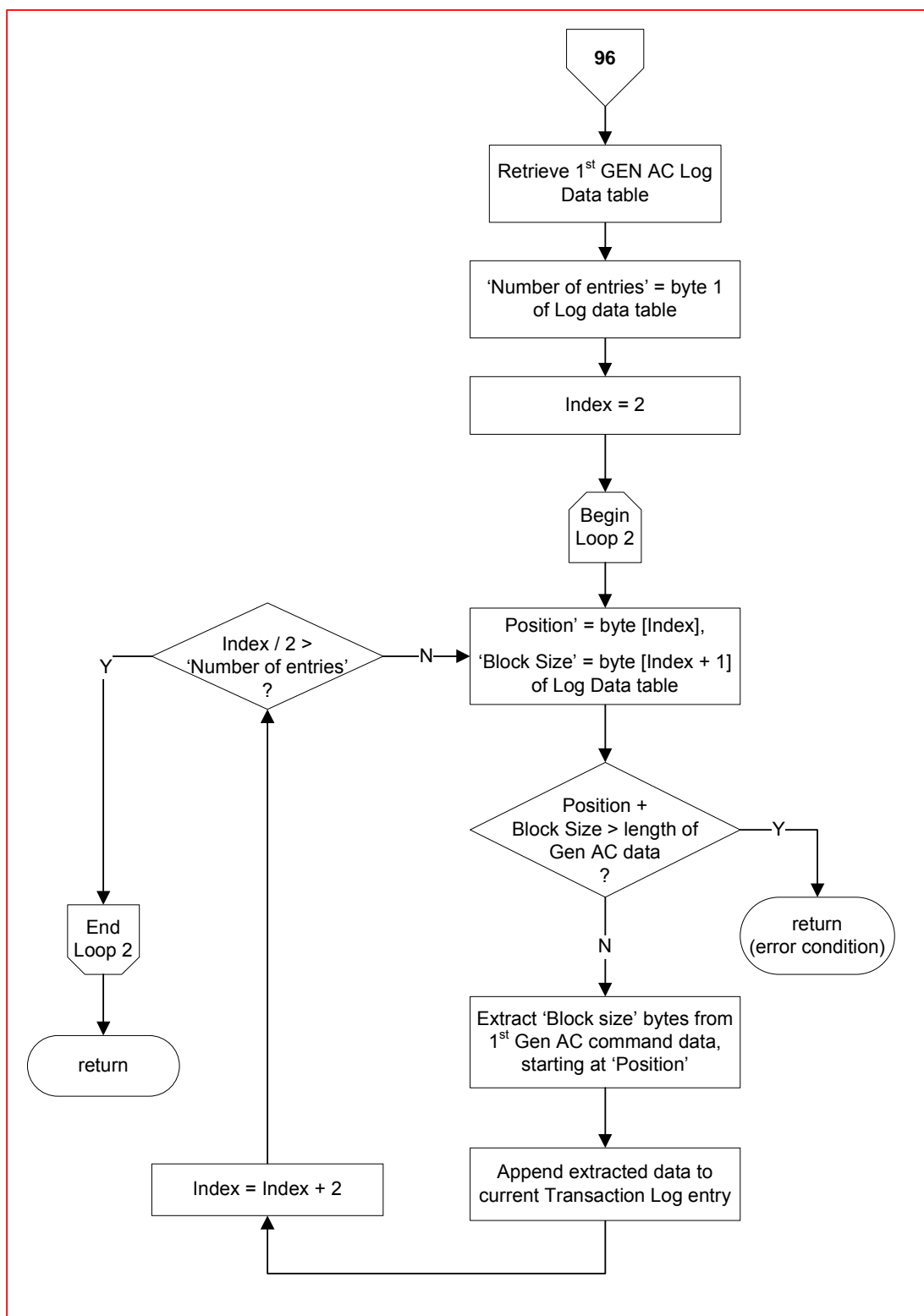
1st GENERATE AC Transaction Logging

15.5.8.3

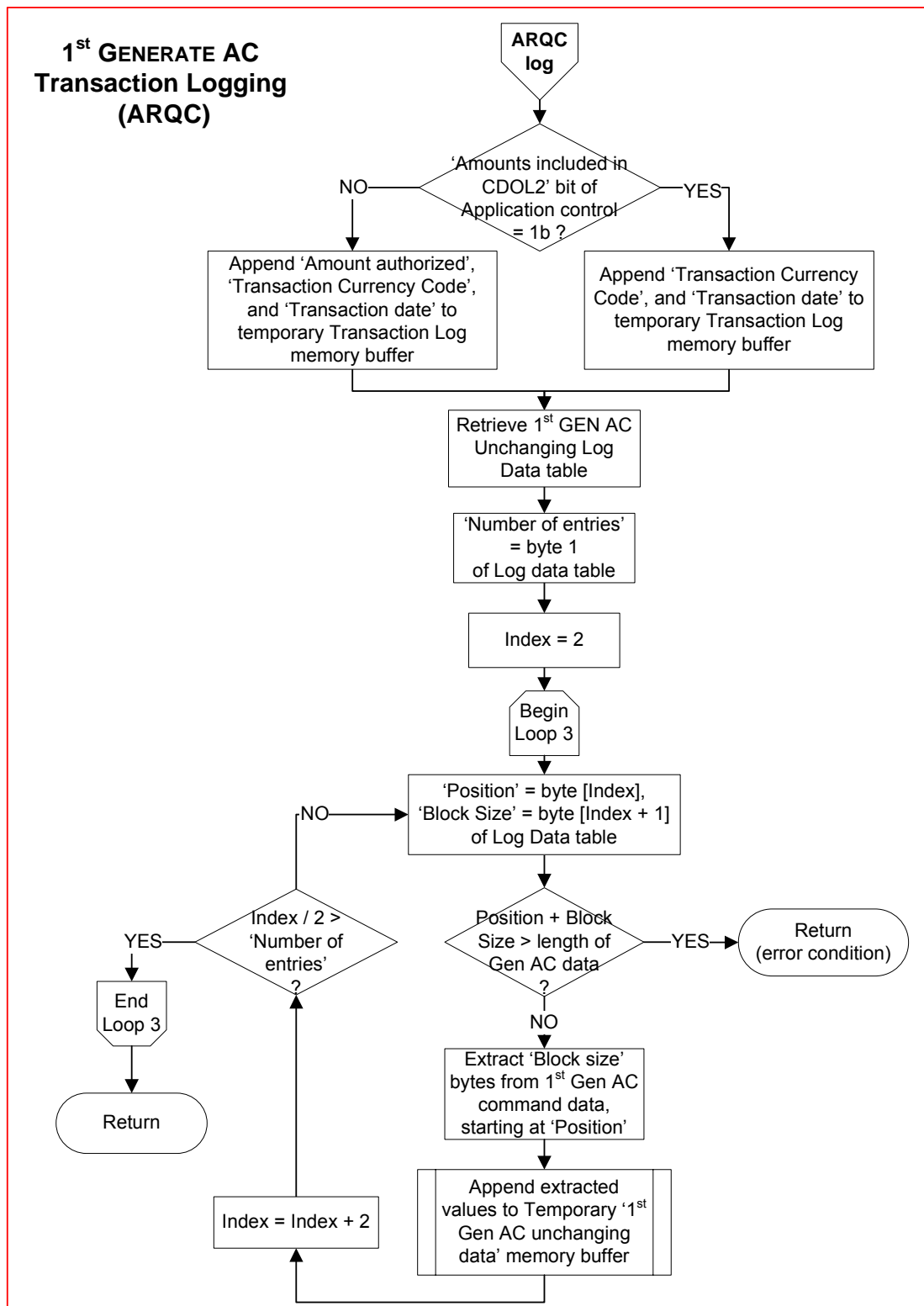


Flow 15-21 First GENERATE AC Transaction Logging

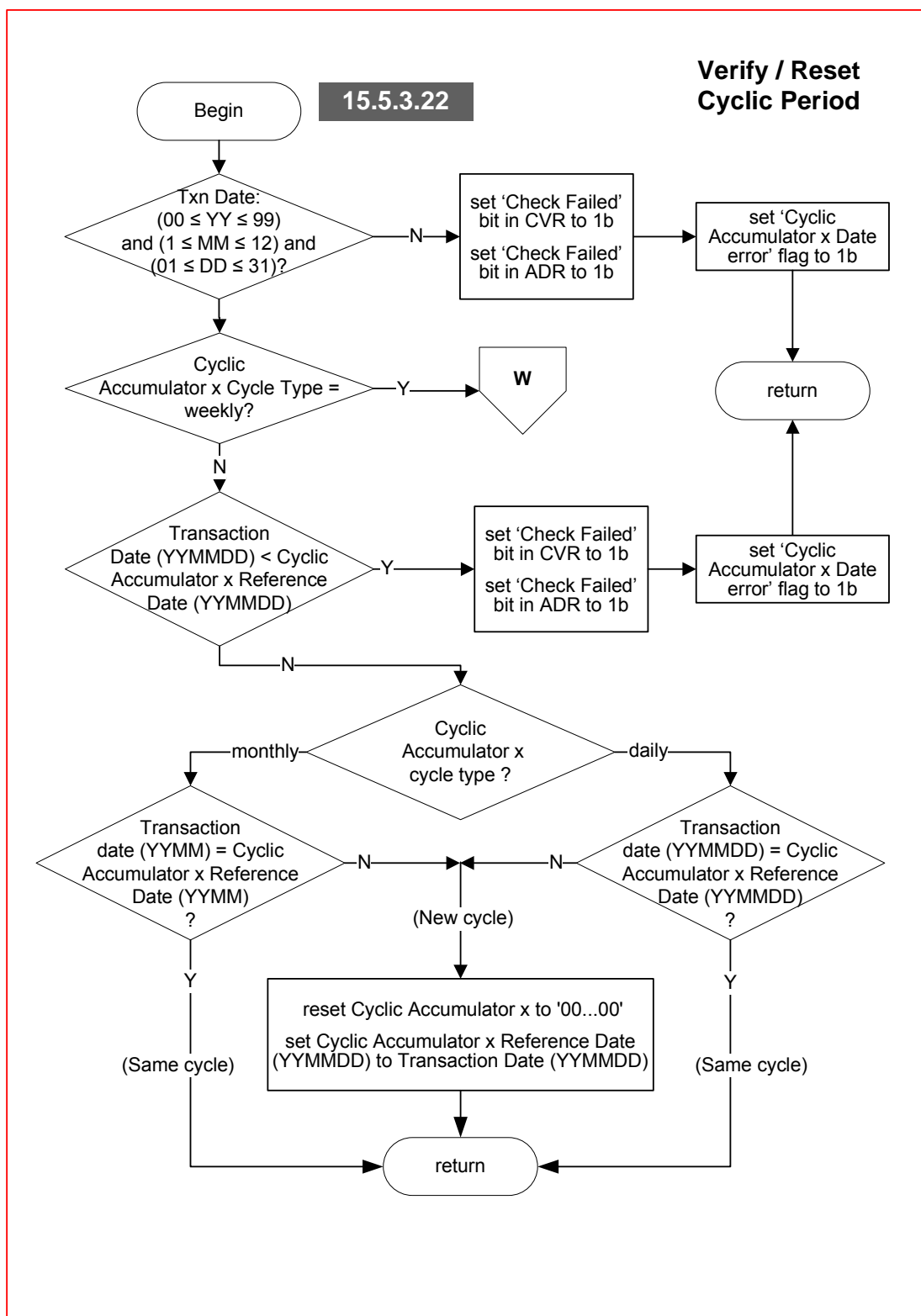
**Flow 15-21.1 First GENERATE AC Transaction Logging, continued**



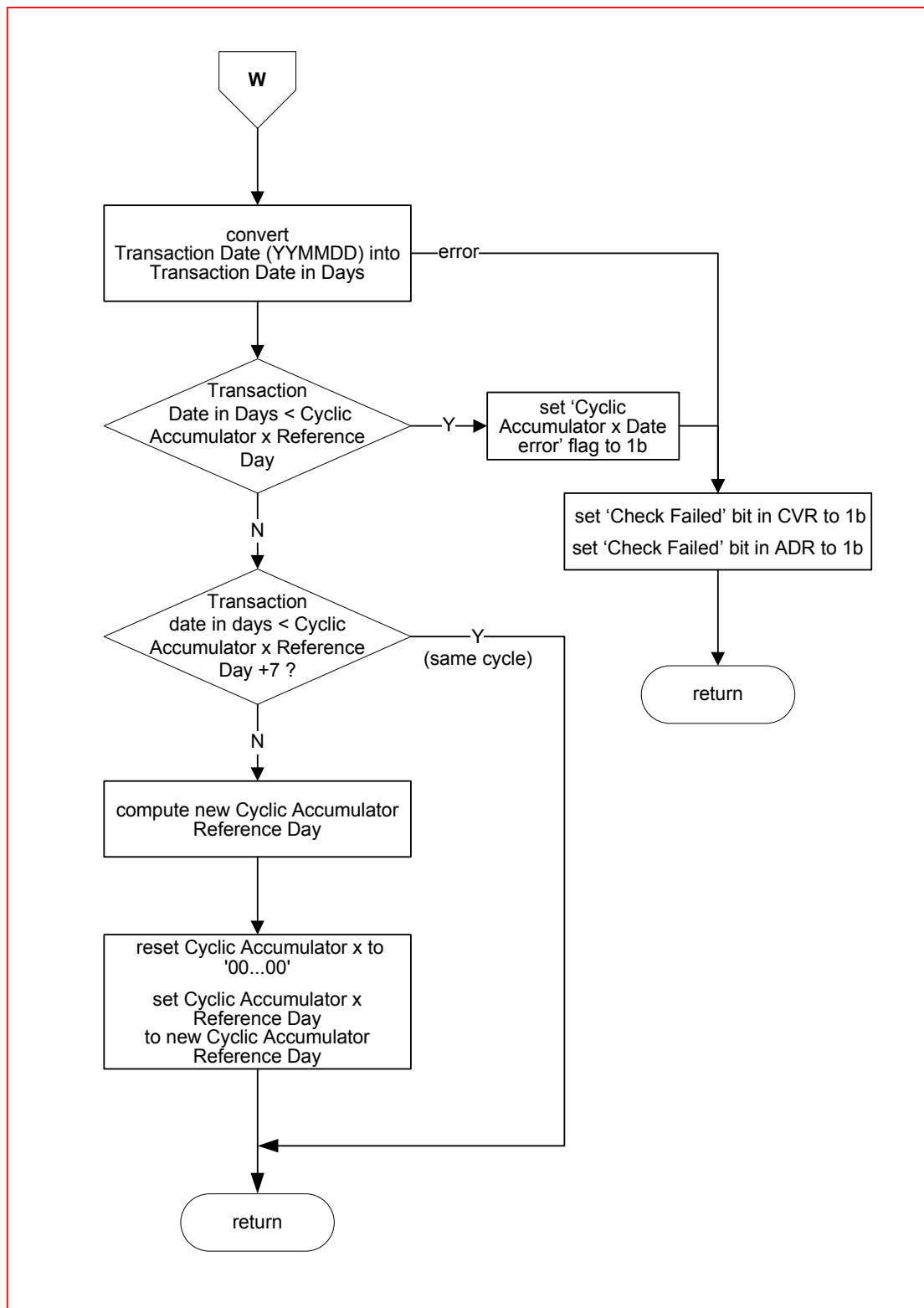
Flow 15-21.2 First GENERATE AC Transaction Logging, continued



Flow 15-21.3 First GENERATE AC Transaction Logging, continued



Flow 15-22 Verify/Reset Cyclic Period

**Flow 15-22.1 Verify/Reset Cyclic Period, continued**

16 Online Processing

This section is organised as follows:

- 16.1 Purpose
- 16.2 Sequence of Execution
 - 16.2.1 Prior Related Processing
 - 16.2.2 Subsequent Related Processing
- 16.3 Card Data
- 16.4 Terminal Data
- 16.5 Commands
- 16.6 Processing
 - 16.6.1 Online Request
 - 16.6.2 Online Response
 - 16.6.3 No Online Response
- 16.7 Function Flow Chart

16.1 Purpose

Online Processing allows the issuer's host computer to review and authorise or decline transactions using the issuer's host-based risk management parameters. In addition to performing traditional online fraud and credit checks, host authorisation systems may perform Online Card Authentication using a card-generated dynamic cryptogram and may consider offline processing results in the authorisation decision.

The response from the issuer may include post-issuance updates to the card and an issuer-generated cryptogram that the card can validate to assure that the response came from the valid issuer. This validation is called Issuer Authentication, and for CPA-compliant applications it is performed as part of Second Card Action Analysis (see section 17.5.2.2, Issuer Authentication Processing).

The CPA application takes no part in online processing, but it does use the results of online processing by the terminal and issuer in subsequent processing. Thus, this section briefly describes the online processing functions that relate to subsequent processing by CPA. Online processing functions that are also performed with magnetic stripe-read and key-entered transactions are not described.

Online processing is performed as described in EMV Book 3, section 10.9 and EMV Book 4, section 6.3.8.

16.2 Sequence of Execution

16.2.1 Prior Related Processing

Initiate Application Processing

The card sends the AIP to the terminal in response to the GET PROCESSING OPTIONS command. The AIP indicates that the card does not support Issuer Authentication using the EXTERNAL AUTHENTICATE command, but combines it with processing of the second GENERATE AC command.

First Card Action Analysis

The card returns the Application Cryptogram in response to the first GENERATE AC command.

16.2.2 Subsequent Related Processing

Second Card Action Analysis

If the authorisation response contains Issuer Authentication Data, the application will perform Issuer Authentication to verify that the response came from the genuine issuer (or its proxy). The card uses the results of Issuer Authentication in deciding whether to approve or decline the transaction, and whether to reset certain counters and indicators. If Issuer Authentication fails, subsequent transactions for the card may be sent online for authorisation until Issuer Authentication is successful.

Issuer-to-Card Script Processing

The terminal sends any issuer script commands received in the online response to the card.

First Card Action Analysis (Subsequent Transaction)

If online processing did not complete on a previous transaction, the card may send the transaction online for an authorisation.

16.3 Card Data

The card data used during Online Processing are listed and described in Table 16-1. For a detailed description of these data and their usage, see Annex L: Data Dictionary.

Data	Description	Template	Tag
Application Cryptogram	The online cryptogram (ARQC) value from the card.	—	'9F26'
Application Interchange Profile (AIP)	The AIP is sent to the terminal by the card during Initiate Application Processing. The AIP indicates that the card supports Issuer Authentication as part of processing the Second GENERATE AC command instead of using the EXTERNAL AUTHENTICATE command.	—	'82'
Application Transaction Counter	Counter of transactions initiated with the card application since the application was put on the card.	—	'9F36'
Cryptogram Information Data (CID)	Contains an indicator of the type of cryptogram. For transactions to be authorised online, the cryptogram type is an ARQC (Authorisation Request Cryptogram). An ARQC is designated by 10b in the first two bits of this field.	—	'9F27'

Table 16-1: Online Processing – Card Data

Data	Description	Template	Tag
Issuer Application Data	<p>Issuer Application Data is a mandatory field used to transmit data to the terminal for input to the online request message or clearing record. The coding of Issuer Application Data is described in Annex L: Data Dictionary. It contains the following data:</p> <ul style="list-style-type: none">• Length Indicator ('0F')• Common Core Indicator (Format Code and Cryptogram Version)• Derivation Key Index (DKI)• Card Verification Results (CVR)• Counters• Length Indicator ('0F')• Profile ID• Issuer discretionary data (which may include the value or accumulator balance of additional counters)	—	'9F10'

Table 16-1: Online Processing – Card Data, continued

16.4 Terminal Data

The online response from the issuer to the terminal contains data used in processing the second GENERATE AC command, as described in section 17. The online response may also contain Issuer Script data as described in section 18.

16.5 Commands

The Application Interchange Profile (AIP) from a CPA-compliant card indicates that the application does not support Issuer Authentication using the EXTERNAL AUTHENTICATE command, so the terminal does not send an EXTERNAL AUTHENTICATE command to the card. If the online response contains Issuer Authentication Data, it is sent to the card in the second GENERATE AC command during Second Card Action Analysis.

NOTE: The 'Issuer Authentication is supported' bit in the AIP is set to 0b to indicate that the card supports Issuer Authentication as part of processing the Second Generate AC command rather than using the External Authenticate command.

16.6 Processing

16.6.1 Online Request

The terminal initiates an online request if it receives an Authorisation Request Cryptogram (ARQC) in the GENERATE AC response from the card after First Card Action Analysis and the terminal supports online authorisations. The online request contains data previously received by the terminal from the card, but the card plays no role in the transmission of the online request to the issuer.

16.6.2 Online Response

If the terminal receives an online response, it will send data received in the online response (such as the Authorisation Response Code and Issuer Authentication Data) to the card during Second Card Action Analysis.

If the transaction is approved online, the terminal will request that the card approve the transaction by requesting a TC type Application Cryptogram in the second GENERATE AC command.

If the transaction is declined online, the terminal will request that the card decline the transaction by requesting an AAC type Application Cryptogram in the second GENERATE AC command.

16.6.3 No Online Response

If the terminal is unable to go online (either because it is not capable of going online, or because there is a problem preventing it from receiving an online response), it will use the TVR, IAC-Default, and TAC-Default to determine the type of Application Cryptogram to request from the card in the second GENERATE AC command. This processing is performed entirely within the terminal using:

- processing rules called the IAC-Default that were received from the card earlier in the transaction, and
- processing rules from the terminal called the TAC-Default.

The terminal compares bits in the IAC-Default and TAC-Default to the corresponding bits in the TVR.

If a TVR bit has the value 1b, and the corresponding bit in either the IAC-Default or TAC-Default also has the value 1b, the terminal will:

- request that the card decline the transaction offline by requesting an AAC type Application Cryptogram in the second GENERATE AC command.
- send the specific Authorisation Response Code Z3 to the card in Second Card Action Analysis to indicate that the terminal was unable to go online.

Otherwise, the terminal will:

- request that the card approve the transaction offline by requesting a TC type Application Cryptogram in the second GENERATE AC command.
- send the specific Authorisation Response Code Y3 to the card in Second Card Action Analysis to indicate that the terminal was unable to go online.

16.7 Function Flow Chart

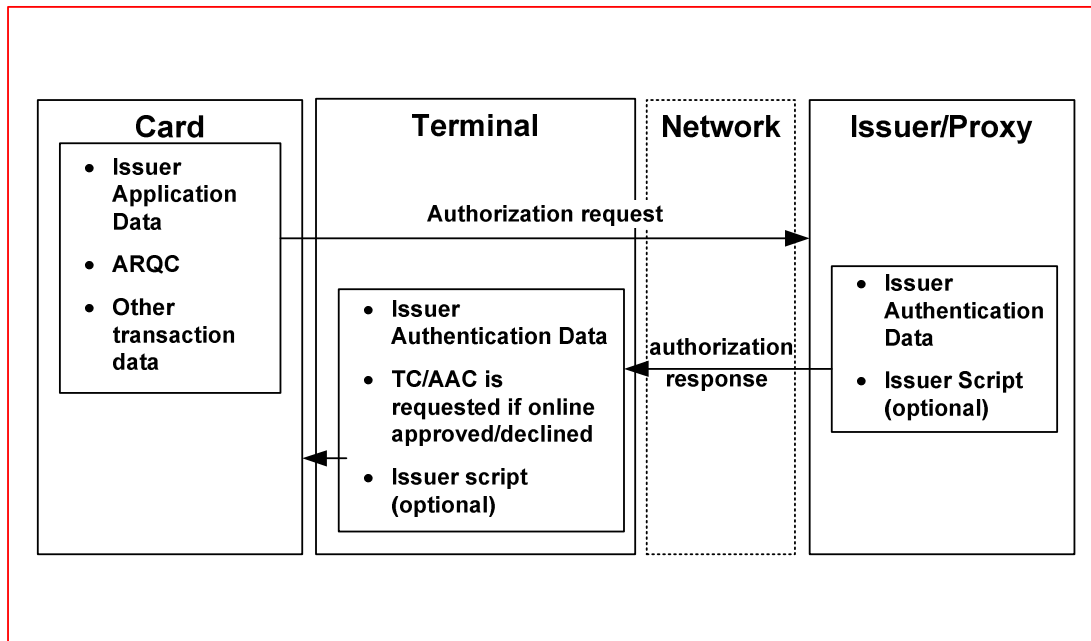


Figure 16-1: Online Processing Flow for Online Response

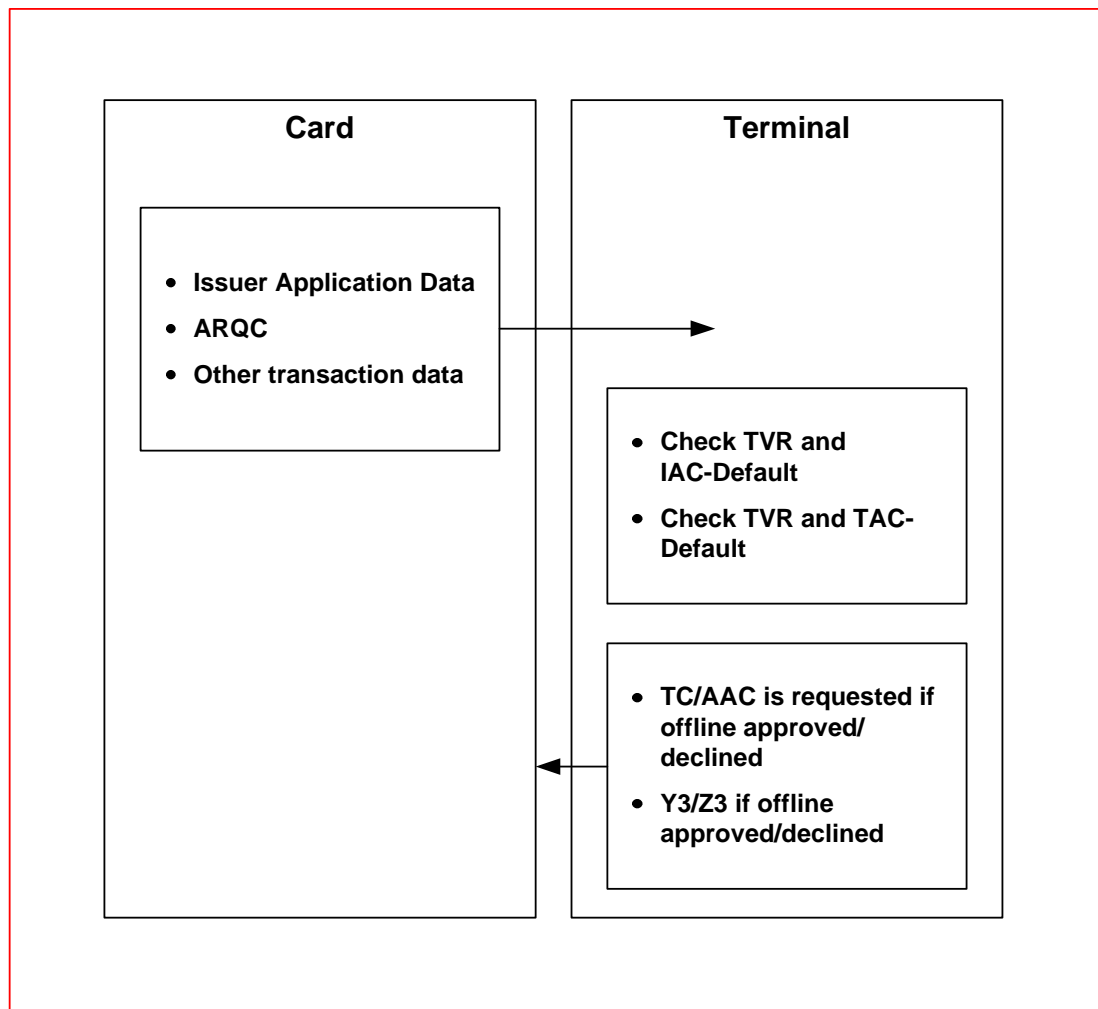


Figure 16-2: Online Processing Flow for No Online Response

17 Second Card Action Analysis

This section is organised as follows:

- 17.1 Purpose
- 17.2 Sequence of Execution
 - 17.2.1 Prior Related Processing
 - 17.2.2 Subsequent Related Processing
- 17.3 Card Data
- 17.4 Terminal Data
- 17.5 Second GENERATE AC Command
 - 17.5.1 Command Coding
 - 17.5.2 Configure Second Card Action Analysis
 - 17.5.3 Online Authorisation Completed
 - 17.5.4 Online Processing Requested, Online Authorisation Not Completed
 - 17.5.5 Application Approves Transaction (TC)
 - 17.5.6 Application Declines Transaction (AAC)
 - 17.5.7 CVR Updates
 - 17.5.8 Respond to GENERATE AC Command
- 17.6 Function Flow Charts

17.1 Purpose

Second Card Action Analysis is performed by the application to conclude transaction processing. Second Card Action Analysis includes the following:

- If Issuer Authentication Data is received, then the application performs Issuer Authentication.
- After an online authorisation, indicators and counters may be reset based upon Issuer Authentication status, application options, and indicators in the online response.
- If online processing was requested and the terminal was unable to go online, then the application performs additional analysis to determine whether the transaction should be approved or declined offline.
- An issuer's online approval may be changed to a decline based upon Issuer Authentication results and application options.
- Indicators and counters are set to reflect what has occurred during transaction processing.

Second Card Action Analysis is performed as described in *EMV Book 3*, section 10.11 and *EMV Book 4*, section 12.2.

17.2 Sequence of Execution

17.2.1 Prior Related Processing

Initiate Application Processing

The card determines the Profile ID of the Profile to be used for processing the transaction. The Profile Control associated with the Profile ID specifies how Card Risk Management is configured for processing the transaction.

Read Application Data

The terminal reads the CDOL2 from the card.

First Card Action Analysis

During First Card Action Analysis the application requests an online authorisation or responds with an offline approval, or an offline decline depending on the type of cryptogram requested by the terminal and the results of card risk management processing.

During Second Card Action Analysis, the terminal only issues a second GENERATE AC command to the card for transactions where the card requested an online authorisation in First Card Action Analysis.

The card does no Second Card Action Analysis processing for transactions where the card responded with an offline approval or offline decline during First Card Action Analysis.

Online Processing

During Online Processing, the terminal sends the authorisation request to the issuer. The issuer validates the application cryptogram, determines whether to approve or decline the transaction online, and whether to use the Card Status Update to update application data or conditions. The Issuer may then generate the Issuer Authentication Data sent to the terminal in the response message. The terminal sends the Issuer Authentication Data to the card in the second GENERATE AC command data because it is listed in CDOL2.

17.2.2 Subsequent Related Processing

First Card Action Analysis (Subsequent Transactions)

On following transactions, the application uses indicators and counters set or reset during Second Card Action Analysis in its processing decisions.

Second Card Action Analysis (Subsequent Transactions)

On following transactions, the application uses indicators and counters set or reset during Second Card Action Analysis in its processing decisions.

17.3 Card Data

The card data used in Second Card Action Analysis are listed and described in Table 17-1. For a detailed description of these data and their usage, see Annex L: Data Dictionary.

NOTE: Several data have multiple instances in the application. This is indicated with the notation Data Name x (for example, Accumulator 1 and Accumulator 2 may be referred to as Accumulator x).

Data	Description	Template	Tag
Accumulator x Balance	Represents the amount of offline spending available for Accumulator x, calculated as: Accumulator x Upper Limit minus Accumulator x value NOTE: This uses the upper limit for Accumulator x in the profile.	—	—
Accumulator Currency Code	A code indicating the currency in which an accumulator is managed.	—	—
Accumulator Profile Control for Accumulator x	Defines the profile-specific behaviour for Accumulator x in the Profile selected for the transaction; including whether to send the value in the Issuer Application Data, and which limit set to use.	—	—
Accumulator x	Represents a cumulative amount of transactions. May include offline approved transactions, and may also include online approved transactions. Transactions can be accumulated if they are in the accumulator currency, or (if currency conversion is allowed for the accumulator) in a currency that can be converted to the accumulator currency using the Currency Conversion Table for Accumulator x.	'BF30'	'DF0x'

Table 17-1: Second Card Action Analysis – Card Data
(continues)

Data	Description		Template	Tag
Accumulator x Control	Identifies the behaviour specific to Accumulator x independent of the Profile selected for the transaction; including which transactions are accumulated, and the currency code in which transactions are accumulated.		'BF32'	'DF0x'
Accumulator x Limit Set	Each Accumulator x has two Limit Sets, with one each of the following in each Limit Set:		'BF30'	'DF1x'
	Accumulator x Lower Limit	The lower of two limits for the maximum value of Accumulator x. A bit is set in the CVR and ADR when the value of Accumulator x exceeds this limit.		
	Accumulator x Upper Limit	The upper of two limits for the maximum value of Accumulator x. A bit is set in the CVR and ADR when the value of Accumulator x exceeds this limit.		
Application Control	Indicators used to activate or de-activate functions in the application.		—	'C1'
Application Cryptogram	<p>A cryptogram value that is returned by the card in the response to the GENERATE AC command. If the card response to the first GENERATE AC command was an Authorisation Request Cryptogram (ARQC), the terminal will send a second GENERATE AC command to the card. The cryptogram type sent in response to the second GENERATE AC command will be one of the following:</p> <ul style="list-style-type: none"> • An Application Authentication Cryptogram (AAC) for declines. • A Transaction Certificate (TC) for approvals. 		—	'9F26'

Table 17-1: Second Card Action Analysis – Card Data, continued

Data	Description		Template	Tag
Application Decisional Results (ADR)	Indicators used internal to the application to identify exception conditions that occurred during the current and previous transactions. If the terminal was unable to go online, the Card Issuer Action Code – Default (CIAC – Default) is compared to the Application Decisional Results to decide whether the transaction should be declined offline.		—	—
Application Transaction Counter (ATC)	A counter of the number of transactions initiated since the application was put on the card.		—	'9F36'
Card Risk Management Data Object List 2 (CDOL2)	A list of tags and lengths of the data elements that the terminal must pass to the card application in the Second GENERATE AC Command Data.		—	'8D'
Card Verification Results (CVR)	Indicates the results of offline processing from current and previous transactions from an application perspective. This data is transmitted online as part of the Issuer Application Data.		—	'9F52'
CIACs Entry x	Identifies the Profile-specific options selected by the issuer for processing transactions with CIACs ID = 'x'. The CIACs Entry chosen for the transaction contains the following CIAC used during second Card Action Analysis:		'BF34'	'DF0x'
	CIAC – Default	Compared to the Application Decisional Results to indicate situations when the issuer specifies a transaction is to be declined if the terminal is unable to go online.		
Counter Profile Control for Counter x	Identifies the counter- and profile-specific behaviour for Counter x in the Profile selected for the transaction; including whether to send the value in the Issuer Application Data, and which limit set to use.		—	—

Table 17-1: Second Card Action Analysis – Card Data, continued

Data	Description	Template	Tag
Counter x	Represents a count of applicable transactions since Counter x was last reset. May include offline approved transactions, international country transactions, non-accumulated transactions, online-approved transactions, or declined transactions.	'BF35'	'DF0x'
Counter x Control	Identifies the behaviour specific to Counter x independent of the Profile used for the transaction, such as which transactions are counted	'BF37'	'DF0x'
Counter x Limit Set	Each Counter x has two Limit Sets, with one each of the following in each Limit Set:	'BF35'	'DF1x'
	Counter x Lower Limit	The lower of two limits for the maximum value of Counter x. A bit is set in the CVR and ADR when the value of Counter x exceeds this limit.	
	Counter x Upper Limit	The upper of two limits for the maximum value of Counter x. A bit is set in the CVR and ADR when the value of Counter x exceeds this limit.	
Cryptogram Information Data (CID)	Returned to the terminal in the GENERATE AC response. The CID designates the type of cryptogram that is being returned. The CID indicates No advice required, and No information given for the Reason/advice/referral code.	—	'9F27'
Currency Conversion Table x	Contains the Target Currency Code and one or more Currency Conversion Parameters that may each be used to convert a transaction in a recognised currency into an approximate value for the transaction in the accumulator currency.	'BF38'	'DF0x'

Table 17-1: Second Card Action Analysis – Card Data, continued

Data	Description	Template	Tag
Cyclic Accumulator Profile Control for Cyclic Accumulator x	Identifies the accumulator- and profile-specific behaviour for Cyclic Accumulator x in the Profile selected for the transaction, including whether to perform the card risk management check, and which limit set to use.	—	—
Cyclic Accumulator x	Represents a cumulative amount of approved transactions processed within a cycle. Includes at least offline approved transactions, but may also include online approved transactions. The cycle may be a day, week, or month. Transactions can be accumulated if they are in the accumulator currency, or in a currency that can be converted to the accumulator currency using the Currency Conversion Table for Cyclic Accumulator x.	'BF42'	'DF0x'
Cyclic Accumulator x Control	Identifies the behaviour specific to Cyclic Accumulator x independent of the Profile used for the transaction, including the Accumulator Currency Code, which transactions are accumulated, and the length of the accumulator cycle.	'BF3A'	'DF0x'
Cyclic Accumulator x Limit	The limit for the value of Cyclic Accumulator x. A bit is set in the ADR when the value of Cyclic Accumulator x exceeds this limit.	—	—
Cyclic Accumulator x Reference Date	For a daily or monthly cyclic accumulator, represents the Transaction Date of the last transaction that reset Cyclic Accumulator x.	'BF42'	'DF1x'
Cyclic Accumulator x Reference Day	The Reference Day is the representation in Days (see Annex E) for the day at the beginning of the week in which Cyclic Accumulator x was last reset.	'BF42'	'DF2x'

Table 17-1: Second Card Action Analysis – Card Data, continued

Data	Description		Template	Tag
First GEN AC Unchanging Log Data Table	Provides information necessary to support optional extension of the data to be logged for a transaction. Identifies data that will not change during the transaction and thus does not need to be requested again from the terminal in the Second GENERATE AC command data. This data is saved in the application from the First GENERATE AC command data in case it is to be logged during processing of the second GENERATE AC command.		'BF40'	'DF03'
Issuer Application Data (IAD)	Informs the issuer about the application. Used to send the CVR and other application information to the issuer.		—	'9F10'
Issuer Country Code	Indicates the country of the issuer.		—	'5F28'
Issuer Options Profile Control x	Identifies the profile-specific options selected by the issuer for processing transactions with Issuer Options Profile Control ID = 'x'.		'BF3B'	'DF0x'
	Second GENERATE AC Command Data Length	Expected value for the length of the Second GENERATE AC Command Data		
Issuer Script Command Counter	An internal application counter that indicates the number of Issuer Script commands successfully processed.		—	—
Limits Entry x	An accumulator limit used as the limit for either the Maximum Transaction Amount or for one of the Cyclic Accumulators in the application.		'BF3C'	'DF0x'
Maximum Transaction Amount (MTA)	A limit on the value of Amount, Authorised for the transaction.		—	—

Table 17-1: Second Card Action Analysis – Card Data, continued

Data	Description	Template	Tag
MTA Profile Control	Identifies the profile-specific behaviour for the Maximum Transaction Amount card risk management check when processing a transaction in the Profile selected for the transaction, including which Currency Conversion Table and limit entry to use.	'BF3D'	'DF0x'
Previous Transaction History (PTH)	Indicators used to store information about previous transactions that is used in card risk management for subsequent transactions.	—	'C7'

Table 17-1: Second Card Action Analysis – Card Data, continued

Data	Description	Template	Tag
Profile Control x	Identifies which application resources are to be used when processing a transaction with Profile ID x. The following contents of Profile Control x are used in Card Action Analysis:	'BF3F'	'DFxx'
	Accumulator Profile Control ID for Accumulator x		Identifies the Accumulator Profile Control that defines the behaviour for Accumulator x when processing a transaction in the Profile.
	CIACs ID		Identifies the CIACs Entry x containing the CIAC – Default to be used when processing a transaction that was unable to go online.
	Counter Profile Control ID for Counter x		Identifies the Counter Profile Control that defines the behaviour for Counter x when processing a transaction in the Profile.
	Cyclic Accumulator Profile Control ID for Cyclic Accumulator x		Identifies the Cyclic Accumulator Profile Control that defines the behaviour for Cyclic Accumulator x when processing a transaction in the Profile.
	Issuer Options Profile Control ID		Identifies the Issuer Options Profile Control to be used when processing the transaction.
	MTA Profile Control ID		Identifies the MTA Profile Control that defines the behaviour for the Maximum Transaction Amount card risk management check when processing a transaction in the Profile.
	VLP Profile Control ID		Identifies the Accumulator Profile Control to be used as the VLP Profile Control that defines the behaviour for the VLP Available Funds when processing a transaction in the Profile.
Profile ID	Identifies the Profile Control to use in configuring the application behaviour for the transaction environment.	—	—

Table 17-1: Second Card Action Analysis – Card Data, continued

Data	Description	Template	Tag
Second GEN AC Log Data Table	Provides information necessary to support optional extension of the data to be logged for a transaction.	'BF40'	'DF02'
Signed Dynamic Application Data	The signature generated by the card at transaction time as part of the response to a GENERATE AC command with CDA processing. The card generates this signature using a hash of dynamic data from the terminal and card. The card computes the Signed Dynamic Application Data with the ICC Private Key. The format of the Signed Dynamic Application Data is shown in <i>EMV Book 2</i> , Table 16.	—	'9F4B'
Source Currency Code	The currency code of the currency from which amounts may be converted using a single Currency Conversion Parameter in the Currency Conversion Table. A Currency Conversion Parameter may only be used if the Transaction Currency Code matches the Source Currency Code in the Currency Conversion Parameter.	—	—
Target Currency Code	The currency code of the currency to which amounts may be converted using a Currency Conversion Table. A Currency Conversion Table may only be used to convert an amount if the Target Currency Code matches the Accumulator Currency Code or MTA Currency Code.	—	—
Transaction Date in Days	Represents the Transaction Date as the number of days since a base date (see Annex E).	—	—
VLP Available Funds	An accumulator that is decremented by the transaction amount when a VLP transaction is approved.	—	'9F79'

Table 17-1: Second Card Action Analysis – Card Data, continued

17.4 Terminal Data

The terminal data used in Second Card Action Analysis are listed and described in Table 17-2. For a detailed description of these data and their usage, see Annex L: Data Dictionary.

Req 17.1 (Use first GENERATE AC data):

If any of the terminal data elements listed below (used for card risk management, Issuer Authentication, or Application Cryptogram generation) was sent in the first GENERATE AC command data but is not present in the second GENERATE AC command data, then the application shall use the value sent in the first GENERATE AC command data for processing the second GENERATE AC command.

Data	Description	Template	Tag
Amount, Authorised	The amount of the current transaction. Used in card risk management and Application Cryptogram generation.	—	'9F02'
Amount, Other	Represents the cashback amount associated with the transaction. Used in card risk management and Application Cryptogram generation.	—	'9F03'
Authorisation Response Code	The values listed below may be provided to the card by the terminal if the transaction did not go online. Other values may be sent by the issuer. The following values are used by the application to determine that the terminal was able to go online. <ul style="list-style-type: none">• Y3 = Unable to go online (offline approved)• Z3 = Unable to go online (offline declined)	—	'8A'

Table 17-2: Second Card Action Analysis – Terminal Data
(continues)

Data	Description	Template	Tag
Issuer Authentication Data	<p>Contains online response data from the issuer, including:</p> <ul style="list-style-type: none"> • Authorisation Response Cryptogram – used to verify that the online response came from the issuer • Card Status Update – used to indicate whether the issuer approves the transaction and to request updates to the application (such as blocking the application or resetting counters) <p>NOTE: Issuer Authentication Data may also contain Proprietary Authentication Data. Support for receipt of Proprietary Authentication Data and functionality associated with the additional data is allowed as additional functionality (see section 19.3.5).</p>	—	'91'
Terminal Country Code	Indicates the country where the terminal is located. Used in card risk management and Application Cryptogram generation.	—	'9F1A'
Terminal Verification Results (TVR)	Used to record offline terminal processing results, such as SDA failure or floor limit exceeded. Used in card risk management and Application Cryptogram generation.	—	'95'
Transaction Currency Code	Indicates the currency code of the transaction. Used in card risk management and Application Cryptogram generation.	—	'5F2A'
Transaction Date	Local date that the transaction was authorised.	—	'9A'
Unpredictable Number	A number used to provide variability and uniqueness to data sent by the application.	—	'9F37'

Table 17-2: Second Card Action Analysis – Terminal Data, continued

17.5 Second GENERATE AC Command

The second GENERATE AC command is used by the terminal to request that the card provide a second Application Cryptogram and the card's decision on further action.

Figure 17-1 shows an overview of Second Card Action Analysis processing and indicates the section where each processing step is described.

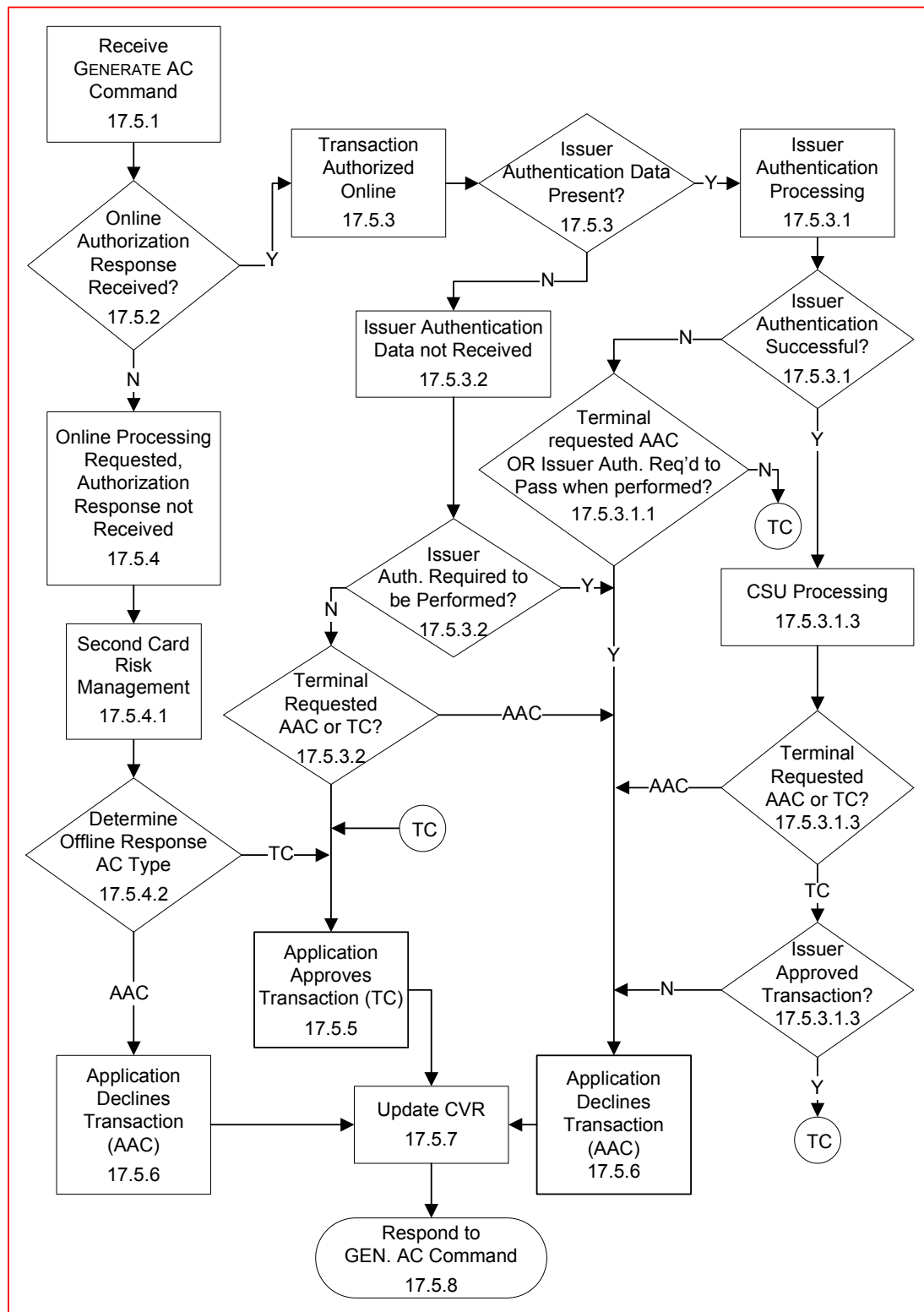


Figure 17-1: Second Card Action Analysis Processing Flow

17.5.1 Command Coding

The P1 parameter of the GENERATE AC command indicates the type of cryptogram the terminal is requesting and whether the transaction is eligible for CDA. Table 17-4 shows the format of P1. The data portion of the command contains the data requested in the CDOL2.

Code	Value
CLA	'80'
INS	'AE'
P1	Reference Control Parameter
P2	'00'
Lc	Var.
Data	Second GENERATE AC Command Data
Le	'00'

Table 17-3: Second GENERATE AC Command Message

NOTE: Second GENERATE AC Command Data is called Transaction-related data in EMV Book 3.

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
x	x							Cryptogram Type
0	0							AAC
0	1							TC
1	0							Not allowed for second GENERATE AC
1	1							RFU
		x						RFU
		x						CDA Requested
		0						CDA Not Requested
		1						CDA Requested
				x	x	x	x	RFU

Table 17-4: Reference Control Parameter Coding for Second GENERATE AC

NOTE: CDA Requested and CDA Not Requested are respectively called CDA signature requested and CDA signature not requested in EMV Book 3.

Combined DDA/AC Generation is only supported by the Dynamic-RSA implementer-option.

Req 17.2 (Interpret command data):

If the 'Amounts included in CDOL2' bit in the Application Control has the value 1b, then Second GENERATE AC Command Data shall be interpreted by the application as consisting of the data elements listed in Table 17-5.

Data Element	Length
<i>Issuer Authentication Data</i>	<i>8</i>
<i>Authorisation Response Code</i>	<i>2</i>
<i>TVR</i>	<i>5</i>
<i>Unpredictable Number</i>	<i>4</i>
<i>Amount, Authorised</i>	<i>6</i>
<i>Amount, Other</i>	<i>6</i>
<i>Second GENERATE AC Extension Data</i>	<i>Var.</i>

Table 17-5: Second GENERATE AC Command Data: Amounts in CDOL2

If the 'Amounts included in CDOL2' bit in the Application Control has the value 0b, then Second GENERATE AC Command Data shall be interpreted by the application as consisting of the data elements listed in Table 17-6.

Data Element	Length
<i>Issuer Authentication Data</i>	<i>8</i>
<i>Authorisation Response Code</i>	<i>2</i>
<i>TVR</i>	<i>5</i>
<i>Unpredictable Number</i>	<i>4</i>
<i>Second GENERATE AC Extension Data</i>	<i>Var.</i>

Table 17-6: Second GENERATE AC Command Data: No Amounts in CDOL2

The application does not use the Second GENERATE AC Extension Data as an individual element, but always as part of the GENERATE AC Command Data. The Second GENERATE AC Extension Data might be used for transaction logging.

Req 17.3 (Support Extension Data length):

At a minimum, the application shall support Second GENERATE AC Extension Data length of up to 32 bytes.

17.5.1.1 Command Format Validation**Req 17.4 (Check cryptogram type):**

If the 'Cryptogram Type' bits in the P1 parameter are set to the value 10b or to the value 11b, then the card shall discontinue processing the command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6A86' (Incorrect Parameters, P1-P2).

Req 17.5 (Check P2):

If the P2 parameter is not set to the value '00', then the card shall discontinue processing the command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6A86' (Incorrect Parameters, P1-P2).

Req 17.6 (Check value of Lc):

If the value of Lc does not equal the value of Second GENERATE AC Command Data Length parameter in the Issuer Options Profile Control for the transaction, then the card shall discontinue processing the command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6700' (Wrong Length).

Req 17.7 (Check value of Lc if Amounts included in CDOL2 = 0b):

*If the 'Amounts included in CDOL2' bit in the Application Control has the value 0b, **and** the value of Lc is less than 19 (the minimum length for CDOL2 related data without amounts included), then the card shall discontinue processing the command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6700' (Wrong Length).*

Req 17.8 (Check value of Lc if Amounts included in CDOL2 = 1b):

*If the 'Amounts included in CDOL2' bit in the Application Control has the value 1b, **and** the value of Lc is less than 31 (the minimum length for CDOL2 related data with amounts included), then the card shall discontinue processing the command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6700' (Wrong Length).*

17.5.2 Configure Second Card Action Analysis

The application only performs Second Card Action Analysis processing when the application requested an online authorisation during First Card Action Analysis.

The Profile ID selected during Initiate Application Processing identifies the Profile Control to be used to configure the application behaviour for Card Action Analysis. See section 15.5.2 for a detailed description of configuring Profile Behaviour.

The Amount, Authorised and Amount, Other may have been updated in the second GENERATE AC command data. If so, the application uses the updated amounts in processing the second GENERATE AC command and generation of the Application Cryptogram, because these are the amounts the terminal will send to the issuer for the second GENERATE AC.

Req 17.9 (Use specified amounts if Amounts Included in CDOL2 = 0b):

If the 'Amounts Included in CDOL2' bit in the Application Control has the value 0b, then the application shall use the Amount, Authorised and Amount, Other received in the First GENERATE AC Command Data for processing the second GENERATE AC command.

Req 17.10 (Use specified amounts if Amounts Included in CDOL2 = 1b):

If the 'Amounts Included in CDOL2' bit in the Application Control has the value 1b, then the application shall use the Amount, Authorised and Amount, Other received in the second Generate AC command data for processing the second Generate AC command.

Req 17.11 (Check CDA Failed):

If the 'CDA Failed' bit in the TVR received in the second GENERATE AC command data has the value 1b, then the application shall set the 'Offline Data Authentication Failed on Previous Transaction' bit in the Previous Transaction History (PTH) to the value 1b.

If a PUT DATA command received before the second GENERATE AC command updated data elements used during processing of the second GENERATE AC command, then the updated values are used during processing of the second GENERATE AC command.

Req 17.12 (Update Profile Configuration for Tag '71' Script):

If any application data used during processing of the second GENERATE AC command was successfully updated by a tag '71' script command received in the current transaction, then the updated values resulting from successfully processing the script command shall be used in processing the second GENERATE AC command.

The transaction date format must be correctly formatted in order for several requirements to be correctly processed. The following requirement describes the conditions for a valid date format.

Req 17.13 (Valid date format):

If **all** of the following are true:

- Transaction Date byte 1 (YY) is in the range '00' to '99'
- **and** Transaction Date byte 2 (MM) is in the range '01' to '12'
- **and** Transaction Date byte 3 (DD) is in the range '01' to '31'

then the Transaction Date format is valid.

An issuer script command may have reconfigured the Cyclic Accumulators (for example, changing monthly accumulator to a weekly accumulator) since the first GENERATE AC command processing was performed, so the application will again check whether the transaction date is correctly formatted, and whether there is a date error for each Cyclic Accumulator *x* that is active for the transaction.

Req 17.14 (Date error for Cyclic Accumulator *x*):

For each Cyclic Accumulator *x* that is active for the transaction, if **any** of the following is true:

- the Transaction Date is not valid
- **or both** of the following are true:
 - the 'Cycle Type' in the Cyclic Accumulator *x* Control has the value 01b (daily) **or** 11b (monthly)
 - **and** the Transaction Date is less than the Cyclic Accumulator *x* Reference Date
- **or both** of the following are true:
 - the 'Cycle Type' in the Cyclic Accumulator *x* Control has the value 10b (weekly)
 - **and** the Transaction Date in Days (see Annex E) is less than the Cyclic Accumulator *x* Reference Day

then Cyclic Accumulator *x* has a Transaction Date error for the transaction, and the application shall:

- set the 'Check Failed' bit in the CVR to the value 1b
- set the 'Check Failed' bit in the ADR to the value 1b

For each Cyclic Accumulator *x* that is active for the transaction and does not have a date error, the application will also determine whether the transaction date is in the current cycle or a new cycle for the cyclic accumulator.

Req 17.15 (Current Cycle for Cyclic Accumulator *x*):

For each value of *x* for which Cyclic Accumulator *x* is active for the transaction, if **any** of the following is true:

- **both** of the following are true:
 - the 'Cycle Type' in the Cyclic Accumulator *x* Control has the value 01b (daily)
 - **and** the Transaction Date is equal to the Cyclic Accumulator *x* Reference Date
- **or both** of the following are true:
 - the 'Cycle Type' in the Cyclic Accumulator *x* Control has the value 10b (weekly)
 - **and** the Transaction Date in Days (see Annex E) is less than the Cyclic Accumulator *x* Reference Day plus 7
- **or both** of the following are true:
 - the 'Cycle Type' in the Cyclic Accumulator *x* Control has the value 11b (monthly)
 - **and** the year and month of the Transaction Date is equal to the year and month of the Cyclic Accumulator *x* Reference Date

then the Transaction Date is in the current cycle for Cyclic Accumulator *x*.

If the transaction is within a new cycle, then the application resets the reference date for the cycle and resets the value of the cyclic accumulator to zero.

Req 17.16 (Reset daily or monthly cyclic accumulator for New Cycle):

For each value of *x* for which Cyclic Accumulator *x* is active for the transaction, if **all** of the following are true:

- Cyclic Accumulator *x* does not have a transaction date error for the transaction (see the "Date error for Cyclic Accumulator *x*" requirement in section 17)
- **and** the Transaction Date is not in the current cycle for Cyclic Accumulator *x*
- **and** the 'Cycle Type' in the Cyclic Accumulator *x* Control has the value 01b (daily) or 11b (monthly)

then the Transaction Date is in a new cycle for Cyclic Accumulator *x*, and the application shall:

- reset the Cyclic Accumulator *x* Reference Date to the Transaction Date
- reset Cyclic Accumulator *x* to zero.

Req 17.17 (Reset weekly cyclic accumulator for New Cycle):

For each value of *x* for which Cyclic Accumulator *x* is active for the transaction, if **all** of the following are true:

- Cyclic Accumulator *x* does not have a transaction date error for the transaction (see the "Date error for Cyclic Accumulator *x*" requirement in section 17)
- **and** the Transaction Date is not in the current cycle for Cyclic Accumulator *x*
- **and** the 'Cycle Type' in the Cyclic Accumulator *x* Control has the value 10b (weekly)

then the Transaction Date is in a new cycle for Cyclic Accumulator *x*, and the application shall:

- reset the Cyclic Accumulator *x* Reference Day to the Transaction Date Reference Day
- reset Cyclic Accumulator *x* to zero.

The type of Authorisation Response Code in this command determines which path the Second Card Action Analysis processing follows:

Req 17.18 (Processing if Authorisation Response Code ≠ Y3 or Z3):

If the Authorisation Response Code in the second GENERATE AC command has a value **other** than Y3 or Z3 (indicating that the online authorisation completed and the terminal received a response from the issuer during Online Processing), then the application shall process the transaction as described in section 17.5.3.

Req 17.19 (Processing if Authorisation Response Code = Y3 or Z3):

If the Authorisation Response Code has a value of Y3 or Z3 (indicating that online authorisation was requested but not completed during Online Processing, either because the terminal did not support online processing or because a response from the issuer was not received), then the application shall process the transaction as described in section 17.5.4.

17.5.3 Online Authorisation Completed

The following processing is performed if the transaction successfully went online (that is, the Authorisation Response Code returned in the second Generate AC command data has a value other than Y3 or Z3).

Req 17.20 (Reset Unable to Go Online for online response received):

When the Second GENERATE AC Command from the terminal contains an Authorisation Response Code indicating that online processing completed (that is, has a value other than Y3 or Z3), the application shall set the 'Unable to Go Online' bit in the PTH to the value 0b.

Req 17.21 (Check Offline Data Authentication Failed):

*If **all** of the following are true:*

- the 'SDA Failed' bit in the TVR received in the Second GENERATE AC Command Data has the value 0b*
- and** the 'DDA Failed' bit in the TVR received in the Second GENERATE AC Command Data has the value 0b*
- and** the 'CDA Failed' bit in the TVR received in the Second GENERATE AC Command Data has the value 0b*

then the application shall reset the 'Offline Data Authentication Failed on Previous Transaction' bit in the PTH to the value 0b.

Req 17.22 (Check whether Issuer Authentication Data received):

If Issuer Authentication Data is received in the Second GENERATE AC Command Data (that is, the Issuer Authentication Data portion of the Second GENERATE AC Command Data is not all zeroes), then the application shall:

- set the 'Issuer Authentication Not Performed' bit in the CVR to the value 0b.*
- set the 'Issuer Authentication Data Not Received in Online Response' bit in the PTH to the value 0b.*
- perform Issuer Authentication processing as described in section 17.5.3.1.*

Otherwise, the application shall:

- set the 'Issuer Authentication Not Performed' bit in the CVR to the value 1b.*
- set the 'Issuer Authentication Data Not Received in Online Response' bit in the PTH to the value 1b.*
- perform the processing in section 17.5.3.2.*

17.5.3.1 Issuer Authentication Data Received

When Issuer Authentication Data is returned in the second GENERATE AC command data, the application performs Issuer Authentication using the following steps:

1. Parse out the CSU from the received Issuer Authentication Data.
2. Using the CSU recovered in step 1 and the ARQC sent in the first GENERATE AC response, generate an ARPC as specified in *EMV Book 2* for a Common Core Definitions application with Cryptogram Version '5'.

NOTE: Issuer Authentication Data may also contain Proprietary Authentication Data. Support for receipt of Proprietary Authentication Data and functionality associated with the additional data is allowed as additional functionality (see section 19.3.5).

3. Compare the ARPC generated in step 2 with the ARPC received in bytes 1 through 4 of the received Issuer Authentication Data.

Req 17.23 (Branch if Issuer Authentication fails):

If Issuer Authentication fails (that is, if the ARPCs do not match), then the application shall continue with the processing in section 17.5.3.1.1.

Req 17.24 (Branch if Issuer Authentication passes):

If Issuer Authentication passes (that is, if the ARPCs match), then the application shall continue with the processing in section 17.5.3.1.2.

17.5.3.1.1 Issuer Authentication Failed

The processing of this section is performed if Issuer Authentication fails.

Req 17.25 (Set Issuer Authentication Failed):

The application shall set:

- the 'Issuer Authentication Failed' bit in the CVR to the value 1b.
- the 'Issuer Authentication Failed' bit in the PTH to the value 1b.

Req 17.26 (Decline if Issuer Authentication Required to Pass):

If the 'Issuer Authentication Required to Pass when performed' bit in the Application Control has the value 1b, then the application shall decline the transaction as described in section 17.5.6 and section 17.5.8.

Req 17.27 (Reset bits if Issuer Authentication failed):

If **both** of the following are true:

- the 'Issuer Authentication Required to Pass when performed' bit in the Application Control has the value 0b,
- and** the 'Issuer Authentication Requirements Apply to Resetting of Non-Velocity-Checking Indicators and Counters' bit in the Application Control has the value 0b,

then the application shall reset bits as indicated in Table 17-7.

reset the following bit:	in:	to the value:	condition
<i>Script Failed</i>	<i>PTH</i>	<i>0b</i>	<i>If no script command failed in this transaction</i>
<i>Script Received</i>	<i>PTH</i>	<i>0b</i>	<i>If no script command was received in this transaction</i>
<i>Go Online on Next Transaction Was Set</i>	<i>CVR</i>	<i>0b</i>	<i>always</i>
<i>Go Online on Next Transaction</i>	<i>PTH</i>	<i>0b</i>	<i>always</i>
<i>Last Online Transaction not Completed</i>	<i>CVR</i>	<i>0b</i>	<i>always</i>
<i>Last Online Transaction not Completed</i>	<i>PTH</i>	<i>0b</i>	<i>always</i>

Table 17-7: Bits to Reset if Non-required Issuer Authentication Fails

Req 17.28 (Decline if AAC requested):

If **both** of the following are true:

- the 'Issuer Authentication Required to Pass when performed' bit in the Application Control has the value 0b,
- and** the terminal requested an AAC type Application Cryptogram in the second Generate AC command,

then the application shall decline the transaction as described in section 17.5.6 and section 17.5.8.

Req 17.29 (Reset accumulators and counters if TC requested):

If **all** of the following are true:

- the 'Issuer Authentication Required to Pass when performed' bit in the Application Control has the value 0b,
- **and** the terminal requested a TC type Application Cryptogram in the second Generate AC command,
- **and** the 'Issuer Authentication Requirements Apply to Resetting of Velocity-Checking Counters' bit in the Application Control has the value 0b,

then:

- for each Accumulator *x* that is active for the transaction:
 - if the 'Reset Accumulator with online response' bit in the Accumulator Profile Control for Accumulator *x* in the profile has the value 1b, then the application shall reset Accumulator *x* to the value 0
- for each Counter *x* that is active for the transaction:
 - if the 'Reset Counter with online response' bit in the Counter Profile Control for Counter *x* in the profile has the value 1b, then the application shall reset Counter *x* to the value 0

Req 17.30 (Reset VLP Available Funds when Issuer Authentication failed):

If **all** of the following are true:

- the 'Issuer Authentication Required to Pass when performed' bit in the Application Control has the value 0b,
- **and** the 'Issuer Authentication Requirements Apply to Resetting of Velocity-Checking Indicators and Counters' bit in the Application Control has the value 0b,
- **and** the terminal requested a TC type Application Cryptogram in the second Generate AC command,
- **and** VLP is supported
- **and** VLP Available Funds is active for the transaction
- **and** the 'Reset Accumulator with online response' bit in the VLP Profile Control has the value 1b,

then the application shall reset the VLP Available Funds to the VLP Funds Limit.

Req 17.31 (Reset Last Online Transaction Date when Issuer Authentication failed):

If **all** of the following are true:

- the 'Issuer Authentication Required to Pass when performed' bit in the Application Control has the value 0b,
- **and** the 'Issuer Authentication Requirements Apply to Resetting of Velocity-Checking Indicators and Counters' bit in the Application Control has the value 0b,
- **and** the terminal requested a TC type Application Cryptogram in the second Generate AC command,
- **and** the 'Reset Maximum Number of Days offline with online response' bit in the Issuer Options Profile Control has the value 1b,
- **and** the Transaction Date format is valid (see Req 17.13)

then:

- the application shall reset the Last Online Transaction Date to the value of the Transaction Date
- if the Transaction Date format is not valid, then the application shall set the 'Check Failed' bit in the CVR to the value 1b

Req 17.32 (Correct cyclic accumulator reference when Issuer Authentication Failed):

For each value of x for which Cyclic Accumulator x is active for the transaction, if **all** of the following are true:

- the 'Issuer Authentication Required to Pass when performed' bit in the Application Control has the value 0b,
- **and** the terminal requested a TC type Application Cryptogram in the second Generate AC command,
- **and** the Transaction Date format is valid (see the "Valid date format" requirement)

then

- if the 'Cycle Type' in the Cyclic Accumulator x Control has the value 01b (daily) or 11b (monthly), **and** the Transaction Date is less than the Cyclic Accumulator x Reference Date; the application shall correct the Cyclic Accumulator Reference Date by setting the Cyclic Accumulator x Reference Date to the value of the Transaction Date
- if the 'Cycle Type' in the Cyclic Accumulator x Control has the value 10b (weekly), **and** the Transaction Date in Days (see Annex E) is less than the Cyclic Accumulator x Reference Day; the application shall correct the Cyclic Accumulator Reference Day by setting the Cyclic Accumulator x Reference Day to the value of the Transaction Date Reference Day

Req 17.33 (Add Amount, Authorised to cyclic accumulator):

*After correcting the Cyclic Accumulator reference date, if necessary (see Req 17.13), if **both** of the following are true:*

- *the 'Issuer Authentication Required to Pass when performed' bit in the Application Control has the value 0b,*
- ***and** the terminal requested a TC type Application Cryptogram in the second Generate AC command,*

then for each value of x for which all of the following are true:

- *Cyclic Accumulator x is active*
- ***and** the 'Allow Accumulation' bit in the Cyclic Accumulator Profile Control for Cyclic Accumulator x in this profile has the value 1b*
- ***and** the 'Include Online Approvals' bit in the Cyclic Accumulator x Control has the value 1b*
- ***and** the Transaction Date format is valid*

then:

- *If the Transaction Currency Code matches the Accumulator Currency Code, then the application shall add Amount, Authorised to Cyclic Accumulator x.*
 - *If adding the Amount, Authorised to the value of Cyclic Accumulator x would exceed the value '99 99 99 99 99 99', then Cyclic Accumulator x shall be set to the value '99 99 99 99 99 99'*
- *Otherwise (the Transaction Currency Code does not match the Accumulator Currency Code), if the Transaction Currency Code matches the Source Currency Code in one of the Currency Conversion Parameters in the Currency Conversion Table active for Cyclic Accumulator x¹, then the application shall add Amount, Authorised converted to the accumulator currency (using the Currency Conversion Table active for Cyclic Accumulator x) to Cyclic Accumulator x.*
 - *If adding the Amount, Authorised converted to the accumulator currency to the value of Cyclic Accumulator x would exceed the value '99 99 99 99 99 99', then Cyclic Accumulator x shall be set to the value '99 99 99 99 99 99'*

¹ If the Currency Conversion Table ID in the Cyclic Accumulator Profile Control for Cyclic Accumulator x for the transaction has the value 'F', then no Currency Conversion Table is active for Cyclic Accumulator x.

Req 17.34 (Approve transaction when Issuer Authentication failed):

If **both** of the following are true:

- the 'Issuer Authentication Required to Pass when performed' bit in the Application Control has the value 0b,
- **and** the terminal requested a TC type Application Cryptogram in the second Generate AC command,

then after completing the other processing in this section, the application shall approve the transaction as described in section 17.5.5 and section 17.5.8.

17.5.3.1.2 Issuer Authentication Passed

The processing of this section is performed if Issuer Authentication passes.

Req 17.35 (Reset bits if Issuer Authentication passed):

The application shall reset bits as indicated in Table 17-8.

reset the following bit:	in:	to the value:	condition
<i>Issuer Authentication Failed</i>	<i>CVR</i>	<i>0b</i>	<i>always</i>
<i>Issuer Authentication Failed</i>	<i>PTH</i>	<i>0b</i>	<i>always</i>
<i>Script Failed</i>	<i>PTH</i>	<i>0b</i>	<i>If no script command failed in this transaction</i>
<i>Script Received</i>	<i>PTH</i>	<i>0b</i>	<i>If no script command was received in this transaction</i>
<i>Last Online Transaction not Completed</i>	<i>CVR</i>	<i>0b</i>	<i>always</i>
<i>Last Online Transaction not Completed</i>	<i>PTH</i>	<i>0b</i>	<i>always</i>

Table 17-8: Bits to Reset if Issuer Authentication Passes

Req 17.36 (Reset VLP Available Funds when Issuer Authentication passed):

If **all** of the following are true:

- VLP is supported
- **and** VLP Available Funds is active for the transaction
- **and** the 'Reset Accumulator with online response' bit in VLP Profile Control has the value 1b,

then the application shall reset the VLP Available Funds to the VLP Funds Limit.

Req 17.37 (Reset Last Online Transaction Date when Issuer Authentication passed):

If the 'Reset Maximum Number of Days offline with online response' bit in the Issuer Options Profile Control has the value 1b:

- *If the Transaction Date date format is valid, then the application shall reset the Last Online Transaction Date in Days to the value of the Transaction Date in Days (see Annex E).*
- *If the Transaction Date format is not valid, the application shall set the 'Check Failed' bit in the CVR to the value 1b*

Req 17.38 (Branch after Issuer Authentication Passed processing):

After completing the other processing described in this section, the application shall perform CSU processing as described in section 17.5.3.1.3.

17.5.3.1.3 CSU Processing

After successful Issuer Authentication, the card has verified that the CSU received in Issuer Authentication Data is valid. The CSU for the Common Payment Application shall be coded as specified in the Common Core Definitions part of *EMV Book 3*, for a Cryptogram Version of '5'.

If the 'Card Block' bit in the CSU has the value 1b, then the application blocks the card.

Req 17.39 (Set Application Blocked):

If the 'Application Block' bit in the CSU has the value 1b, then the application shall set the 'Application Blocked' bit in the PTH to the value 1b.

This blocks the application; which causes the card to respond to all subsequent SELECT commands with status bytes indicating that the selected file is invalidated, and to respond to all subsequent Generate AC commands with an AAC type Application Cryptogram.

If the 'Update PIN Try Counter' bit in the CSU has the value 1b, then the application resets the PIN Try Counter to the value contained in the 'PIN Try Counter' bits of the CSU.

Req 17.40 (Reset Go Online on Next Transaction):

If the 'Set Go Online on Next Transaction' bit in the CSU has the value 1b, then the application shall:

- *set the 'Go Online on Next Transaction Was Set' bit in the CVR to the value 1b.*
- *set the 'Go Online on Next Transaction' bit in the PTH to the value 1b.*

Otherwise, the application shall:

- *reset the 'Go Online on Next Transaction Was Set' bit in the CVR to the value 0b.*
- *reset the 'Go Online on Next Transaction' bit in the PTH to the value 0b.*

The CSU can be used to control behaviour of velocity-checking counters and accumulators in the card during second Card Action Analysis. The issuer may personalise default controls for the offline counters to be used in place of the controls received in the CSU when the 'CSU Created by Proxy for the Issuer' bit in the CSU indicates that the CSU did not originate from the Issuer.

Req 17.41 (CSU processing, reset accumulators and counters to zero):

If **both** of the following are true:

- the 'Update Counters' bits in the CSU have the value 10b (reset counters to 0),
- **and either** of the following is true:
 - the 'CSU Created by Proxy for the Issuer' bit in the CSU has the value 0b (the CSU was not created by a Proxy),
 - **or** the 'Use Default Update Counters to Control Offline Counters if CSU is Generated by Issuer Proxy' bit in the Application Control has the value 0b (use the CSU bits even if the CSU was created by a Proxy)

then:

- for each Accumulator *x* that is active for the transaction:
 - if the 'Reset Accumulator with online response' bit in the Accumulator Profile Control for Accumulator *x* in the profile has the value 1b, then the application shall reset Accumulator *x* to the value 0
- for each Counter *x* that is active for the transaction:
 - if the 'Reset Counter with online response' bit in the Counter Profile Control for Counter *x* in the profile has the value 1b, then the application shall reset Counter *x* to the value 0

Req 17.42 (CSU processing, set accumulators and counters to Upper Limit):

If **both** of the following are true:

- the 'Update Counters' bits in the CSU have the value 01b (set counters to their Upper Limit),
- **and either** of the following is true:
 - the 'CSU Created by Proxy for the Issuer' bit in the CSU has the value 0b (the CSU was not created by a Proxy),
 - **or** the 'Use Default Update Counters to Control Offline Counters if CSU is Generated by Issuer Proxy' bit in the Application Control has the value 0b (use the CSU bits even if the CSU was created by a Proxy)

then:

- for each Accumulator *x* that is active for the transaction:
 - if the 'Reset Accumulator with online response' bit in the Accumulator Profile Control for Accumulator *x* in the profile has the value 1b, then the application shall set Accumulator *x* to the value of Accumulator *x* Upper Limit
- for each Counter *x* that is active for the transaction:
 - if the 'Reset Counter with online response' bit in the Counter Profile Control for Counter *x* in the profile has the value 1b, then the application shall set Counter *x* to the value of Counter *x* Upper Limit

Req 17.43 (CSU processing, add transaction to accumulators and counters):

If **both** of the following are true:

- the 'Update Counters' bits in the CSU have the value 11b (add online transaction to the counters),
- **and either** of the following is true:
 - the 'CSU Created by Proxy for the Issuer' bit in the CSU has the value 0b (the CSU was not created by a Proxy),
 - **or** the 'Use Default Update Counters to Control Offline Counters if CSU is Generated by Issuer Proxy' bit in the Application Control has the value 0b (use the CSU bits even if the CSU was created by a Proxy)

then:

- For each value of x for which Accumulator x is active **and** the 'Allow Accumulation' bit in the Accumulator Profile Control for Accumulator x in this profile has the value 1b:
 - If the Transaction Currency Code matches the Accumulator Currency Code, then the application shall add Amount, Authorised to Accumulator x.
 - If adding the Amount, Authorised to the value of Accumulator x would exceed the value '99 99 99 99 99 99', then Accumulator x shall be set to the value '99 99 99 99 99 99'
 - Otherwise (the Transaction Currency Code does not match the Accumulator Currency Code), if the Transaction Currency Code matches the Source Currency Code in one of the Currency Conversion Parameters in the Currency Conversion Table active for Accumulator x², then the application shall add Amount, Authorised converted to the accumulator currency (using the Currency Conversion Table active for Accumulator x) to Accumulator x.
 - If adding the Amount, Authorised converted to the accumulator currency to the value of Accumulator x would exceed the value '99 99 99 99 99 99', then Accumulator x shall be set to the value '99 99 99 99 99 99'
- For each value of x for which **all** of the following are true:
 - Counter x is active
 - **and** the 'Allow Counting' bit in the Counter Profile Control for Counter x has the value 1b
 - **and** the value of Counter x is less than 'FF'
 - **and either** of the following is true:
 - the 'Include Only If International' bit in the Counter x Control has the value 0b
 - **or** the Terminal Country Code does not match the Issuer Country Code

² If the Currency Conversion Table ID in the Accumulator Profile Control for Accumulator x for the transaction has the value 'F', then no Currency Conversion Table is active for Accumulator x.

- **and either** of the following is true:
 - the 'Include Only If Not Accumulated' bit in the Counter x Control has the value 0b
 - **or** for **all** values of y, the transaction could not be accumulated in Accumulator y because one of the conditions listed in Table 17-9 is not true

the application shall increment Counter x by one.

Accumulator y is active for the Profile.
The 'Allow Accumulation' bit in the Accumulator Profile Control for Accumulator y in this profile has the value 1b.
The 'Issuer Approves Online Transaction' bit of the received CSU has the value 1b
Either of the following is true: <ul style="list-style-type: none">• the Transaction Currency Code matches the Accumulator Currency Code• or the Transaction Currency Code does not match the Accumulator Currency Code and the Transaction Currency Code matches the Source Currency Code in one of the Currency Conversion Parameters in the Currency Conversion Table active for Accumulator y³

Table 17-9: Conditions for Accumulating Transaction using CSU Approval

³ If the Currency Conversion Table ID in the Accumulator Profile Control for Accumulator y for the transaction has the value 'F', then no Currency Conversion Table is active for Accumulator y.

Req 17.44 (Default CSU processing, reset accumulators and counters to zero):

If **all** of the following are true:

- the 'CSU Created by Proxy for the Issuer' bit in the CSU has the value 1b (the CSU was created by a Proxy)
- **and** the 'Use Default Update Counters to Control Offline Counters if CSU is Generated by Issuer Proxy' bit in the Application Control has the value 1b (use the Default Update Counters in the Application Control if the CSU was created by a Proxy)
- **and** the 'Default Update Counters' bits in the Application Control have the value 10b (reset counters to 0):

then:

- for each Accumulator *x* that is active for the transaction:
 - if the 'Reset Accumulator with online response' bit in the Accumulator Profile Control for Accumulator *x* in the profile has the value 1b, then the application shall reset Accumulator *x* to the value 0
- for each Counter *x* that is active for the transaction:
 - if the 'Reset Counter with online response' bit in the Counter Profile Control for Counter *x* in the profile has the value 1b, then the application shall reset Counter *x* to the value 0

Req 17.45 (Default CSU processing, set accumulators and counters to Upper Limit):

If **all** of the following are true:

- the 'CSU Created by Proxy for the Issuer' bit in the CSU has the value 1b (the CSU was created by a Proxy)
- **and** the 'Use Default Update Counters to Control Offline Counters if CSU is Generated by Issuer Proxy' bit in the Application Control has the value 1b (use the Default Update Counters in the Application Control if the CSU was created by a Proxy)
- **and** the 'Default Update Counters' bits in the Application Control have the value 01b (set counters to their Upper Limit):

then:

- for each Accumulator *x* that is active for the transaction:
 - if the 'Reset Accumulator with online response' bit in the Accumulator Profile Control for Accumulator *x* in the profile has the value 1b, then the application shall set Accumulator *x* to the value of Accumulator *x* Upper Limit
- for each Counter *x* that is active for the transaction:
 - if the 'Reset Counter with online response' bit in the Counter Profile Control for Counter *x* in the profile has the value 1b, then the application shall set Counter *x* to the value of Counter *x* Upper Limit

Req 17.46 (Default CSU processing, add transaction to accumulators and counters):

If **all** of the following are true:

- the 'CSU Created by Proxy for the Issuer' bit in the CSU has the value 1b (the CSU was created by a Proxy)
- **and** the 'Use Default Update Counters to Control Offline Counters if CSU is Generated by Issuer Proxy' bit in the Application Control has the value 1b (use the Default Update Counters in the Application Control if the CSU was created by a Proxy)
- **and** the 'Default Update Counters' bits in the Application Control have the value 11b (add online transaction to the counters):

then:

- For each value of x for which Accumulator x is active **and** the 'Allow Accumulation' bit in the Accumulator Profile Control for Accumulator x in this profile has the value 1b:
 - If the Transaction Currency Code matches the Accumulator Currency Code, then the application shall add Amount, Authorised to Accumulator x.
 - If adding the Amount, Authorised to the value of Accumulator x would exceed the value '99 99 99 99 99 99', then Accumulator x shall be set to the value '99 99 99 99 99 99'
 - Otherwise (the Transaction Currency Code does not match the Accumulator Currency Code), if the Transaction Currency Code matches the Source Currency Code in one of the Currency Conversion Parameters in the Currency Conversion Table active for Accumulator x⁴, then the application shall add Amount, Authorised converted to the accumulator currency (using the Currency Conversion Table active for Accumulator x) to Accumulator x.
 - If adding the Amount, Authorised converted to the accumulator currency to the value of Accumulator x would exceed the value '99 99 99 99 99 99', then Accumulator x shall be set to the value '99 99 99 99 99 99'
- For each value of x for which **all** of the following are true:
 - Counter x is active
 - **and** the 'Allow Counting' bit in the Counter Profile Control for Counter x has the value 1b:
 - **and** the value of Counter x is less than 'FF'

⁴ If the Currency Conversion Table ID in the Accumulator Profile Control for Accumulator x for the transaction has the value 'F', then no Currency Conversion Table is active for Accumulator x.

- **and either** of the following is true:
 - the 'Include Only If International' bit in the Counter x Control has the value 0b
 - **or** the Terminal Country Code does not match the Issuer Country Code
- **and either** of the following is true:
 - the 'Include Only If Not Accumulated' bit in the Counter x Control has the value 0b
 - **or** for **all** values of y, the transaction could not be accumulated in Accumulator y because one of the conditions listed in Table 17-9 is not true:

the application shall increment Counter x by one.

Req 17.47 (Correct cyclic accumulator reference when Issuer Authentication Passed):

For each value of x for which Cyclic Accumulator x is active for the transaction, if **all** of the following are true:

- the terminal requested a TC type Application Cryptogram
- **and** the 'Issuer Approves Online Transaction' bit of the received CSU has the value 1b.
- **and** the Transaction Date format is valid (see Req 17.13)

then:

- if the 'Cycle Type' in the Cyclic Accumulator x Control has the value 01b (daily) or 11b (monthly), **and** the Transaction Date is less than the Cyclic Accumulator x Reference Date; the application shall correct the Cyclic Accumulator Reference Date by setting the Cyclic Accumulator x Reference Date to the value of the Transaction Date
- if the 'Cycle Type' in the Cyclic Accumulator x Control has the value 10b (weekly), **and** the Transaction Date in Days (see Annex E) is less than the Cyclic Accumulator x Reference Day; the application shall correct the Cyclic Accumulator Reference Day by setting the Cyclic Accumulator x Reference Day to the value of the Transaction Date Reference Day

Req 17.48 (Accumulate and use CSU to decide to approve transaction):

If **both** of the following are true:

- the terminal requested a TC type Application Cryptogram
- **and** the 'Issuer Approves Online Transaction' bit of the received CSU has the value 1b.

then:

- For each value of x for which **all** of the following are true:
 - Cyclic Accumulator x is active
 - **and** the 'Allow Accumulation' bit in the Cyclic Accumulator Profile Control for Cyclic Accumulator x in this profile has the value 1b
 - **and** the 'Include Online Approvals' bit in the Cyclic Accumulator x Control has the value 1b
 - **and** the Transaction Date format is valid (see Req 17.13)

then:

- if the Transaction Currency Code matches the Accumulator Currency Code, then the application shall add Amount, Authorised to Cyclic Accumulator x.
 - If adding the Amount, Authorised to the value of Cyclic Accumulator x would exceed the value '99 99 99 99 99 99', then Cyclic Accumulator x shall be set to the value '99 99 99 99 99 99'
- Otherwise (the Transaction Currency Code does not match the Accumulator Currency Code), if the Transaction Currency Code matches the Source Currency Code in one of the Currency Conversion Parameters in the Currency Conversion Table active for Cyclic Accumulator x⁵, then the application shall add Amount, Authorised converted to the accumulator currency (using the Currency Conversion Table active for Cyclic Accumulator x) to Cyclic Accumulator x.
 - If adding the Amount, Authorised converted to the accumulator currency to the value of Cyclic Accumulator x would exceed the value '99 99 99 99 99 99', then Cyclic Accumulator x shall be set to the value '99 99 99 99 99 99'
- The application shall approve the transaction as described in section 17.5.5 and section 17.5.8.

⁵ If the Currency Conversion Table ID in the Cyclic Accumulator Profile Control for Cyclic Accumulator x for the transaction has the value 'F', then no Currency Conversion Table is active for Cyclic Accumulator x.

Req 17.49 (Use CSU to decide to decline transaction):

*If **either** of the following is true:*

- *the terminal requested an AAC type Application Cryptogram*
- ***or** the 'Issuer Approves Online Transaction' bit of the received CSU has the value 0b.*

then the application shall decline the transaction as described in section 17.5.6 and section 17.5.8.

17.5.3.2 Issuer Authentication Data not Received

The application performs the following when the application has successfully gone online (that is, the Authorisation Response Code returned in the second GENERATE AC command data does not indicate the terminal was unable to go online) and no Issuer Authentication Data was returned in the second GENERATE AC command data (that is, the Issuer Authentication Data returned in the second GENERATE AC command data was all zero).

Req 17.50 (Decline transaction if Issuer Authentication Data required and not received):

If the 'Issuer Authentication Required to be Performed' bit in the Application Control has the value 1b, then the application shall decline the transaction as specified in section 17.5.6 and section 17.5.8.

Req 17.51 (Reset bits if Issuer Authentication not received):

*If **both** of the following are true:*

- *the 'Issuer Authentication Required to be Performed' bit in the Application Control has the value 0b,*
- ***and** the 'Issuer Authentication Requirements apply to Resetting of Non-Velocity-Checking Indicators and Counters' bit in the Application Control has the value 0b,*

then the application shall reset bits as indicated in Table 17-10.

reset the following bit:	in:	to the value:	condition
<i>Issuer Authentication Failed</i>	<i>CVR</i>	<i>0b</i>	<i>always</i>
<i>Issuer Authentication Failed</i>	<i>PTH</i>	<i>0b</i>	<i>always</i>
<i>Script Failed</i>	<i>PTH</i>	<i>0b</i>	<i>If no script command failed in this transaction</i>
<i>Script Received</i>	<i>PTH</i>	<i>0b</i>	<i>If no script command was received in this transaction</i>
<i>Go Online on Next Transaction Was Set</i>	<i>CVR</i>	<i>0b</i>	<i>always</i>
<i>Go Online on Next Transaction</i>	<i>PTH</i>	<i>0b</i>	<i>always</i>
<i>Last Online Transaction not Completed</i>	<i>CVR</i>	<i>0b</i>	<i>always</i>
<i>Last Online Transaction not Completed</i>	<i>PTH</i>	<i>0b</i>	<i>always</i>

Table 17-10: Bits to Reset if Issuer Authentication Data Not Present

Req 17.52 (Decline transaction when Issuer Authentication not required):

If the terminal requested an AAC type Application Cryptogram in the second GENERATE AC command, then after completing the other processing in this section, the application shall decline the transaction as described in section 17.5.6 and section 17.5.8.

Req 17.53 (Reset velocity-checking indicators when Issuer Authentication not required):

*If **all** of the following are true:*

- *the 'Issuer Authentication Required to be Performed' bit in the Application Control has the value 0b,*
- ***and** the terminal requested a TC type Application Cryptogram in the second GENERATE AC command,*
- ***and** the 'Issuer Authentication Requirements Apply to Resetting of Velocity-Checking Counters' bit in the Application Control has the value 0b,*

then:

- *For each value of x for which **both** of the following are true:*
 - *Accumulator x is active*
 - ***and** the 'Reset Accumulator with online response' bit in the Accumulator Profile Control for Accumulator x has the value 1b,*
the application shall reset Accumulator x to the value 0.
- *For each value of x for which **both** of the following are true:*
 - *Counter x is active*
 - ***and** the 'Reset Counter with online response' bit in the Counter Profile Control for Counter x has the value 1b,*
the application shall reset Counter x to the value 0.
- *If **all** of the following are true:*
 - *VLP is supported*
 - ***and** VLP Available Funds is active for the transaction*
 - ***and** the 'Reset Accumulator with online response' bit in the VLP Profile Control has the value 1b,*
then the application shall reset the VLP Available Funds to the VLP Funds Limit.
- *If the 'Reset Maximum Number of Days offline with online response' bit in the Issuer Options Profile Control has the value 1b:*
 - *If the Transaction Date format is valid, then the application shall reset the Last Online Transaction Date to the value of the Transaction Date.*
 - *If the Transaction Date format is not valid, then the application shall set the 'Check Failed' bit in the CVR to the value 1b*

Req 17.54 (Correct cyclic accumulator reference when Issuer Authentication not required):

*If the terminal requested a TC type Application Cryptogram in the second Generate AC command **and** the 'Issuer Authentication Required to be Performed' bit in the Application Control has the value 0b, then for each value of x for which Cyclic Accumulator x is active for the transaction **and** the Transaction Date format is valid:*

- if the 'Cycle Type' in the Cyclic Accumulator x Control has the value 01b (daily) or 11b (monthly), **and** the Transaction Date is less than the Cyclic Accumulator x Reference Date; the application shall correct the Cyclic Accumulator Reference Date by setting the Cyclic Accumulator x Reference Date to the value of the Transaction Date*
- if the 'Cycle Type' in the Cyclic Accumulator x Control has the value 10b (weekly), **and** the Transaction Date in Days (see Annex E) is less than the Cyclic Accumulator x Reference Day; the application shall correct the Cyclic Accumulator Reference Day by setting the Cyclic Accumulator x Reference Day to the value of the Transaction Date Reference Day*

Req 17.55 (Convert and accumulate amount when Issuer Authentication not required):

For each value of *x* for which **all** of the following are true:

- the 'Issuer Authentication Required to be Performed' bit in the Application Control has the value 0b,
- **and** the terminal requested a TC type Application Cryptogram in the second GENERATE AC command,
- **and** Cyclic Accumulator *x* is active
- **and** the 'Allow Accumulation' bit in the Cyclic Accumulator Profile Control for Cyclic Accumulator *x* in this profile has the value 1b
- **and** the 'Include Online Approvals' bit in the Cyclic Accumulator *x* Control has the value 1b
- **and** the Transaction Date format is valid

then:

- If the Transaction Currency Code matches the Accumulator Currency Code, then the application shall add Amount, Authorised to Cyclic Accumulator *x*.
 - If adding the Amount, Authorised to the value of Cyclic Accumulator *x* would exceed the value '99 99 99 99 99 99', then Cyclic Accumulator *x* shall be set to the value '99 99 99 99 99 99'
- Otherwise (the Transaction Currency Code does not match the Accumulator Currency Code), then the application shall add Amount, Authorised converted to the accumulator currency (using the Currency Conversion Table active for Cyclic Accumulator *x*⁶) to Cyclic Accumulator *x*.
 - If adding the Amount, Authorised converted to the accumulator currency to the value of Cyclic Accumulator *x* would exceed the value '99 99 99 99 99 99', then Cyclic Accumulator *x* shall be set to the value '99 99 99 99 99 99'

Req 17.56 (Approve transaction when Issuer Authentication not required):

If **both** of the following are true:

- the 'Issuer Authentication Required to be Performed' bit in the Application Control has the value 0b,
- **and** the terminal requested a TC type Application Cryptogram in the second GENERATE AC command,

then after completing the other processing described in this section, the application shall approve the transaction as described in section 17.5.5 and section 17.5.8.

⁶ If the Currency Conversion Table ID in the Cyclic Accumulator Profile Control for Cyclic Accumulator *x* for the transaction has the value 'F', then no Currency Conversion Table is active for Cyclic Accumulator *x*.

17.5.4 Online Processing Requested, Online Authorisation Not Completed

Req 17.57 (Set bits when online processing not completed):

When the second GENERATE AC command from the terminal contains an Authorisation Response Code indicating that online processing was requested but not completed ('Y3' or 'Z3'), the application shall:

- *Set the 'Unable to Go Online' bit in the CVR to the value 1b*
- *Set the 'Unable to Go Online' bit in the ADR to the value 1b*
- *Set the 'Unable to Go Online' bit in the PTH to the value 1b*
- *Set the 'Issuer Authentication Not Performed' bit in the CVR to the value 1b*
- *If the terminal requested an AAC, then:*
 - *For each value of x for which all of the following are true:*
 - *Counter x is active*
 - **and** *the 'Allow Counting' bit in the Counter x Profile Control for Counter x has the value 1b,*
 - **and** *the 'Include Offline Declines' bit in the Counter x Control has the value 1b,*
 - **and** *the value of Counter x is less than 'FF',**then the application shall increment Counter x by one.*
 - **and** *the application shall decline the transaction as described in section 17.5.6 and section 17.5.8.*
- *If the terminal requested a TC, the application shall perform the card risk management checks specified in section 17.5.4.1, and then determine whether to approve or decline the transaction, as specified in section 17.5.4.2.*

17.5.4.1 Second Card Risk Management

When the online authorisation was not completed (that is, the Authorisation Response Code returned in the second GENERATE AC command data has the value 'Y3' or 'Z3'), the application performs each mandatory card risk management check and the optional checks that are activated by the issuer before determining whether to approve or decline the transaction in the second GENERATE AC command.

Table 17-11 summarises the Card Risk Management checks provided and describes the result if the condition being checked for occurs. The section that describes the check is also indicated.

Risk Management Check	Result (if condition occurs)	See section:
Maximum Transaction Amount Check	If the MTA is active in the Profile, sets an ADR bit to indicate that the value of the transaction exceeds the limit.	17.5.4.1.1
Accumulator x Lower Limit Exceeded Check	If Accumulator x is active in the Profile, sets an ADR bit to indicate that the Accumulator x Lower Limit has been exceeded.	17.5.4.1.2
Accumulator x Upper Limit Exceeded Check	If Accumulator x is active in the Profile, sets an ADR bit to indicate that the Accumulator x Upper Limit has been exceeded.	17.5.4.1.3
Counter x Lower Limit Exceeded Check	If Counter x is active in the Profile, sets an ADR bit to indicate that the Counter x Lower Limit has been exceeded.	17.5.4.1.4
Counter x Upper Limit Exceeded Check	If Counter x is active in the Profile, sets an ADR bit to indicate that the Counter x Upper Limit has been exceeded.	17.5.4.1.5
Cyclic Accumulator x Limit Exceeded Check	If Cyclic Accumulator x is active in the Profile, sets an ADR bit to indicate that the Cyclic Accumulator x Limit has been exceeded.	17.5.4.1.6

Table 17-11: Card Risk Management Checks

The card performs each of the following Card Risk Management checks to see whether the condition has occurred, then proceeds to the next check. The application may perform the checks in any order, but all checks that are to be performed shall be completed before the application determines the type of Application Cryptogram to be sent in the GENERATE AC response.

NOTE: The Issuer Authentication Failed, Issuer Authentication not Performed, and Unable to Go Online checks for this transaction are updated in other sections of Second Card Action Analysis processing.

NOTE: For some of the following checks, CCD requires that the terminal request a TC. The condition that the terminal must have requested a TC has been met because the following checks are not performed if the terminal has requested an AAC (see section 17.5.4, Req 17.57).

17.5.4.1.1 Maximum Transaction Amount Check

This issuer-optional check determines whether the limit for the Maximum Transaction Amount has been exceeded. This enables the application to consider this information when deciding whether to approve or decline the transaction offline.

The MTA Profile Control ID in the Profile Control selected for the transaction is used to activate this check.

Req 17.58 (Check MTA):

If the MTA Check is active (that is, the MTA Profile Control ID in the Profile Control has a value other than 'F'), then the application shall perform this check.

Req 17.59 (Set MTA Exceeded):

*If **either** of the following is true:*

- the Transaction Currency Code matches the MTA Currency Code **and** the Amount, Authorised is greater than the MTA*
- or** the Transaction Currency Code does not match the MTA Currency Code **and** the Transaction Currency Code matches the Source Currency Code in one of the Currency Conversion Parameters in the Currency Conversion Table active for the Maximum Transaction Amount⁷ **and** the amount converted to the MTA currency⁸ (using Amount, Authorised and the Currency Conversion Table active for the Maximum Transaction Amount⁹) is greater than the MTA*

then the application shall set the 'MTA exceeded' bit in the ADR to the value 1b.

Otherwise, the application shall reset the 'MTA exceeded' bit in the ADR to the value 0b.

17.5.4.1.2 Accumulator x Lower Limit Exceeded Check

This issuer-optional check determines whether the lower limit for a cumulative amount of offline transactions has been exceeded, and enables the application to consider this information when deciding whether to approve or decline the transaction offline.

This check is performed for each Accumulator x that is active for the transaction.

⁷ If the Currency Conversion Table ID in the MTA Profile Control for the transaction has the value 'F', then no Currency Conversion Table is active for the MTA.

⁸ If the converted Amount, Authorised would overflow the maximum value for an amount ('99 99 99 99 99 99'), then instead compare the value '99 99 99 99 99 99' (rather than the converted amount) to the MTA.

⁹ If the Currency Conversion Table ID in the MTA Profile Control has the value 'F', then no Currency Conversion Table is active for the MTA.

Req 17.60 (Check accumulator lower limit):

For each value of *x* for which Accumulator *x* is active for the transaction, if **any** of the following is true:

- the value of Accumulator *x* is greater than the Accumulator *x* Lower Limit
- **or all** of the following are true:
 - the 'Allow Accumulation' bit in the Accumulator Profile Control for Accumulator *x* in this profile has the value 1b
 - **and** the 'Include Offline Approvals' bit in the Accumulator *x* Control has the value 1b
 - **and** the Transaction Currency Code matches the Accumulator Currency Code
 - **and** the sum¹⁰ of the value of Accumulator *x* and Amount, Authorised is greater than the Accumulator *x* Lower Limit
- **or all** of the following are true:
 - the 'Allow Accumulation' bit in the Accumulator Profile Control for Accumulator *x* in this profile has the value 1b
 - **and** the 'Include Offline Approvals' bit in the Accumulator *x* Control has the value 1b
 - **and** the Transaction Currency Code does not match the Accumulator Currency Code¹¹
 - **and** the Transaction Currency Code matches the Source Currency Code in any of the Currency Conversion Parameters in the Currency Conversion Table active for Accumulator *x*¹²
 - **and** the sum¹³ of the value of Accumulator *x* and the amount converted to the accumulator currency (using Amount, Authorised and the Currency Conversion Table active for Accumulator *x*) is greater than the Accumulator *x* Lower Limit

then the application shall set the 'Accumulator *x* Lower Limit Exceeded' bit in the ADR to 1b.

Otherwise (that is, Accumulator *x* has not exceeded its Lower Limit), then the application shall reset the 'Accumulator *x* Lower Limit Exceeded' bit in the ADR to 0b.

¹⁰ If adding the Amount, Authorised to the value of Accumulator *x* would cause the accumulator to overflow, then instead compare the value '99 99 99 99 99 99' (rather than the sum of Amount, Authorised and the value of Accumulator *x*) to the Accumulator *x* Lower Limit.

¹¹ This condition is necessary because a Currency Conversion Parameter can be disabled by setting its Currency Code to the Accumulator Currency Code.

¹² If the Currency Conversion Table ID in the Accumulator Profile Control for Accumulator *x* for the transaction has the value 'F', then no Currency Conversion Table is active for Accumulator *x*.

¹³ If adding the converted Amount, Authorised to the value of Accumulator *x* would cause the accumulator to overflow, then instead compare the value '99 99 99 99 99 99' (rather than the sum of the converted Amount, Authorised and the value of Accumulator *x*) to the Accumulator *x* Lower Limit.

17.5.4.1.3 Accumulator x Upper Limit Exceeded Check

This issuer-optional check determines whether the upper limit for a cumulative amount of offline transactions has been exceeded, and enables the application to consider this information when deciding whether to approve or decline the transaction offline.

This check is performed for each Accumulator x that is active for the transaction.

Req 17.61 (Check accumulator upper limit):

*For each value of x for which Accumulator x is active for the transaction, if **any** of the following is true:*

- *the value of Accumulator x is greater than the Accumulator x Upper Limit*
- ***or all** of the following are true:*
 - *the 'Allow Accumulation' bit in the Accumulator Profile Control for Accumulator x in this profile has the value 1b*
 - ***and** the 'Include Offline Approvals' bit in the Accumulator x Control has the value 1b*
 - ***and** the Transaction Currency Code matches the Accumulator Currency Code*
 - ***and** the sum¹⁴ of the value of Accumulator x and Amount, Authorised is greater than the Accumulator x Upper Limit*
- ***or all** of the following are true:*
 - *the 'Allow Accumulation' bit in the Accumulator Profile Control for Accumulator x in this profile has the value 1b*
 - ***and** the 'Include Offline Approvals' bit in the Accumulator x Control has the value 1b*
 - ***and** the Transaction Currency Code does not match the Accumulator Currency Code*
 - ***and** the Transaction Currency Code matches the Source Currency Code in one of the Currency Conversion Parameters in the Currency Conversion Table active for Accumulator x¹⁵*
 - ***and** the sum¹⁶ of the value of Accumulator x and the amount converted to the accumulator currency (using Amount, Authorised and the Currency Conversion Table active for Accumulator x) is greater than the Accumulator x Upper Limit*

then the application shall set the 'Accumulator x Upper Limit Exceeded' bit in the ADR to 1b.

Otherwise (that is, Accumulator x has not exceeded its Upper Limit), then the application shall reset the 'Accumulator x Upper Limit Exceeded' bit in the ADR to 0b.

¹⁴ If adding the Amount, Authorised to the value of Accumulator x would cause the accumulator to overflow, then instead compare the value '99 99 99 99 99 99' (rather than the sum of Amount, Authorised and the value of Accumulator x) to the Accumulator x Upper Limit.

¹⁵ If the Currency Conversion Table ID in the Accumulator Profile Control for Accumulator x for the transaction has the value 'F', then no Currency Conversion Table is active for Accumulator x.

¹⁶ If adding the converted Amount, Authorised to the value of Accumulator x would cause the accumulator to overflow, then instead compare the value '99 99 99 99 99 99' (rather than the sum of the converted Amount, Authorised and the value of Accumulator x) to the Accumulator x Upper Limit.

17.5.4.1.4 Counter x Lower Limit Exceeded Check

This issuer-optional check determines whether the lower limit for a count of offline transactions has been exceeded, and enables the application to consider this information when deciding whether to approve or decline the transaction offline.

This check is performed for each Counter x that is active for the transaction.

Req 17.62 (Check counter lower limit):

*For each value of x for which Counter x is active for the transaction, if **either** of the following is true:*

- *Counter x is greater than the Counter x Lower Limit*
- ***or all** of the following are true:*
 - *the 'Allow Counting' bit in the Counter Profile Control for Counter x has the value 1b*
 - ***and** the 'Include Offline Approvals' bit in the Counter x Control has the value 1b*
 - ***and either** of the following is true:*
 - *the 'Include Only If International' bit in the Counter x Control has the value 0b*
 - ***or** the Terminal Country Code does not match the Issuer Country Code*
 - ***and either** of the following is true:*
 - *the 'Include Only If Not Accumulated' bit in the Counter x Control has the value 0b*
 - ***or** for **all** values of y, the transaction could not be accumulated in Accumulator y because one of the conditions listed in Table 17-12 is not true*
 - ***and** the value of Counter x + 1 is greater than the Counter x Lower Limit¹⁷*

then the application shall set the 'Counter x Lower Limit Exceeded' bit in the ADR to 1b.

Otherwise (that is, Counter x has not exceeded its Lower Limit), if Counter x is active for the transaction then the application shall reset the 'Counter x Lower Limit Exceeded' bit in the ADR to 0b.

¹⁷ If incrementing Counter x would cause the counter to overflow, then instead compare the value 'FF' (rather than the value of Counter x plus 1) to the Counter x Lower Limit.

<i>Accumulator y is active for the Profile.</i>
<i>The 'Allow Accumulation' bit in the Accumulator Profile Control for Accumulator y in this profile has the value 1b.</i>
<i>The 'Include Offline Approvals' bit in the Accumulator y Control has the value 1b.</i>
<i>The 'terminal requested a TC in the second GENERATE AC command</i>
<p>Either of the following is true:</p> <ul style="list-style-type: none"> the Transaction Currency Code matches the Accumulator Currency Code or the Transaction Currency Code does not match the Accumulator Currency Code and the Transaction Currency Code matches the Source Currency Code in one of the Currency Conversion Parameters in the Currency Conversion Table active for Accumulator y¹⁸

Table 17-12: Conditions for Accumulating Offline Approved Transaction

¹⁸ If the Currency Conversion Table ID in the Accumulator Profile Control for Accumulator y for the transaction has the value 'F', then no Currency Conversion Table is active for Accumulator y.

17.5.4.1.5 Counter x Upper Limit Exceeded Check

This issuer-optional check determines whether an upper limit for the count of offline transactions has been exceeded, and enables the application to consider this information when deciding whether to approve or decline the transaction offline.

This check is performed for each Counter x that is active for the transaction.

Req 17.63 (Check counter upper limit):

*For each value of x for which Counter x is active for the transaction, if **either** of the following is true:*

- *Counter x is greater than the Counter x Upper Limit*
- ***or all** of the following are true:*
 - *the 'Allow Counting' bit in the Counter Profile Control for Counter x has the value 1b*
 - ***and** the 'Include Offline Approvals' bit in the Counter x Control has the value 1b*
 - ***and either** of the following is true:*
 - *the 'Include Only If International' bit in the Counter x Control has the value 0b*
 - ***or** the Terminal Country Code does not match the Issuer Country Code*
 - ***and either** of the following is true:*
 - *the 'Include Only If Not Accumulated' bit in the Counter x Control has the value 0b*
 - ***or** for **all** values of y, the transaction could not be accumulated in Accumulator y because one of the conditions listed in Table 17-12 is not true:*
 - ***and** the value of Counter x + 1 is greater than the Counter x Upper Limit¹⁹*

then the application shall set the 'Counter x Upper Limit Exceeded' bit in the ADR to 1b.

Otherwise (that is, Counter x has not exceeded its Upper Limit), if Counter x is active for the transaction then the application shall reset the 'Counter x Upper Limit Exceeded' bit in the ADR to 0b.

¹⁹ If incrementing Counter x would cause the counter to overflow, then instead compare the value 'FF' (rather than the value of Counter x plus 1) to the Counter x Upper Limit.

17.5.4.1.6 Cyclic Accumulator x Limit Exceeded Check

This issuer-optional check determines whether the limit for the cumulative amount of approved transactions within a single cycle (day, week, or month) has been exceeded. This enables the application to consider this information when deciding whether to approve or decline the transaction offline.

This check is performed for each Cyclic Accumulator x that is active for the transaction.

Then the application checks whether the transaction is within the current cycle.

- The transaction is within the current day if the Transaction Date (YYMMDD) matches the Reference Date (YYMMDD).
- The transaction is within the current week if the Transaction Date is before the start of the week following the current cycle.
- The transaction is within the current month if the Transaction Date (YYMM) matches the Reference Date (YYMM).

Req 17.64 (Check whether transaction within current cycle):

*For each value of x for which Cyclic Accumulator x is active for the transaction **and** the 'Allow Accumulation' bit in the Cyclic Accumulator Profile Control for Cyclic Accumulator x in this profile has the value 1b, if **any** of the following is true:*

- **both** of the following are true:
 - the 'Cycle Type' in the Cyclic Accumulator x Control has the value 01b (daily)
 - **and** the Transaction Date is equal to the Cyclic Accumulator x Reference Date
- **or both** of the following are true:
 - the 'Cycle Type' in the Cyclic Accumulator x Control has the value 10b (weekly)
 - **and** the Transaction Date in Days (see Annex E) is less than the Cyclic Accumulator x Reference Day plus 7
- **or both** of the following are true:
 - the 'Cycle Type' in the Cyclic Accumulator x Control has the value 11b (monthly)
 - **and** the year and month of the Transaction Date is equal to the year and month of the Cyclic Accumulator x Reference Date

then the transaction is within the current cycle.

If the transaction is within the current cycle, and the limit for the cycle has been exceeded, then the application will set an ADR bit.

Req 17.65 (Set bit if cyclic accumulator limit already exceeded):

*For each value of x for which Cyclic Accumulator x is active for the transaction, if **both** of the following are true:*

- *the transaction is within the current cycle*
- ***and** the value of Cyclic Accumulator x is greater than the Cyclic Accumulator x Limit*

then the application shall:

- *set the 'Cyclic Accumulator x Limit Exceeded' bit in the ADR to 1b*
- *discontinue processing this check.*

If the current transaction could be accumulated in Cyclic Accumulator x, then the application checks whether accumulating the transaction would result in exceeding the limit for the cycle. If the limit would be exceeded, then the application will set an ADR bit.

Req 17.66 (Set bit if cyclic accumulator limit newly exceeded):

*For each value of x for which Cyclic Accumulator x is active for the transaction **and** the 'Allow Accumulation' bit in the Cyclic Accumulator Profile Control for Cyclic Accumulator x in this profile has the value 1b, if **either** of the following is true:*

- the Transaction Currency Code matches the Accumulator Currency Code **and** the sum²⁰ of the value of Cyclic Accumulator x and Amount, Authorised is greater than the Cyclic Accumulator x Limit*
- or** the Transaction Currency Code does not match the Accumulator Currency Code **and** the Transaction Currency Code matches the Source Currency Code in one of the Currency Conversion Parameters in the Currency Conversion Table active for Cyclic Accumulator x²¹ **and** the sum²² of the value of Cyclic Accumulator x and Amount, Authorised converted to the accumulator currency (using the Currency Conversion Table active for Cyclic Accumulator x) is greater than the Cyclic Accumulator x Limit*

then the application shall:

- set the 'Cyclic Accumulator x Limit Exceeded' bit in the ADR to 1b.*
- discontinue processing this check.*

Req 17.67 (Reset bit if Cyclic Accumulator Limit no longer exceeded):

*For each value of x for which Cyclic Accumulator x is active for the transaction **and** Cyclic Accumulator x has not exceeded its Limit, the application shall reset the 'Cyclic Accumulator x Limit Exceeded' bit in the ADR to 0b.*

17.5.4.1.7 Issuer-discretionary bit 1 and Issuer-discretionary bit 2

The setting of Issuer-discretionary bit 1 and Issuer-discretionary bit 2 is beyond the scope of this specification.

²⁰ If adding the Amount, Authorised to the value of Cyclic Accumulator x would cause the cyclic accumulator to overflow, then instead compare the value '99 99 99 99 99 99' (rather than the sum of Amount, Authorised and the value of Cyclic Accumulator x) to the Cyclic Accumulator x Limit.

²¹ If the Currency Conversion Table ID in the Cyclic Accumulator Profile Control for Cyclic Accumulator x for the transaction has the value 'F', then no Currency Conversion Table is active for Cyclic Accumulator x.

²² If adding the converted Amount, Authorised to the value of Cyclic Accumulator x would cause the cyclic accumulator to overflow, then instead compare the value '99 99 99 99 99 99' (rather than the sum of the converted Amount, Authorised and the value of Cyclic Accumulator x) to the Cyclic Accumulator x Limit.

17.5.4.2 Determine Offline Response Type

When the transaction is unable to go online, the Application Decisional Results (ADR), Application Control, and Card Issuer Action Codes (the CIAC-Default) data elements are used to determine whether the application will approve or decline the transaction.

NOTE: The processing in this section is only performed if the terminal requests a TC. If the terminal has requested an AAC, the transaction is always declined.

The application compares the CIAC – Default and the ADR.

- If a CIAC – Default bit and the corresponding ADR bit are both set to the value 1b, then the application continues processing as an offline decline as described in section 17.5.4.2.1.
- Otherwise, the application continues processing as an offline approval as described in section 17.5.4.2.2.

17.5.4.2.1 Application Declines Transaction Offline (Unable to Go Online)

Req 17.68 (Determine transaction is offline decline):

*If any CIAC – Default bit and the corresponding ADR bit are **both** set to the value 1b, then the application shall treat the transaction as an offline decline.*

Req 17.69 (Increment counter when offline decline):

If the application treats the transaction as an offline decline, then for each Counter x that is active:

- *if the 'Allow Counting' bit in the Counter Profile Control for Counter x has the value 1b **and** the 'Include Offline Declines' bit in the Counter x Control has the value 1b **and** the value of Counter x is less than 'FF', then the application shall increment Counter x by one.*

Req 17.70 (Continue offline transaction as a decline):

If the application treats the transaction as an offline decline, then the application shall decline the transaction as described in section 17.5.6 and section 17.5.8.

17.5.4.2.2 Application Approves Transaction Offline (Unable to Go Online)

Req 17.71 (Determine transaction is offline approval):

If there is no CIAC – Default bit set to the value 1b with the corresponding ADR bit also set to the value 1b, then the application shall treat the transaction as an offline approval.

Req 17.72 (Accumulate amount when approve transaction offline):

*If the application treats the transaction as an offline approval, then for each value of x for which **all** of the following are true:*

- *Accumulator x is active*
- ***and** the 'Allow Accumulation' bit in the Accumulator Profile Control for Accumulator x in this profile has the value 1b*
- ***and** the 'Include Offline Approvals' bit in the Accumulator x Control has the value 1b:*

the application shall:

- *If the Transaction Currency Code matches the Accumulator Currency Code, then the application shall add Amount, Authorised to Accumulator x.*
 - *If adding the Amount, Authorised to the value of Accumulator x would exceed the value '99 99 99 99 99 99', then Accumulator x shall be set to the value '99 99 99 99 99 99'*
- *Otherwise (the Transaction Currency Code does not match the Accumulator Currency Code), if the Transaction Currency Code matches the Source Currency Code in one of the Currency Conversion Parameters in the Currency Conversion Table active for Accumulator x²³, then the application shall add Amount, Authorised converted to the accumulator currency (using the Currency Conversion Table active for Accumulator x) to Accumulator x.*
 - *If adding the Amount, Authorised converted to the accumulator currency to the value of Accumulator x would exceed the value '99 99 99 99 99 99', then Accumulator x shall be set to the value '99 99 99 99 99 99'*

²³ If the Currency Conversion Table ID in the Accumulator Profile Control for Accumulator x for the transaction has the value 'F', then no Currency Conversion Table is active for Accumulator x.

Req 17.73 (Increment counter when approve transaction offline):

*If the application treats the transaction as an offline approval, then for each value of x for which **all** of the following are true:*

- *Counter x is active*
- ***and** the 'Allow Counting' bit in the Counter Profile Control for Counter x has the value 1b*
- ***and** the value of Counter x is less than 'FF'*
- ***and** the 'Include Offline Approvals' bit in the Counter x Control has the value 1b*
- ***and either** of the following is true:*
 - *the 'Include Only If International' bit in the Counter x Control has the value 0b*
 - ***or** the Terminal Country Code does not match the Issuer Country Code*
- ***and either** of the following is true:*
 - *the 'Include Only If Not Accumulated' bit in the Counter x Control has the value 0b*
 - ***or** for **all** values of y, the transaction could not be accumulated in Accumulator y because one of the conditions listed in Table 17-12 is not true:*

the application shall increment Counter x by one.

Req 17.74 (Add amount to cyclic accumulator when approve transaction offline):

*If the application treats the transaction as an offline approval, then for each value of x for which **all** of the following are true:*

- *Cyclic Accumulator x is active*
- ***and** the 'Allow Accumulation' bit in the Cyclic Accumulator Profile Control for Cyclic Accumulator x in this profile has the value 1b*
- ***and** the Transaction Date format is valid*
- ***and either** of the following is true:*
 - ***both** of the following are true:*
 - *the 'Cycle Type' in the Cyclic Accumulator x Control has the value 01b (daily) or 11b (monthly)*
 - ***and** the Transaction Date is not less than the Cyclic Accumulator x Reference Date,*
 - ***both** of the following are true:*
 - *the 'Cycle Type' in the Cyclic Accumulator x Control has the value 10b (weekly)*
 - ***and** the Transaction Date in Days (see Annex E) is not less than the Cyclic Accumulator x Reference Day,*

the application shall:

- *If the Transaction Currency Code matches the Accumulator Currency Code, then the application shall add Amount, Authorised to Cyclic Accumulator x.*
 - *If adding the Amount, Authorised to the value of Cyclic Accumulator x would exceed the value '99 99 99 99 99 99', then Cyclic Accumulator x shall be set to the value '99 99 99 99 99 99'*
- *Otherwise (the Transaction Currency Code does not match the Accumulator Currency Code), if the Transaction Currency Code matches the Source Currency Code in one of the Currency Conversion Parameters in the Currency Conversion Table active for Accumulator x²⁴, then the application shall add Amount, Authorised converted to the accumulator currency (using the Currency Conversion Table active for Cyclic Accumulator x) to Cyclic Accumulator x.*
 - *If adding the Amount, Authorised converted to the accumulator currency to the value of Cyclic Accumulator x would exceed the value '99 99 99 99 99 99', then Cyclic Accumulator x shall be set to the value '99 99 99 99 99 99'*

²⁴ If the Currency Conversion Table ID in the Cyclic Accumulator Profile Control for Cyclic Accumulator x for the transaction has the value 'F', then no Currency Conversion Table is active for Cyclic Accumulator x.

Req 17.75 (Reset Offline Data Authentication Failed on Previous Transaction):

*If the application treats the transaction as an offline approval **and** each of the following bits in the TVR provided by the terminal in the Second GENERATE AC Command Data has the value 0b:*

- *SDA Failed*
- *DDA Failed*
- *CDA Failed*

then the application shall reset the 'Offline Data Authentication Failed on Previous Transaction' bit in PTH to the value 0b.

Req 17.76 (Continue offline transaction as an approval):

If the application treats the transaction as an offline approval, then after completing the other processing in this section, the application shall approve the transaction as described in section 17.5.5 and section 17.5.8.

17.5.5 Application Approves Transaction (TC)

When the transaction is to be approved, the application returns a Transaction Certificate (TC) type Application Cryptogram in the response to the second GENERATE AC command. The application sets bits in the CVR to indicate the type of Application Cryptogram to the issuer, and also updates the CVR bits as necessary to indicate the current conditions in the application.

Req 17.77 (Update CVR bits when approve transaction):

Prior to building the Issuer Application Data data element and generating the response cryptogram, the application shall:

- *update CVR bits as described in section 17.5.7.*
- *set the 'Application Cryptogram Type Returned in Second GENERATE AC' bits in the CVR to the value 01b (TC).*
- *set the 'CDA Performed' bit in the CVR to the value 1b if the terminal requested CDA (that is, if the 'CDA Requested' bit of the P1 parameter in the GENERATE AC command from the terminal was set to 1b).*

Prior to responding to the GENERATE AC command, the application sets the Cryptogram Information Data (CID) to the value '40' (TC) to indicate to the terminal that the Application Cryptogram in the GENERATE AC response is a TC.

17.5.6 Application Declines Transaction (AAC)

When the transaction is to be declined, the application returns an AAC type Application Cryptogram in the second GENERATE AC response. The application sets bits in the CVR to indicate the type of Application Cryptogram to the issuer, and also updates the CVR bits as necessary to indicate the current conditions in the application.

Req 17.78 (Update CVR bits when decline transaction):

Prior to generating the response cryptogram, the application shall:

- *update CVR bits as described in section 17.5.7, before building the IAD for the second GENERATE AC response*
- *set the 'Application Cryptogram Type Returned in Second GENERATE AC' bits in the CVR to the value 00b (AAC).*

Prior to responding to the GENERATE AC command, the application sets the Cryptogram Information Data (CID) to the value '00' (AAC) to indicate to the terminal that the Application Cryptogram in the GENERATE AC response is an AAC.

17.5.7 CVR Updates

The application performs the following in order to update the indicators in the CVR before building the Issuer Application Data to be returned in the second GENERATE AC response.

17.5.7.1 Lower Offline Transaction Count Limit Exceeded Check

This issuer-optional check provides the issuer notification that a lower limit for the count of offline transactions has been exceeded.

Req 17.79 (Check lower offline transaction count):

*For each value of x for which Counter x is active in the Profile **and** Counter x is greater than Counter x Lower Limit, the application shall set the 'Lower Offline Transaction Count Limit Exceeded' bit in the CVR to the value 1b.*

Otherwise (that is, none of the active Counter x counters have exceeded their Lower Limit), the application shall reset the 'Lower Offline Transaction Count Limit Exceeded' bit in the CVR to the value 0b.

17.5.7.2 Upper Offline Transaction Count Limit Exceeded Check

This issuer-optional check provides the issuer notification that an upper limit for the count of offline transactions has been exceeded.

Req 17.80 (Check upper offline transaction count):

*For each value of x for which Counter x is active in the Profile **and** Counter x is greater than Counter x Upper Limit, the application shall set the 'Upper Offline Transaction Count Limit Exceeded' bit in the CVR to the value 1b.*

Otherwise (that is, none of the active Counter x counters have exceeded their Upper Limit), the application shall reset the 'Upper Offline Transaction Count Limit Exceeded' bit in the CVR to the value 0b.

17.5.7.3 Lower Cumulative Offline Amount Limit Exceeded Check

This issuer-optional check provides the issuer notification that a lower limit for the cumulative amount of offline transactions has been exceeded.

Req 17.81 (Check lower cumulative offline amount):

*For each value of x for which Accumulator x is active in the Profile **and** Accumulator x is greater than the Accumulator x Lower Limit, the application shall set the 'Lower Cumulative Offline Amount Limit Exceeded' bit in the CVR to the value 1b.*

Otherwise (that is, none of the active Accumulator x accumulators have exceeded their Lower Limit), the application shall reset the 'Lower Cumulative Offline Amount Limit Exceeded' bit in the CVR to the value 0b.

17.5.7.4 Upper Cumulative Offline Amount Limit Exceeded Check

This issuer-optional check provides the issuer notification that an upper limit for the cumulative amount of offline transactions has been exceeded.

Req 17.82 (Check lower cumulative offline amount):

*For each value of x for which Accumulator x is active in the Profile **and** Accumulator x is greater than the Accumulator x Upper Limit, the application shall set the 'Upper Cumulative Offline Amount Limit Exceeded' bit in the CVR to the value 1b.*

Otherwise (that is, none of the active Accumulator x accumulators have exceeded their Upper Limit), the application shall reset the 'Upper Cumulative Offline Amount Limit Exceeded' bit in the CVR to the value 0b.

17.5.7.5 PIN Try Counter

This mandatory check provides the issuer notification of the number of PIN tries remaining in the PIN Try Counter.

Req 17.83 (Copy PIN Try Counter):

The application shall set the 'Low Order Nibble of PIN Try Counter' bits in the CVR to the value of the low-order nibble of the PIN Try Counter, using identical bit settings.

17.5.7.6 PIN Try Limit Exceeded

This mandatory check provides the issuer notification that the PIN Try Limit has been exceeded (in either the current or a preceding transaction), and enables the application decision regarding transaction disposition to consider this information.

Req 17.84 (Check PIN Try Limit):

If the value of the PIN Try Counter is zero, then the application shall set the 'PIN Try Limit Exceeded' bit in the CVR to 1b; otherwise, the application shall reset the bit to the value 0b.

17.5.7.7 Issuer Script Processing Failed

This mandatory check provides the issuer with notification that issuer script processing failed either in a previous online transaction (without the conditions to reset the indicator being met), or in the current transaction during processing of a script command received before the second GENERATE AC command.

Req 17.85 (Set Issuer Script Processing Failed):

The Application shall set the 'Issuer Script Processing Failed' bit in the CVR to the value of the 'Script Failed' bit in the PTH.

17.5.7.8 Number of Issuer Script Commands

This mandatory check provides the issuer with a count of the number of issuer script commands processed.

Req 17.86 (Copy Number of Issuer Script Commands):

The application shall set the 'Number of Issuer Script Commands Containing Secure Messaging Processed' bits in the CVR to the value of the Issuer Script Command Counter.

17.5.8 Respond to GENERATE AC Command

17.5.8.1 Build Issuer Application Data

Req 17.87 (Build IAD):

The application shall build the Issuer Application Data to be sent in the response, coded as specified in the CCD Part of EMV Book 3, Annex C.7, for a CCD-compliant application with a Format Code of 'A', with:

- *For each Accumulator x that is active for the transaction **and** for which the 'Send Accumulator in IAD' bit in the Accumulator Profile Control for Accumulator x has the value 1b:*
 - *If the 'Send Accumulator Balance' bit in the Accumulator Profile Control for Accumulator x has the value 1b, then the value (Accumulator x Upper Limit minus Accumulator x) shall be sent.*
 - *Otherwise (the 'Send Accumulator Balance' = 0b), the value of Accumulator x shall be sent.*
- *The profile-specific requirements shown in Table 17-13.*
- *If the 'Encipher Counters Portion of IAD' bit in the Issuer Options Profile Control has the value 1b, the Counters portion of Issuer Application Data shall be enciphered (see section 20) before generating the Application Cryptogram.*

IAD Byte	Description	Value
1	Length	'0F'
2	CCI	set to the value of the Profile CCI in the Issuer Options Profile Control for the transaction ('A5' for CCD-compliant profiles)
3	DKI	Set to the value of the Profile DKI in the Issuer Options Profile Control for the transaction (issuer-discretionary)
4-8	CVR	set by application processing
9-16	Counters	<p>Begins with the following:</p> <ul style="list-style-type: none"> If Accumulator 1 is active for the transaction and the 'Send Accumulator in IAD' bit in the Accumulator Profile Control for Accumulator 1 has the value 1b, then Accumulator 1 (Value or Balance) is sent in Counters bytes 1-6. Otherwise, if Accumulator 2 is active for the transaction and the 'Send Accumulator in IAD' bit in the Accumulator Profile Control for Accumulator 2 has the value 1b, then Accumulator 2 (Value or Balance) is sent in Counters bytes 1-6. Otherwise, if VLP Available Funds is active for the transaction and the 'Send Accumulator in IAD' bit in the VLP Profile Control has the value 1b, then VLP Available Funds is sent in Counters bytes 1-6. <p>The remaining bytes of Counters shall contain the values of each Counter x that is active for the transaction and for which the 'Send Counter in IAD' bit in the Counter Profile Control for Counter x has the value 1b, in priority order based upon the counter number (that is, the value of x for Counter x), with the lowest numbered counter having the highest priority.</p> <p>The default value for these bytes is personalised in bytes 9-16 of the Default Issuer Application Data. Any portion of these bytes not filled with an accumulator or counter should use the default value.</p>
17	Length	'0F'

Table 17-13: Issuer Application Data for Second GENERATE AC
(continues)

IAD Byte	Description	Value
18	Profile ID	Profile ID used for the transaction
19-32	Issuer-Discretionary	<p>If more than one accumulator is to be sent in the IAD, these bytes contain the remaining accumulators that were not sent in bytes 9-16, in the order shown:</p> <p>The remaining bytes shall contain the values of each Counter <i>x</i> not included in bytes 9-16 that is active for the transaction and for which the 'Send Counter in IAD' bit in the Counter Profile Control for Counter <i>x</i> has the value 1b, in priority order based upon the counter number (that is, the value of <i>x</i> for Counter <i>x</i>), with the lowest numbered counter having the highest priority.</p> <p>The default value for these bytes is personalised in bytes 19-32 of the Default Issuer Application Data. Any portion of these bytes not filled with an accumulator or counter should use the default value.</p> <p>NOTE: Issuers may request specific data in this field, but this functionality is outside the scope of this specification.</p>

Table 17-13: Issuer Application Data for Second Generate AC, continued

See Figure 15-2 for examples illustrating the result of building the IAD with accumulators and counters.

17.5.8.2 Generate Application Cryptogram

The application generates an Application Cryptogram using the data provided by the terminal and data from the card.

Data requirements, key requirements, and the algorithms used in the Application Cryptogram generation process are as detailed in Table CCD-3 and section 8 of the CCD part of *EMV Book 2*, for a CCD-compliant application with Cryptogram Version of '5'.

NOTE: Support for Cryptogram Versions with a value in the range '0' to '3' is beyond the scope of this specification. A Profile that uses a Cryptogram Version in this range is not CCD-compliant.

17.5.8.3 Log Transaction

If the issuer chooses to log transactions, the application appends information to the Transaction Log.

Req 17.88 (Update Transaction Log):

*Prior to responding to the GENERATE AC command, if **both** of the following are true:*

- *the 'Log Transactions' bit in the Issuer Options Profile Control has the value 1b*
- ***and any** of the following is true:*
 - ***all** of the following are true (all approvals are logged):*
 - *the response is a TC type Application Cryptogram*
 - ***and** the 'Log Approved Transactions' bit in the Application Control has the value 1b*
 - ***and** the 'Log Offline Only' bit in the Application Control has the value 0b (log all approved transactions)*
 - *or **all** of the following are true (online approvals are not logged):*
 - *the response is a TC type Application Cryptogram*
 - ***and** the 'Log Approved Transactions' bit in the Application Control has the value 1b*
 - ***and** the 'Log Offline Only' bit in the Application Control has the value 1b*
 - ***and** the terminal was unable to go online (that is, the ARC was Y3)*
 - *or **both** of the following are true (declined transactions are logged):*
 - *the response is an AAC type Application Cryptogram*
 - ***and** the 'Log Declined Transactions' bit in the Application Control has the value 1b*

then the application shall append to the Transaction Log the value only (omitting the tag and length) for the data elements listed in Table 17-14, in the order shown.

Data Element	Condition
<i>Amount, Authorised</i>	<i>always</i>
<i>Transaction Currency Code</i>	<i>always</i>
<i>Transaction Date</i>	<i>always</i>
<i>CVR</i>	<i>if the 'Log the CVR' bit in Application Control is set to the value 1b</i>
<i>ATC</i>	<i>if the 'Log the ATC' bit in Application Control is set to the value 1b</i>
<i>CID</i>	<i>if the 'Log the CID' bit in Application Control is set to the value 1b</i>
<i>Data extracted from the First GENERATE AC Command Data using the First GEN AC Unchanging Log Data Table</i>	<i>if present</i>
<i>Data extracted from the Second GENERATE AC Command Data using the Second GEN AC Log Data Table</i>	<i>if present</i>

Table 17-14: Data Elements to Log

17.5.8.4 Return GENERATE AC Response

If **both** of the following are true:

- CDA processing is requested by the terminal (that is, the 'CDA Requested' bit of the P1 parameter in the GENERATE AC command from the terminal was set to 1b)
- **and** the application is responding with a TC type Application Cryptogram

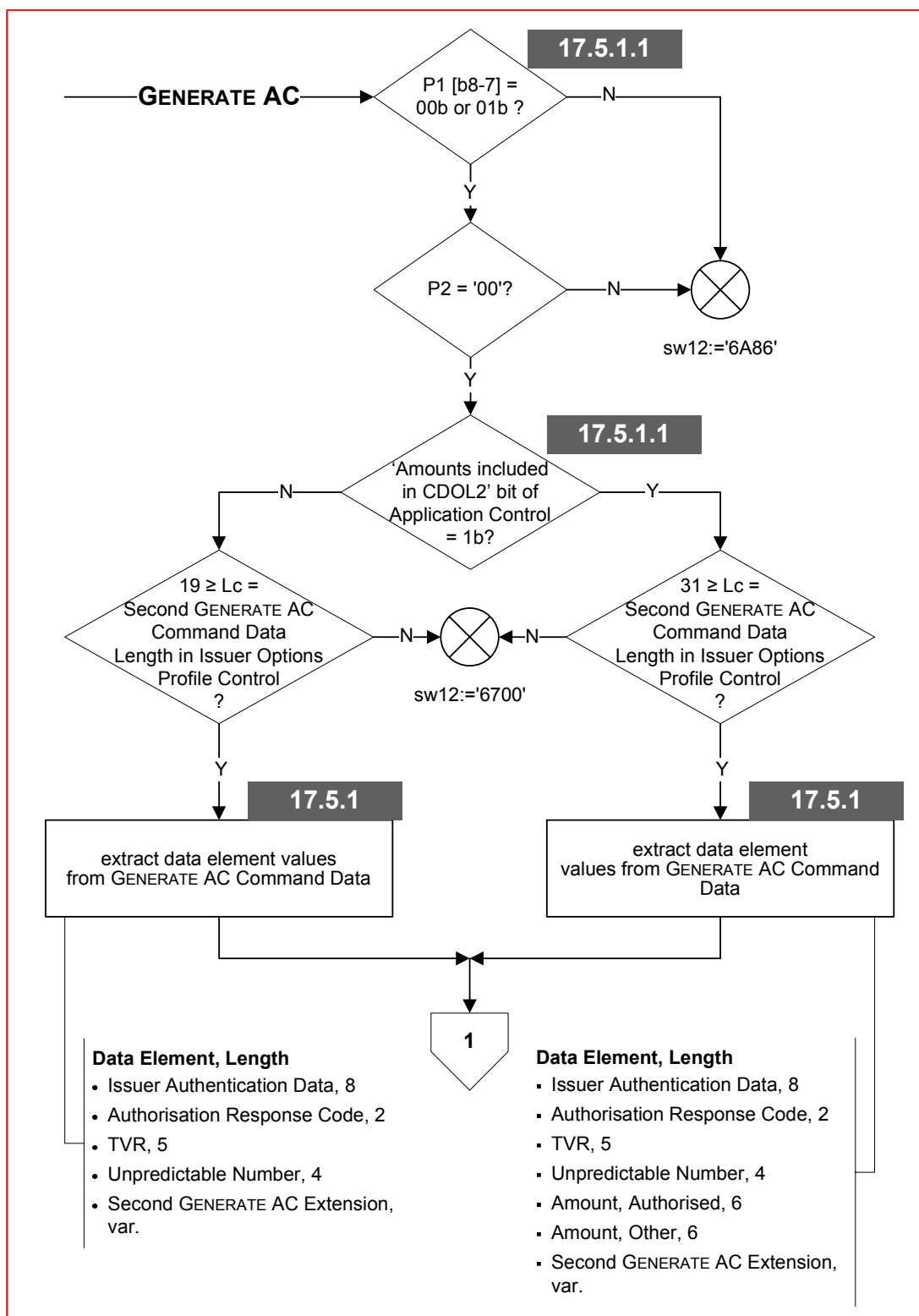
then the application:

- generates a dynamic signature from the Application Cryptogram as described in *EMV Book 2*, section 6.6.1
- returns the second GENERATE AC response as described in the CCD part of *EMV Book 2*, section 6.6.1 and Table CCD-1.

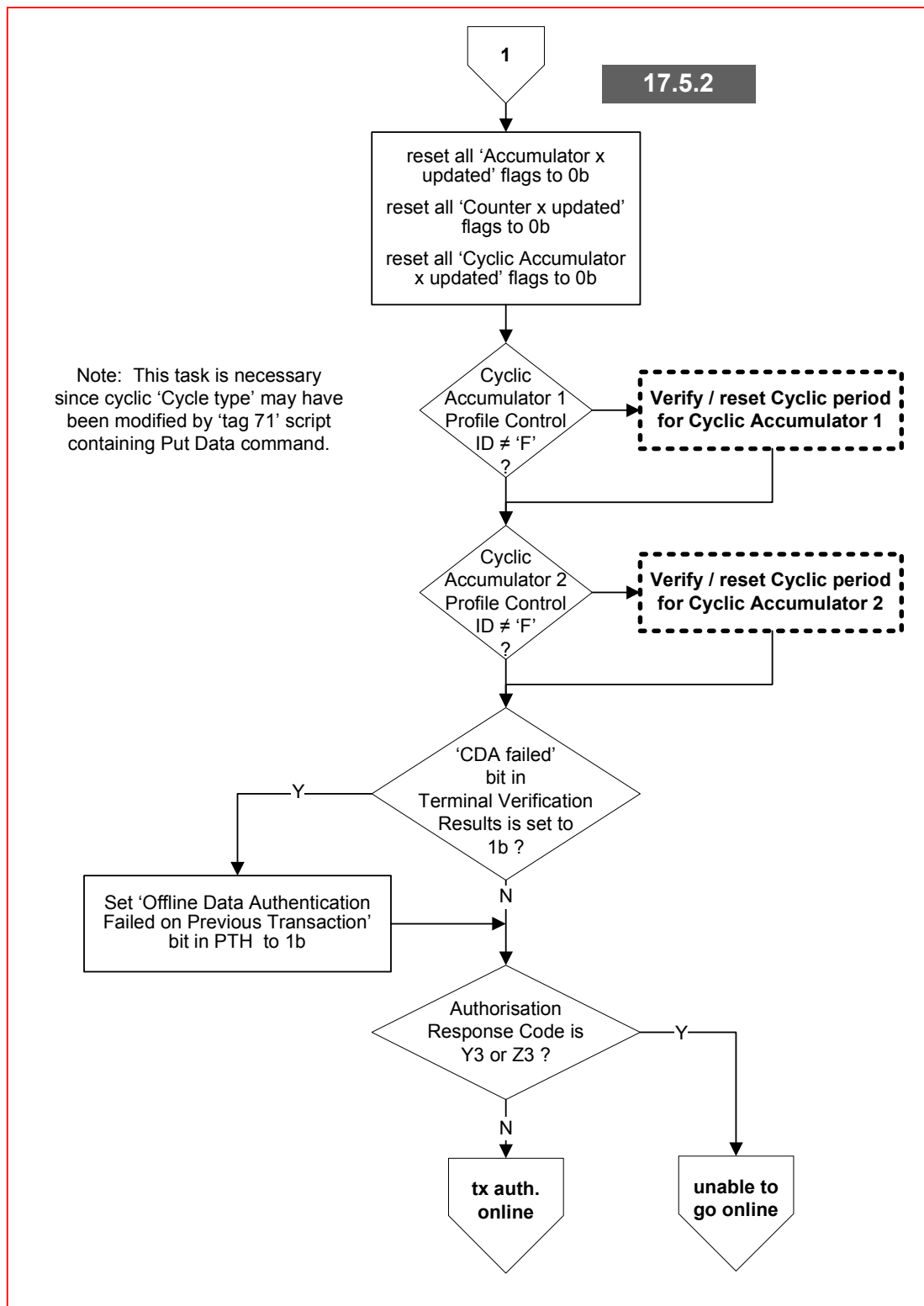
Otherwise, the application returns the second GENERATE AC response as described in the CCD part of *EMV Book 3*, section 6.5.5.4 and Table CCD-2.

17.6 Function Flow Charts

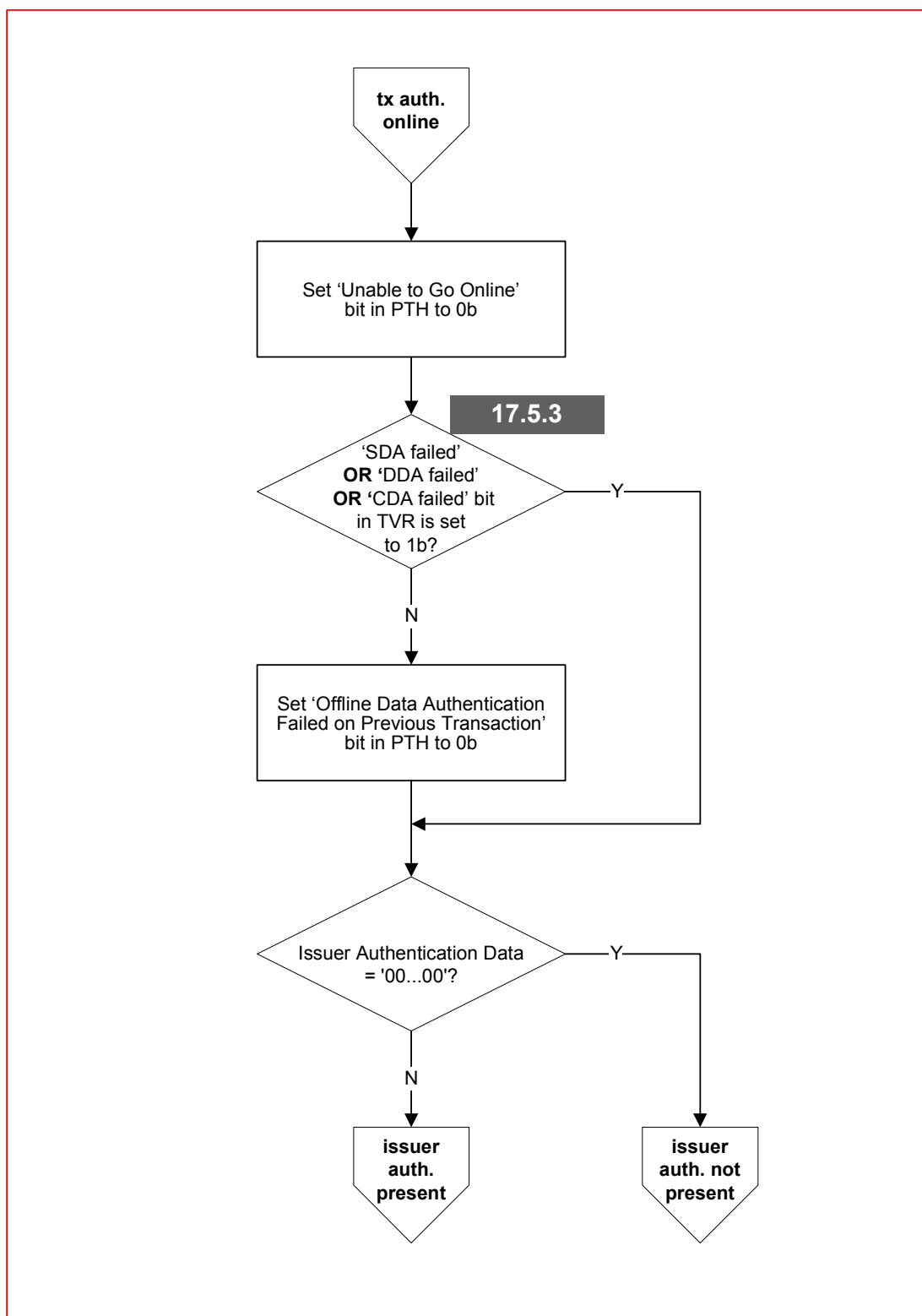
The following flow shows how an application could perform Second Card Action Analysis processing.



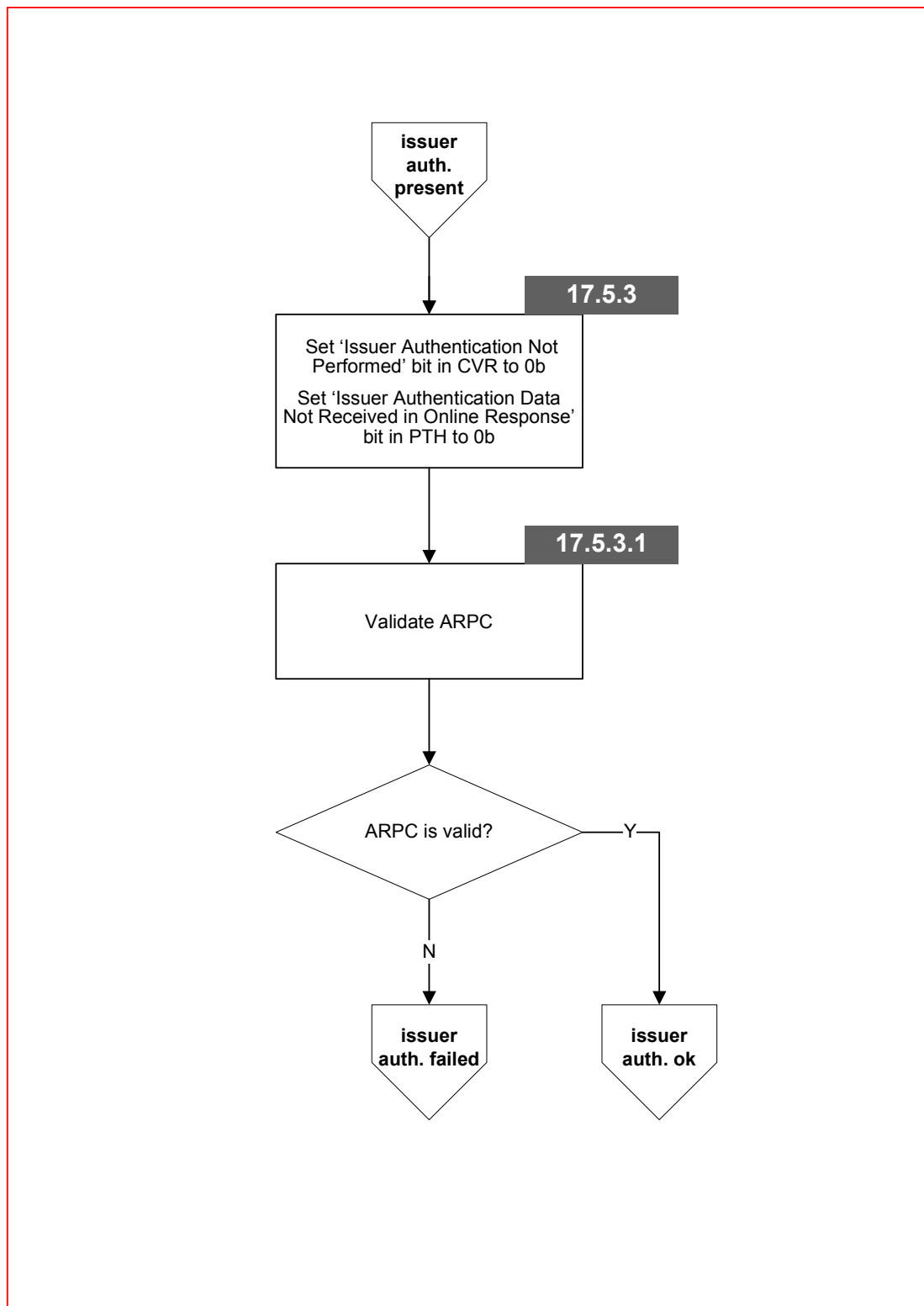
Flow 17-1 Second GENERATE AC Initial Flow

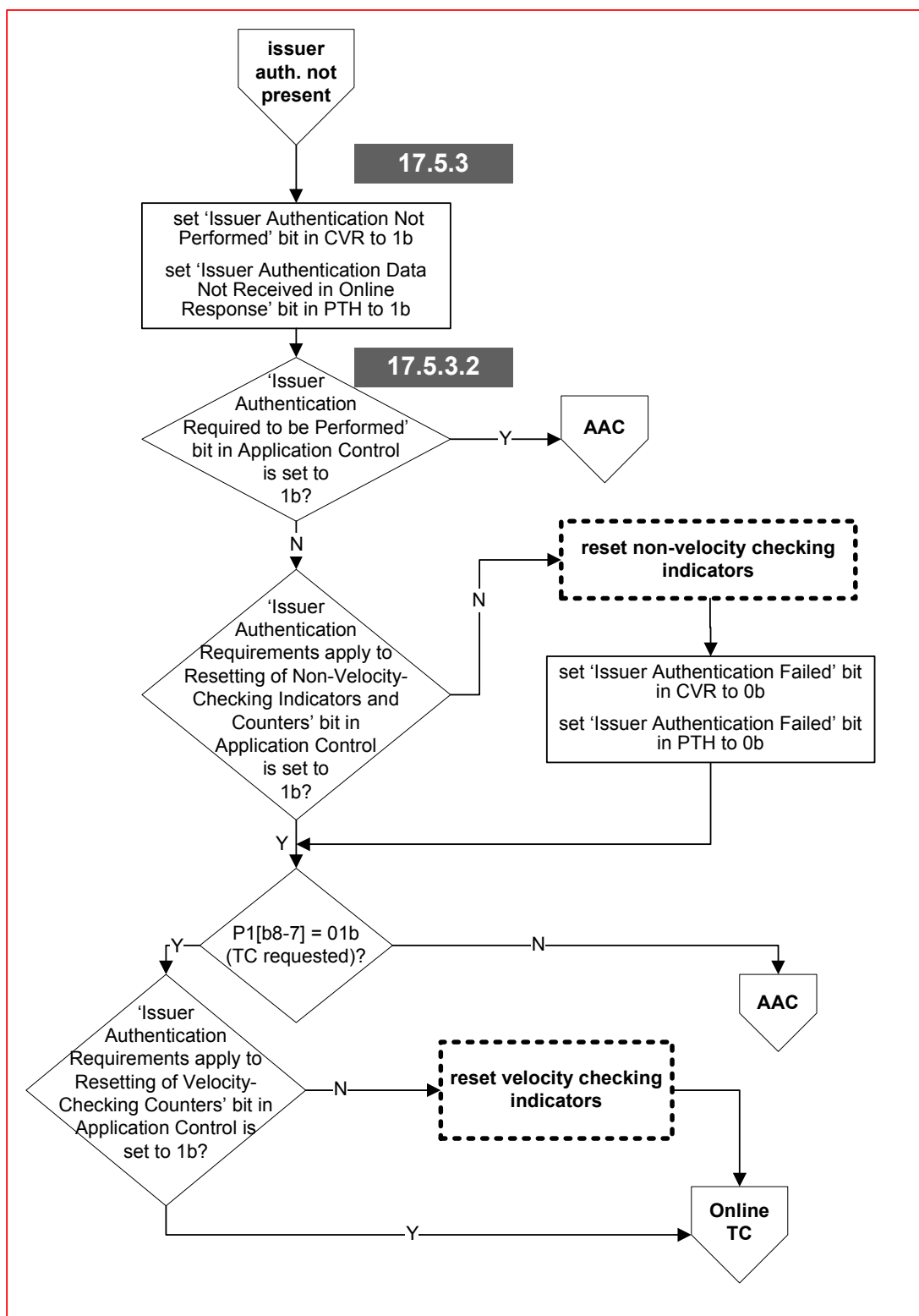


Flow 17-1.1 Second GENERATE AC Initial Flow, continued

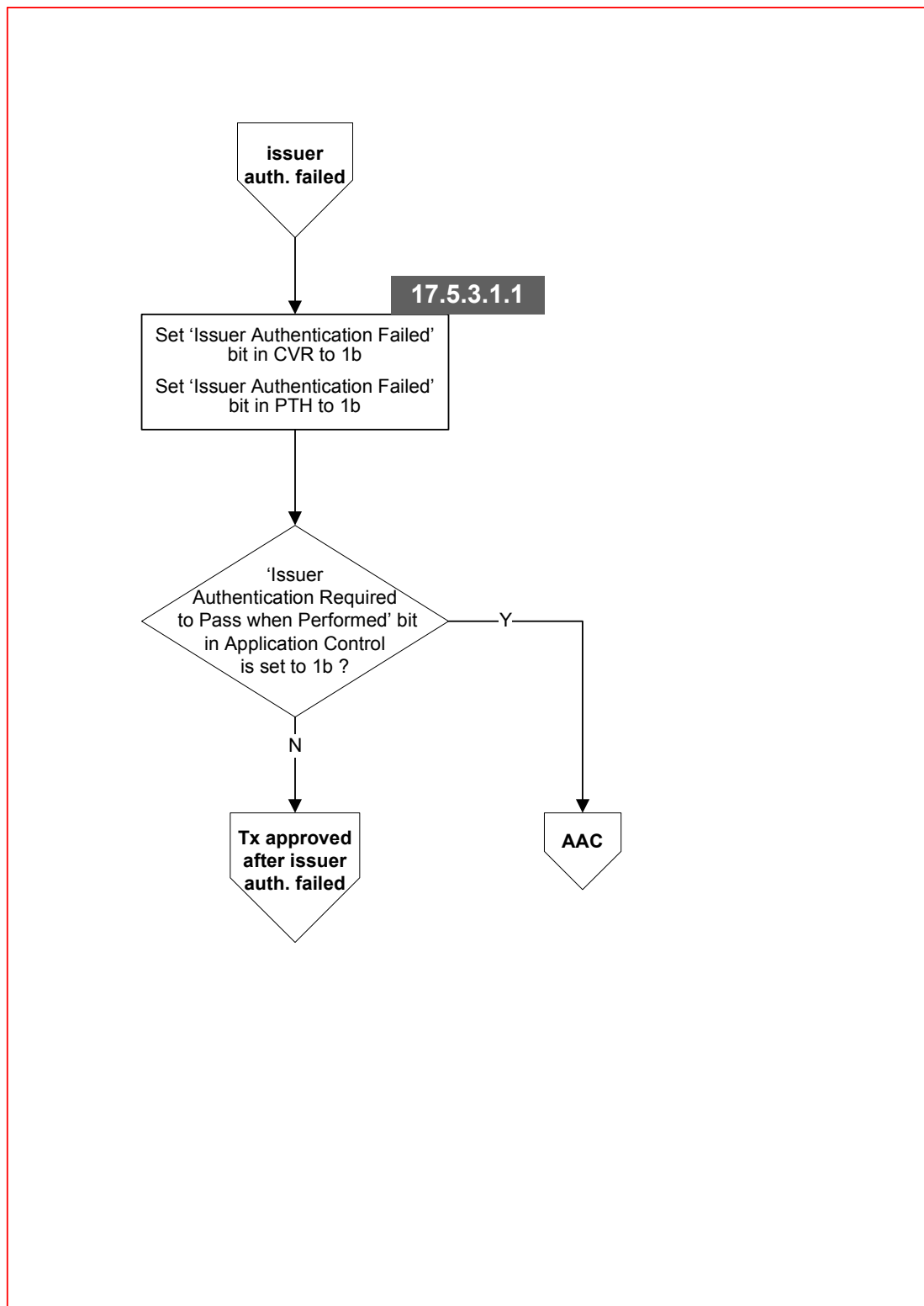


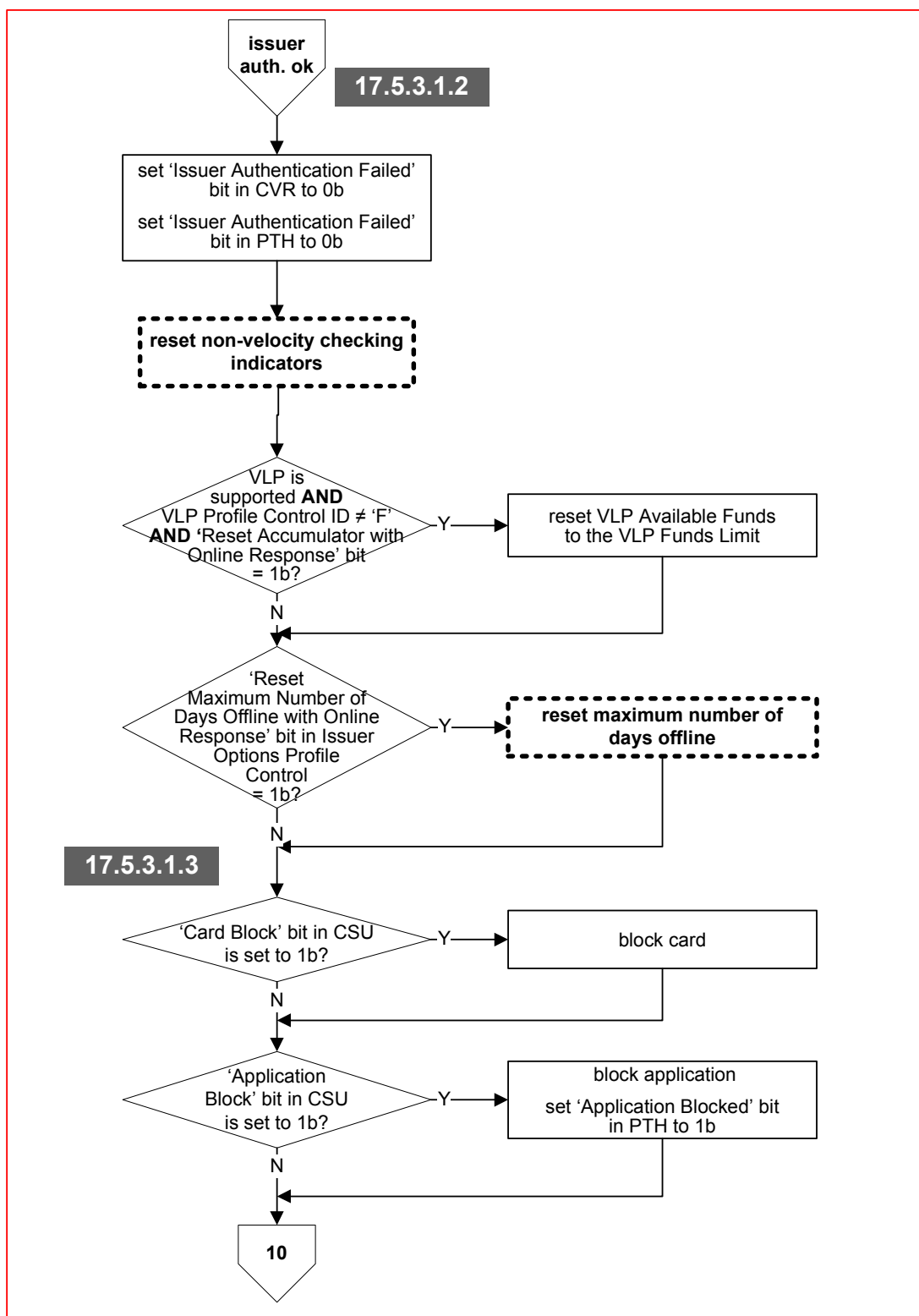
Flow 17-2 Transaction Authorization Online

**Flow 17-3 Issuer Authentication Present**

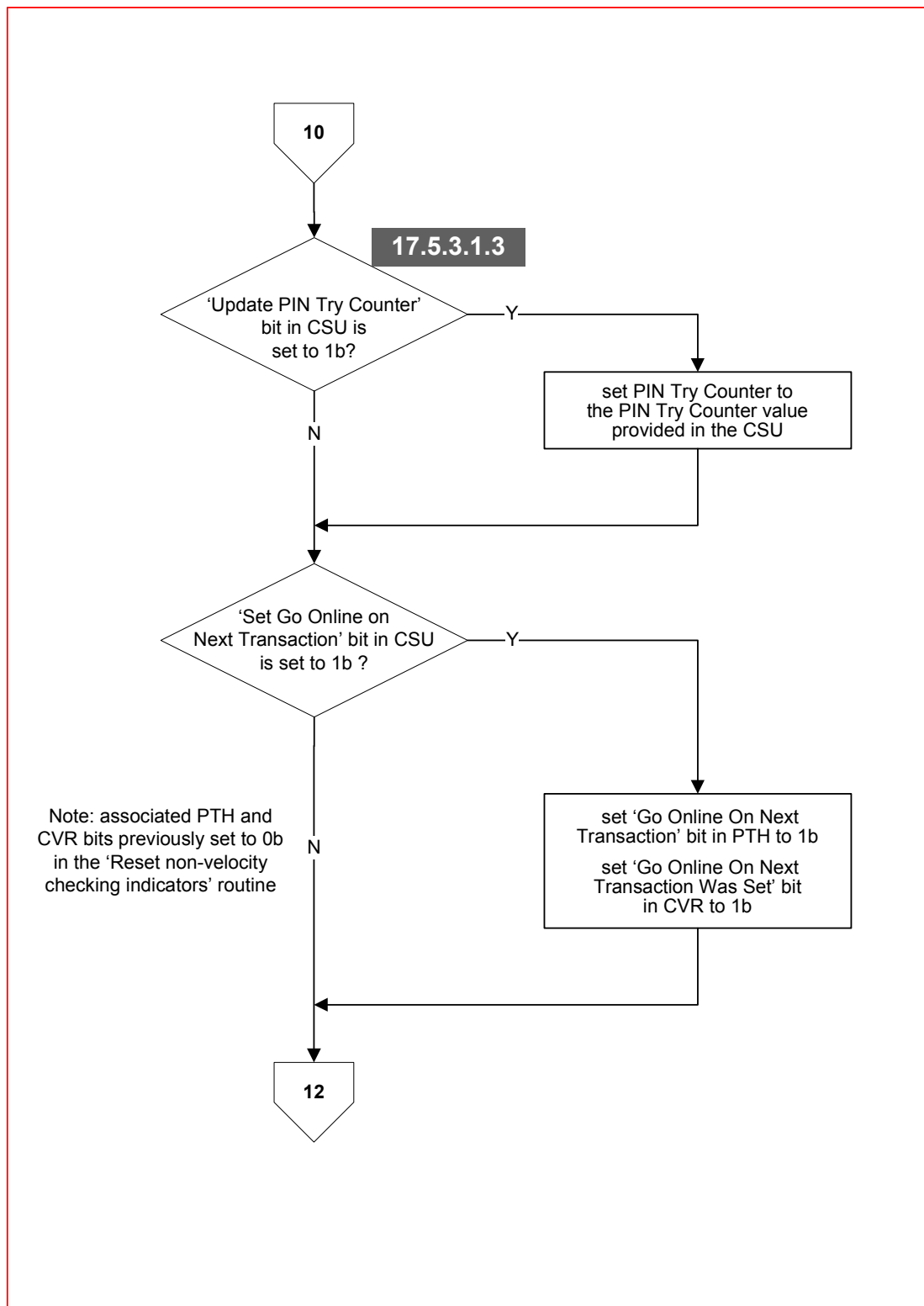


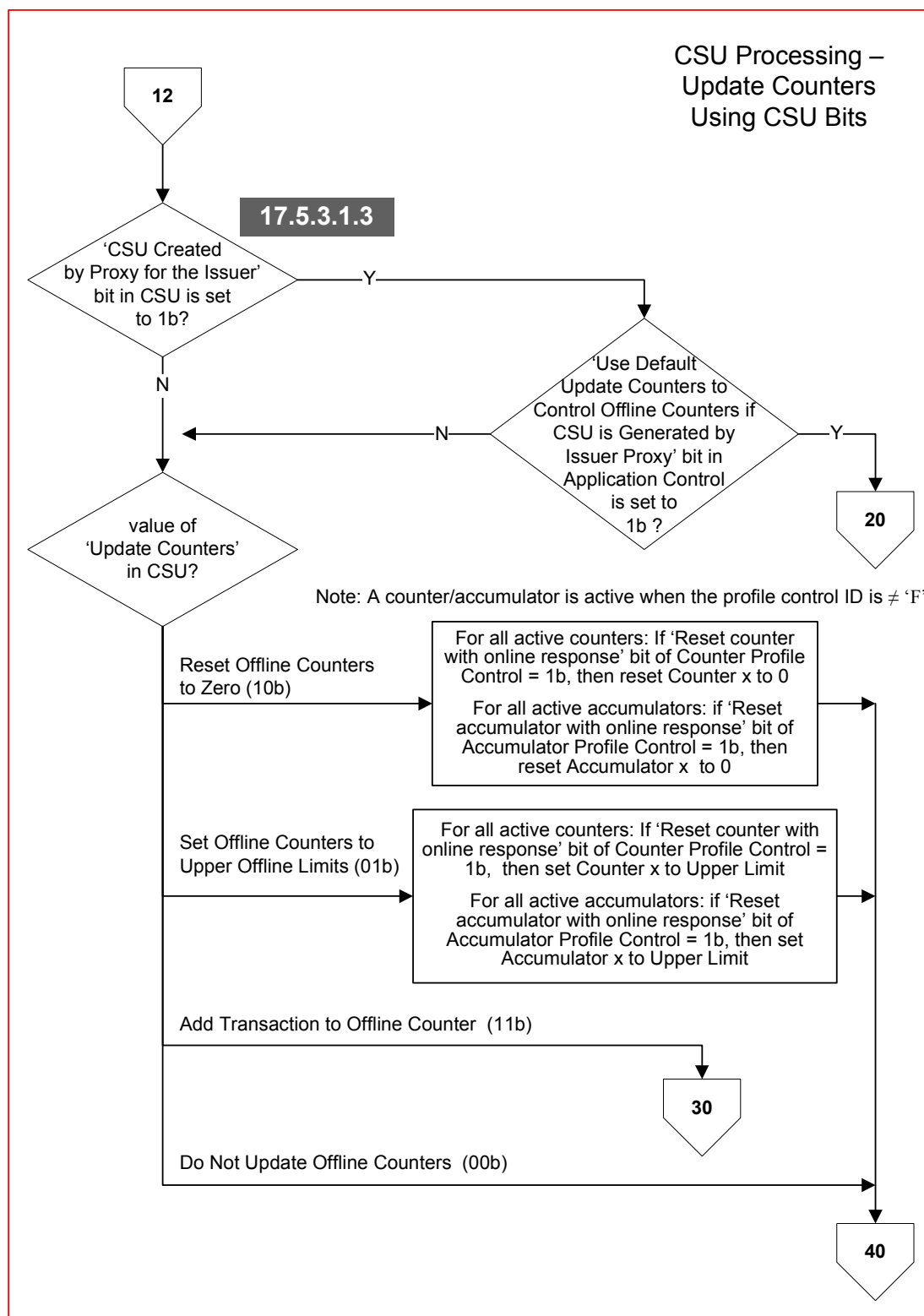
Flow 17-4 Issuer Authentication Not Present

**Flow 17-5 Issuer Authentication Failed**



Flow 17-6 Issuer Authentication Okay

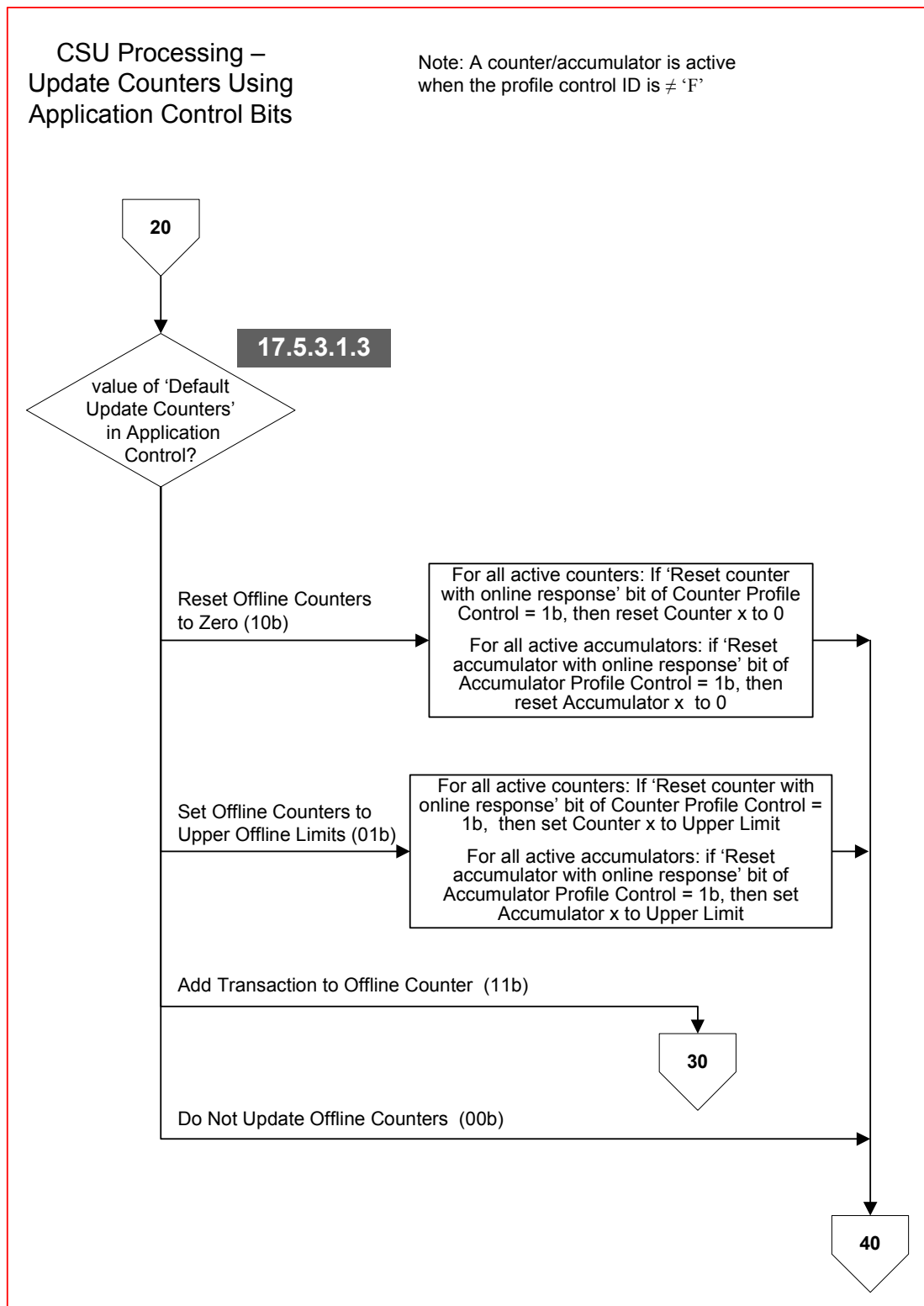
**Flow 17-6.1 Issuer Authentication Okay, continued**

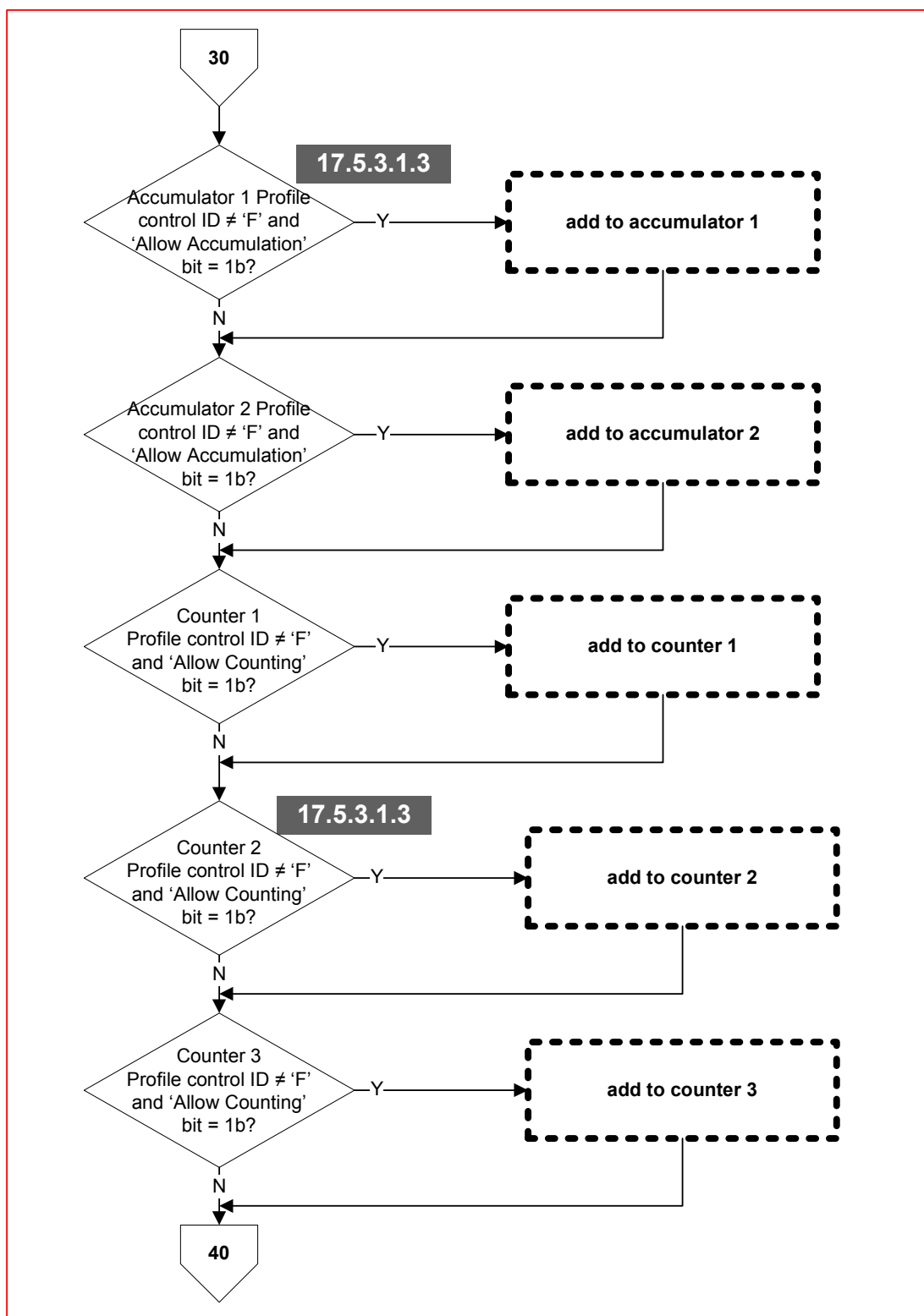


Flow 17-6.2 Issuer Authentication Okay, continued

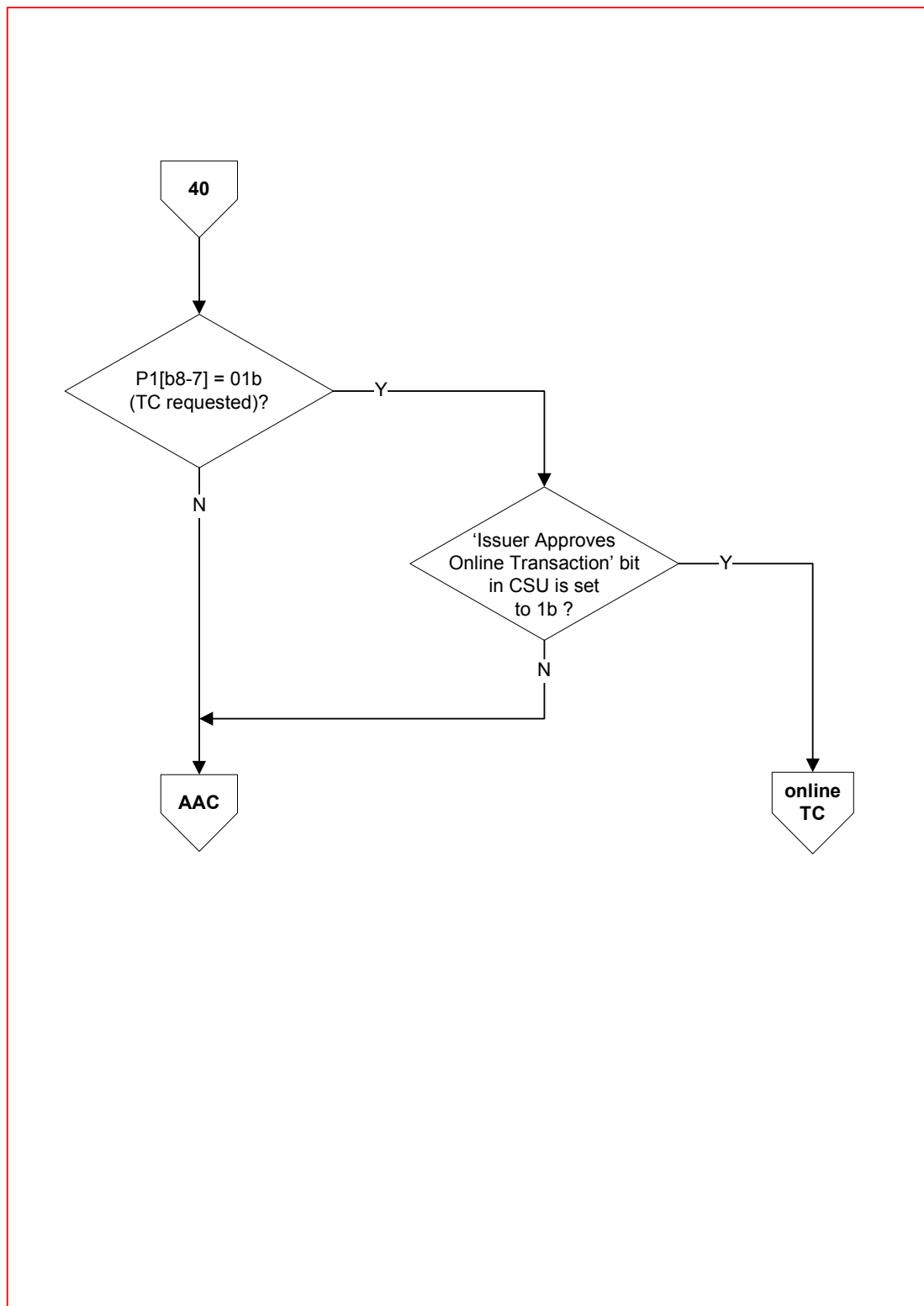
**CSU Processing –
Update Counters Using
Application Control Bits**

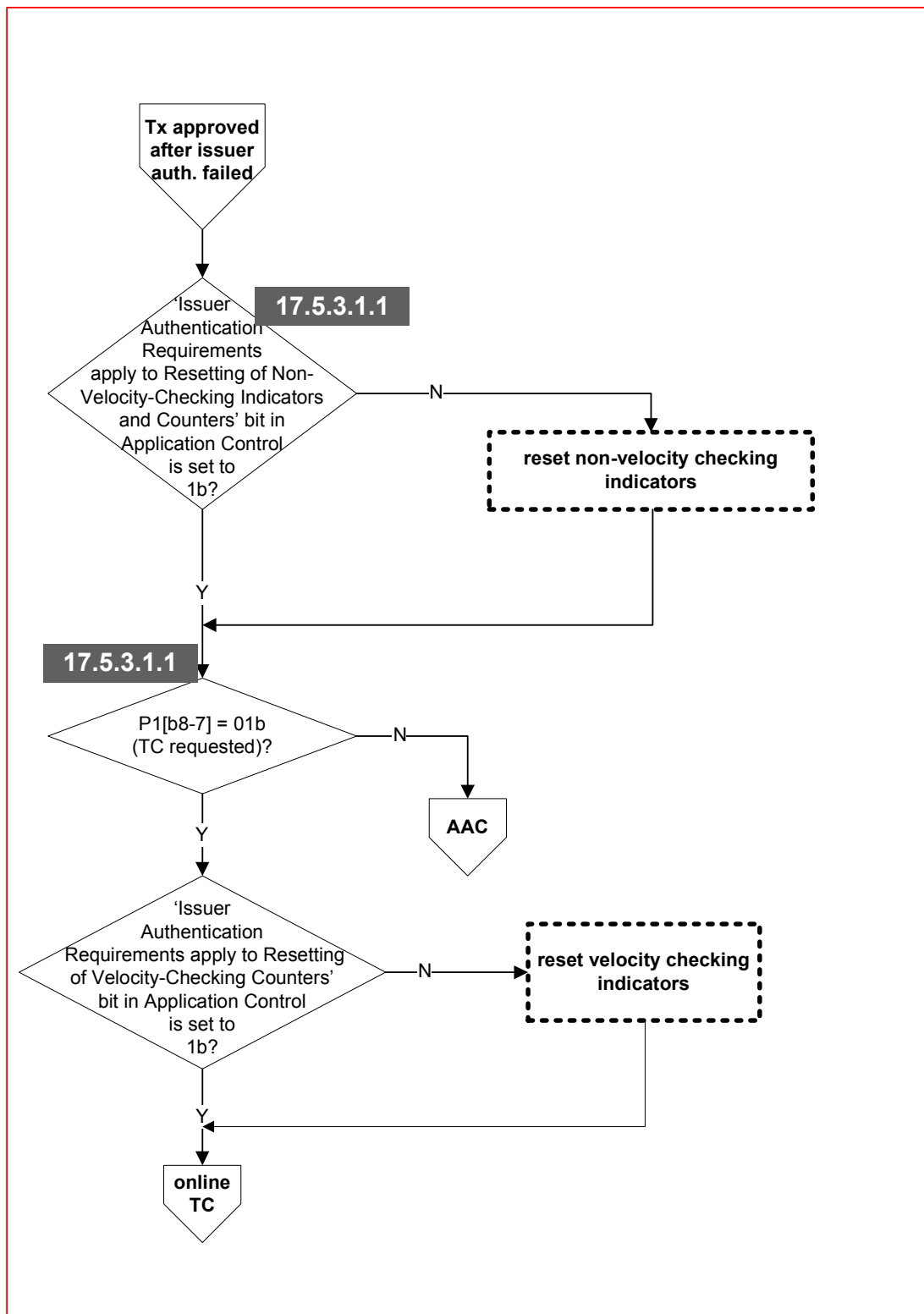
Note: A counter/accumulator is active
when the profile control ID is ≠ 'F'

**Flow 17-6.3 Issuer Authentication Okay, continued**

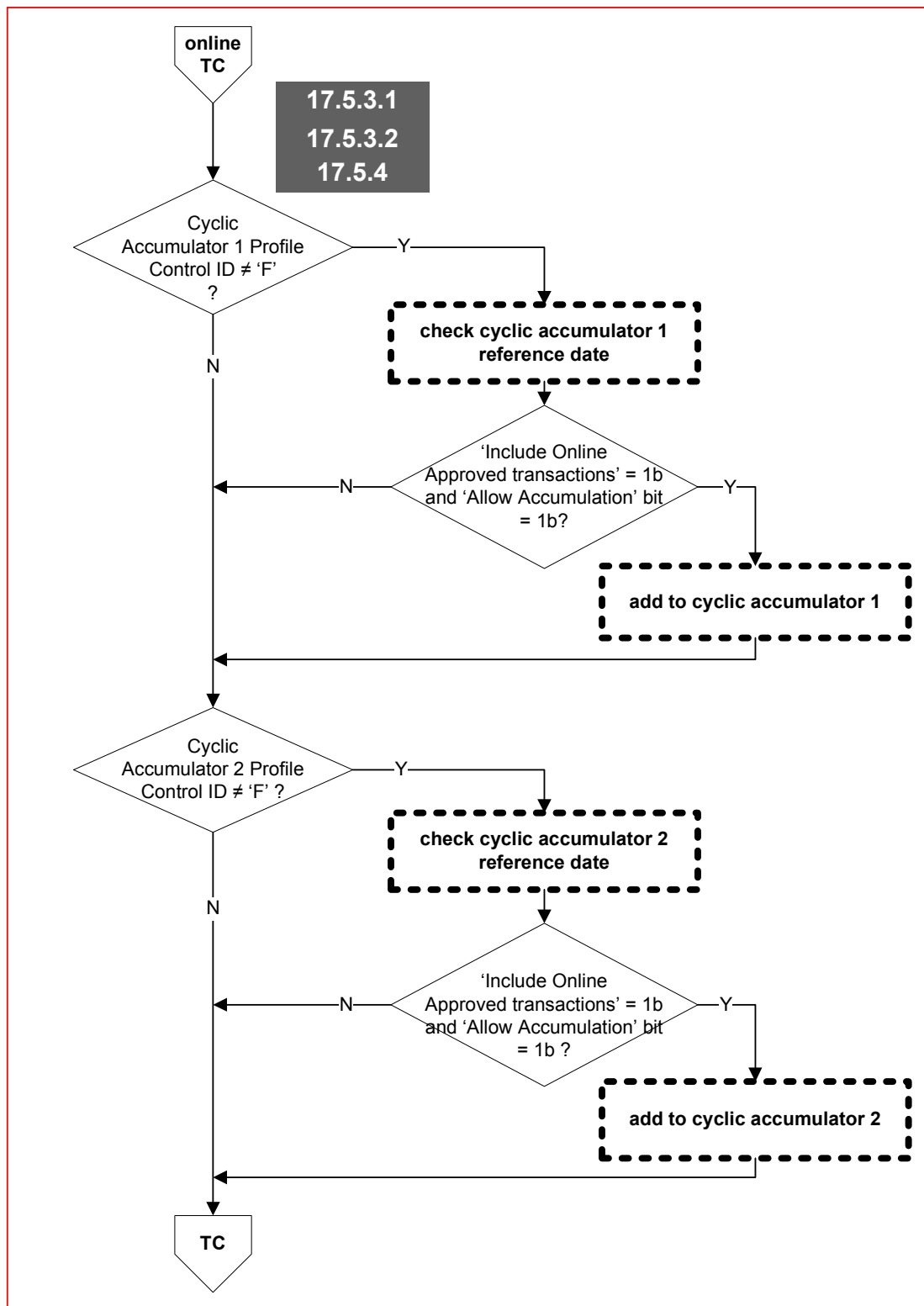


Flow 17-6.4 Issuer Authentication Okay, continued

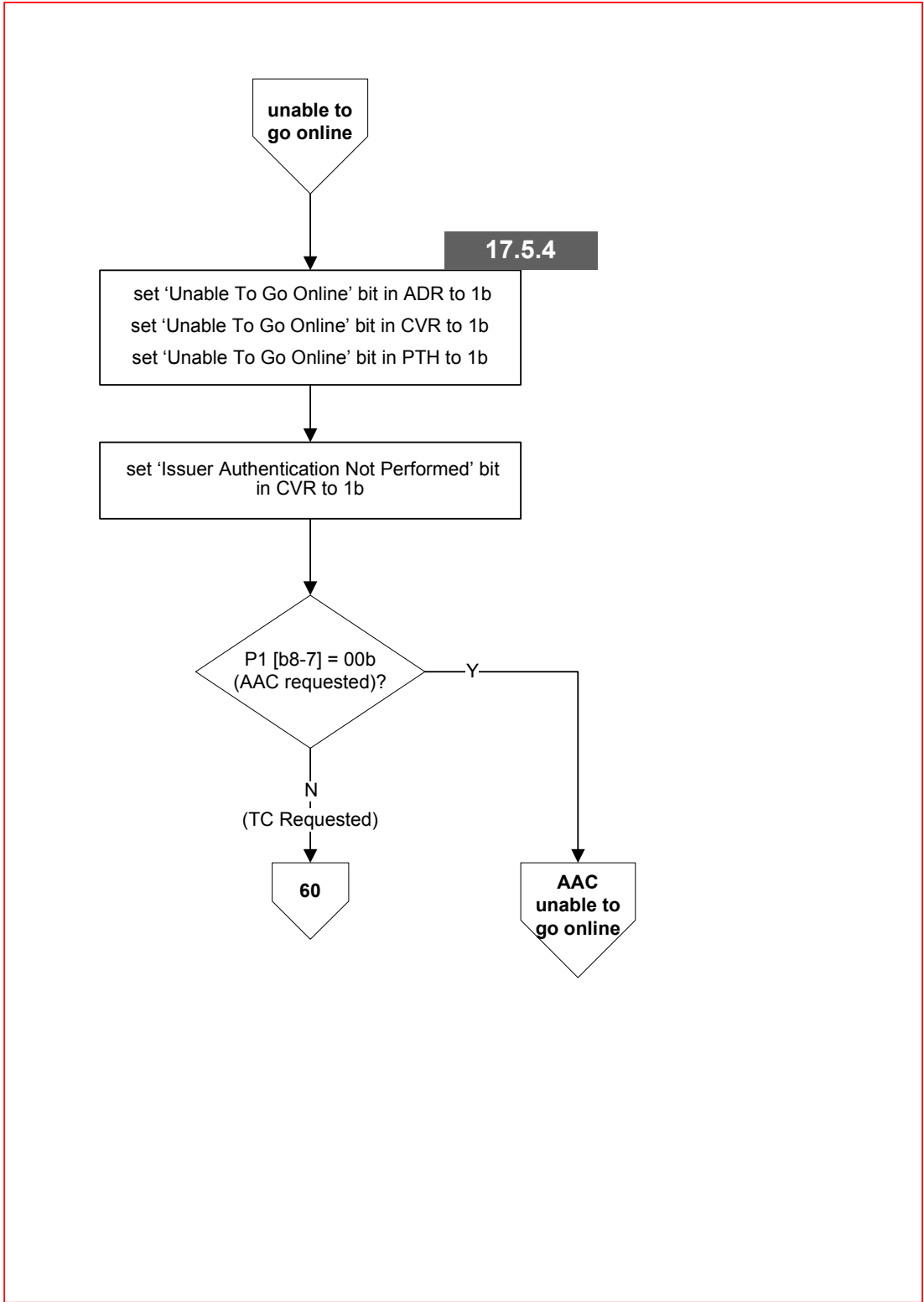
**Flow 17-6.5 Issuer Authentication Okay, continued**



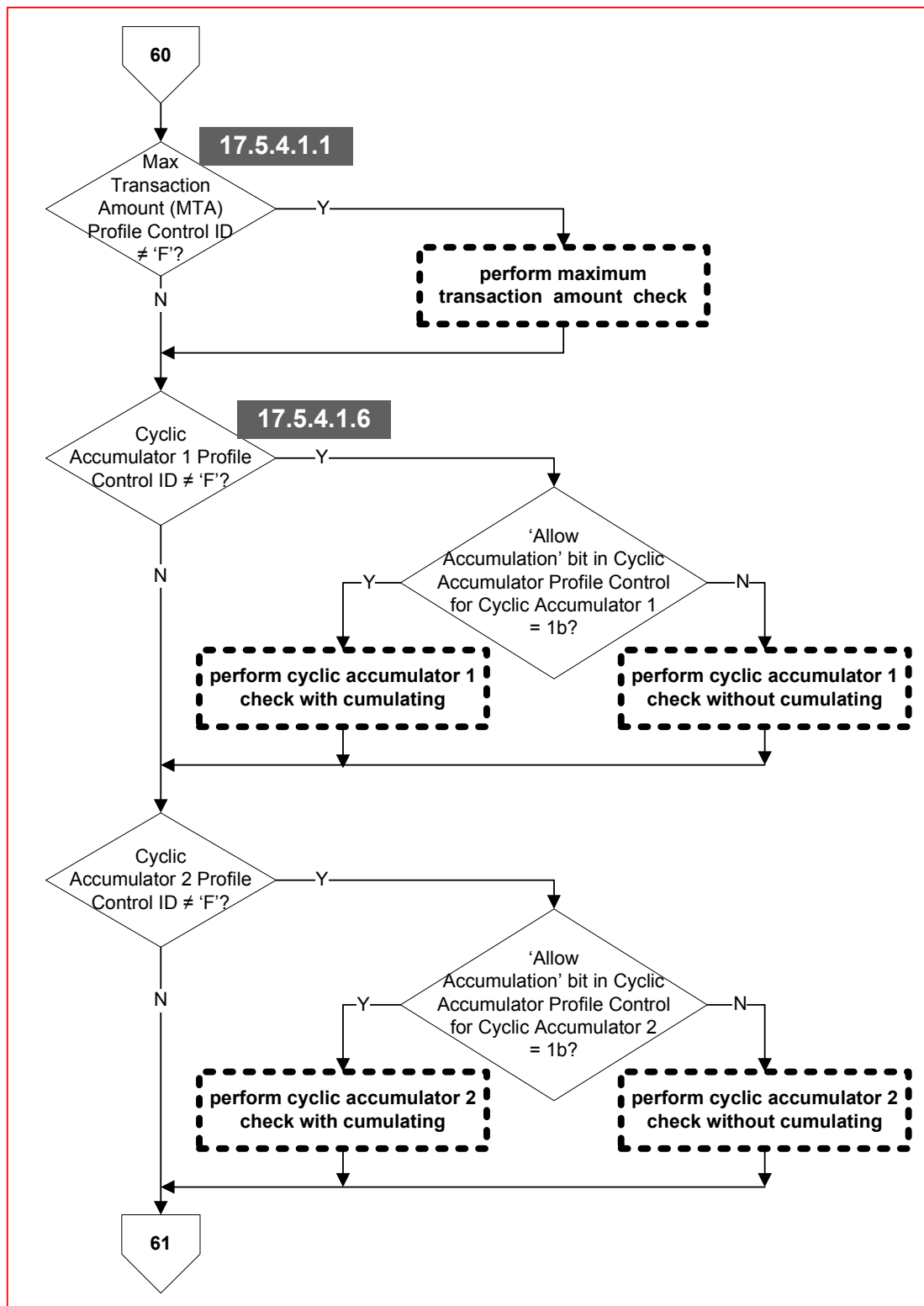
Flow 17-7 Transaction Approved after Issuer Authentication Failed



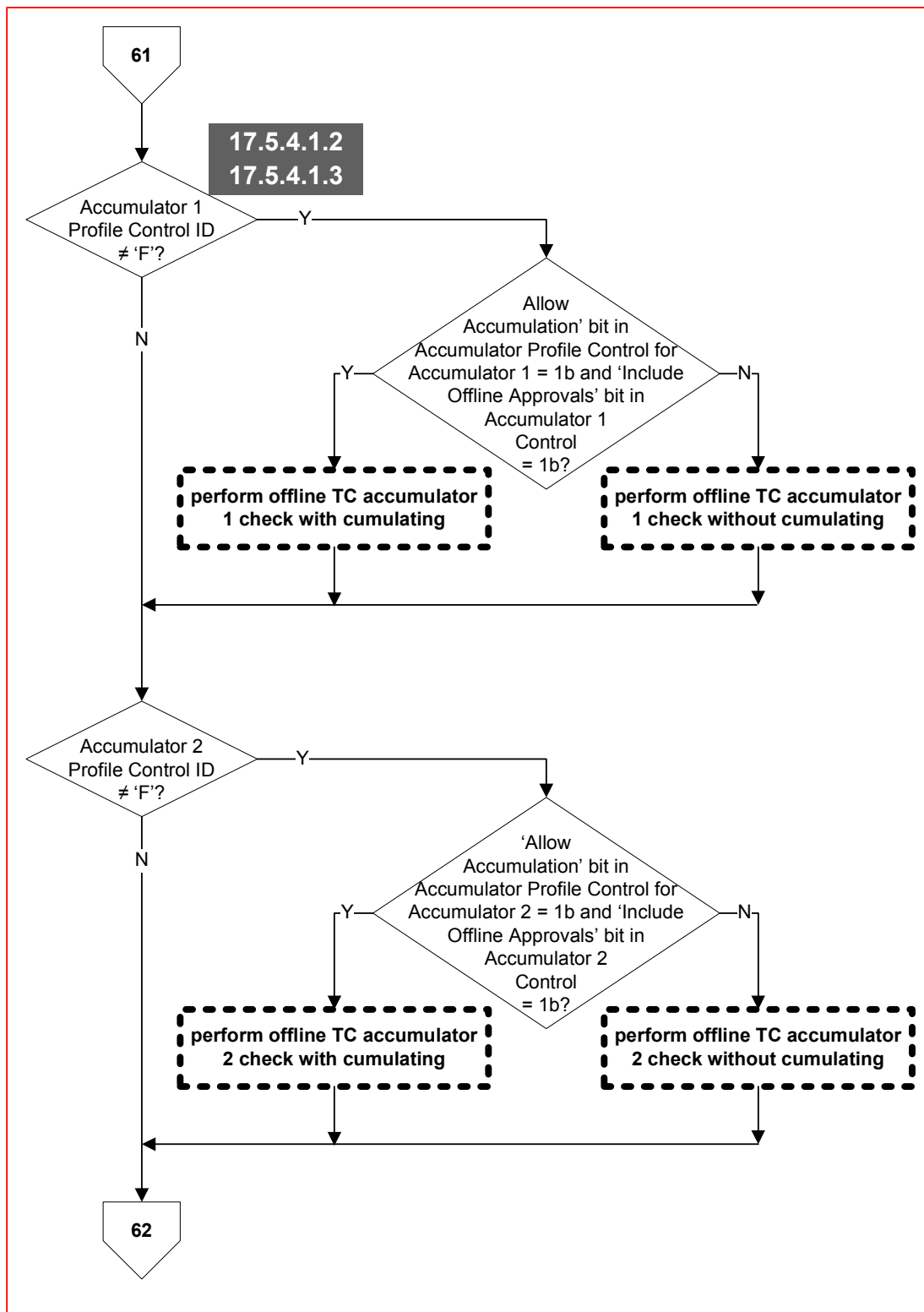
Flow 17-8 Online TC



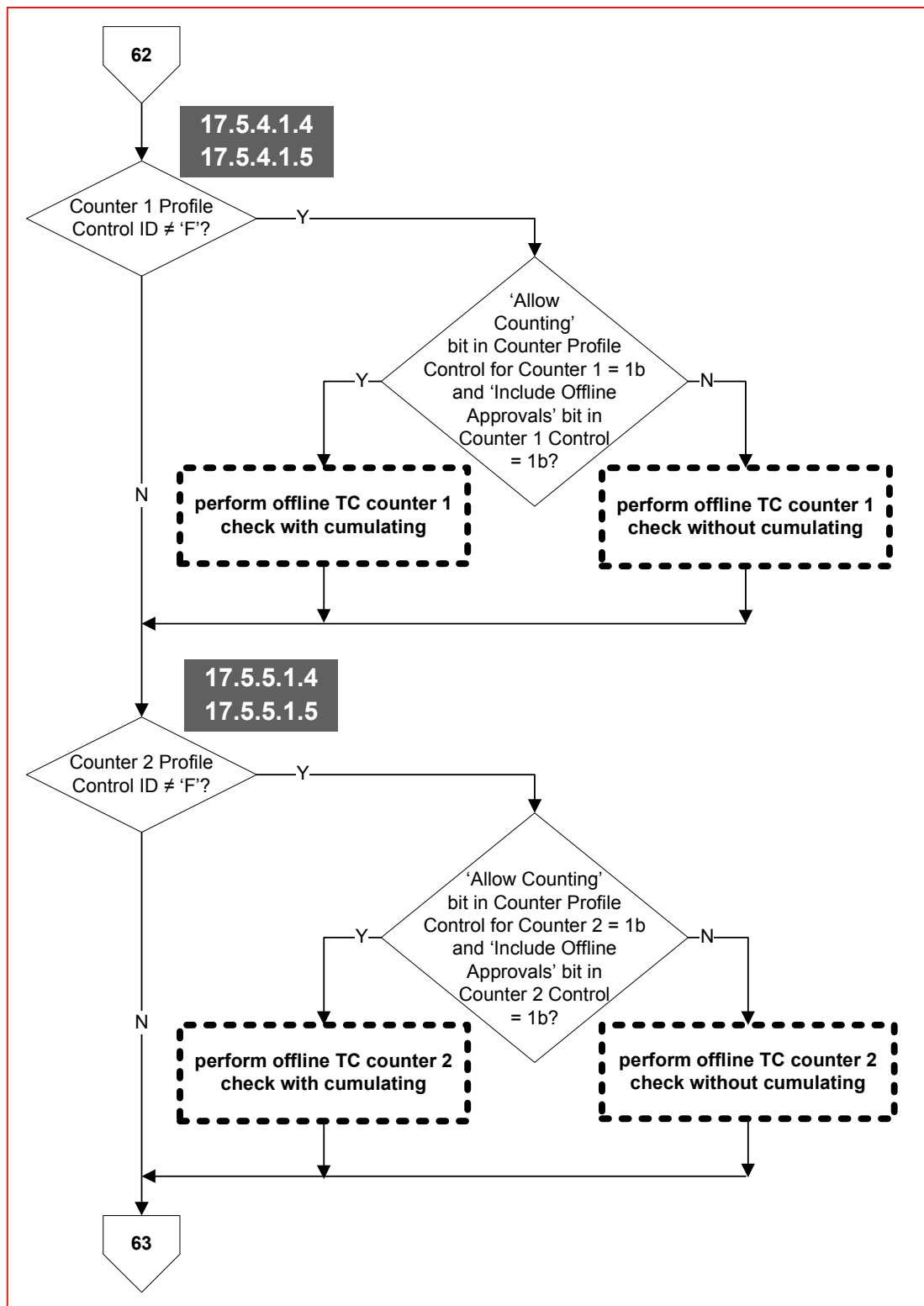
Flow 17-9 Unable To Go Online



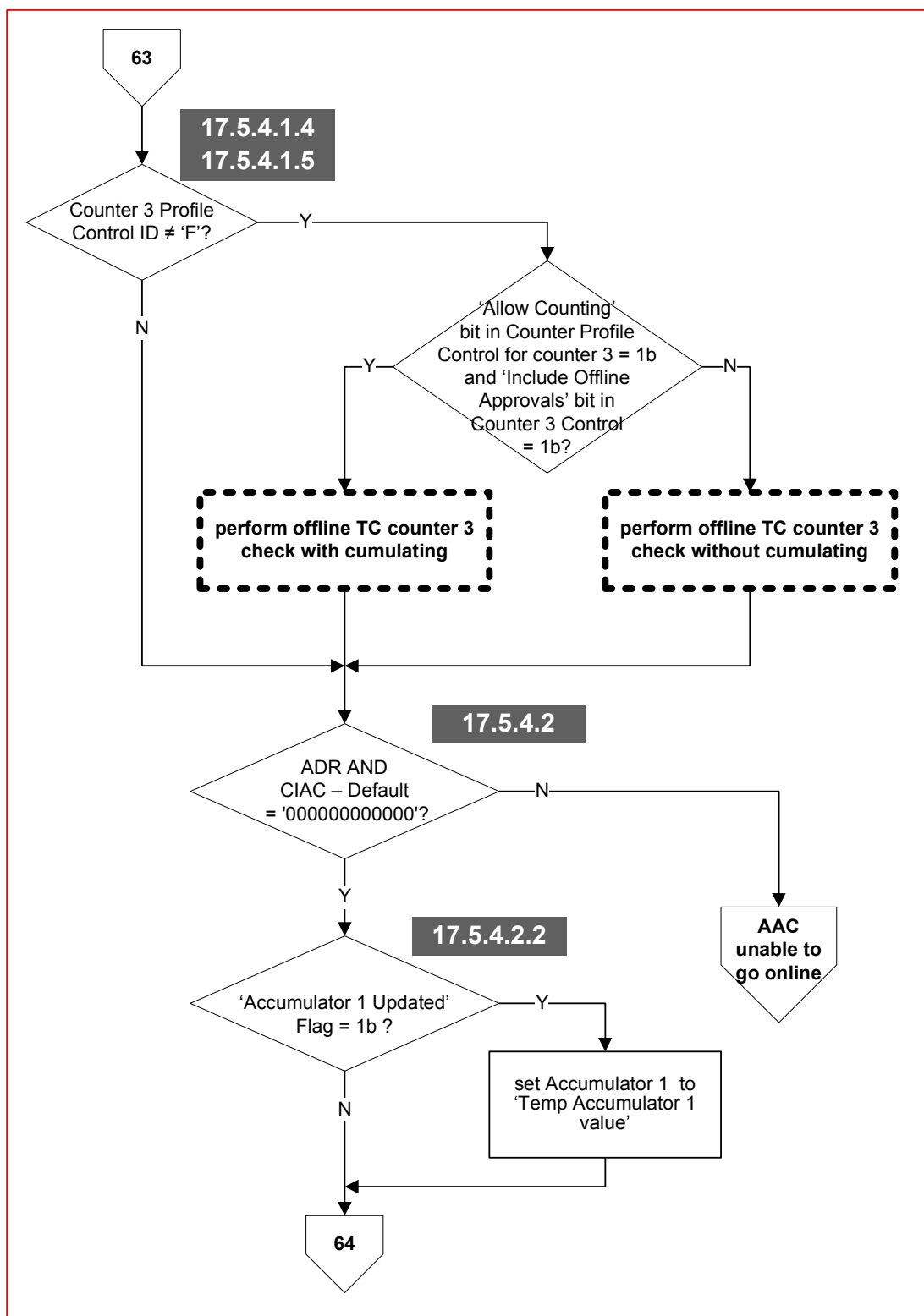
Flow 17-9.1 Unable To Go Online, continued



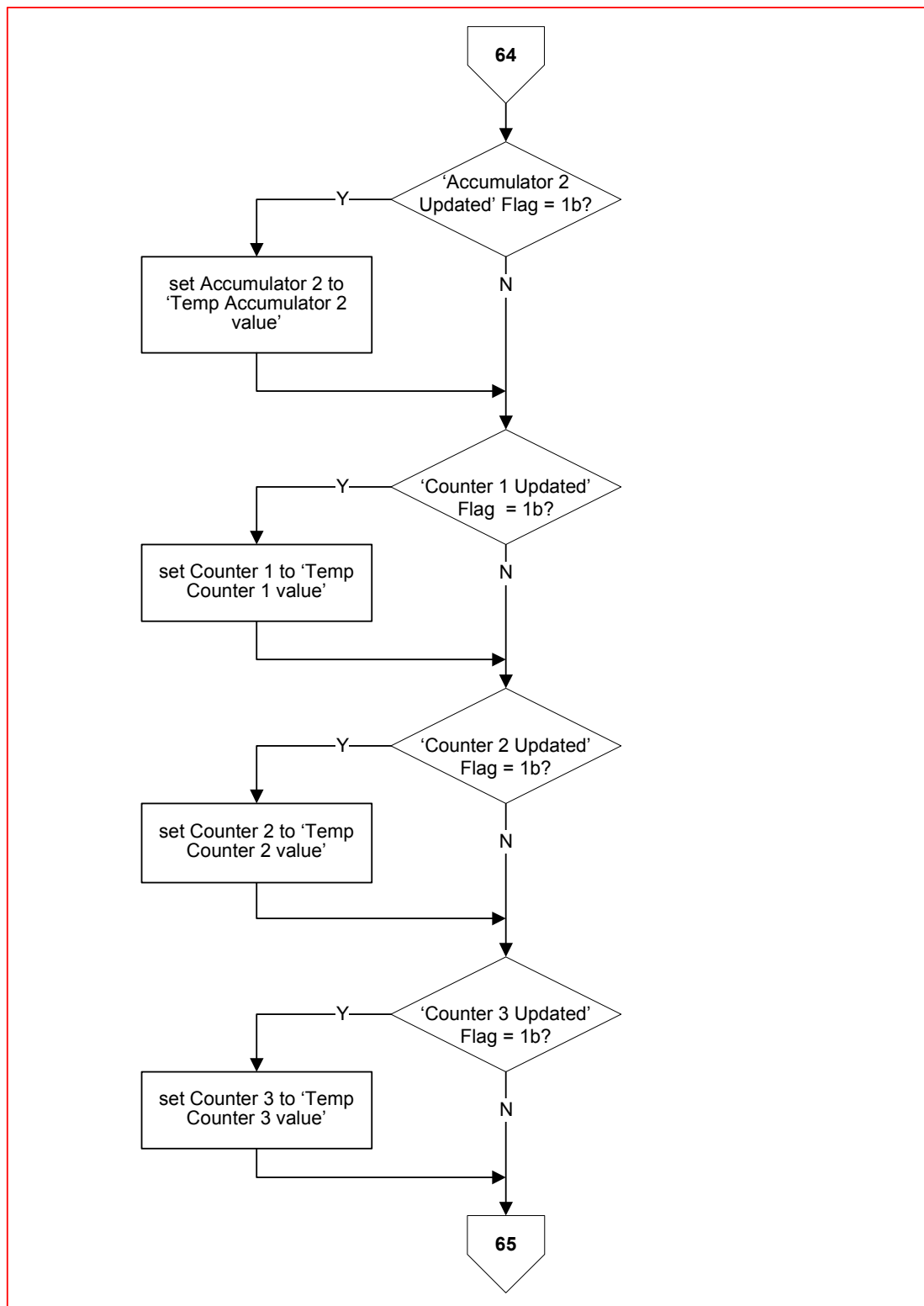
Flow 17-9.2 Unable To Go Online, continued

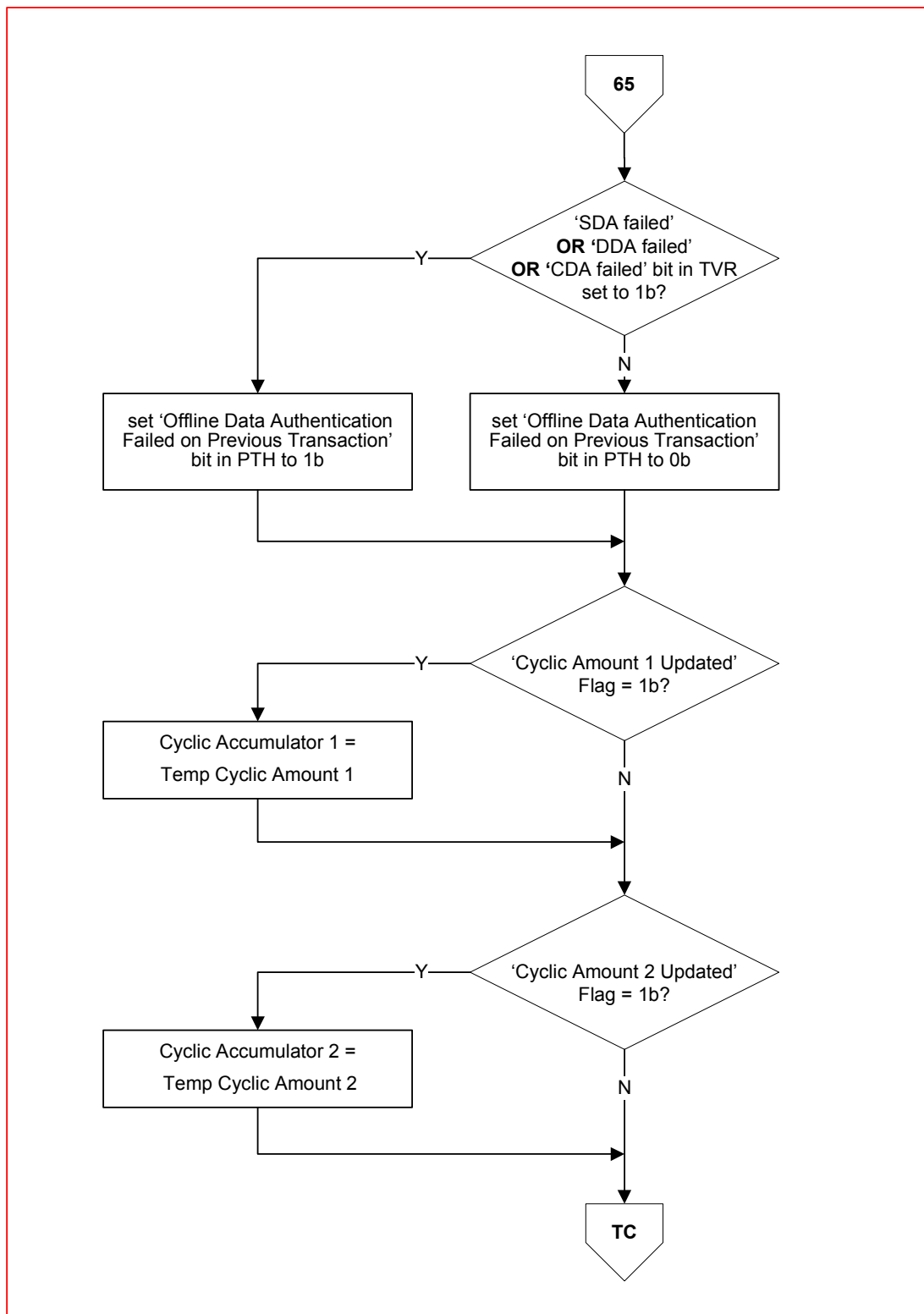


Flow 17-9.3 Unable To Go Online, continued

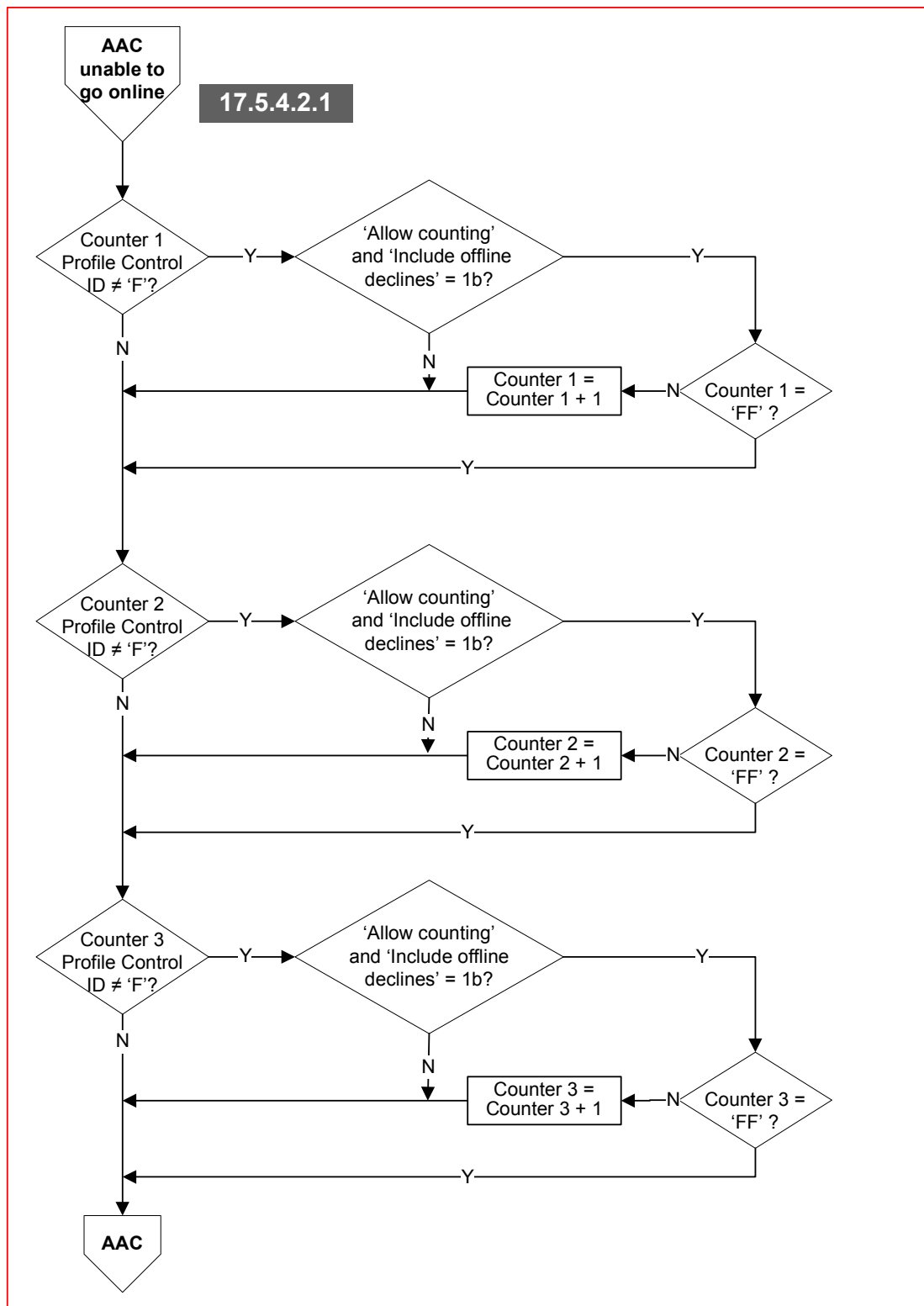


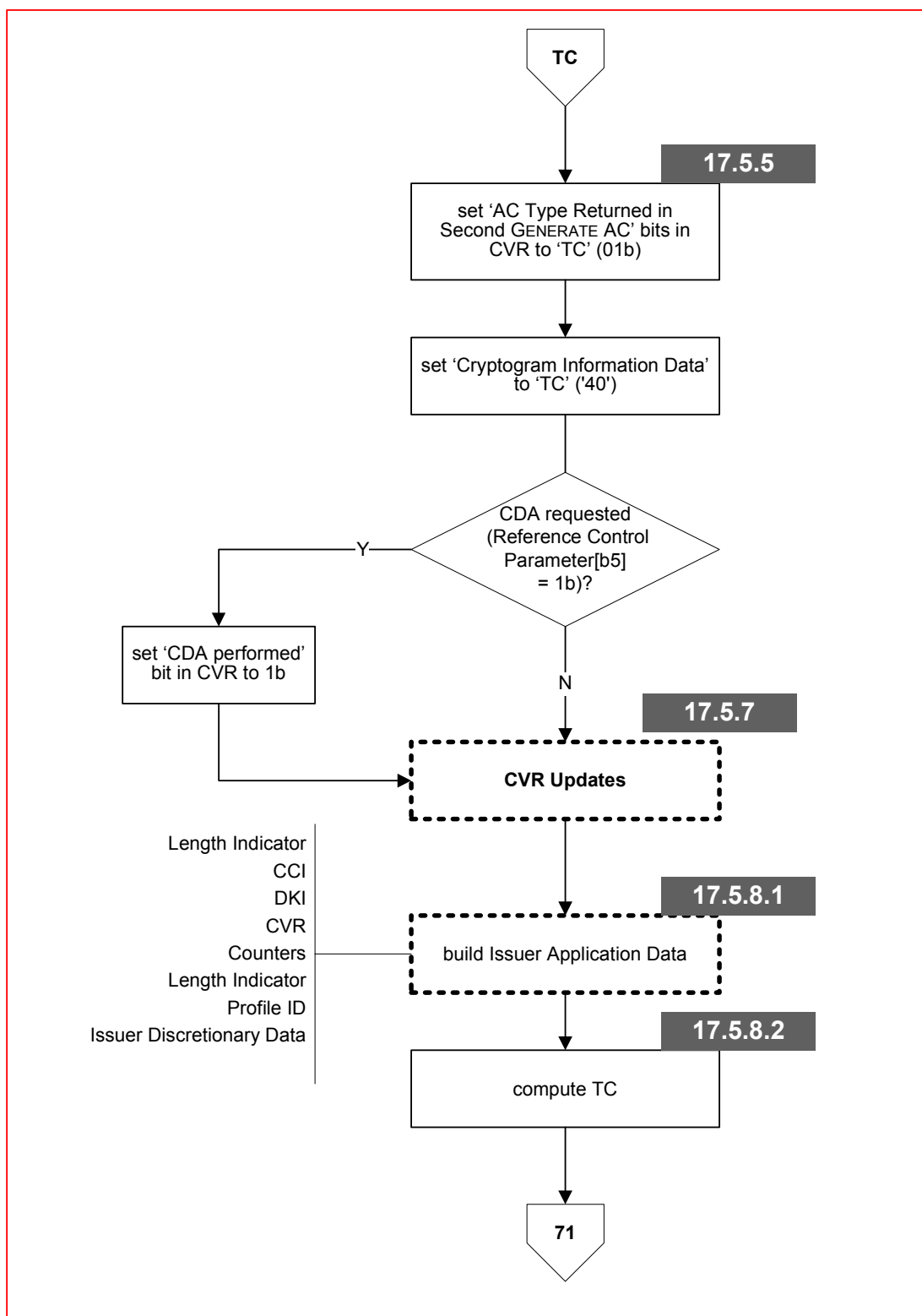
Flow 17-9.4 Unable To Go Online, continued

**Flow 17-9.5 Unable To Go Online, continued**

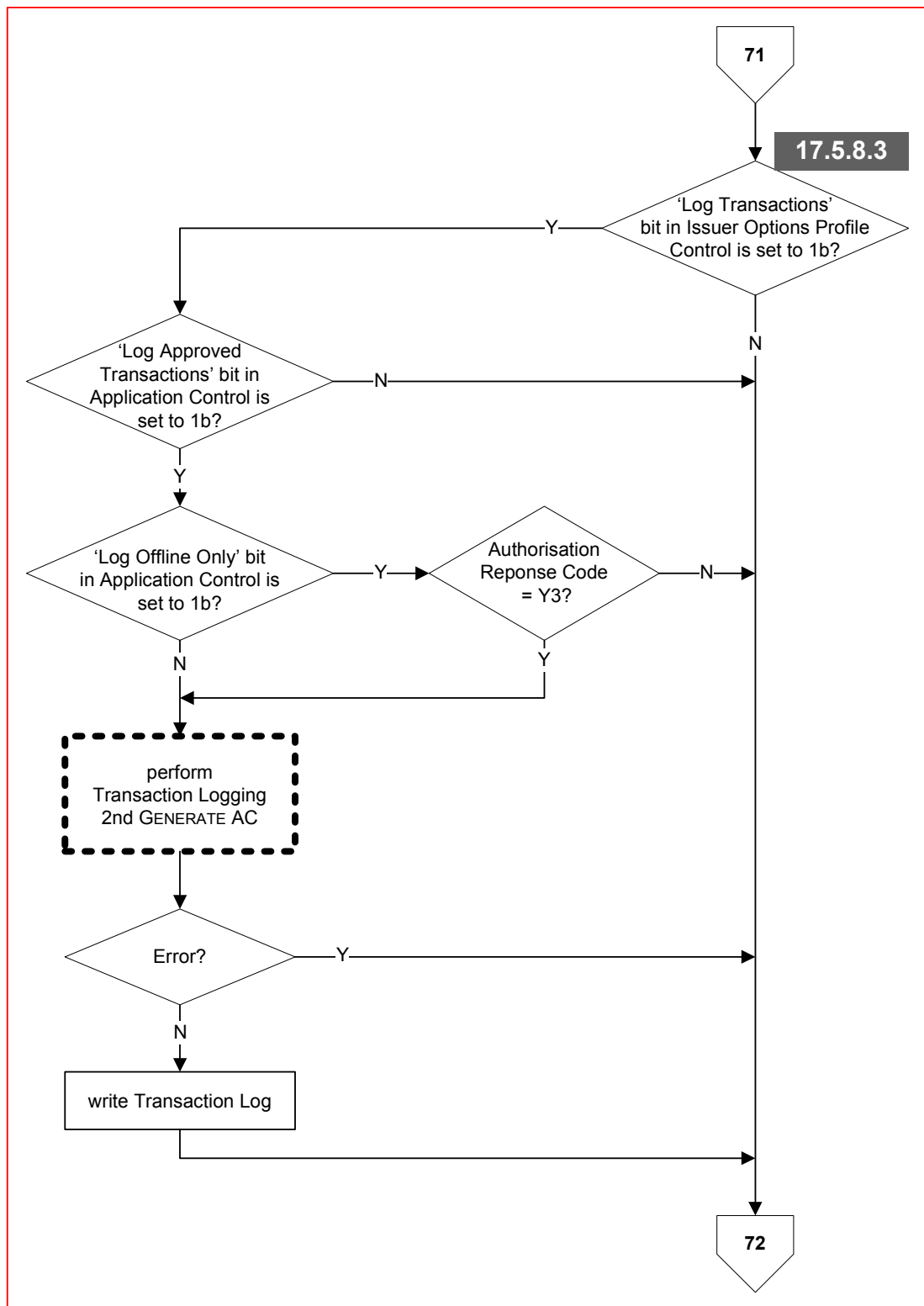


Flow 17-9.6 Unable To Go Online, continued

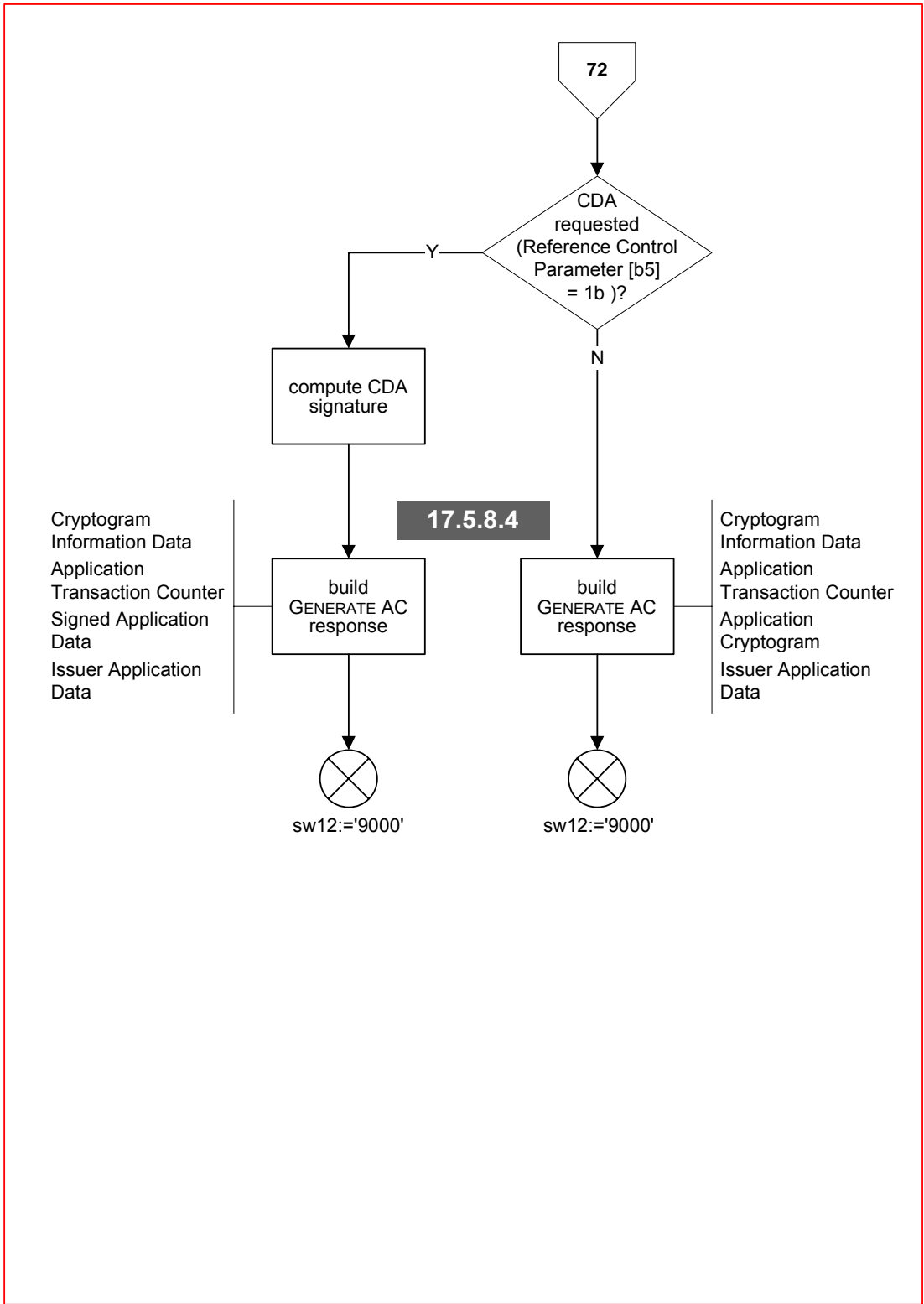
**Flow 17-10 AAC Unable To Go Online**



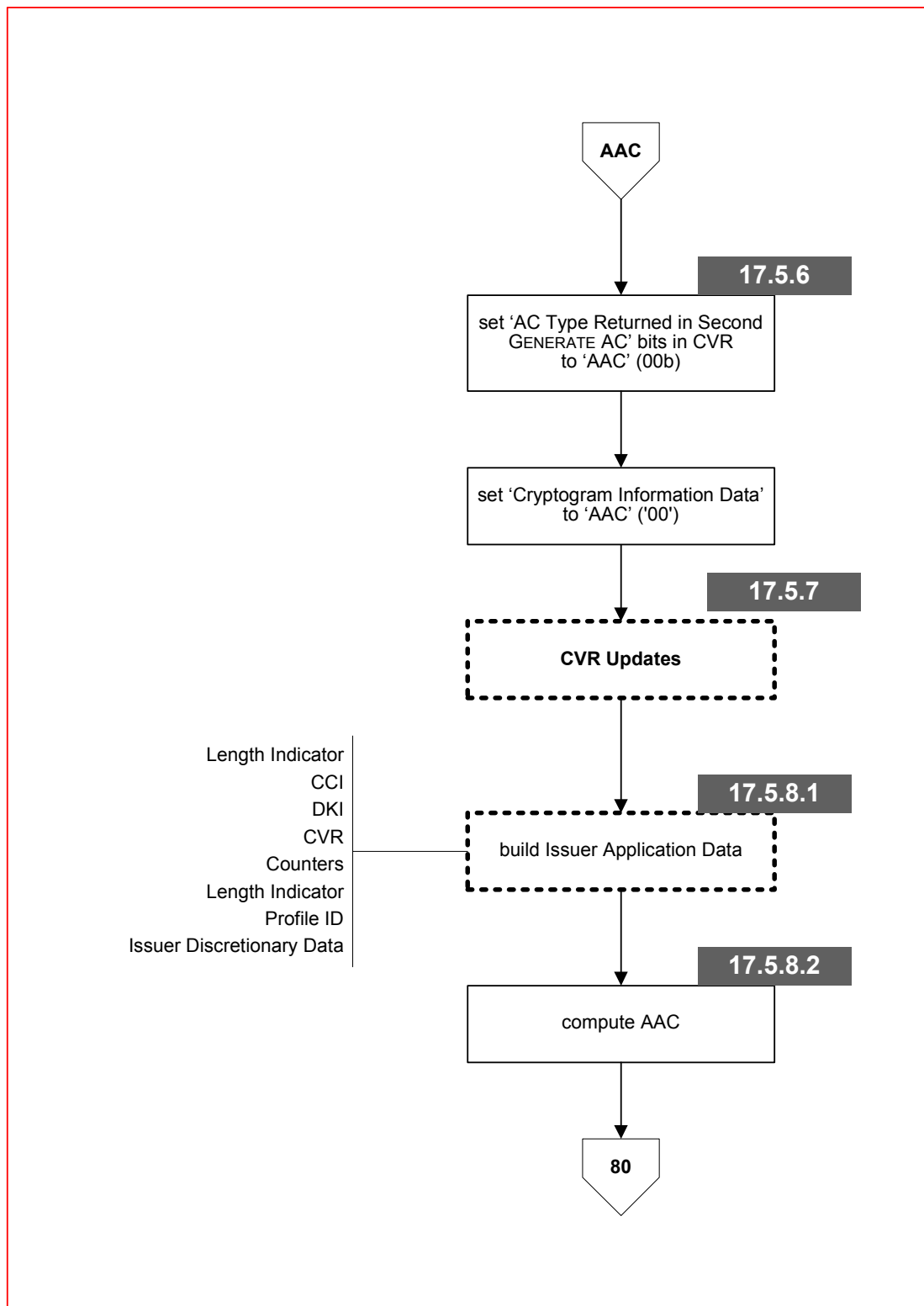
Flow 17-11 TC



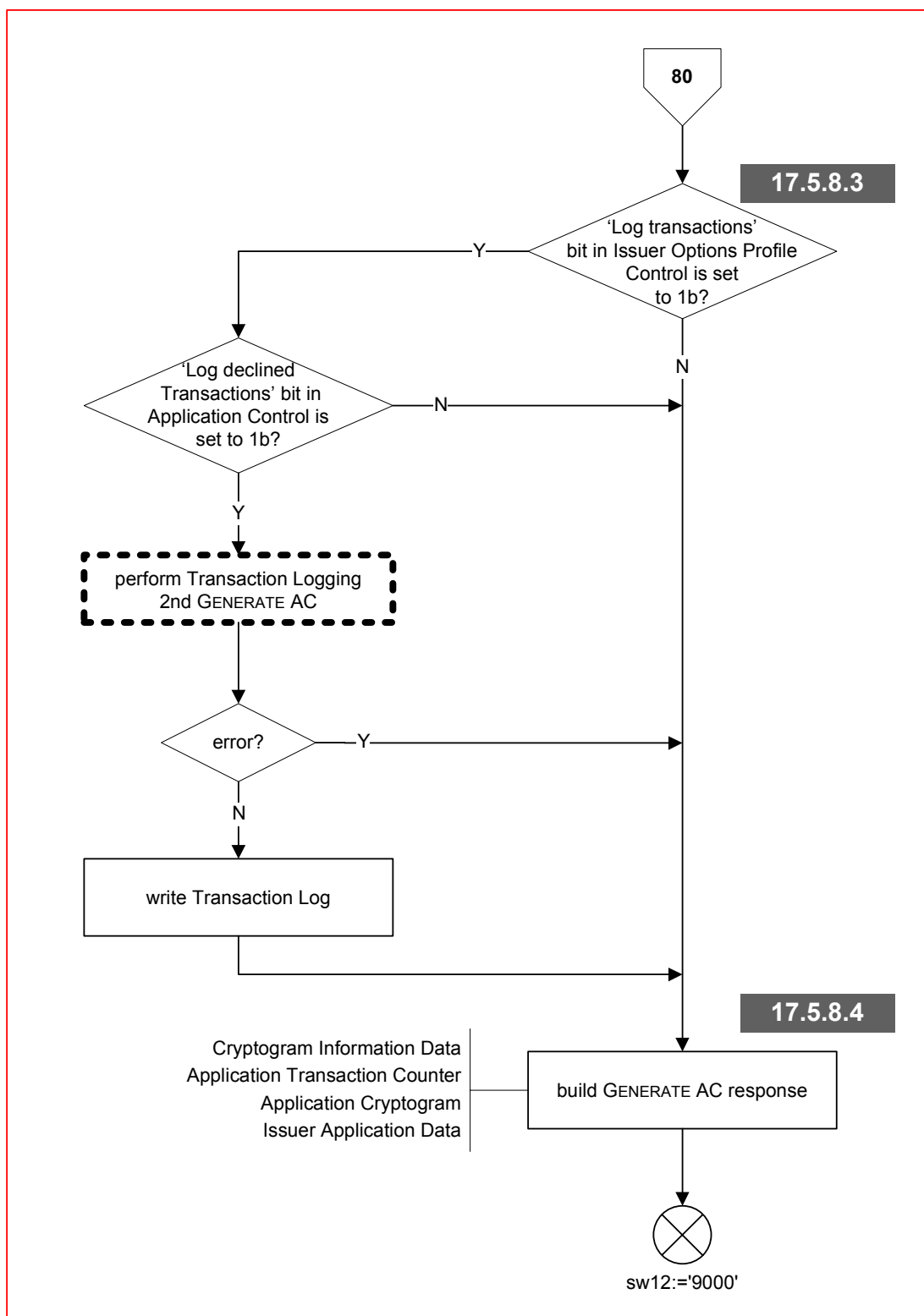
Flow 17-11.1 TC, continued



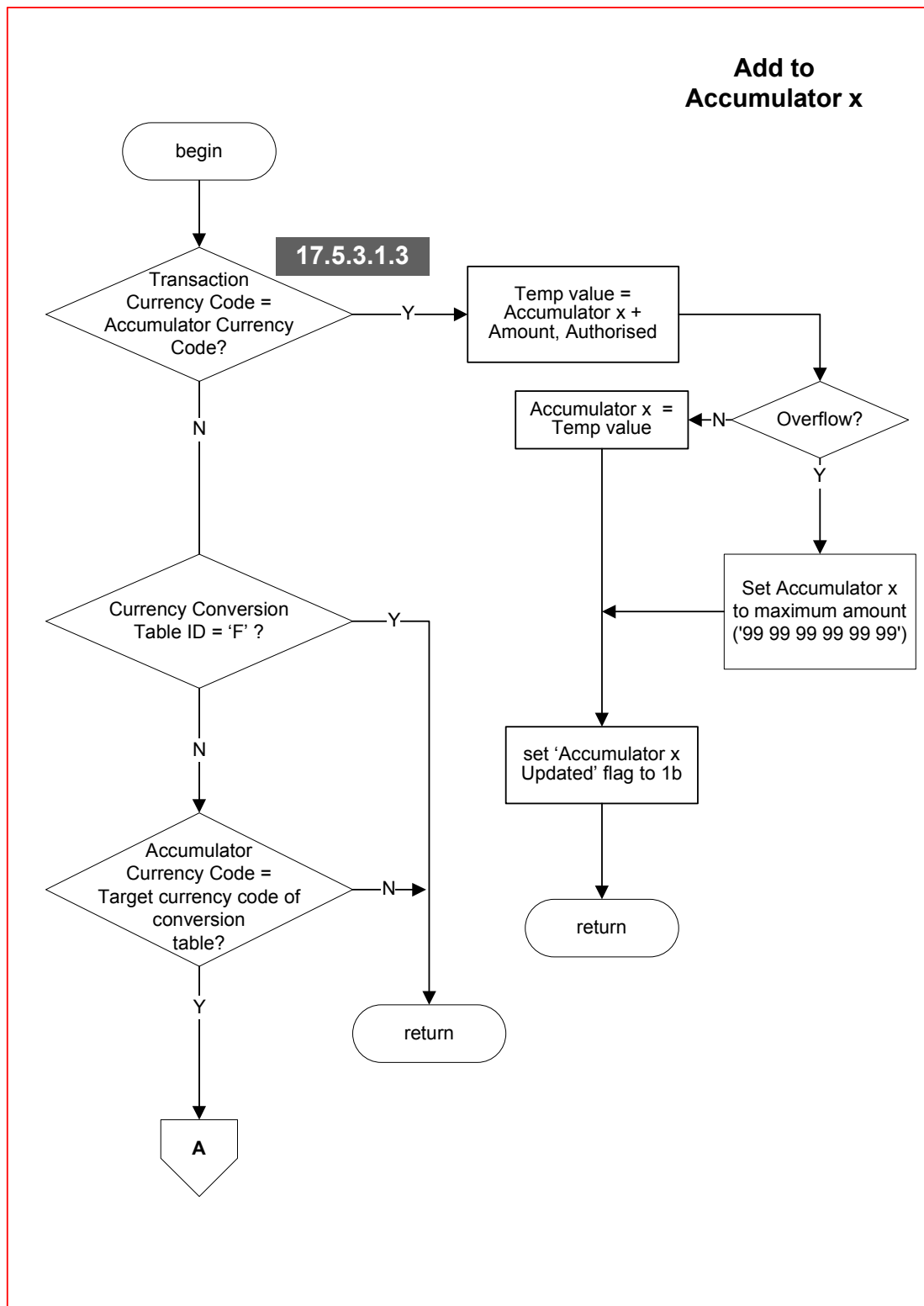
Flow 17-11.2 TC, continued



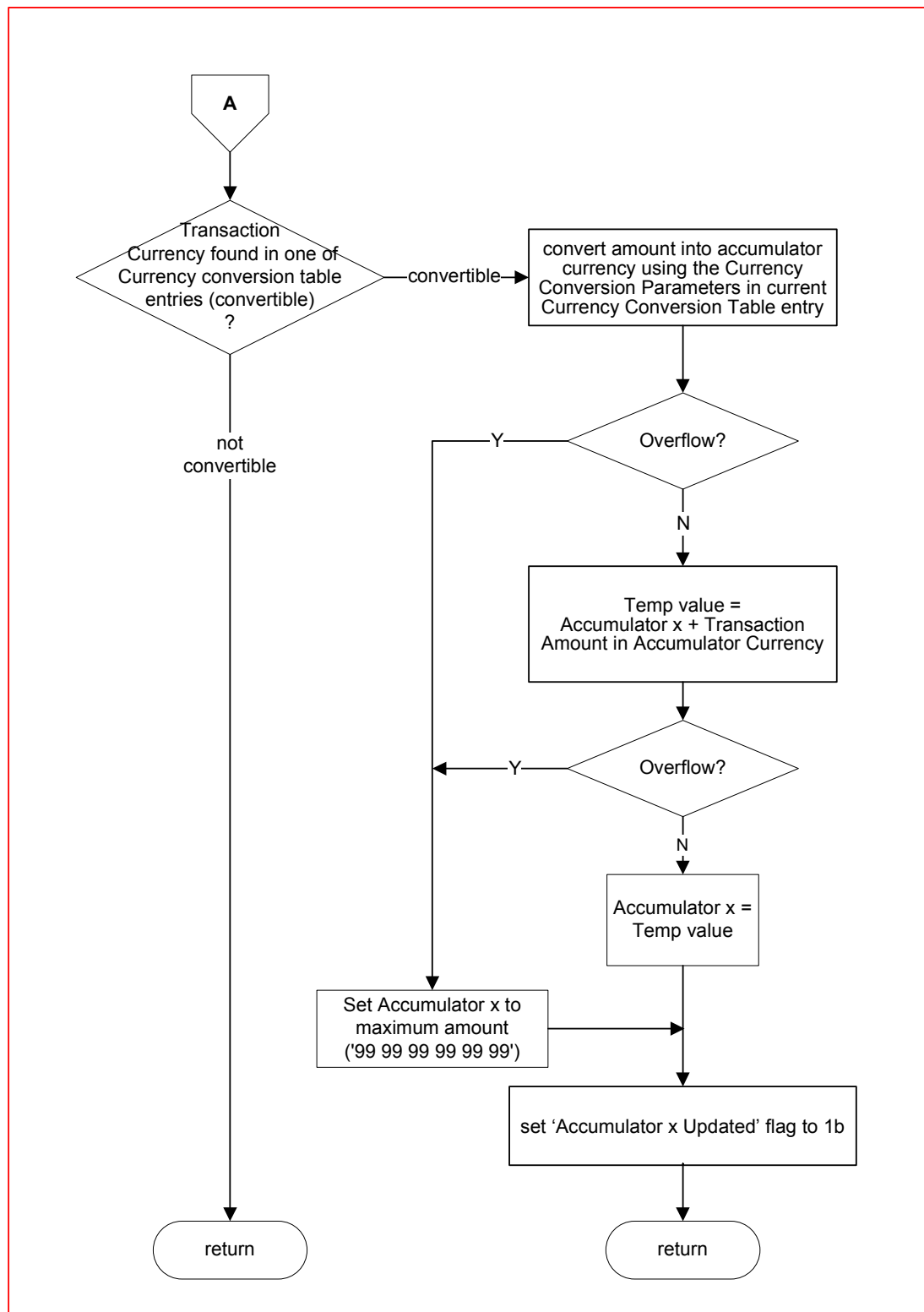
Flow 17-12 AAC



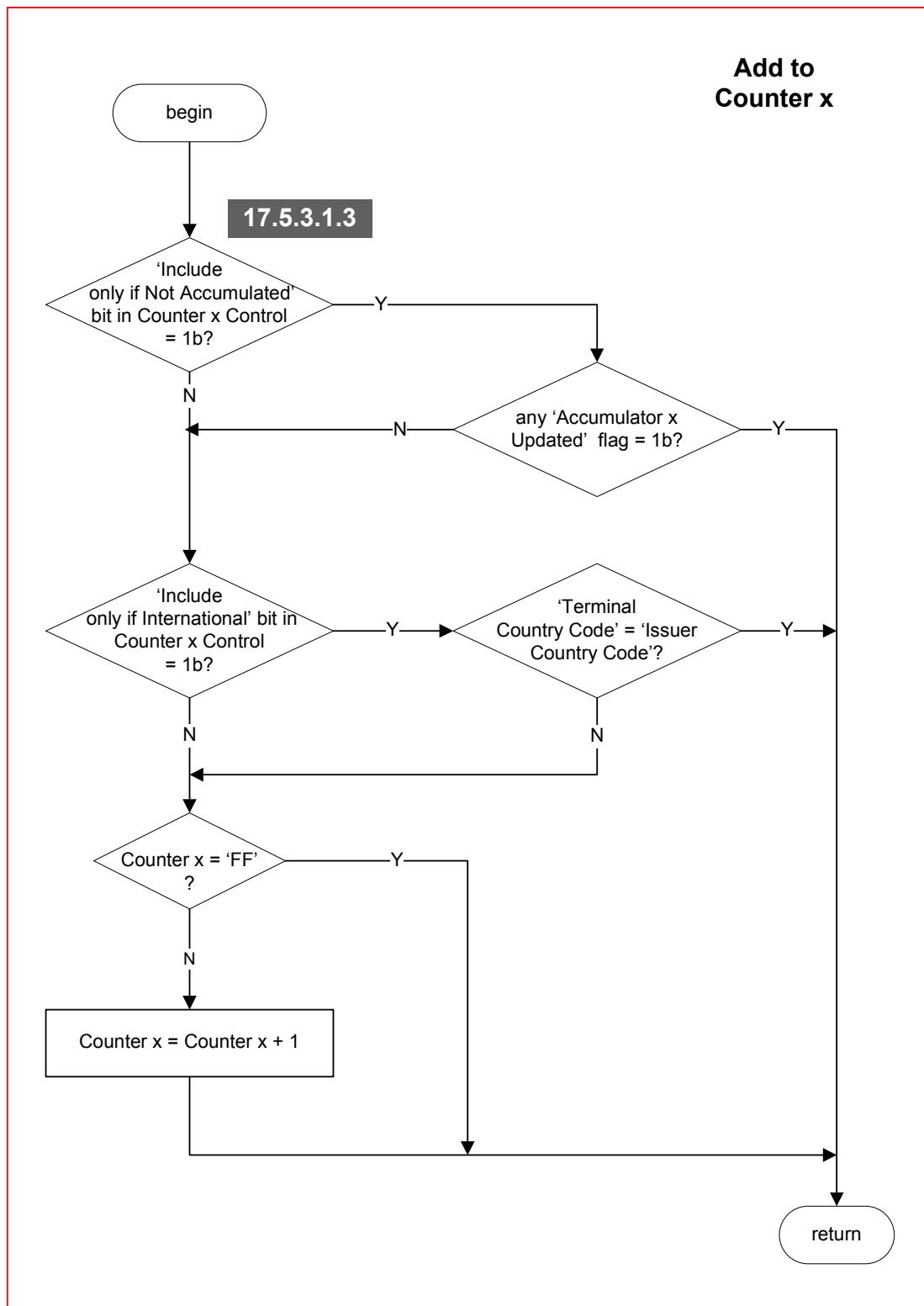
Flow 17-12.1 AAC, continued



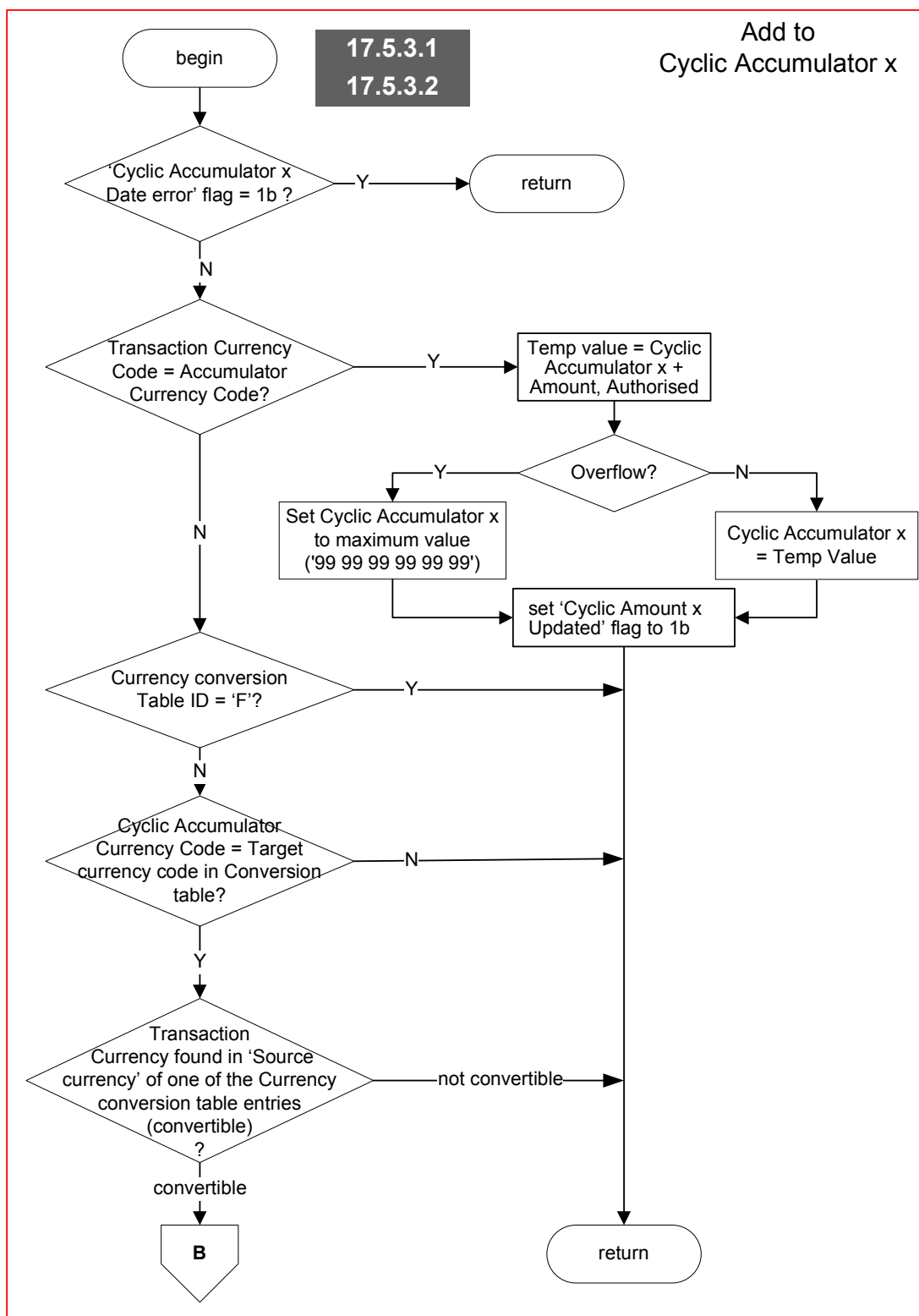
Flow 17-13 Add to Accumulator x



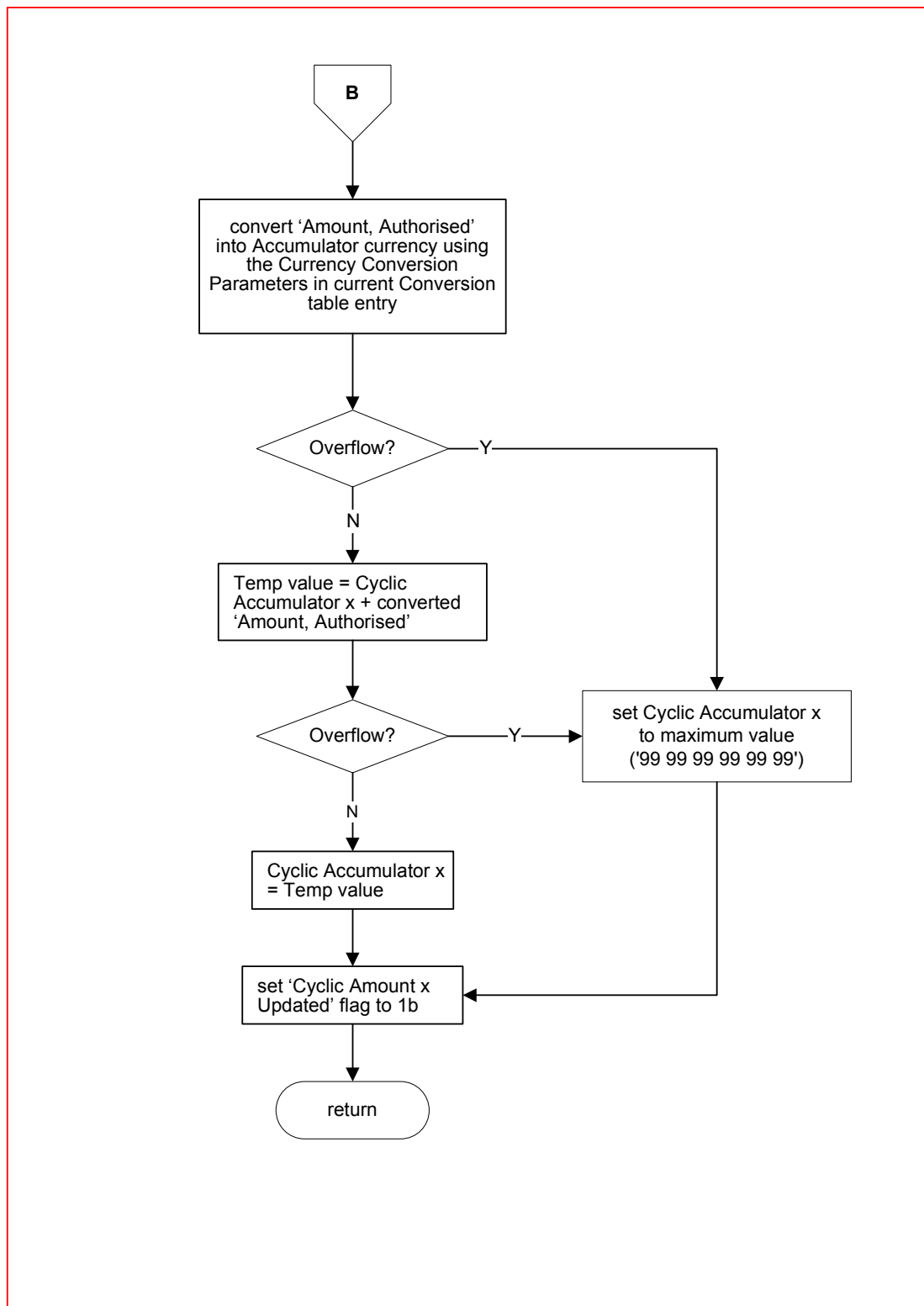
Flow 17-13.1 Add to Accumulator x, continued



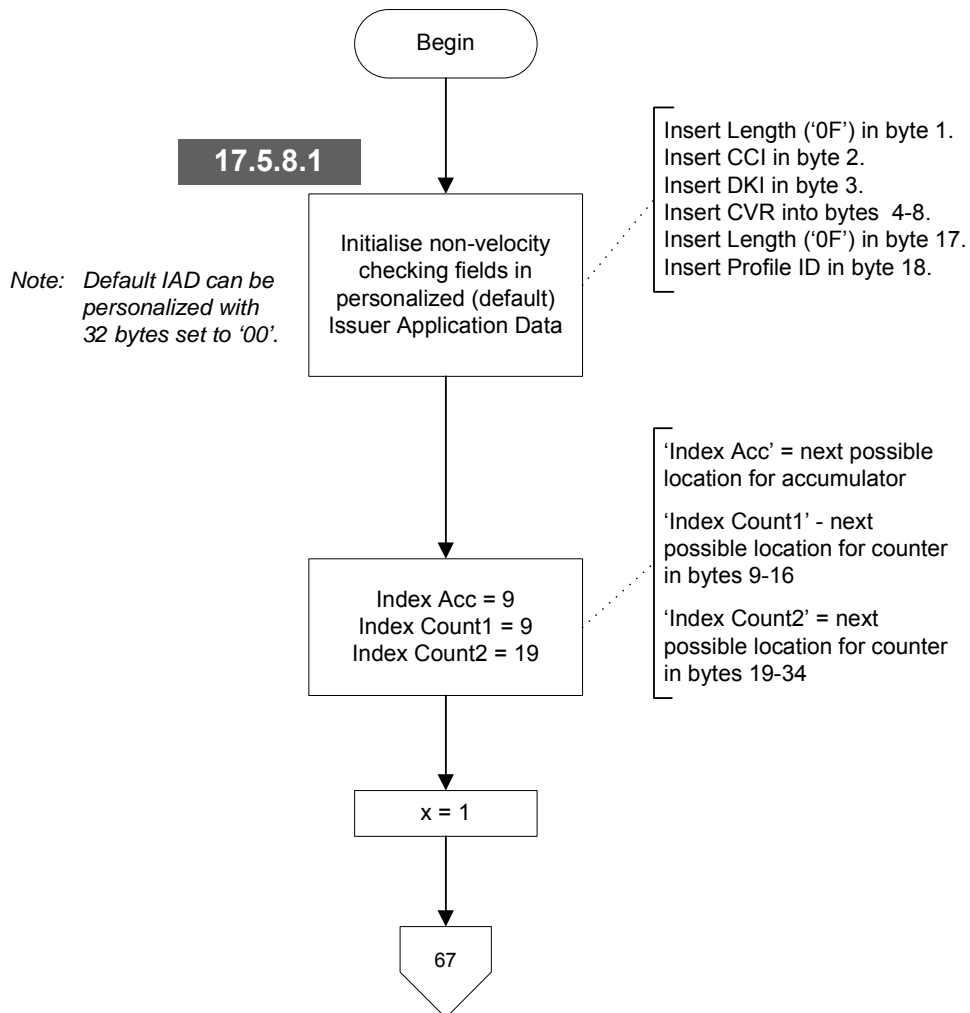
Flow 17-14 Add to Counter x



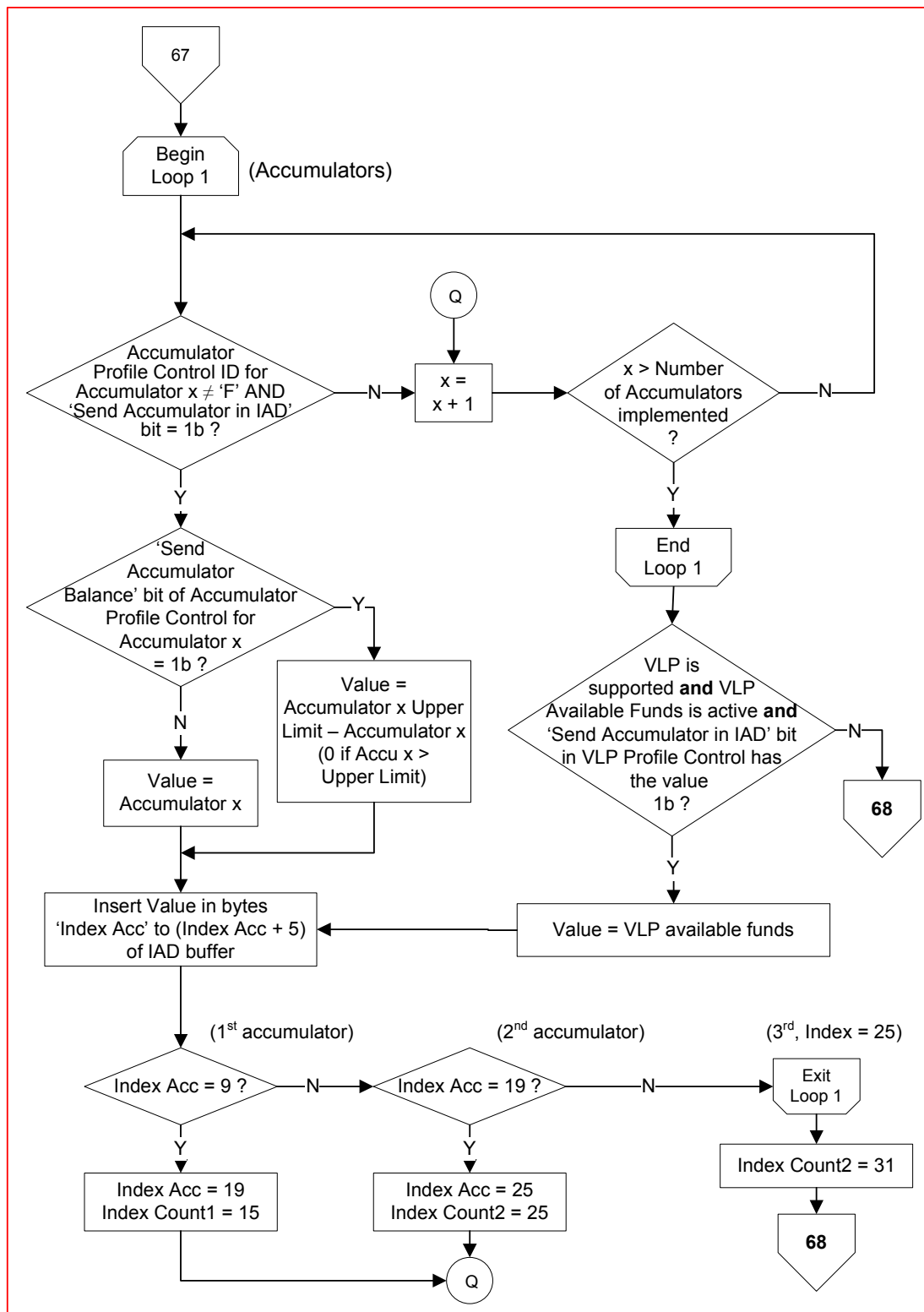
Flow 17-15 Add to Cyclic Accumulator x

**Flow 17-15.1 Add to Cyclic Accumulator x, continued**

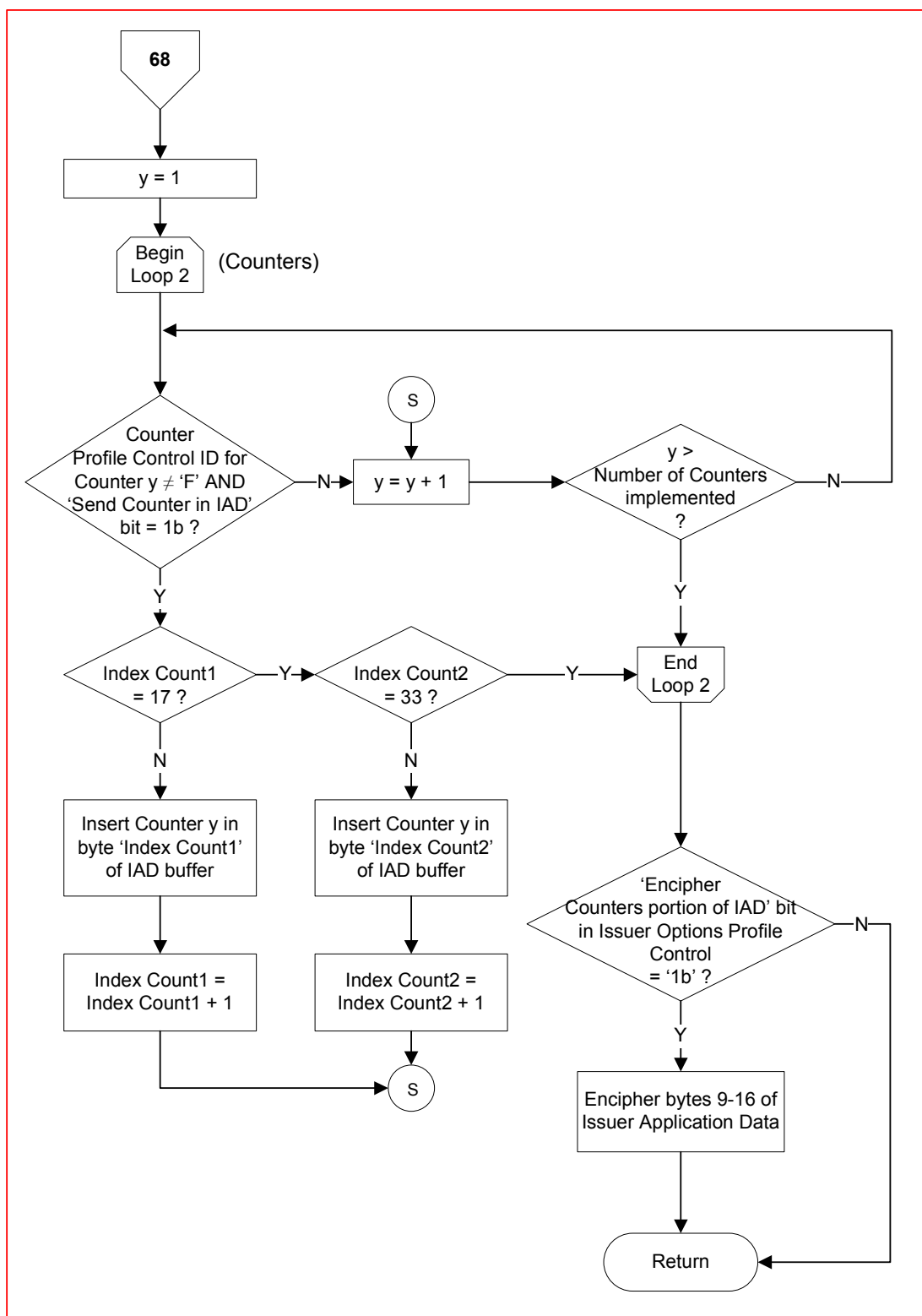
Build Issuer Application Data



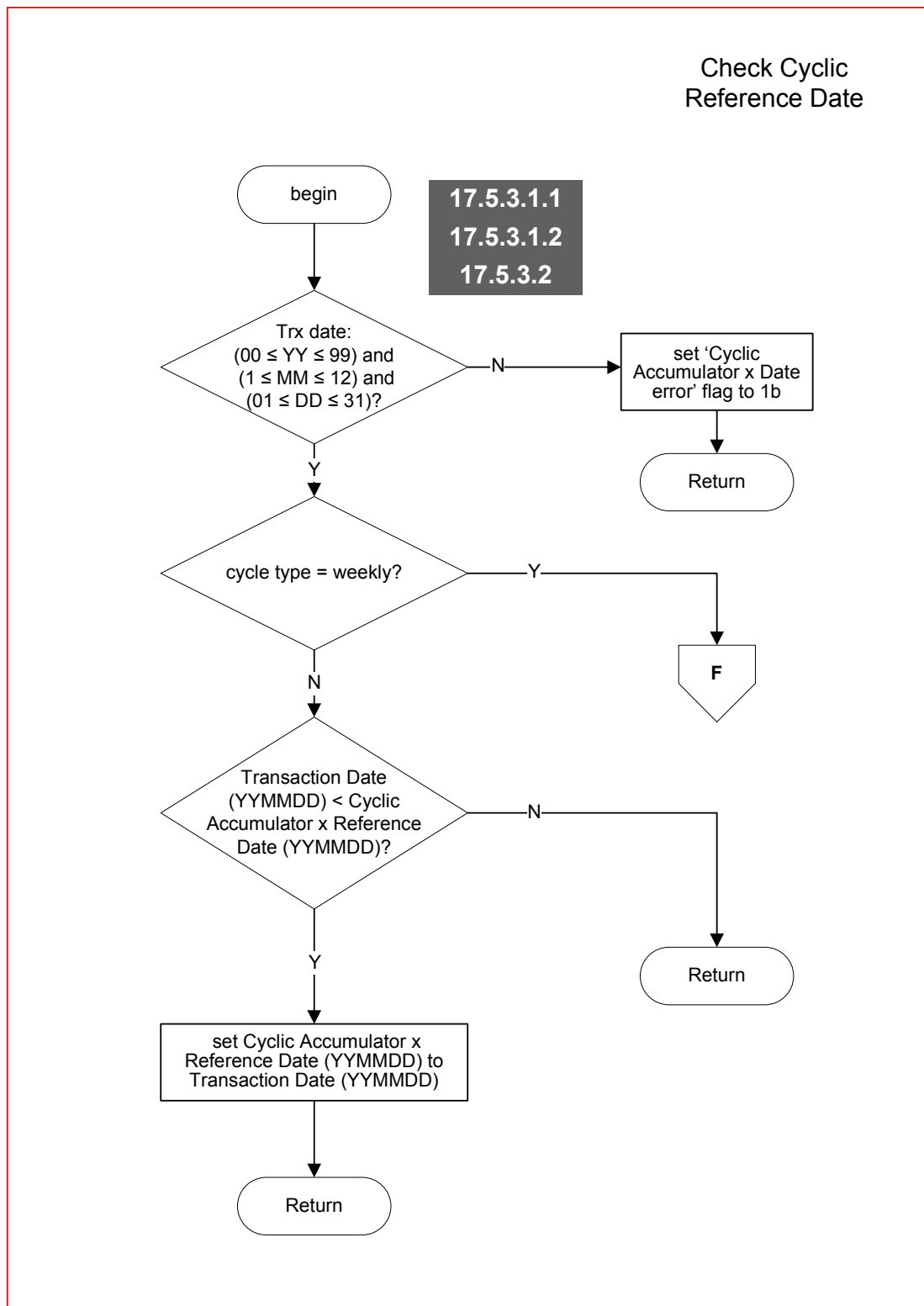
Flow 17-16 Build Issuer Application Data

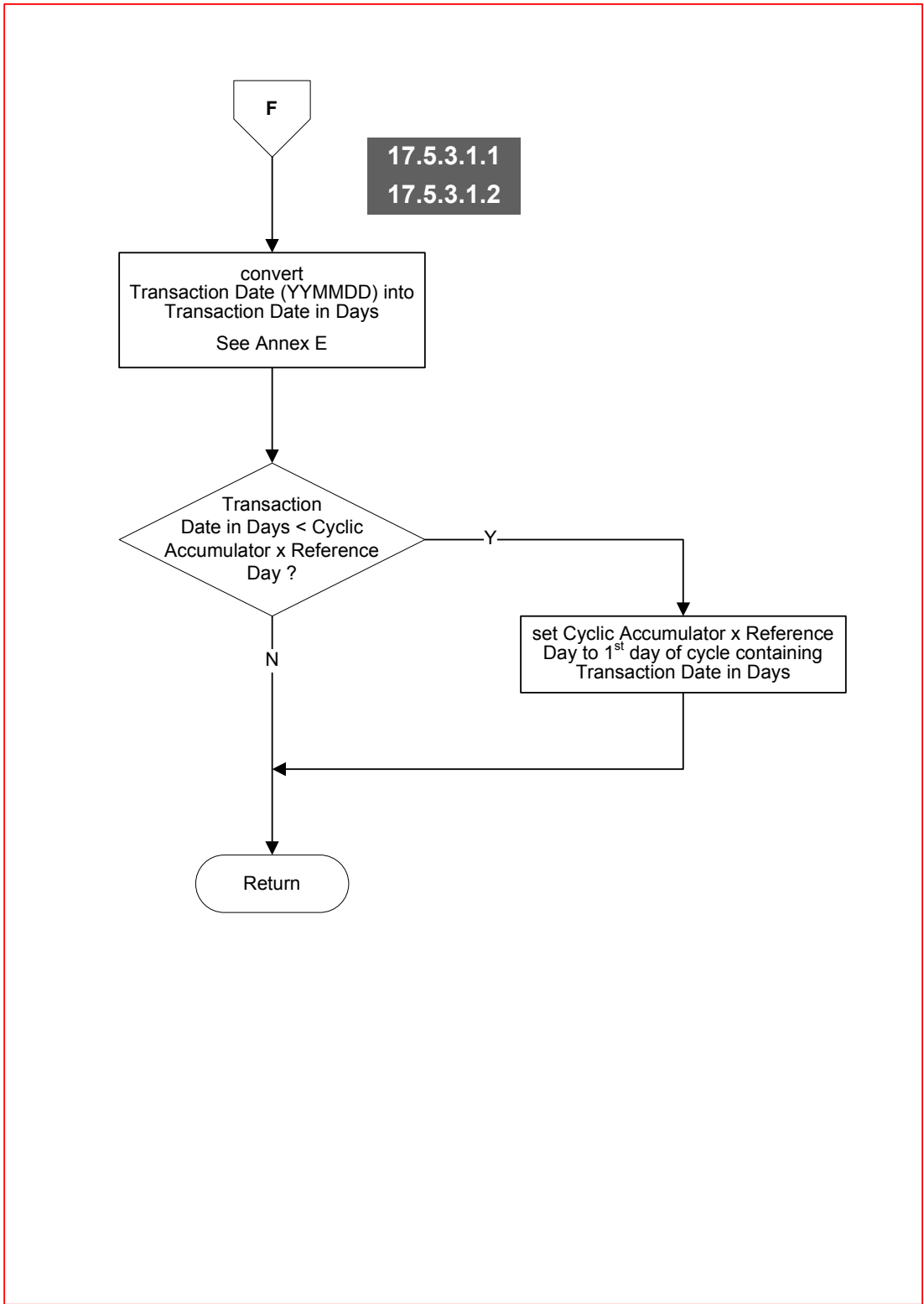


Flow 17-16.1 Build Issuer Application Data, continued

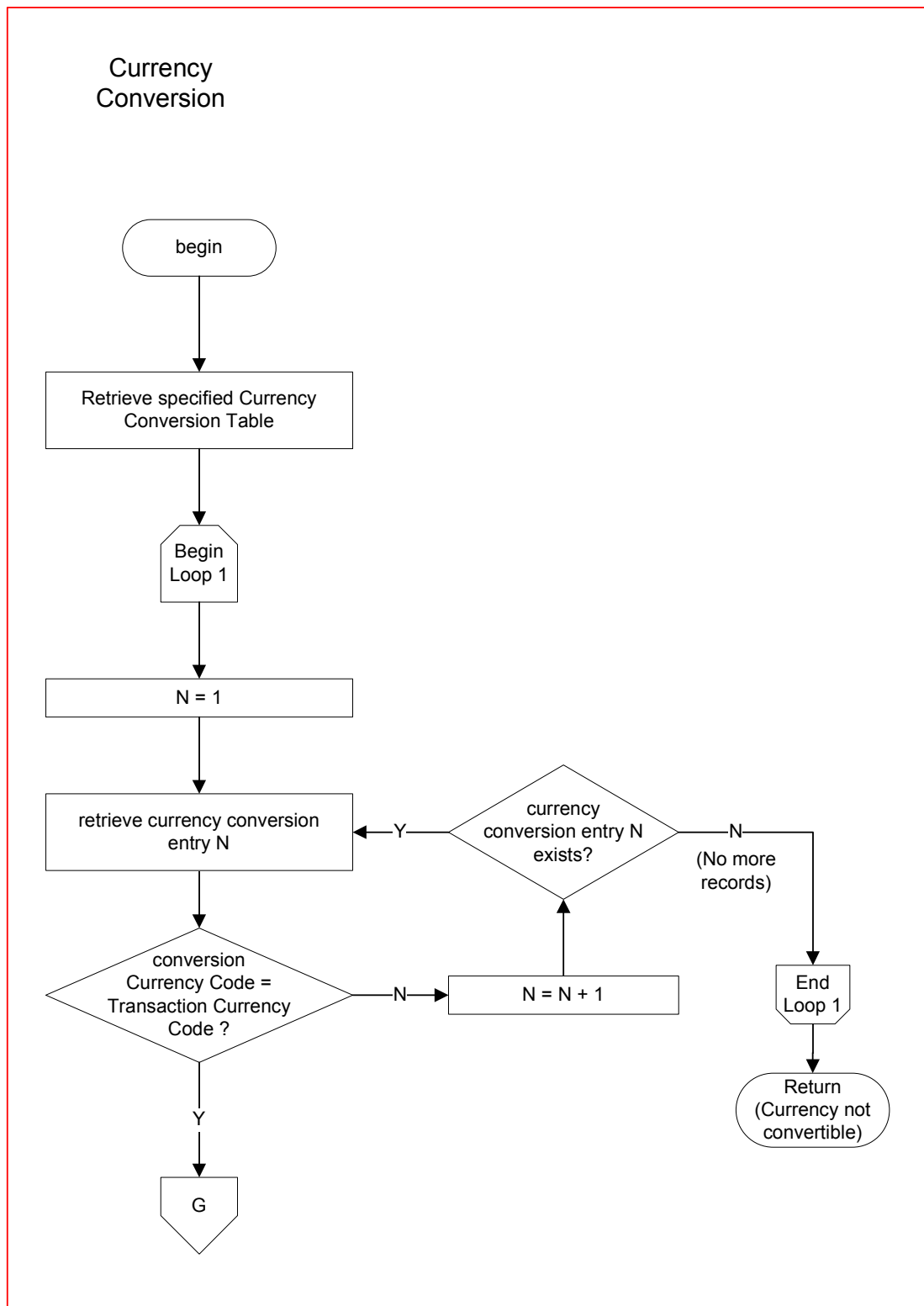


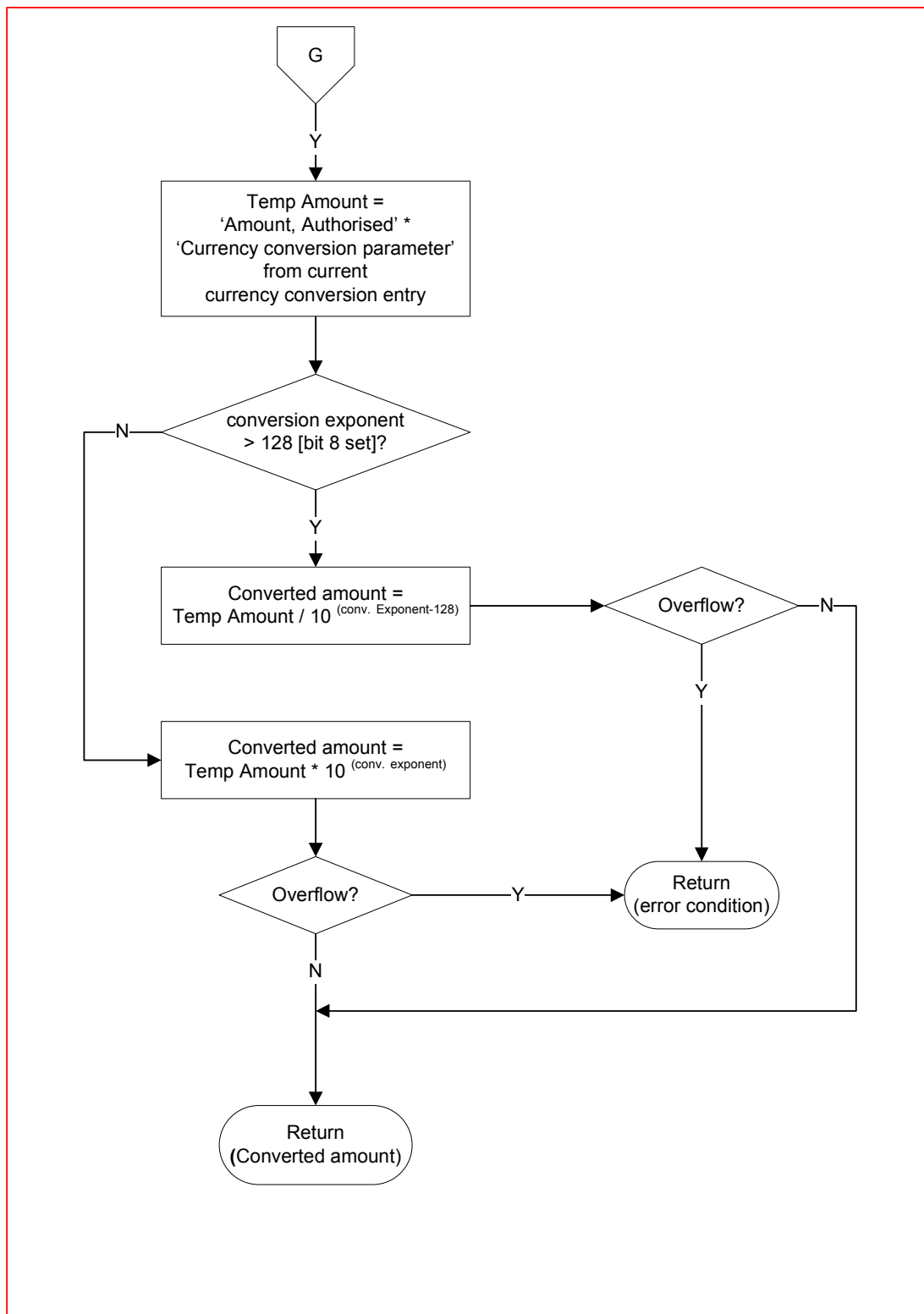
Flow 17-16.2 Build Issuer Application Data, continued

**Flow 17-17 Check Cyclic Reference Date**

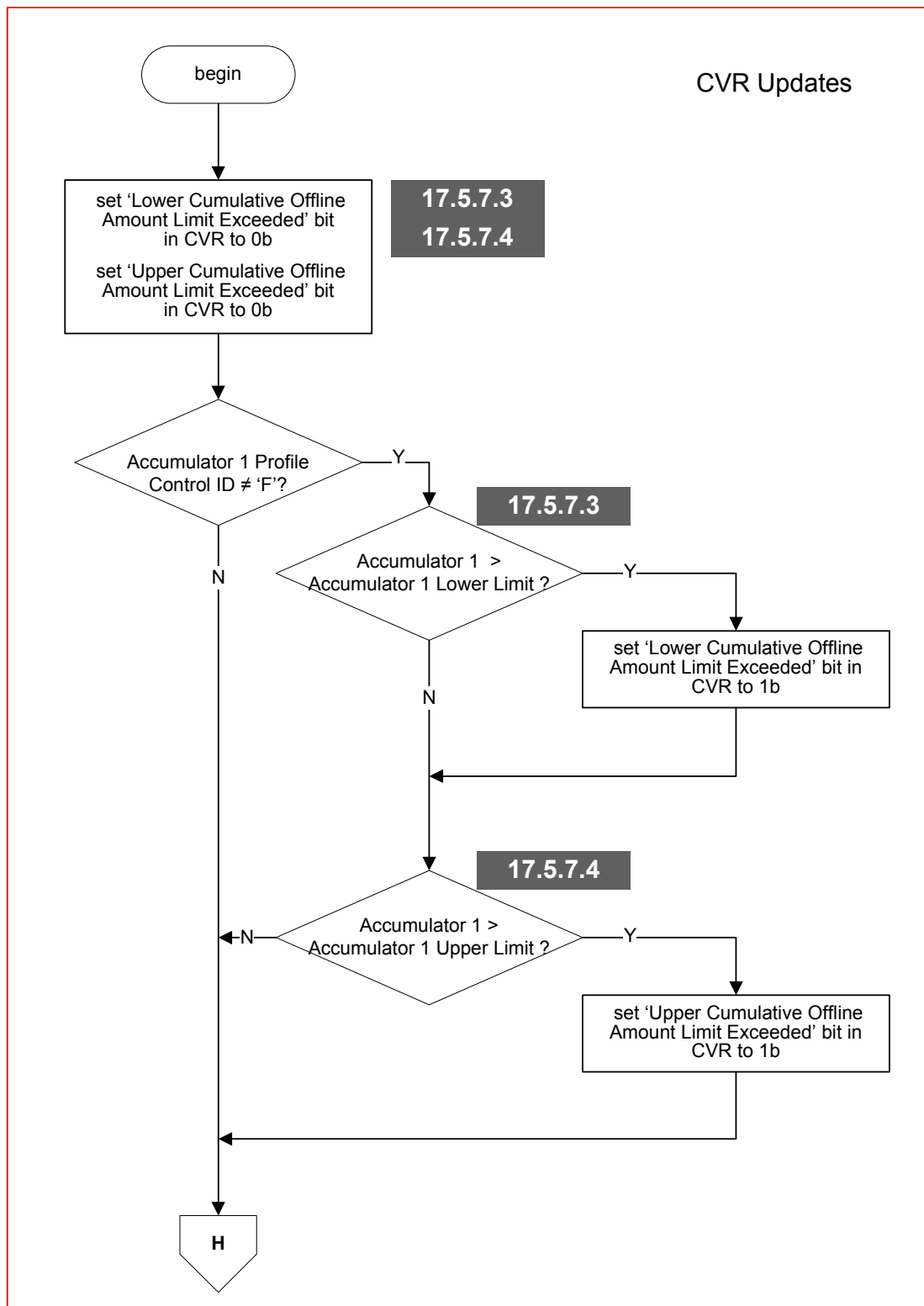


Flow 17-17.1 Check Cyclic Reference Date, continued

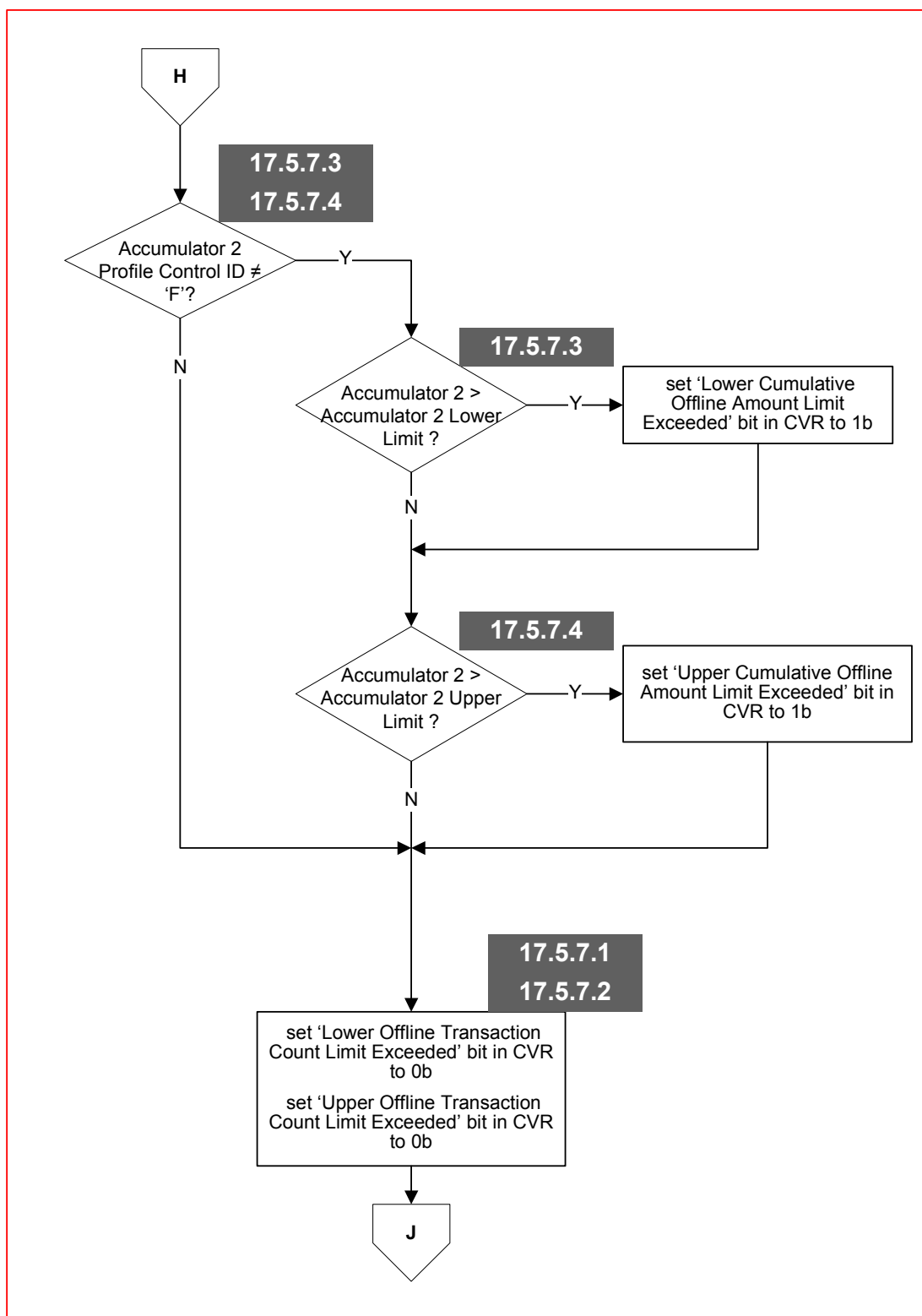
**Flow 17-18 Currency Conversion**



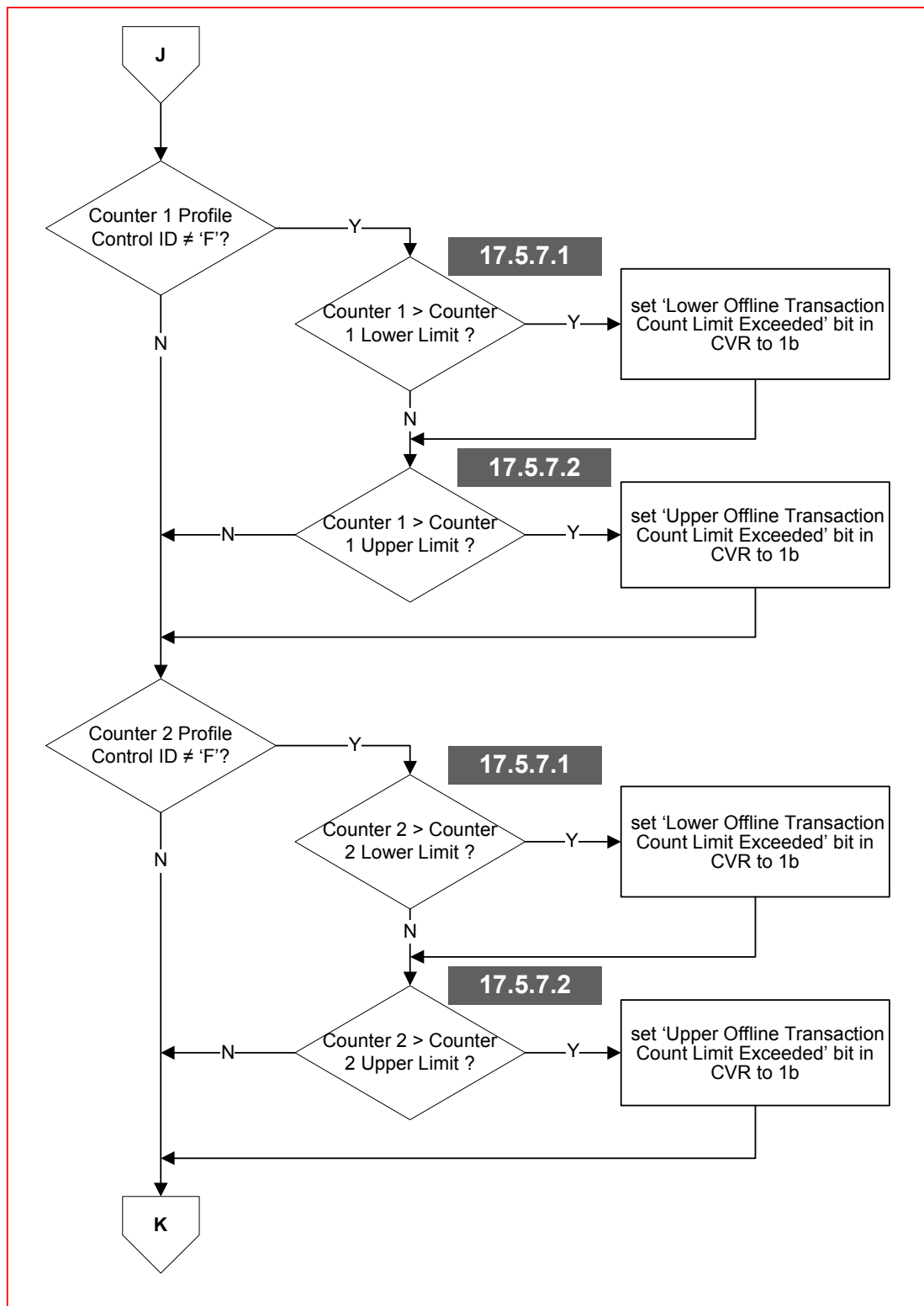
Flow 17-18.1 Currency Conversion, continued



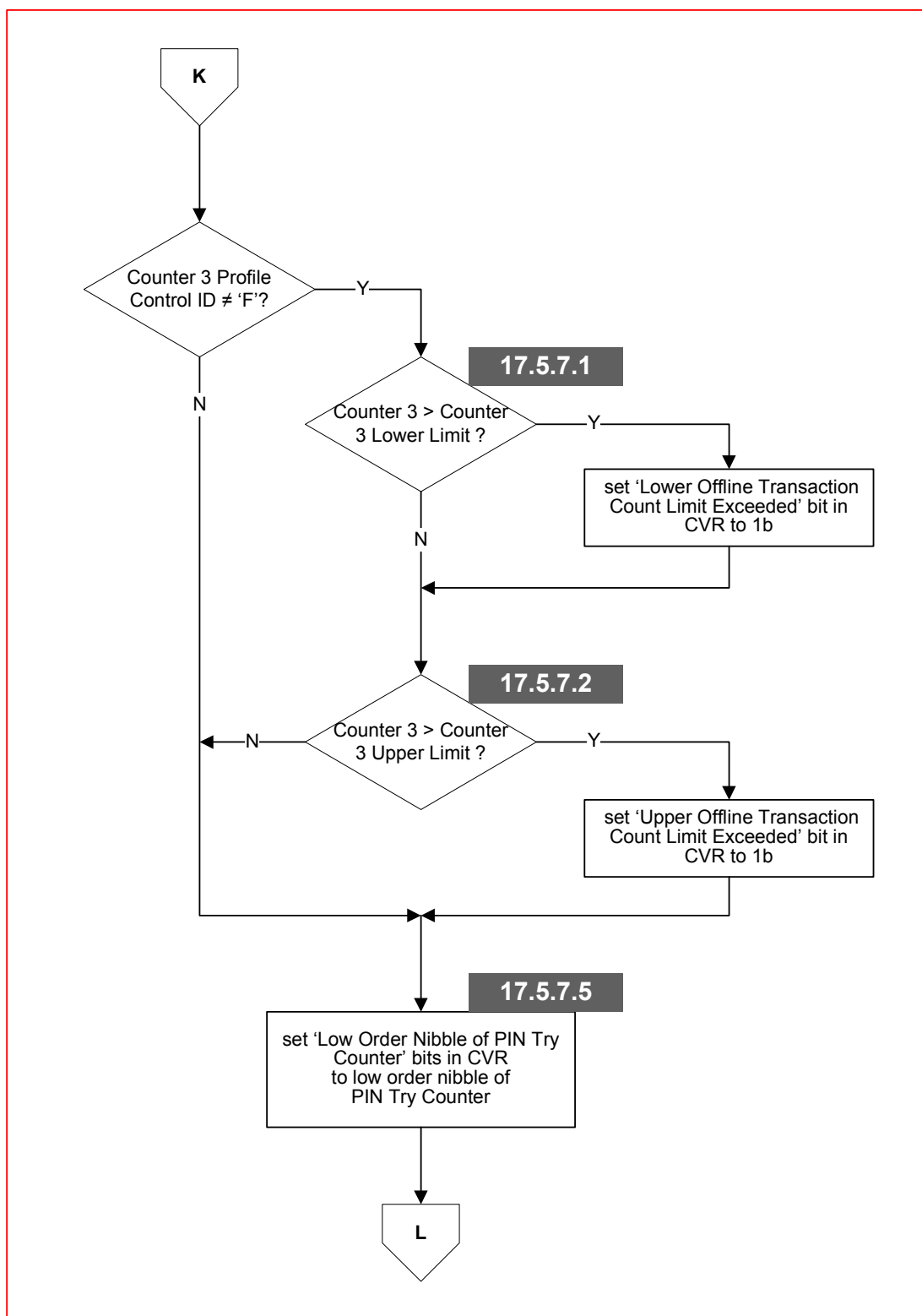
Flow 17-19 CVR Updates



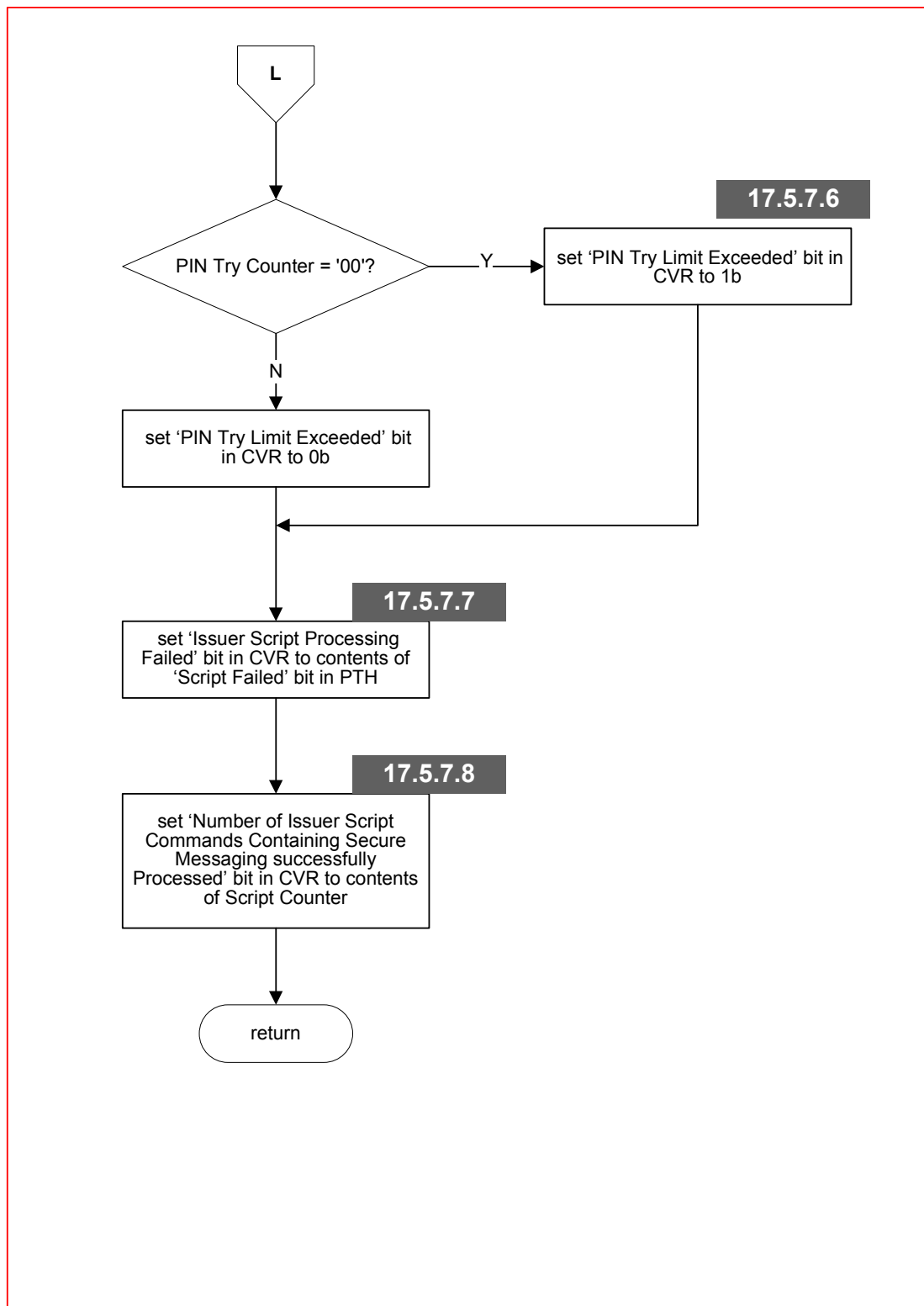
Flow 17-19.1 CVR Updates, continued

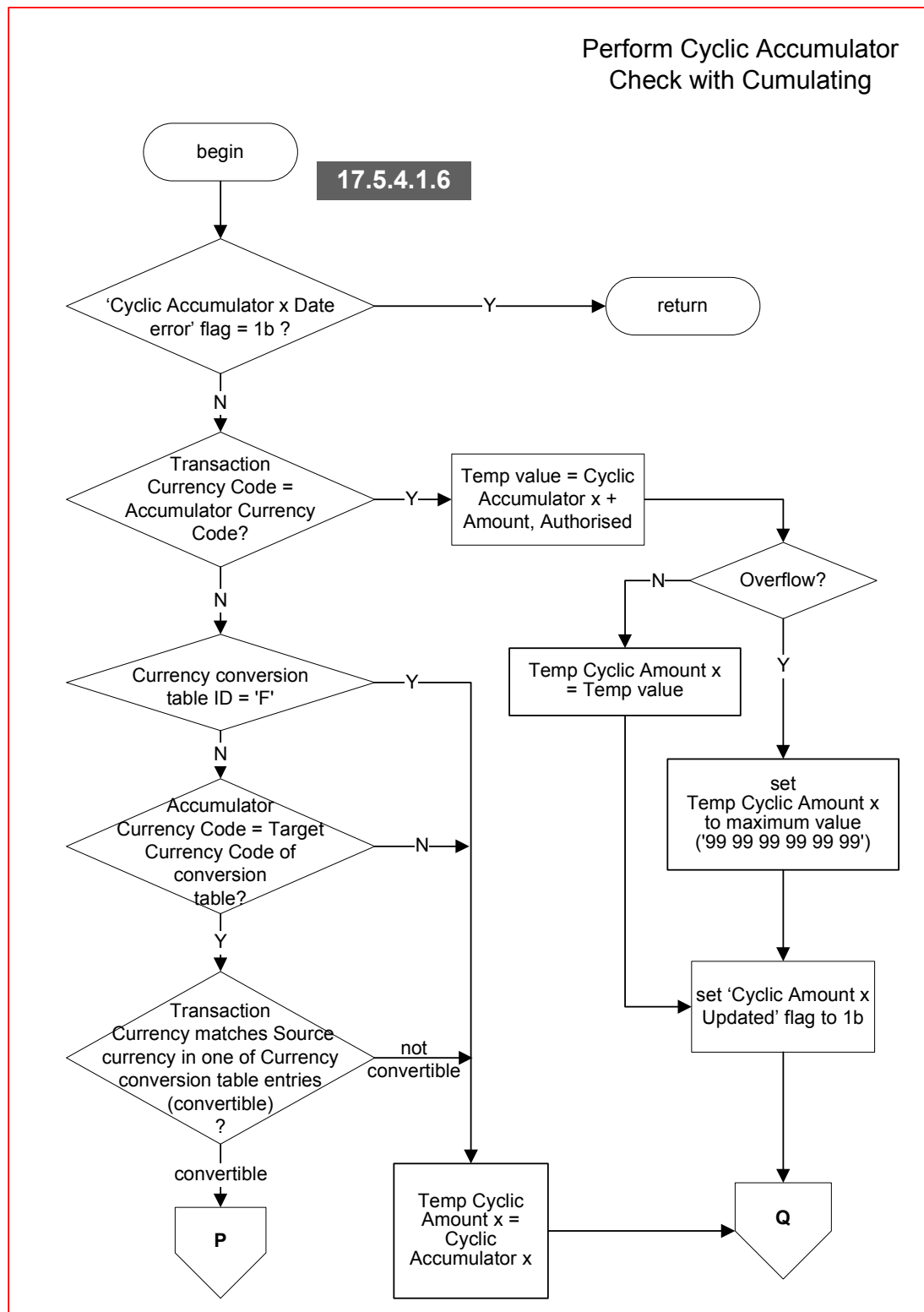


Flow 17-19.2 CVR Updates, continued

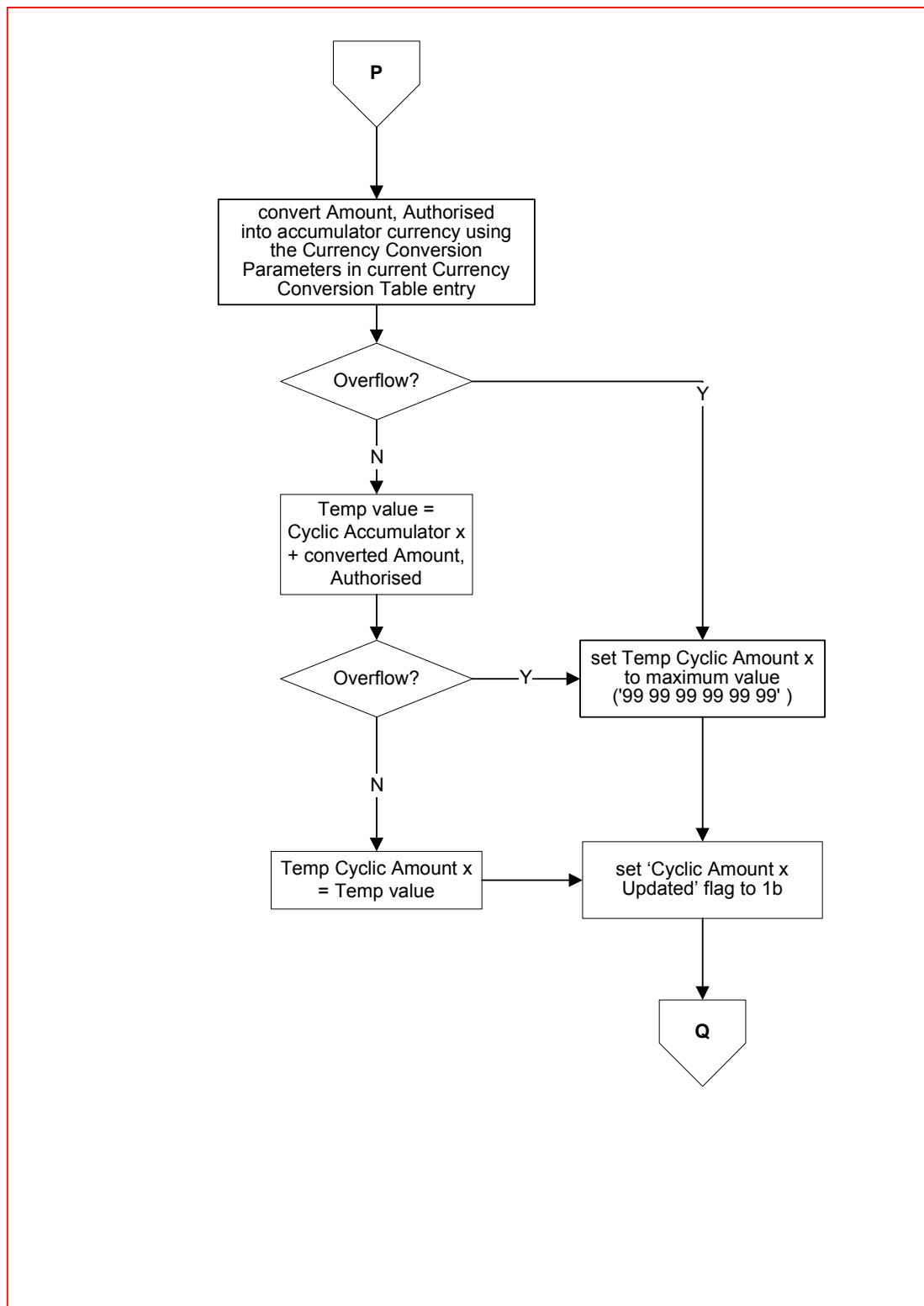


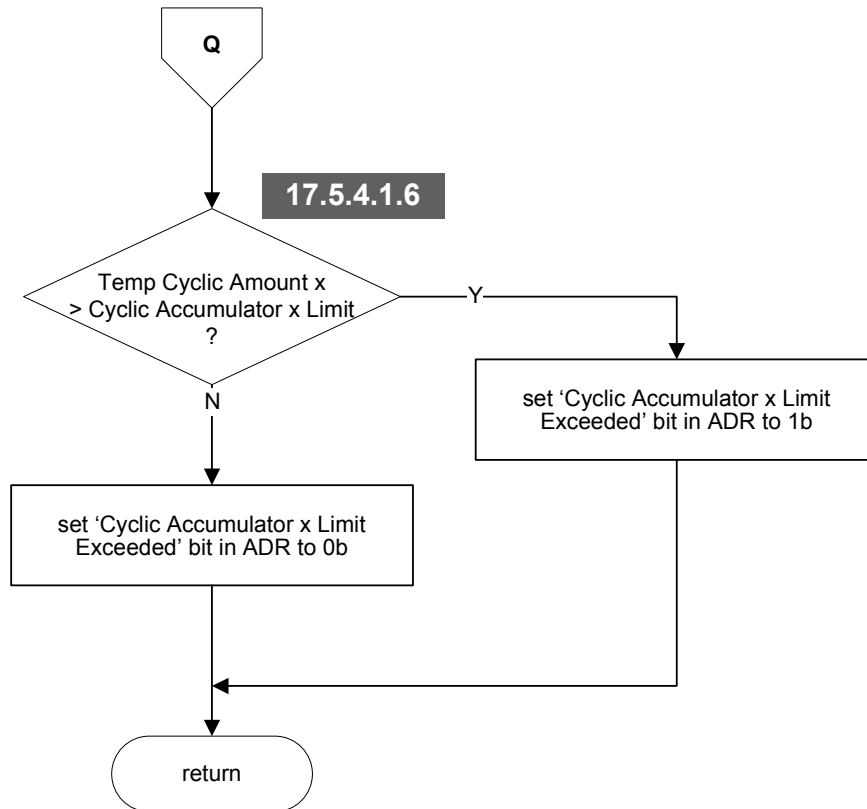
Flow 17-19.3 CVR Updates, continued

**Flow 17-19.4 CVR Updates, continued**

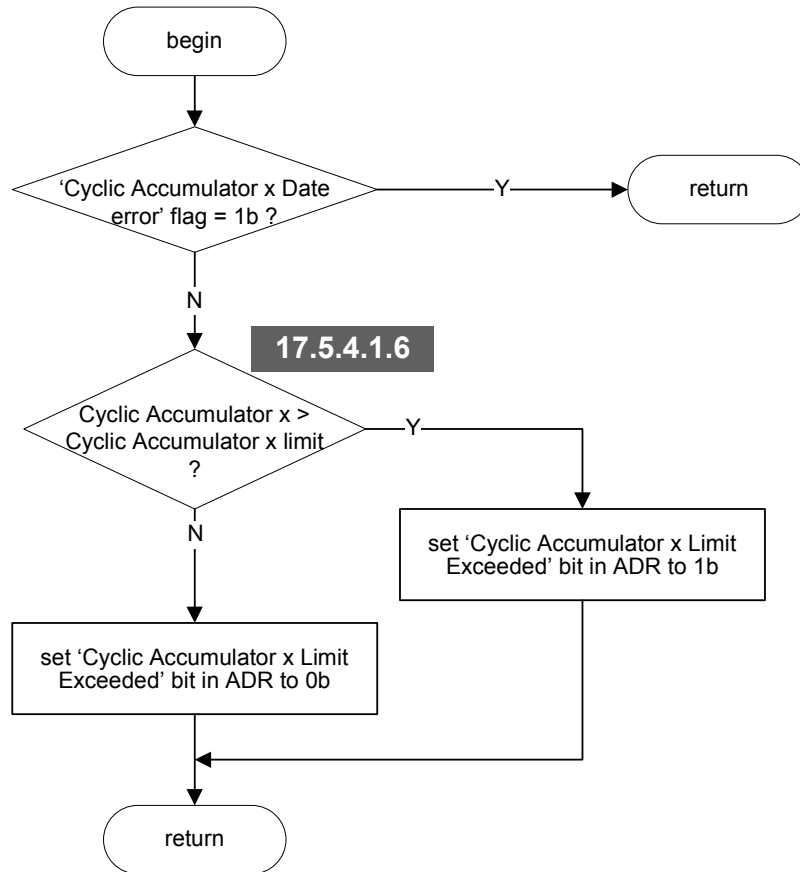


Flow 17-20 Cyclic Accumulator Check with Cumulating

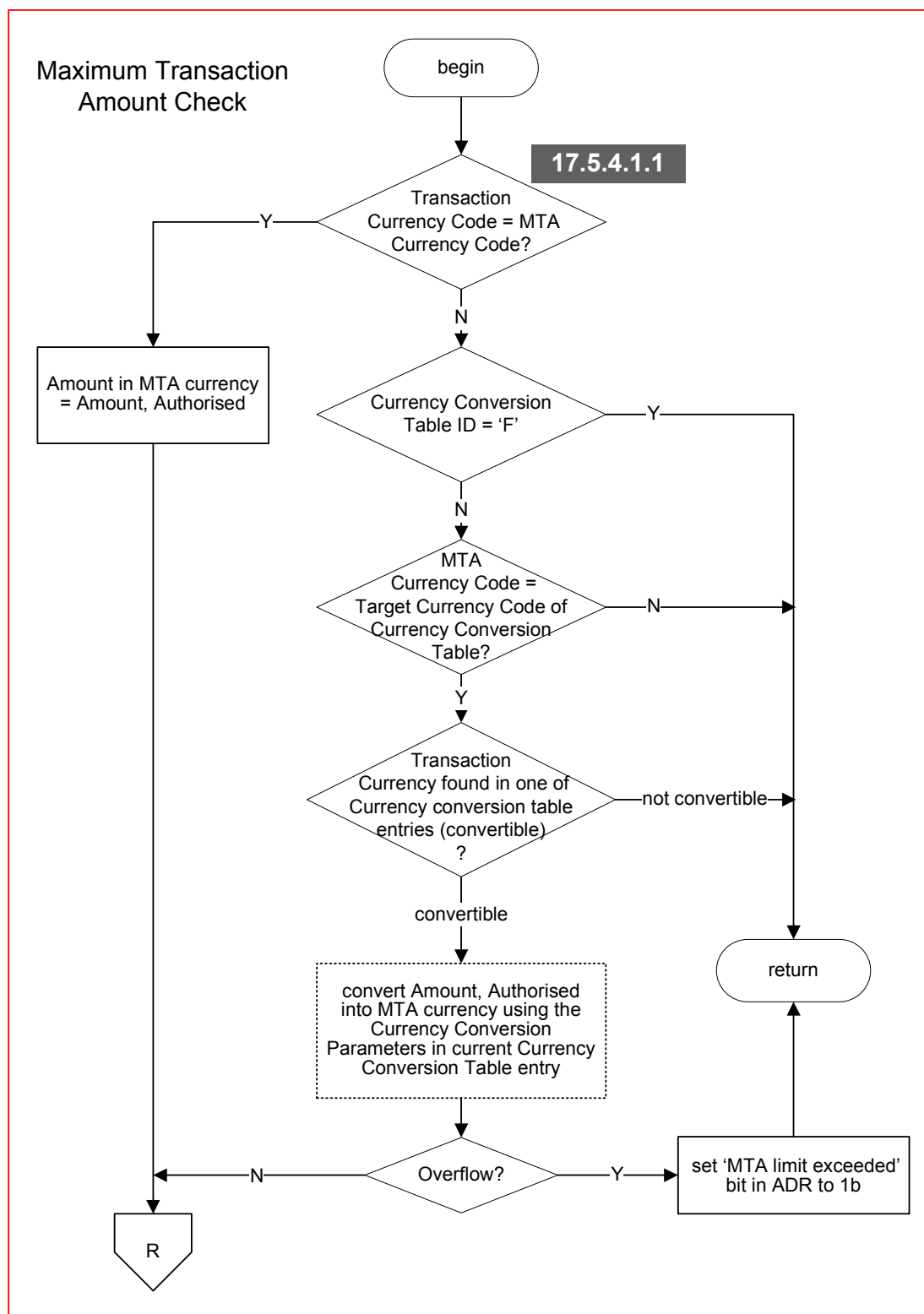
**Flow 17-20.1 Cyclic Accumulator Check with Cumulating, continued**



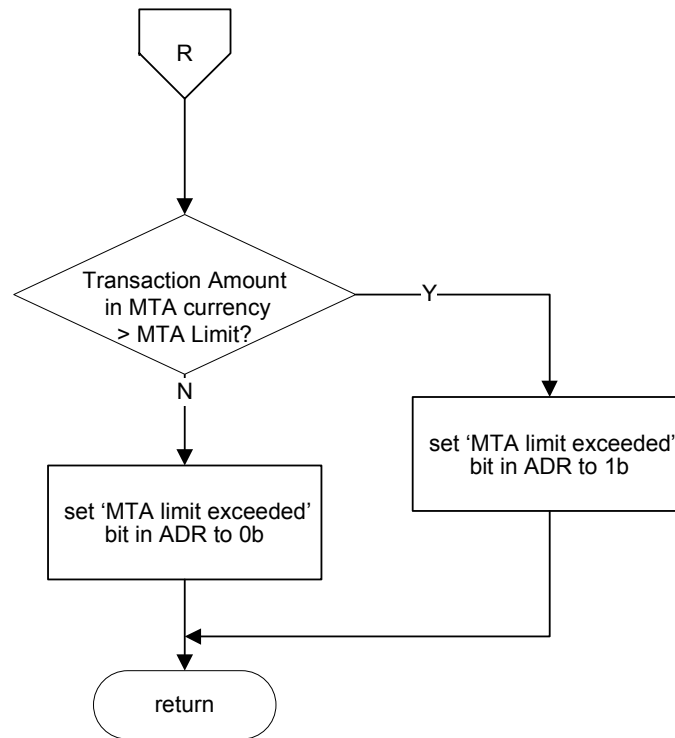
Flow 17-20.2 Cyclic Accumulator Check with Cumulating, continued

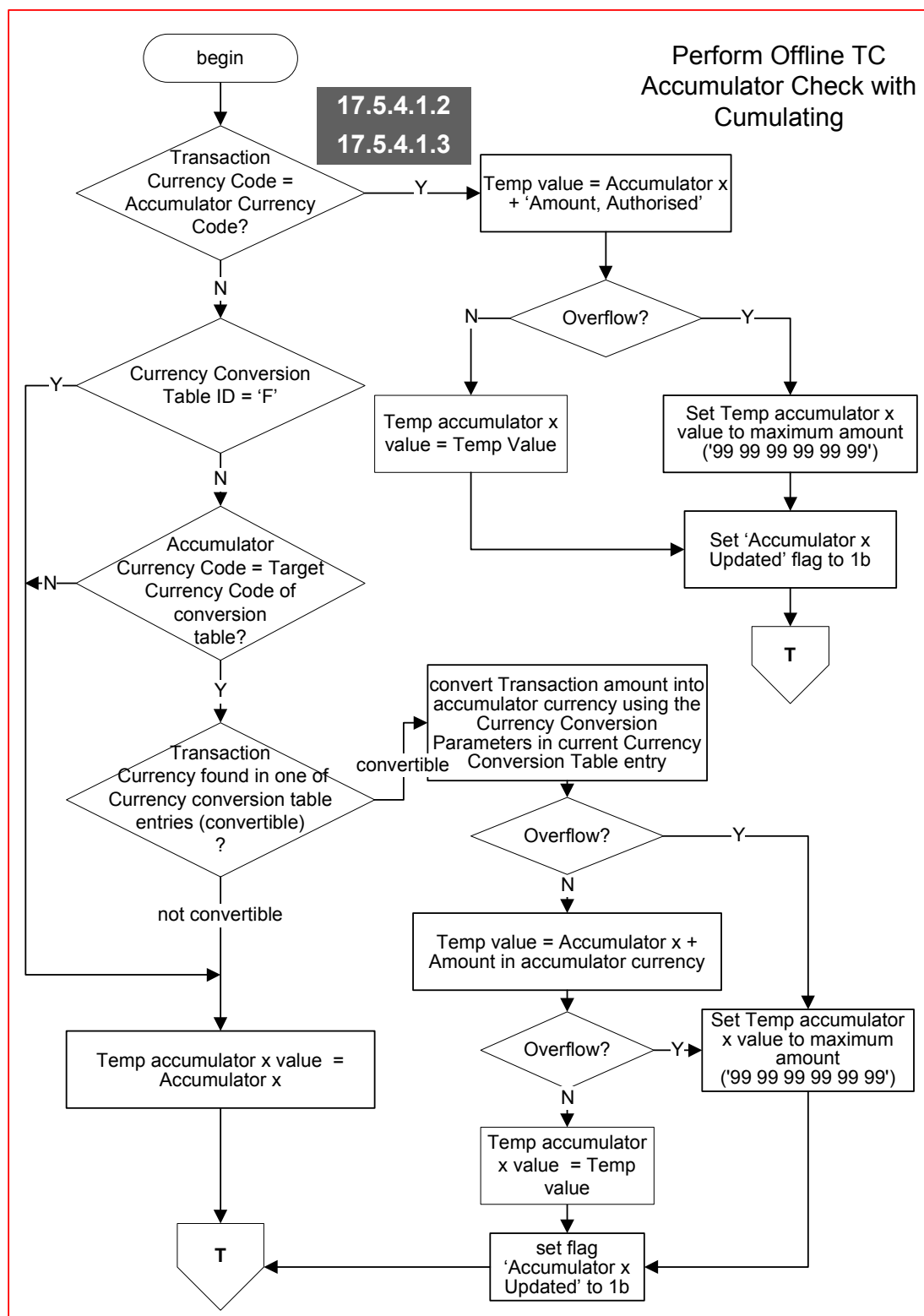
Perform Cyclic Accumulator
Check Without Cumulating

Flow 17-21 Cyclic Accumulator Check Without Cumulating

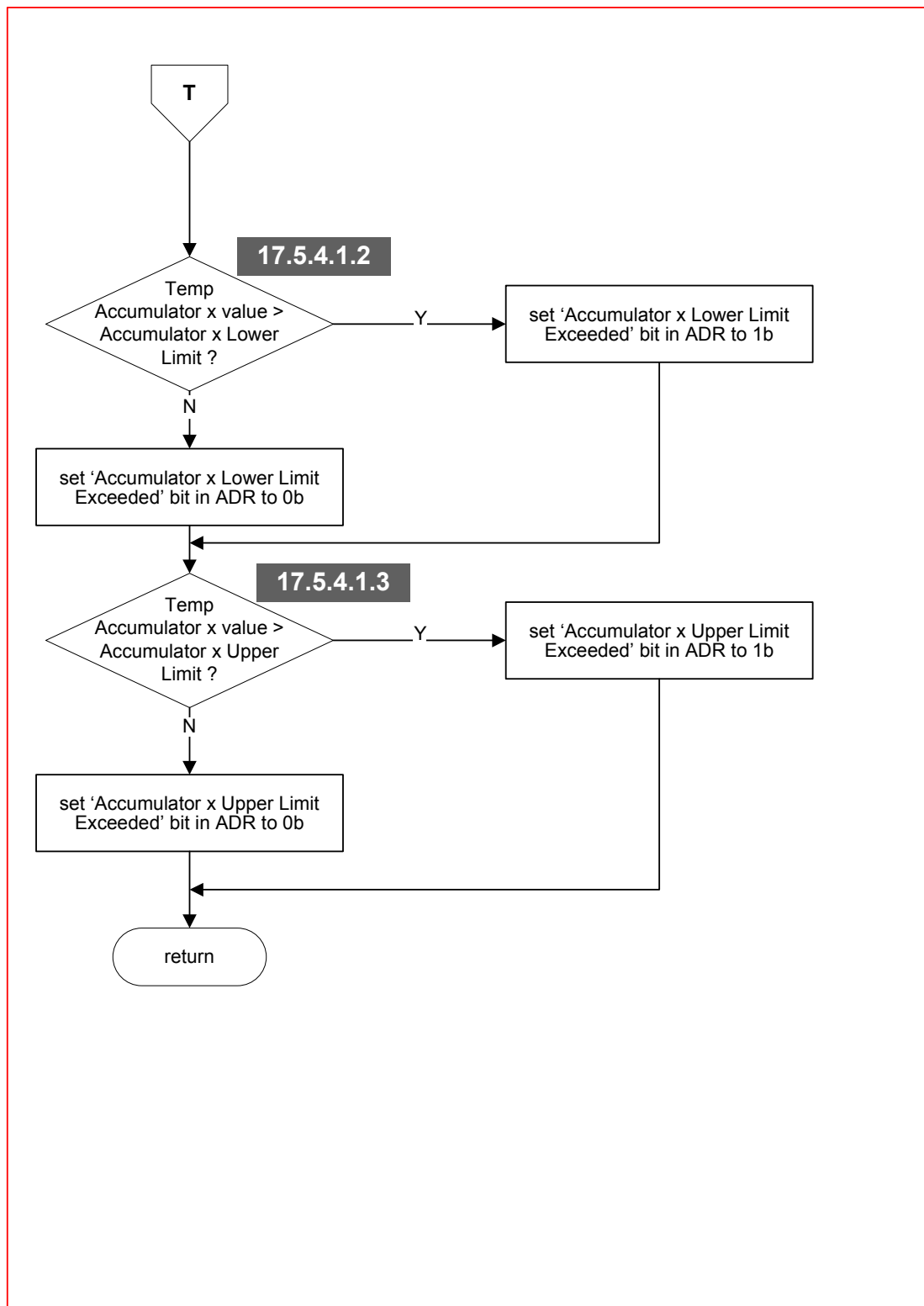


Flow 17-22 Maximum Transaction Amount Check

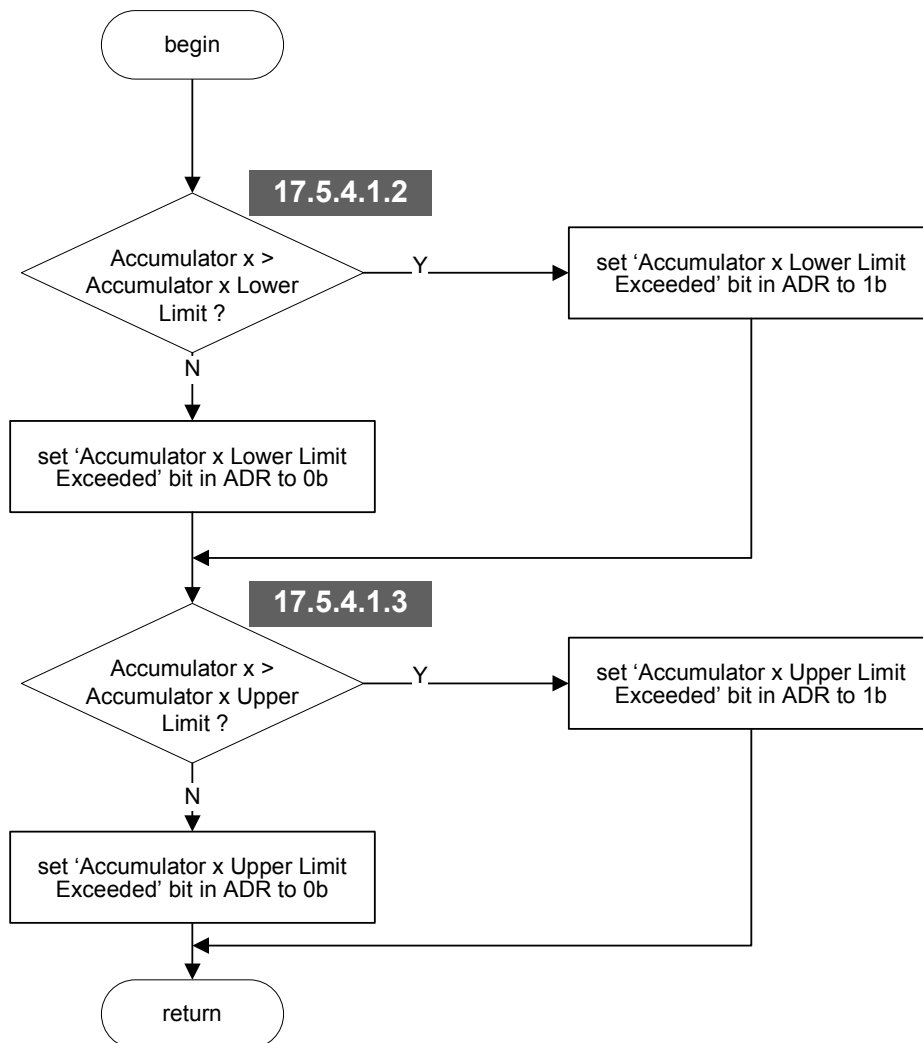
**Flow 17-22.1 Maximum Transaction Amount Check, continued**



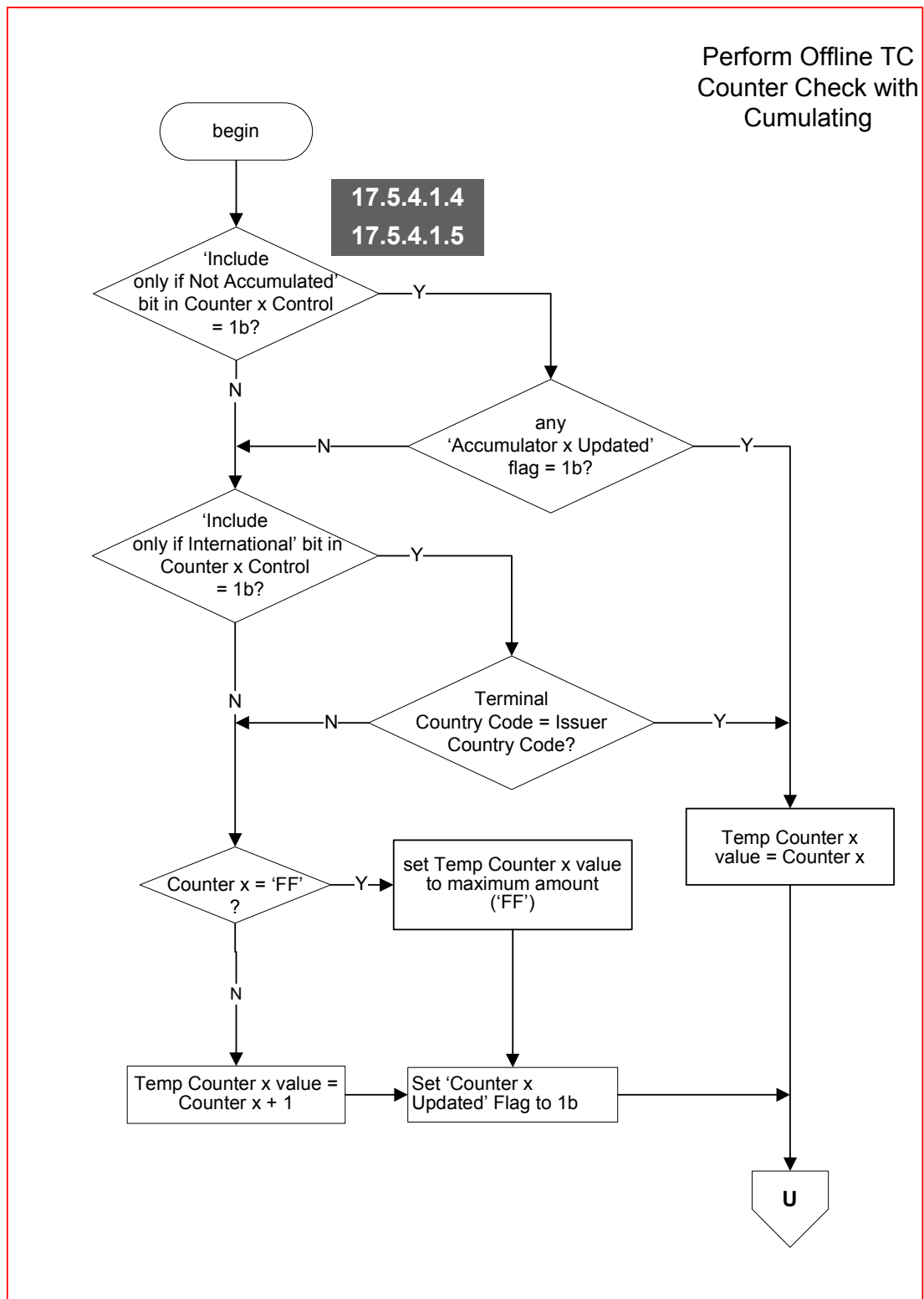
Flow 17-23 Offline TC Accumulator Check with Cumulating

**Flow 17-23.1 Offline TC Accumulator Check with Cumulating, continued**

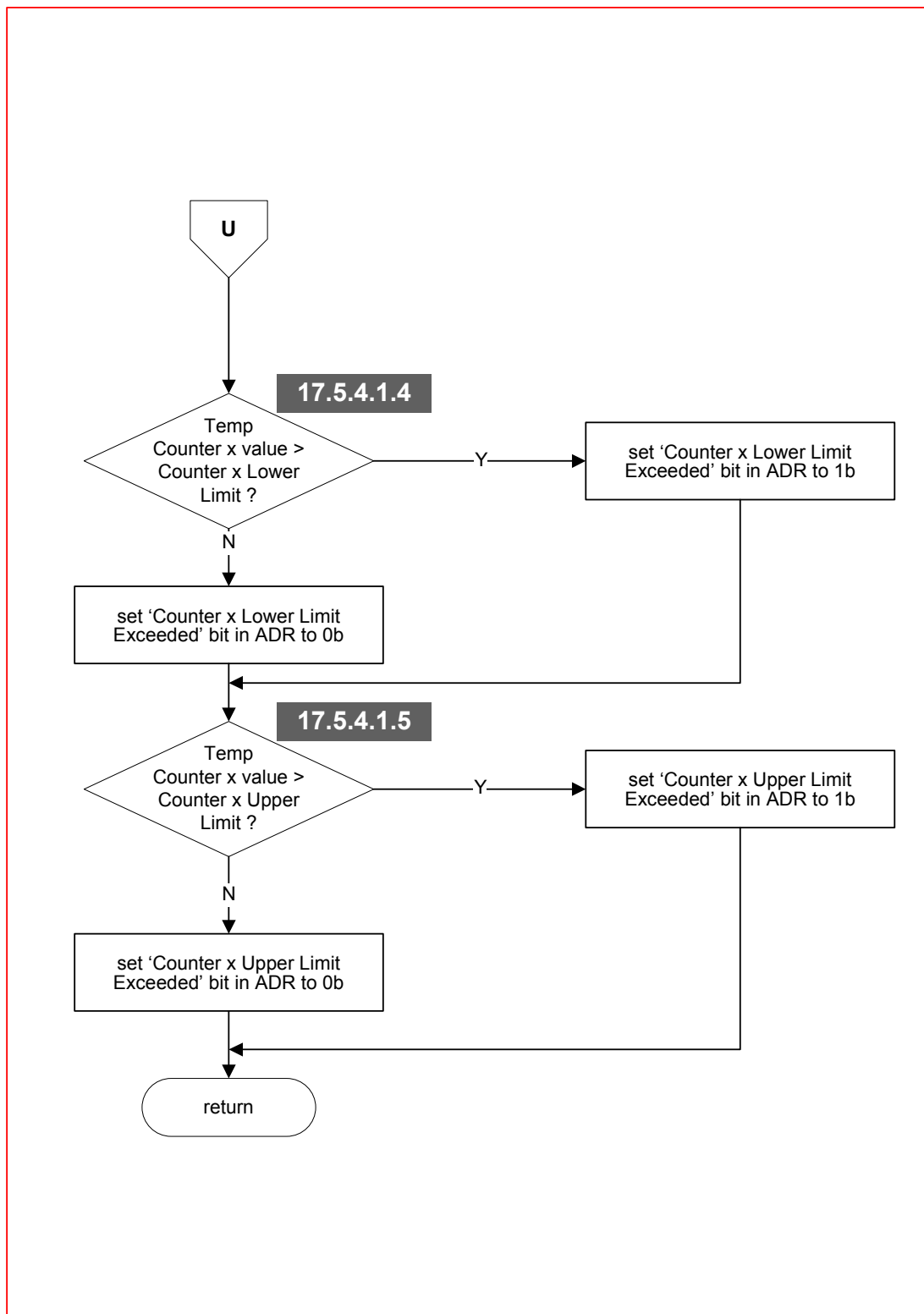
Perform Offline TC
Accumulator Check
Without Cumulating



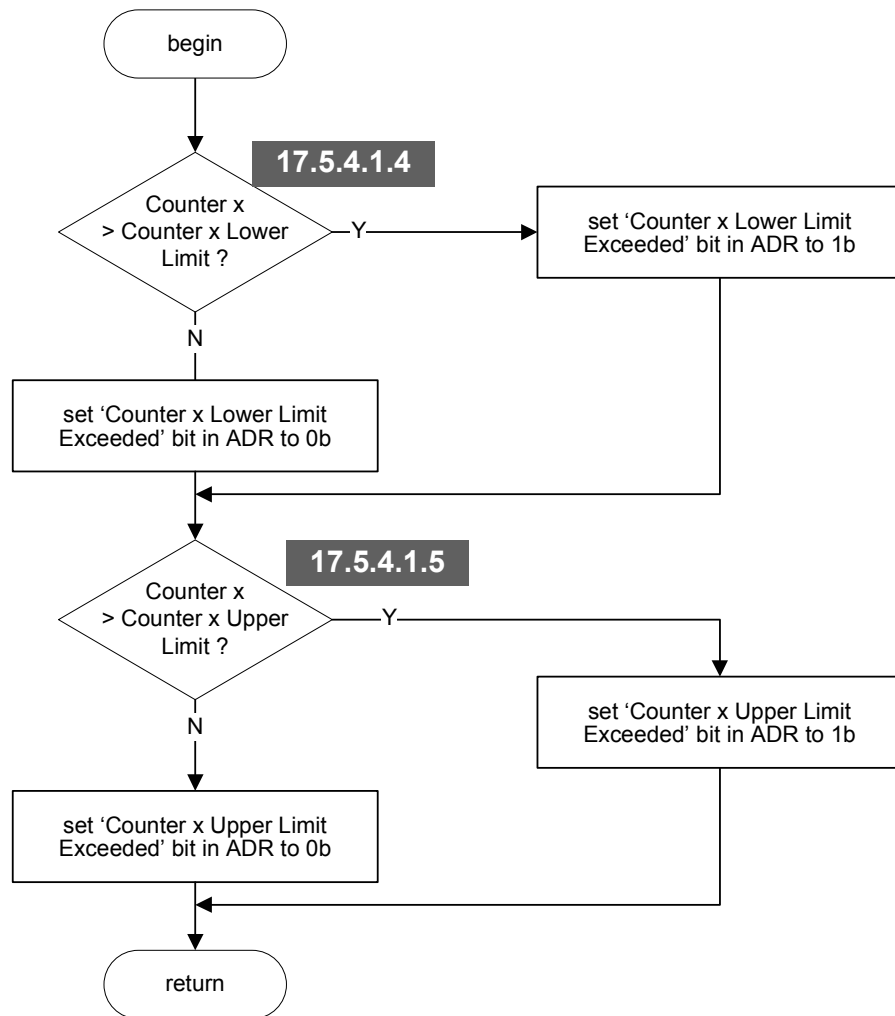
Flow 17-24 Offline TC Accumulator Check Without Cumulating



Flow 17-25 Offline TC Counter Check with Cumulating

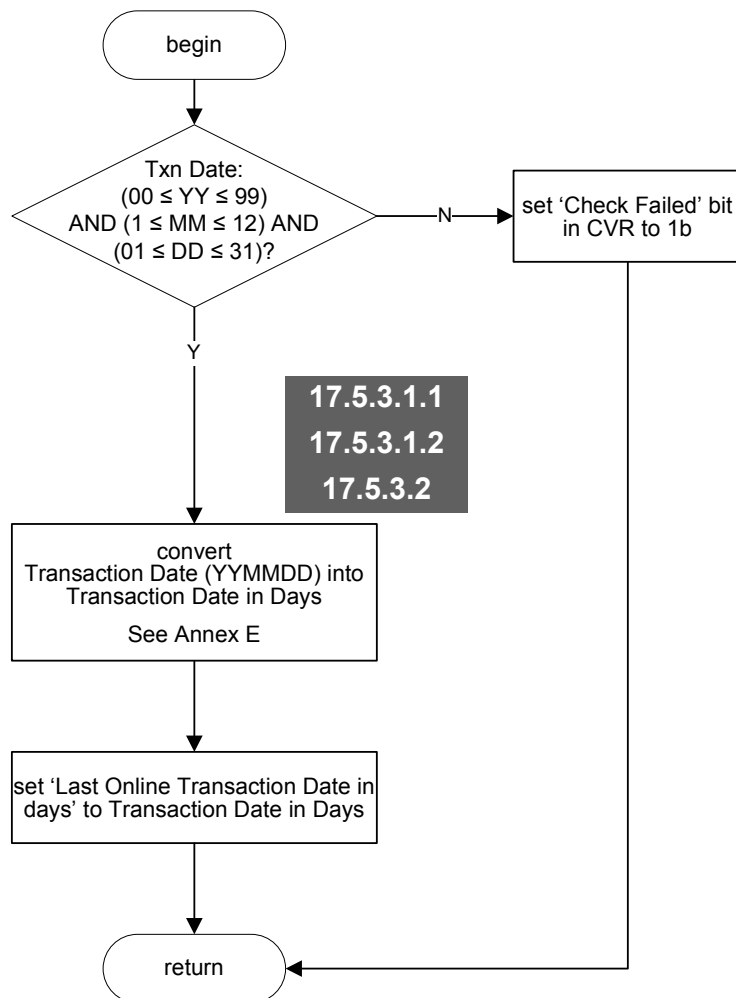


Flow 17-25.1 Offline TC Counter Check with Cumulating, continued

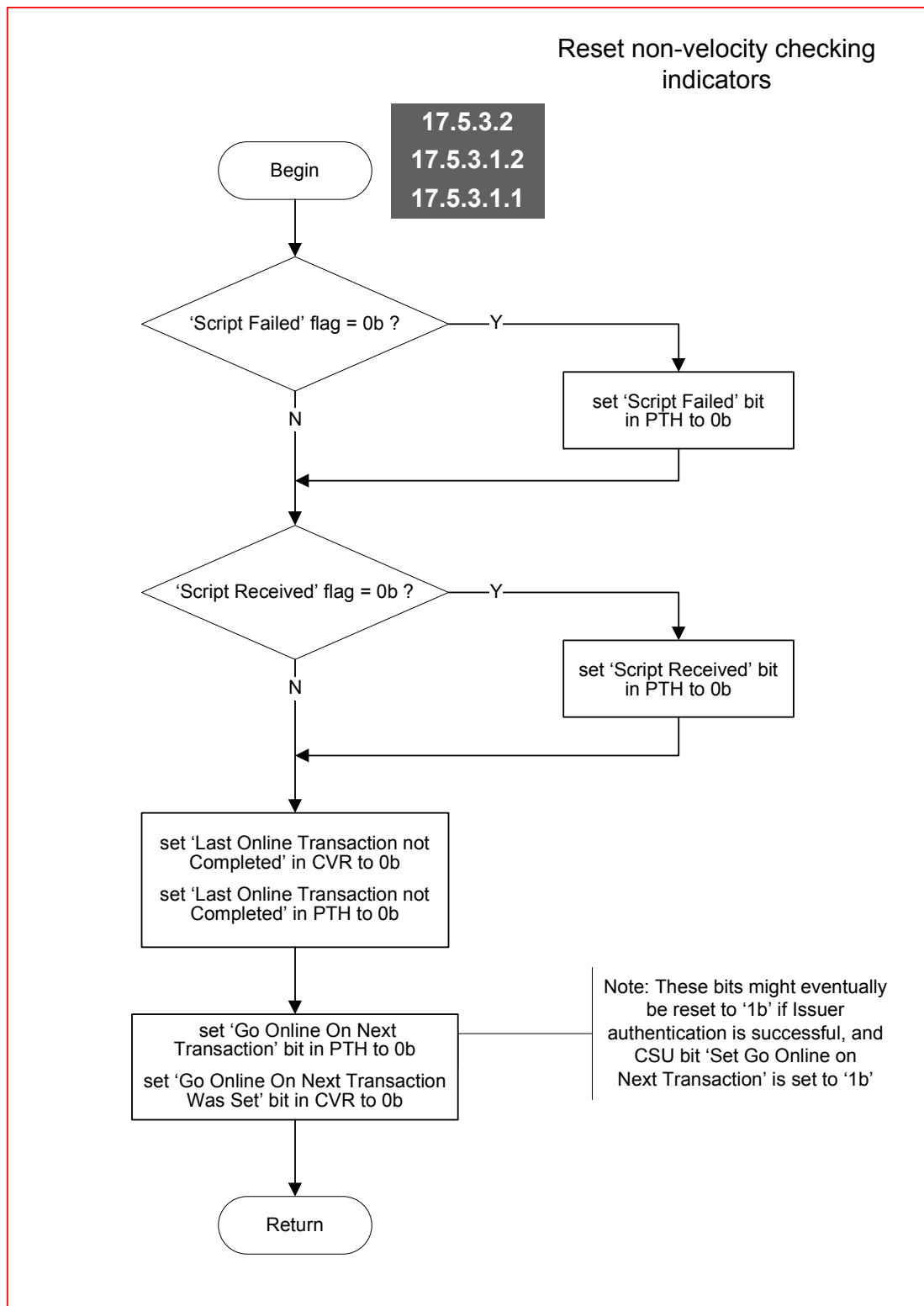
Perform Offline TC
Counter Check Without
Cumulating

Flow 17-26 Offline TC Counter Check Without Cumulating

Reset Maximum Number of Days Offline

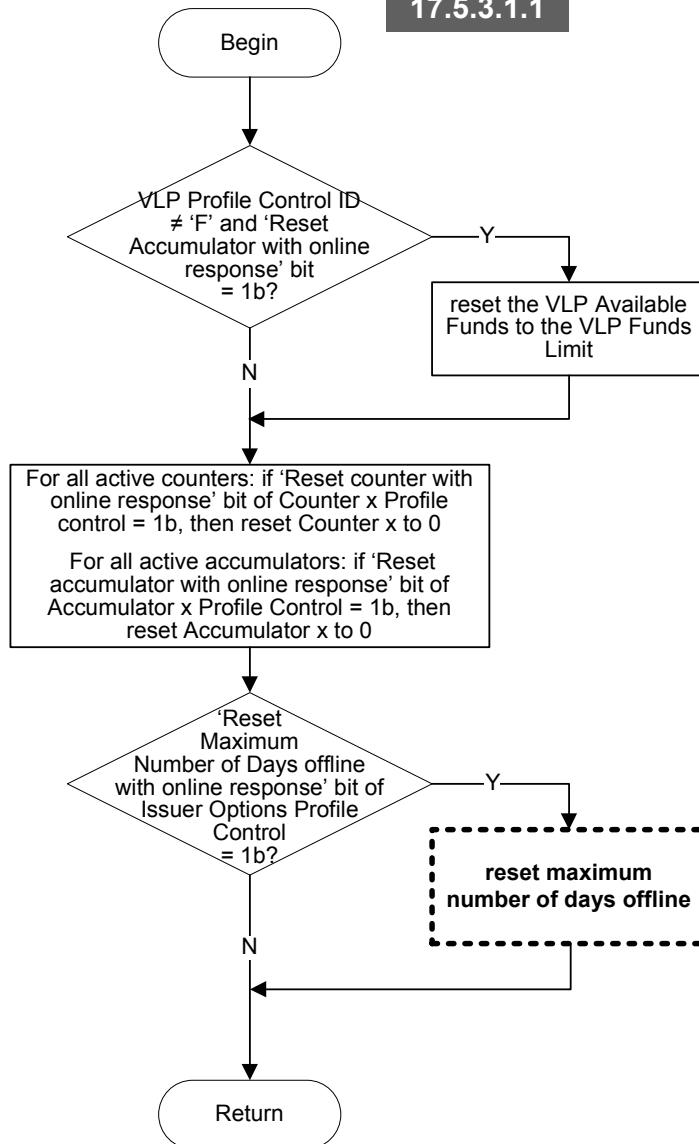


Flow 17-27 Reset Maximum Number of Days Offline

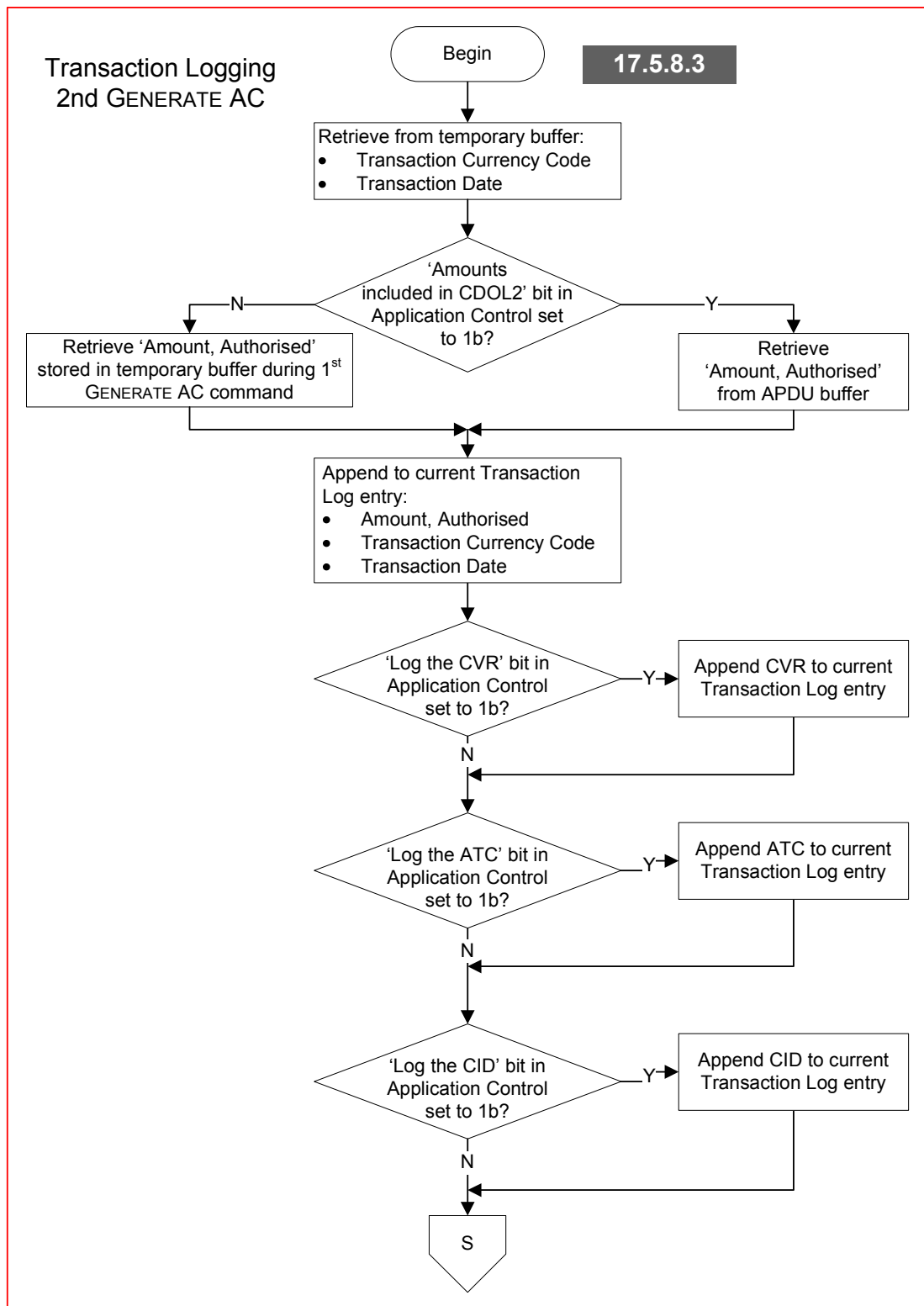
**Flow 17-28 Reset Non-velocity Checking Indicators**

Reset Velocity Checking Indicators

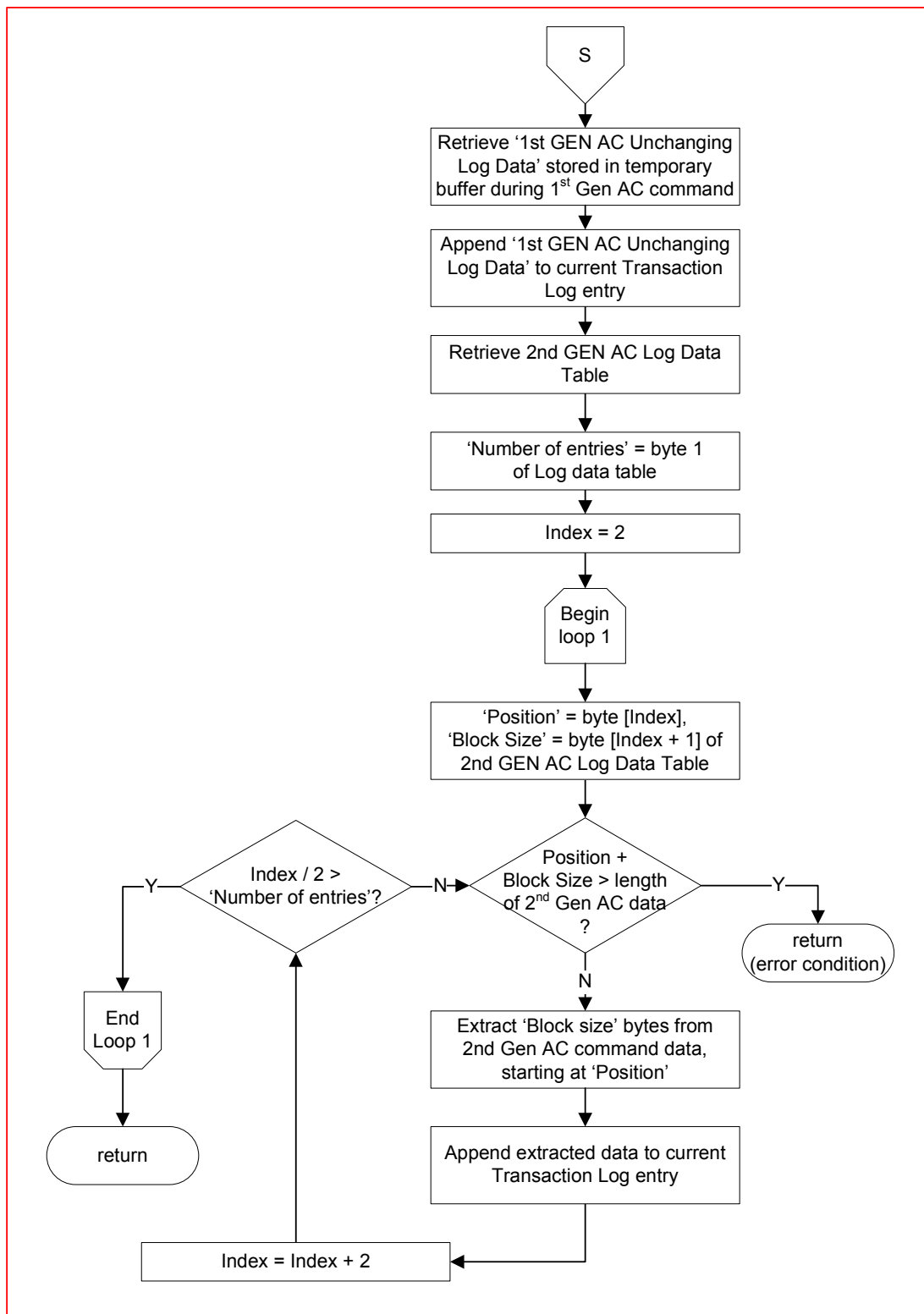
17.5.3.2
17.5.3.1.1



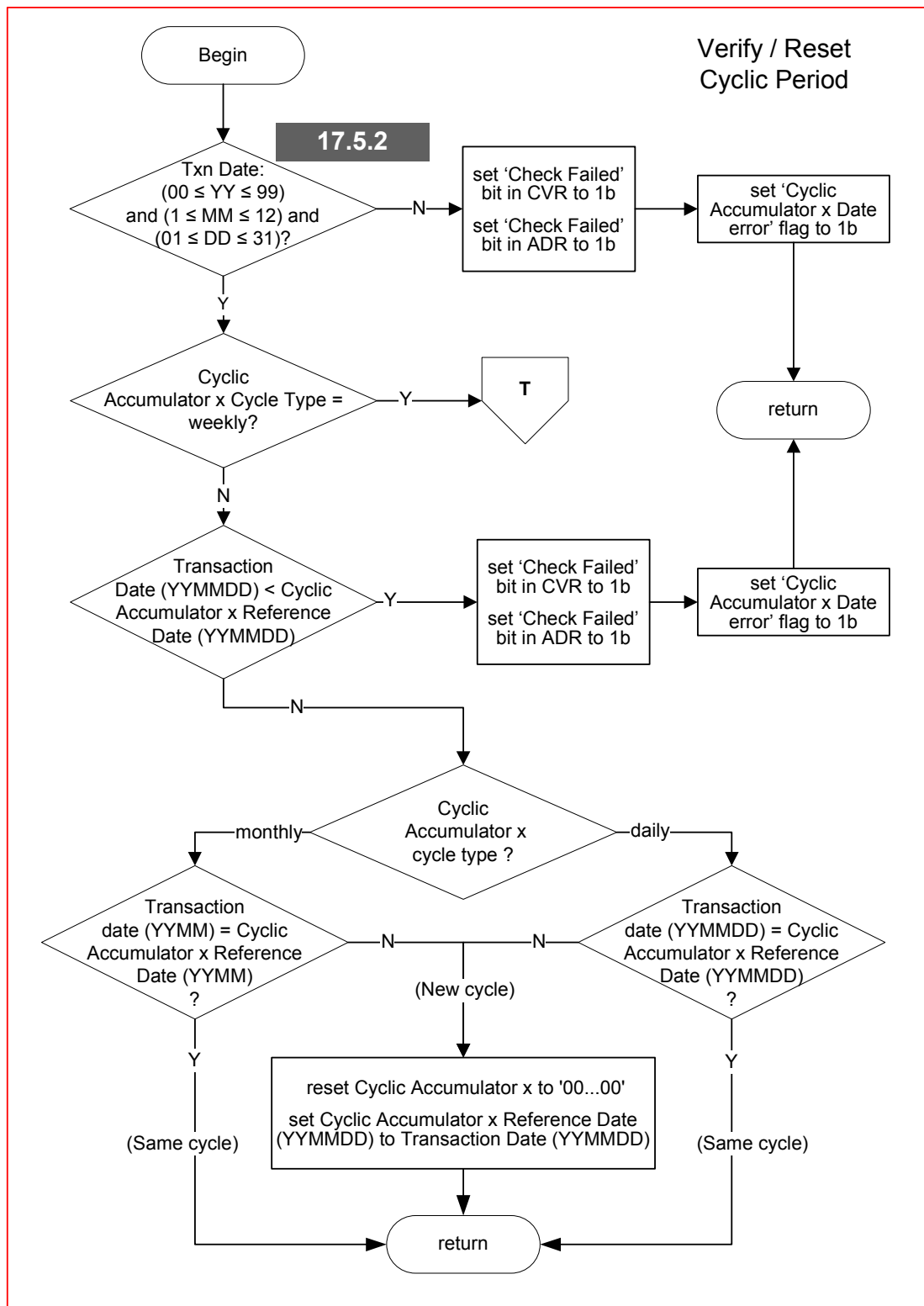
Flow 17-29 Reset Velocity Checking Indicators

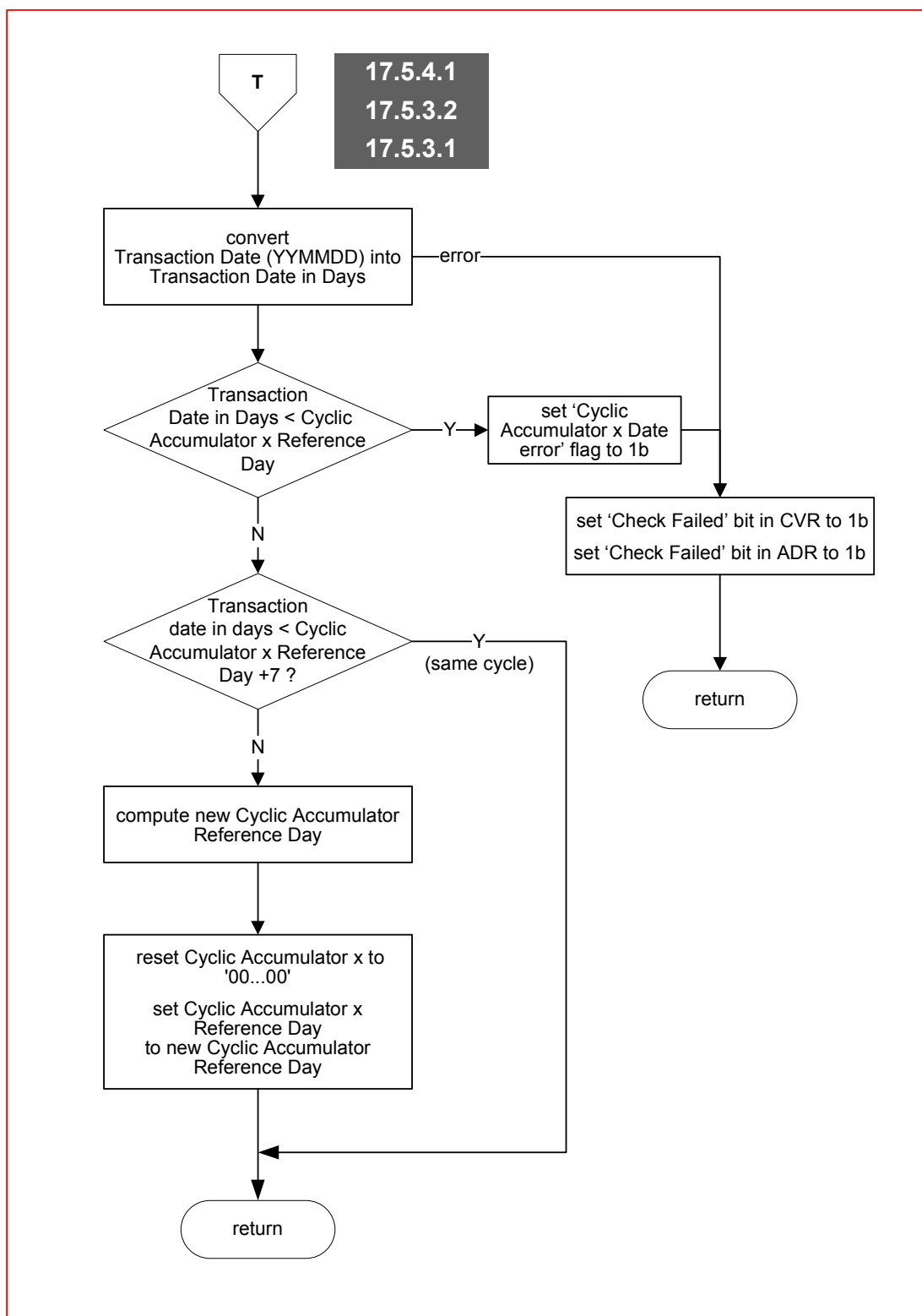


Flow 17-30 Transaction Logging Second GENERATE AC



Flow 17-30.1 Transaction Logging Second GENERATE AC, continued

**Flow 17-31 Verify/Reset Cyclic Period**



Flow 17-31.1 Verify/Reset Cyclic Period, continued

18 Issuer Script Command Processing

This section is organised as follows:

- 18.1 Purpose
- 18.2 Sequence of Execution
 - 18.2.1 Prior Related Processing
 - 18.2.2 Subsequent Related Processing
- 18.3 Card Data
- 18.4 Terminal Data
 - 18.4.1 Authorisation Response Data
- 18.5 Overview
 - 18.5.1 Authorisation Response Message
 - 18.5.2 Issuer-to-Card Script Processing
 - 18.5.3 Card Secure Messaging
 - 18.5.4 Resulting Indicators
 - 18.5.5 Script Commands Supported
- 18.6 APPLICATION UNBLOCK Command
 - 18.6.1 APPLICATION UNBLOCK Command Coding
 - 18.6.2 APPLICATION UNBLOCK Command Processing
 - 18.6.3 APPLICATION UNBLOCK Flow
- 18.7 PIN CHANGE/UNBLOCK Command
 - 18.7.1 PIN CHANGE/UNBLOCK Command Coding
 - 18.7.2 PIN CHANGE/UNBLOCK Processing
 - 18.7.3 PIN CHANGE/UNBLOCK Flow
- 18.8 PUT DATA Command
 - 18.8.1 PUT DATA Command Coding
 - 18.8.2 PUT DATA Processing
 - 18.8.3 PUT DATA Flow
- 18.9 UPDATE RECORD Command
 - 18.9.1 UPDATE RECORD Command Coding
 - 18.9.2 UPDATE RECORD Processing
 - 18.9.3 UPDATE RECORD Flow

18.1 Purpose

Issuer-to-Card Script Processing enables issuers to modify card parameters on cards (such as personalised data) without reissuing the card. With this function, the issuer transmits commands in issuer scripts contained in the authorisation response message. The terminal passes these commands to the card, where they are executed if secure messaging requirements are satisfied.

Issuer-to-Card Script Processing is performed as described in *EMV Book 3*, section 6 and section 10.10; and *EMV Book 4*, section 6.3.9. Secure messaging is performed as described in section 9 of the CCD part of EMV Book 2.

18.2 Sequence of Execution

18.2.1 Prior Related Processing

The online response received by the terminal from the acquirer may contain an issuer script to be processed during Issuer-to-Card Script Processing.

Online Processing

Script commands are sent to the card before the Second GENERATE AC command if the script template received by the terminal uses tag '71'.

Second Card Action Analysis

Script commands are sent to the card after the Second GENERATE AC command if the script template received by the terminal uses tag '72'.

18.2.2 Subsequent Related Processing

Second Card Action Analysis (current transaction)

If a script command is received before the Second GENERATE AC command, then during Second Card Action Analysis:

- The card sets the 'Number of issuer script commands containing secure messaging processed' bits in the Card Verification Results (CVR).
- If the Issuer-to-Card Script Processing failed, then the card sets the 'Issuer script processing failed' bit in the CVR.

Card Action Analysis (subsequent transactions)

During Card Action Analysis for the card's next transaction:

- The card sets the 'Number of issuer script commands containing secure messaging processed' bits in the CVR.
- The card sets the 'Script received' bit in the ADR to allow the application to go online because the card received a script command on a previous transaction.
- If the Issuer-to-Card Script Processing failed, then the card:
 - sets the 'Issuer script processing failed' bit in the ADR to allow the application to go online or decline offline because script processing failed on a previous transaction
 - and sets the 'Issuer script processing failed' bit in the CVR to indicate to the issuer that script processing failed on a previous transaction.

Second Card Action Analysis (subsequent transactions)

The indication of issuer script failure and issuer script received may be reset after online transactions based on the results of Issuer Authentication and issuer options in the card.

18.3 Card Data

The card data used in Issuer-to-Card Script Processing are listed and described in Table 18-1. For a detailed description of these data and their usage, see Annex L: Data Dictionary.

Data	Description	Template	Tag
Application Decisional Results	Used internal to the application to indicate exception conditions that occurred during the current and previous transactions. The Application Decisional Results are used during GENERATE AC command processing when determining whether the transaction should be declined offline or go online.	—	—
Card Verification Results (CVR)	<p>In second Card Action Analysis of the current transaction if the issuer script command was received before the second GENERATE AC command, or in first Card Action Analysis of subsequent transactions; the Card Action Analysis function fills in the following CVR subfields:</p> <ul style="list-style-type: none"> • Number of issuer script commands containing secure messaging processed — Set from the value in the Issuer Script Command Counter. • Issuer script processing failed — Set to indicate that processing of a script command failed. 	—	'9F52'

Table 18-1: Issuer-to-Card Script Processing – Card Data

Data	Description	Template	Tag
Issuer Script Command Counter	The Issuer Script Command Counter is used by the card to count each command containing secure messaging that was successfully processed either before or after the second GENERATE AC command.	—	—
Previous Transaction History (PTH)	<p>Contains indicators used to store information about previous transactions that is used in Card Risk Management for subsequent transactions. The PTH contains three script-related indicators:</p> <ul style="list-style-type: none">• Script received• Application blocked• Script failed <p>On subsequent transactions the 'Script received' and 'Script failed' indicators may be reset during Second Card Action Analysis.</p>	—	'C7'

Table 18-1: Issuer-to-Card Script Processing – Card Data, continued

18.4 Terminal Data

The terminal data used in Issuer-to-Card Script Processing are listed and described in Table 18-2. For a detailed description of these data and their usage, see Annex L: Data Dictionary.

Data	Description	Template	Tag
Issuer Script Results	Issuer Script Results contains the results of script processing and might be included in the clearing message	—	—
Terminal Verification Results (TVR)	The TVR contains two script-related indicators: <ul style="list-style-type: none"> • Issuer script failed before final GENERATE AC command • Issuer script failed after final GENERATE AC command¹ 	—	'95'
Transaction Status Information (TSI)	The TSI contains a flag indicating that Issuer-to-Card Script Processing was performed	—	'9B'

Table 18-2: Issuer-to-Card Script Processing – Terminal Data

¹ This indicator is always zero when the second GENERATE AC application cryptogram is generated, so the issuer should use the value zero for this bit when validating the second GENERATE AC application cryptogram.

18.4.1 Authorisation Response Data

If Issuer-to-Card Script Processing is to occur, then the issuer includes the data listed and described in Table 18-3 in the authorisation response:

Data	Description
Issuer Script Template	CPA allows only one issuer script template per transaction, but that script template may be either Tag '71' or Tag '72'. <ul style="list-style-type: none">• Tag '71' identifies Issuer Script Template 1 and contains proprietary issuer data for transmission to the card before the second GENERATE AC command.• Tag '72' identifies Issuer Script Template 2 and contains proprietary issuer data for transmission to the card after the second GENERATE AC command.
Issuer Script Identifier	The Issuer Script Identifier is a number used by the issuer to uniquely identify the issuer script.
Issuer Script Commands	Each issuer script command in the script is in BER-TLV format with a tag of '86'.

Table 18-3: Issuer-to-Card Script Processing – Authorisation Response Data

The terminal sends the individual commands in the issuer script to the card either before or after the Second GENERATE AC command, as indicated by the Issuer Script Template Tag. After each individual command, the terminal analyses the status returned by the command. If the card response indicates an error occurred, the terminal will terminate the processing of the issuer script.

18.5 Overview

Issuer scripts are processed in the following manner:

18.5.1 Authorisation Response Message

At most one issuer script template may be sent to the terminal in the authorisation response message for a CPA-compliant card. That Issuer Script Template may be either tag '71' or tag '72'.

- Tag '71' identifies Issuer Script Template 1 and contains issuer script commands to be transmitted to the card before the second Generate AC command.
- Tag '72' identifies Issuer Script Template 2 and contains issuer script commands to be transmitted to the card after the second Generate AC command.

The issuer script commands defined in this section are used to perform the following updates:

- Unblocking the application
- Changing the offline PIN
- Unblocking the offline PIN
- Updating card parameters

The originator of an issuer script command is assumed to be the card issuer. If an entity other than the issuer originates the commands, the same requirements apply.

The Card Block and Application Block capabilities using the CSU are supported in CPA.

18.5.2 Issuer-to-Card Script Processing

All issuer script commands require secure messaging. The application considers all commands received with secure messaging to be script commands.

Section 18.5.4 describes the setting of indicators relating to Issuer-to-Card Script processing when an issuer script command is transmitted to the card.

18.5.3 Card Secure Messaging

The objective of secure messaging is to ensure data confidentiality, message integrity, and issuer authentication. Message integrity and issuer authentication are achieved using a MAC. Data confidentiality is achieved using encipherment of the confidential data such as an offline PIN.

CPA authenticates the issuer before processing an issuer script command using secure messaging. Issuer Authentication (as described in section 17) need not be performed for script processing.

Req 18.1 (Issuer Authentication not required to process script commands):

The card shall not require that the Issuer Authentication described in section 17.5.3.1 of this specification be performed and passed in order to execute script commands.

Req 18.2 (Secure messaging required in scripts that update information):

Any issuer script command to update, reset, change, or alter in any way information in the application shall:

- *support secure messaging, and*
- *require that secure messaging be successfully performed.*

Once a MAC error has occurred, the card will reject subsequent script commands received in the same transaction (see section 20.3). In the flow diagrams the fulfilment of this requirement makes use of the 'Script failed' flag but other implementations are allowed.

Secure messaging for issuer script commands uses Secure Messaging Format 1 as specified for CCD-compliant applications in *EMV Book 2*, section 9.

18.5.3.1 Message Authentication (MACing)

Message Authentication (MACing) shall be used to authenticate the issuer as the originator of the issuer script command and to ensure that the command has not been altered after being sent by the issuer.

The MAC is generated using all the command header and the command data. The MAC is generated after encipherment of any confidential data in the command. The integrity of a command, including the data component contained in the command data field, if present, is ensured using secure messaging.

Req 18.3 (4-byte MAC support):

The CPA application shall support 4-byte MACs.

NOTE: Support for MACs of length 5 to 8 bytes is allowed as additional functionality. See section 19.3.5.

NOTE: CPA Secure Messaging includes chaining MACs from one command to the next within a script.

18.5.3.2 Data Confidentiality

Data encipherment is used to ensure the confidentiality of the data required for the command. Data encipherment occurs prior to generation of the command's MAC.

18.5.4 Resulting Indicators

The card uses the Issuer Script Command Counter to count each command containing secure messaging that was successfully processed² either before or after the second GENERATE AC command.

The card increments the Issuer Script Command Counter by one if the script command is successfully processed.

The card uses the 'Script received' bit in PTH to record that a script command has been received.

Req 18.4 (Definition of script command received):

The card shall set the 'Script received' bit in PTH to 1b when the card receives a script command and the command format is valid (that is, the command passes the command validation described in section 6.3, and the CLA byte of the command header indicates secure messaging).

The card uses the 'Script failed' bit in PTH to record that processing of a script command has failed.

Req 18.5 (Definition of script failed):

If a script command which passes requirement 18.4 is received either before or after the second GENERATE AC command, and the command is not successfully processed; then the card:

- *shall set the 'Script failed' bit in PTH to the value 1b,*
- *shall return an error status in the command response*
- *shall reject subsequent script commands received in the same transaction*

Examples of reasons the command may not be successfully processed include:

- Format validation failed
- Secure messaging failed (for example, the calculated MAC was not equal to the MAC in the command, or the length of MAC is not supported by the application)
- Secure messaging passed but processing of the command failed

² A script command has been successfully processed if the card will respond with SW1 SW2 = '9000' or a warning ('62xx' or '63xx').

18.5.5 Script Commands Supported

The issuer script commands supported by all implementations of CPA that may be performed using Issuer-to-Card Script Processing are listed below. Additional script commands may be supported by a CPA implementation but are beyond the scope of this specification.

Req 18.6 (Supported script commands):

All implementations of CPA shall support the following issuer script commands:

- *APPLICATION UNBLOCK*
- *PIN CHANGE/UNBLOCK*
- *PUT DATA*
- *UPDATE RECORD*

18.6 APPLICATION UNBLOCK Command

The APPLICATION UNBLOCK command is used to remove the restrictions placed on the application when the application was blocked. For example, the application may have been blocked when the 'Application Block' bit in a successfully-recovered CSU was set to the value 1b.

Unblocking the application through the use of the APPLICATION UNBLOCK command means that the application no longer is required to respond to all GENERATE AC commands with an AAC and that the SELECT response during Application Selection is no longer required to be SW1SW2 = '6283'.

In this version of CPA, unblocking of an application should occur only at a special device as designated by the issuer. Since unblocking the application is performed at a special device, the transaction processing flow for unblocking the application is beyond the scope of this specification; it has only to comply with the sequence of commands described in section 6.2.2.

The APPLICATION UNBLOCK command is performed as described in *EMV Book 3*, section 6.5.2. The command received from the terminal includes the secure messaging MAC in the command data field.

The card receives the APPLICATION UNBLOCK command from the terminal. If the MAC is valid and the Parameter Bytes P1 and P2 contain the value '00', then the application sets the 'Application blocked' bit in the PTH to the value 0b before sending the APPLICATION UNBLOCK response to the terminal.

18.6.1 APPLICATION UNBLOCK Command Coding

Code	Value
CLA	'8C'
INS	'18'
P1	'00'
P2	'00'
New Lc	'06'
Data	Secured Command Data Field
Le	not present

Table 18-4: APPLICATION UNBLOCK Command Message

Req 18.7 (APPLICATION UNBLOCK script command received):

The application shall set the 'Script received' bit in the PTH to the value 1b.

18.6.1.1 APPLICATION UNBLOCK Command Format Validation

Req 18.8 (Check P1 value for APPLICATION UNBLOCK):

If P1 is not '00', then the card:

- *shall set the 'Script failed' bit in PTH to the value 1b,*
- *shall discontinue processing the APPLICATION UNBLOCK command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6A86' (Incorrect Parameters P1 P2).*

Req 18.9 (Check P2 value for APPLICATION UNBLOCK):

If P2 is not '00', then the card:

- *shall set the 'Script failed' bit in PTH to the value 1b,*
- *shall discontinue processing the APPLICATION UNBLOCK command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6A86' (Incorrect Parameters P1 P2).*

18.6.2 APPLICATION UNBLOCK Command Processing

The command data (Secured Command Data Field) for Application Unblock contains only the MAC data, as shown in Figure 18-1.

'8E'	'04'	MAC (4 bytes)
------	------	---------------

Figure 18-1: Command Data Format if Only MAC Data is Present

Req 18.10 (Check command data length for APPLICATION UNBLOCK):

If New Lc has a value other than '06', then the card:

- shall set the 'Script failed' bit in the PTH to the value 1b
- shall discontinue processing the APPLICATION UNBLOCK command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6700' (Wrong Length).

Req 18.11 (Check MAC tag for APPLICATION UNBLOCK):

If the first byte of the Command Data has a value other than '8E' (MAC tag), then the card:

- shall set the 'Script failed' bit in the PTH to the value 1b
- shall discontinue processing the APPLICATION UNBLOCK command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6987' (Expected secure messaging data objects missing).

Req 18.12 (Check MAC length for APPLICATION UNBLOCK):

If the second byte of the Command Data has a value other than '04' (MAC length), then the card:

- shall set the 'Script failed' bit in the PTH to the value 1b
- shall discontinue processing the APPLICATION UNBLOCK command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6988' (Incorrect secure messaging data objects).

The application verifies the MAC.

Req 18.13 (Verify MAC and unblock application):

If the MAC verification is not successful, then the application:

- shall set the 'Script failed' bit in the PTH to the value 1b
- shall discontinue processing the APPLICATION UNBLOCK command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6982' (Security status not satisfied).

If the MAC verification is successful, then the application shall:

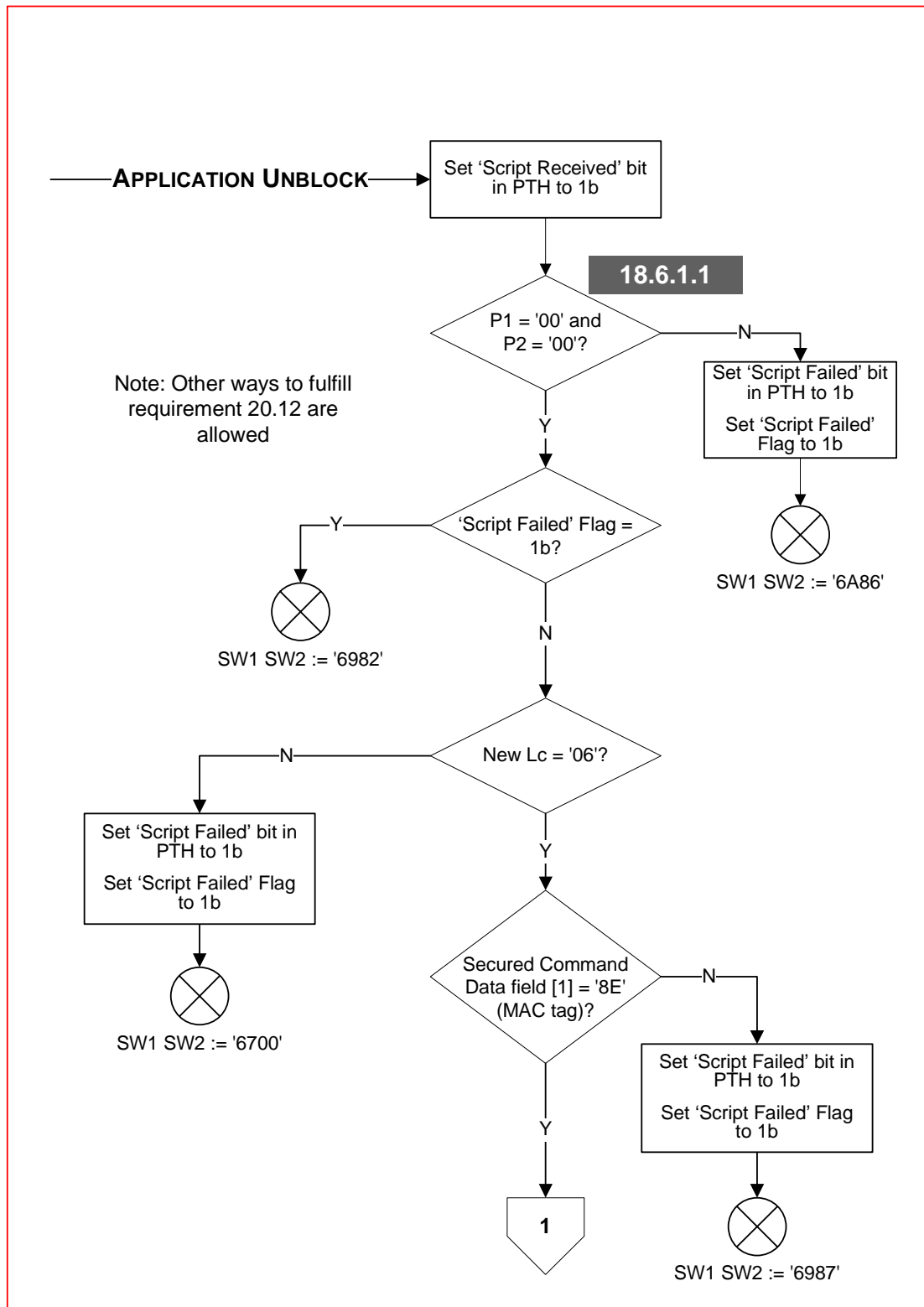
- be unblocked
- set the 'Application Blocked' bit in the PTH to the value 0b
- increment by one the Issuer Script Command Counter³
- respond with SW1 SW2 = '9000'

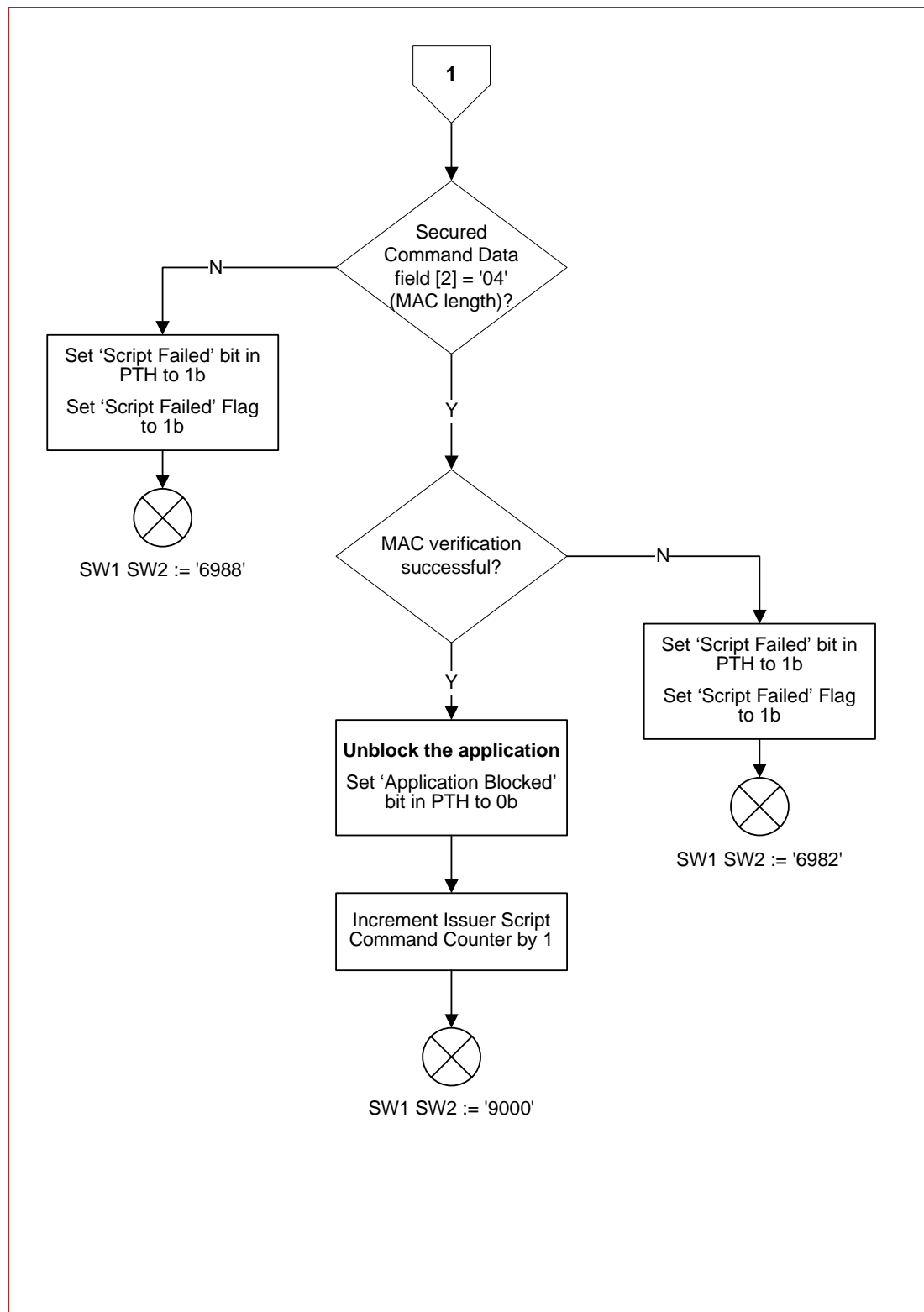
18.6.3 APPLICATION UNBLOCK Flow

Figure 18-2 illustrates the flow of the APPLICATION UNBLOCK command.

³ The Issuer Script Command Counter is for information only and is cyclic: '0F'+1='00'.

Figure 18-2: APPLICATION UNBLOCK Flow





18.7 PIN CHANGE/UNBLOCK Command

The PIN CHANGE/UNBLOCK command provides the issuer the capability either to simultaneously change and unblock the Reference PIN or to unblock the Reference PIN. The Reference PIN is unblocked by resetting the PIN Try Counter to the value of the PIN Try Limit.

Changing the PIN

If the card's Reference PIN is to be changed:

- The plaintext PIN Block format is as shown in EMV Book 3, section 6.5.12.2.
- The PIN data is enciphered as described in EMV Book 2, part IV section 9.3, with the Padding Indicator byte set to the value '01'. This means that an additional 8 byte block having the value '80 00 00 00 00 00 00 00' is appended to the PIN-block before encipherment.

Whenever the card's Reference PIN is changed, the card implicitly unblocks the PIN, since the successful completion of the PIN CHANGE/UNBLOCK command automatically resets the PIN Try Counter to the PIN Try Limit.

Regardless of the method used, PIN change should only be performed within a secure environment controlled by the issuer.

18.7.1 PIN CHANGE/UNBLOCK Command Coding

Code	Value
CLA	'8C'
INS	'24'
P1	'00'
P2	'00' or '02'
New Lc	'06' or '19'
Data	PIN related data (Enciphered PIN data component, if present, and MAC data component)
Le	not present

Table 18-5: PIN CHANGE / UNBLOCK Command Message

Req 18.14 (PIN CHANGE/UNBLOCK script command received):

The application shall set the 'Script received' bit in the PTH to the value 1b.

18.7.1.1 PIN Change/Unblock Command Format Validation

Req 18.15 (Check P1 value for PIN CHANGE/UNBLOCK):

If P1 is not '00', then the card:

- *shall set the 'Script failed' bit in PTH to the value 1b,*
- *shall discontinue processing the command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6A86' (Incorrect Parameters P1 P2).*

Req 18.16 (Check P2 value for PIN CHANGE/UNBLOCK):

If P2 is neither '00' nor '02', then the card:

- *shall set the 'Script failed' bit in PTH to the value 1b,*
- *shall discontinue processing the command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6A86' (Incorrect Parameters P1 P2).*

18.7.2 PIN CHANGE/UNBLOCK Processing

Req 18.17 (Check whether to Change the PIN or Unblock the PIN):

If P2 is '00', continue with the processing described in section 18.7.2.1. Otherwise, if P2 is '02', continue with the processing described in section 18.7.2.2.

18.7.2.1 Unblock PIN

The processing in this section applies if P2 is '00'.

The command data (PIN related data) for unblocking the PIN contains only the MAC data, as shown in Figure 18-1.

Req 18.18 (Check command data length for PIN UNBLOCK):

If New Lc has a value other than '06', then the card:

- *shall set the 'Script failed' bit in the PTH to the value 1b*
- *shall discontinue processing the PIN CHANGE/ UNBLOCK command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6700' (Wrong Length).*

Req 18.19 (Check MAC tag for PIN UNBLOCK):

If the first byte of the PIN Related Data has a value other than '8E' (MAC tag), then the card:

- *shall set the 'Script failed' bit in the PTH to the value 1b*
- *shall discontinue processing the PIN CHANGE/ UNBLOCK command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6987' (Expected secure messaging data objects missing).*

Req 18.20 (Check MAC length for PIN UNBLOCK):

If the second byte of the PIN Related Data has a value other than '04' (MAC length), then the card:

- *shall set the 'Script failed' bit in the PTH to the value 1b*
- *shall discontinue processing the PIN CHANGE/ UNBLOCK command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6988' (Incorrect secure messaging data objects).*

The application verifies the MAC.

Req 18.21 (Verify MAC and reset PIN Try Counter):

If the MAC verification is successful, then the card shall:

- *set the PIN Try Counter to the value of the PIN Try Limit*
- *increment by one the Issuer Script Command Counter⁴*
- *respond with SW1 SW2 = '9000'.*

If the MAC verification is not successful, then the card:

- *shall set the 'Script failed' bit in the PTH to the value 1b*
- *shall discontinue processing the PIN CHANGE/ UNBLOCK command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6982' (Security status not satisfied)*

⁴ The Issuer Script Command Counter is for information only and is cyclic: '0F'+1='00'.

18.7.2.2 Change PIN

The processing in this section applies if P2 is '02'.

The plaintext PIN Block before encipherment for confidentiality is coded as shown in *EMV Book 3*, Table 24. It is padded with '80 00 00 00 00 00 00 00' and then enciphered as specified for a CCD-compliant application in *EMV Book 2*, section 9.3. For an illustration of the process for recovering the New PIN Block, see Figure 18-3.

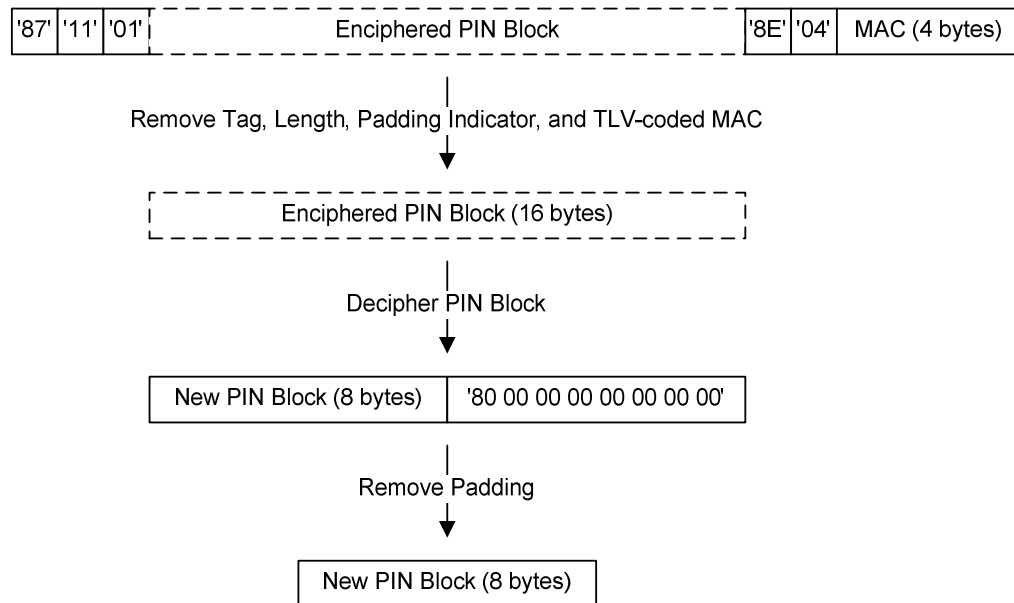


Figure 18-3: Recovering New PIN Block from PIN CHANGE/UNBLOCK Command Data

Req 18.22 (Check command data length for PIN CHANGE):

If New Lc has a value other than '19', then the card:

- shall set the 'Script failed' bit in the PTH to the value 1b
- shall discontinue processing the PIN CHANGE/ UNBLOCK command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6700' (Wrong length).

Req 18.23 (Check secure messaging template tag for PIN CHANGE):

If the first byte of the PIN Related Data has a value other than '87', then the card shall:

- set the 'Script failed' bit in the PTH to the value 1b
- discontinue processing the PIN CHANGE/ UNBLOCK command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6987' (Expected secure messaging data objects missing).

Req 18.24 (Check length of secure messaging for PIN CHANGE):

If the second byte of the PIN Related Data has a value other than '11', then the card:

- *shall set the 'Script failed' bit in the PTH to the value 1b*
- *shall discontinue processing the PIN CHANGE/ UNBLOCK command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6988' (Incorrect secure messaging data objects).*

Req 18.25 (Check Padding Indicator for PIN CHANGE):

If the third byte of the PIN Related Data has a value other than '01' (Padding Indicator), then the card:

- *shall set the 'Script failed' bit in the PTH to the value 1b*
- *shall discontinue processing the PIN CHANGE/ UNBLOCK command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6988' (Incorrect secure messaging data objects).*

Req 18.26 (Check MAC tag for PIN CHANGE):

If the twentieth byte of the PIN Related Data has a value other than '8E' (MAC tag), then the card:

- *shall set the 'Script failed' bit in the PTH to the value 1b*
- *shall discontinue processing the PIN CHANGE/ UNBLOCK command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6987' (Expected secure messaging data objects missing).*

Req 18.27 (Check MAC length for PIN CHANGE):

If the twenty-first byte of the PIN Related Data has a value other than '04' (MAC length), then the card:

- *shall set the 'Script failed' bit in the PTH to the value 1b*
- *shall discontinue processing the PIN CHANGE/ UNBLOCK command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6988' (Incorrect secure messaging data objects).*

The application verifies the MAC.

Req 18.28 (MAC verification failed for PIN CHANGE):

If the MAC verification is not successful, then the card:

- *shall set the 'Script failed' bit in the PTH to the value 1b*
- *shall discontinue processing the PIN CHANGE/ UNBLOCK command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6982' (Security status not satisfied)*

Otherwise the card shall decipher bytes 4-19 of the command data to recover the New PIN Block.

Req 18.29 (Verify PIN Block format and change Reference PIN):

If **all** of the following are true:

- *byte 1, bits b8-b5 of the New PIN Block (the control field) has the value '2'*
- **and** *byte 1, bits b4-b1 of the New PIN Block (the PIN length) has a value greater than or equal to '4'*
- **and** *byte 1, bits b4-b1 of the New PIN Block (the PIN length) has a value less than or equal to 'C'*
- **and** *all the Filler digits of the New PIN Block have the value 'F'*

then the card shall:

- *update the Reference PIN*
- *set the PIN Try Counter to the value of the PIN Try Limit*
- *increment by one the Issuer Script Command Counter⁵*
- *respond with SW1 SW2 = '9000'.*

Otherwise the card:

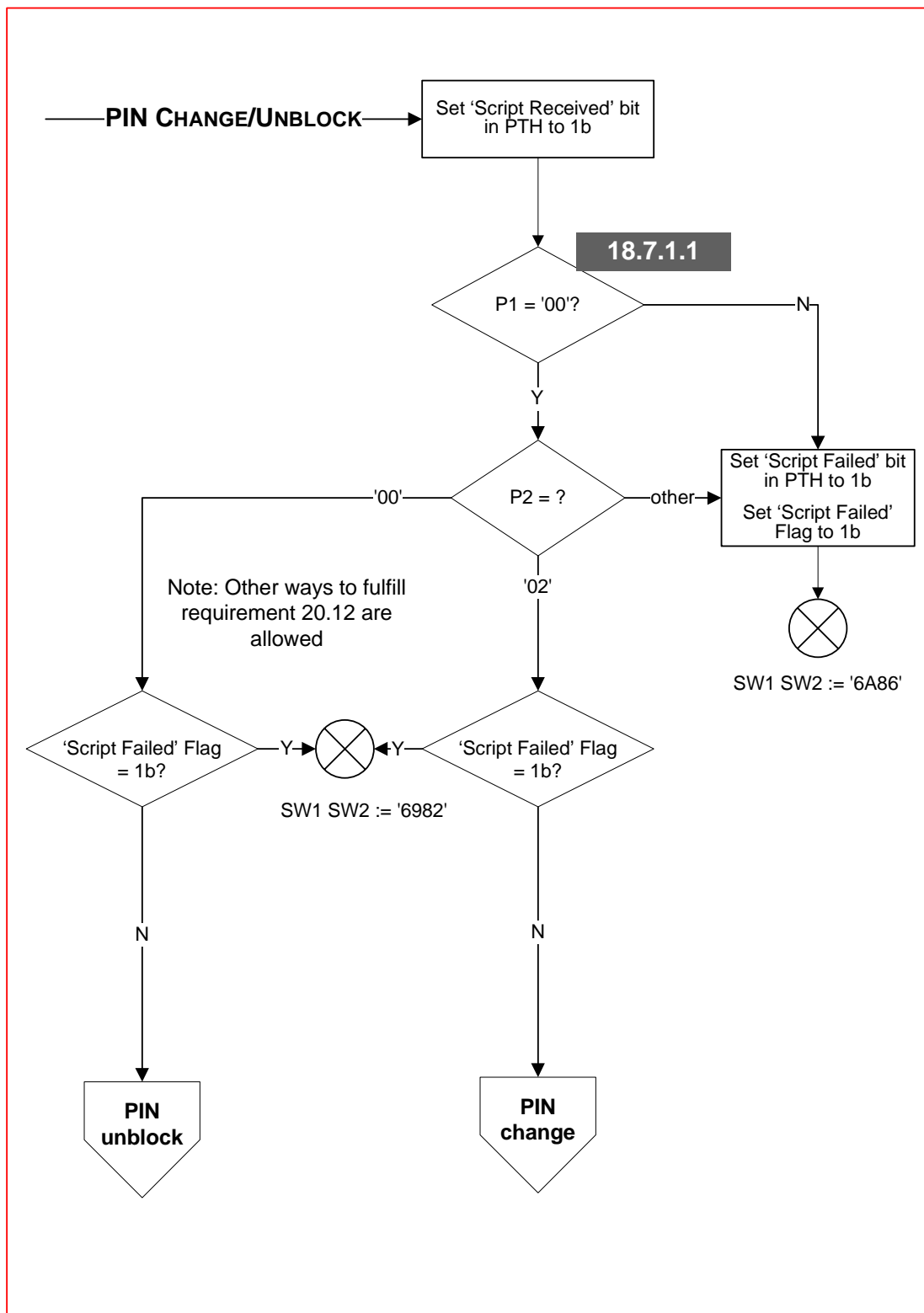
- *shall set the 'Script failed' bit in the PTH to the value 1b*
- *shall discontinue processing the PIN CHANGE/ UNBLOCK command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6988' (Incorrect secure messaging data objects)*

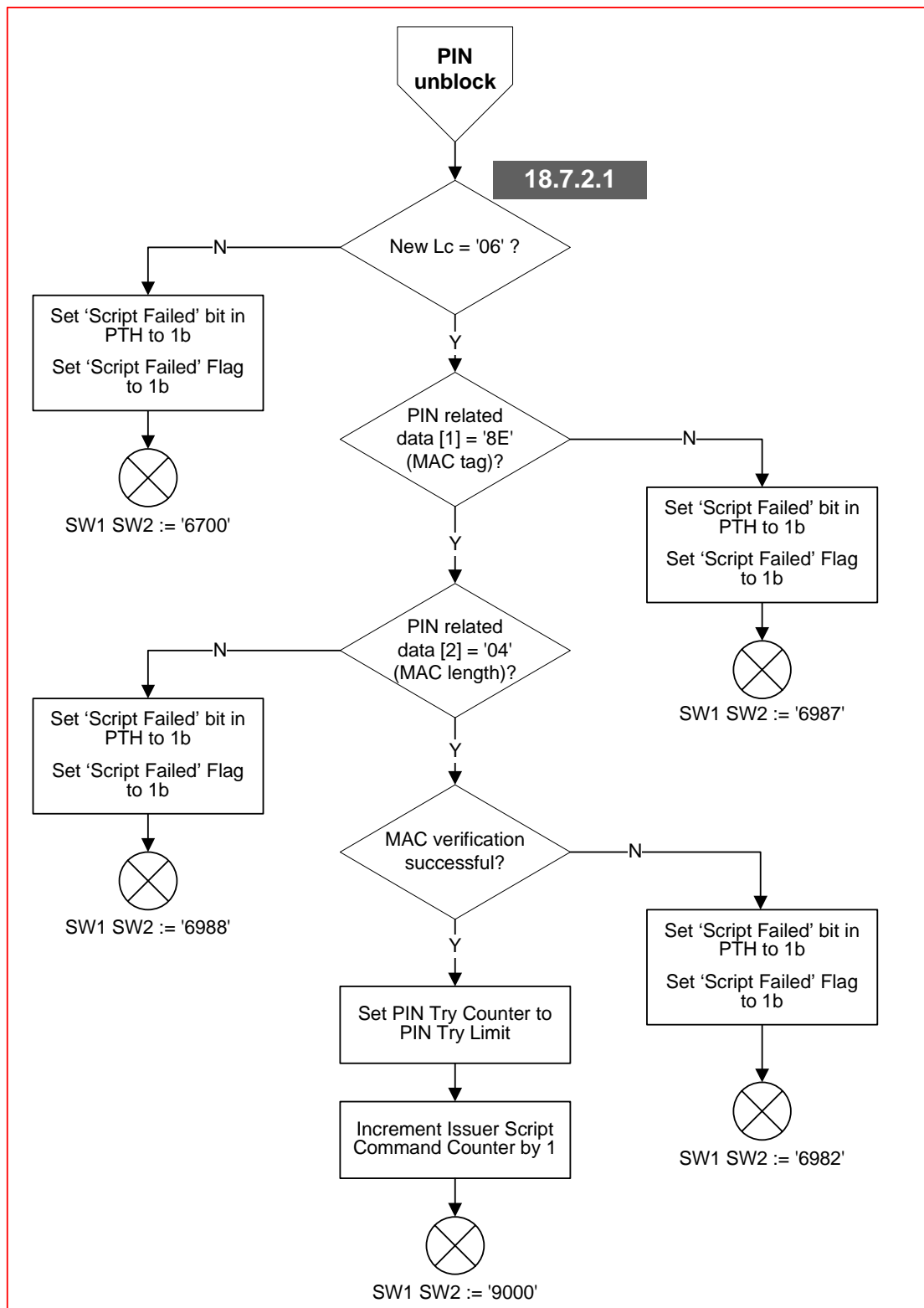
18.7.3 PIN CHANGE/UNBLOCK Flow

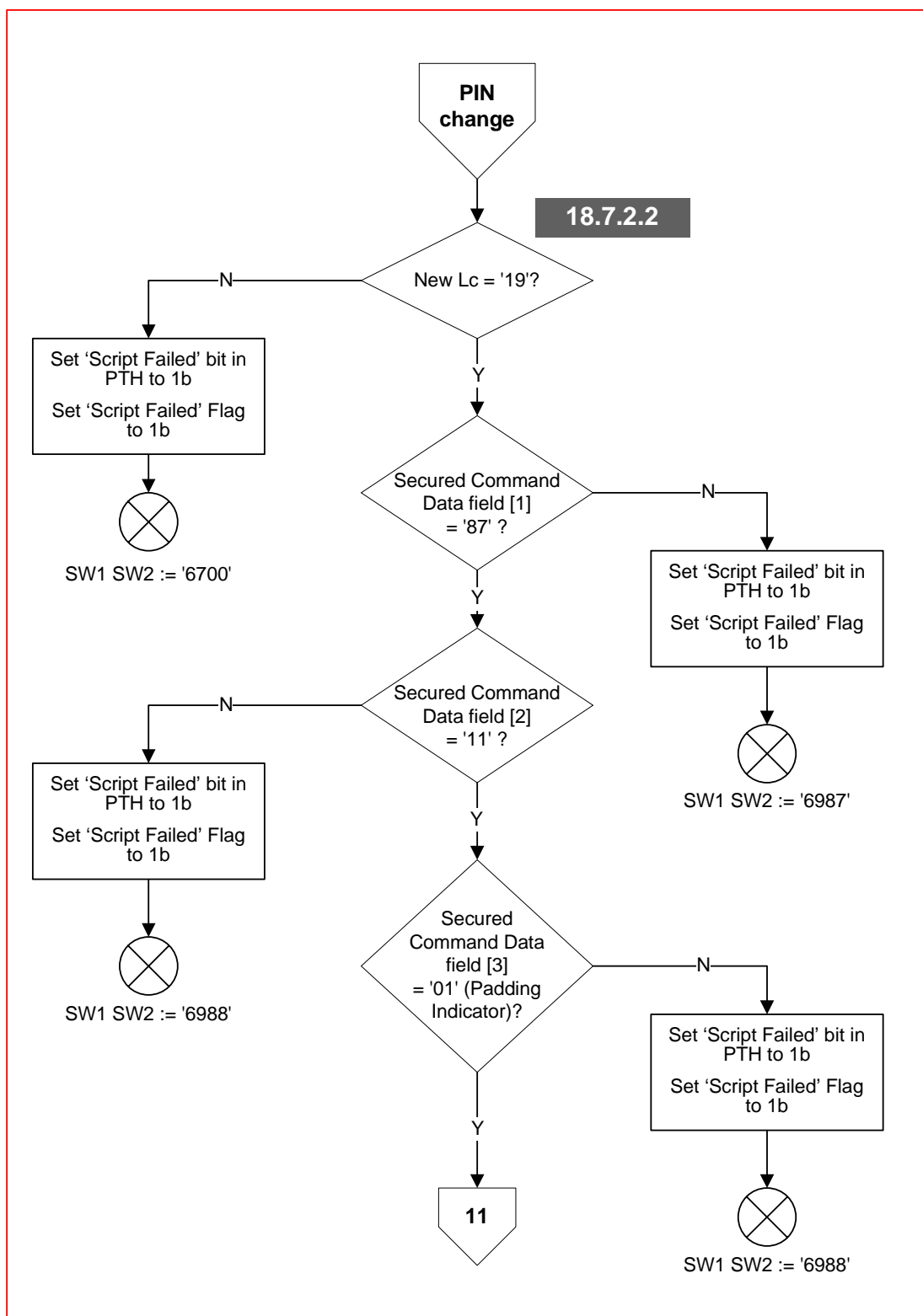
Figure 18-4 illustrates the flow of the PIN CHANGE/UNBLOCK command.

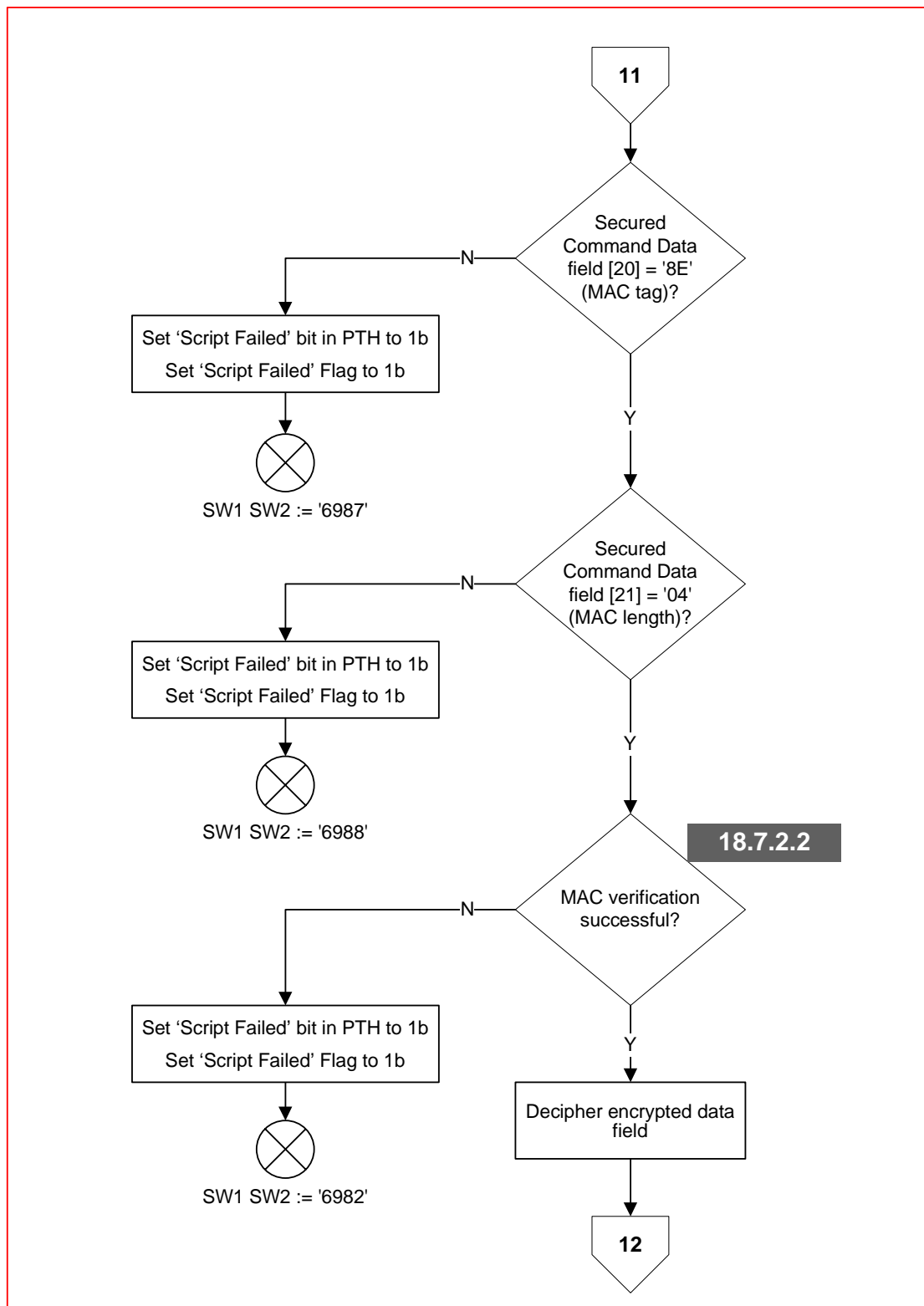
⁵ The Issuer Script Command Counter is for information only and is cyclic: '0F'+1='00'.

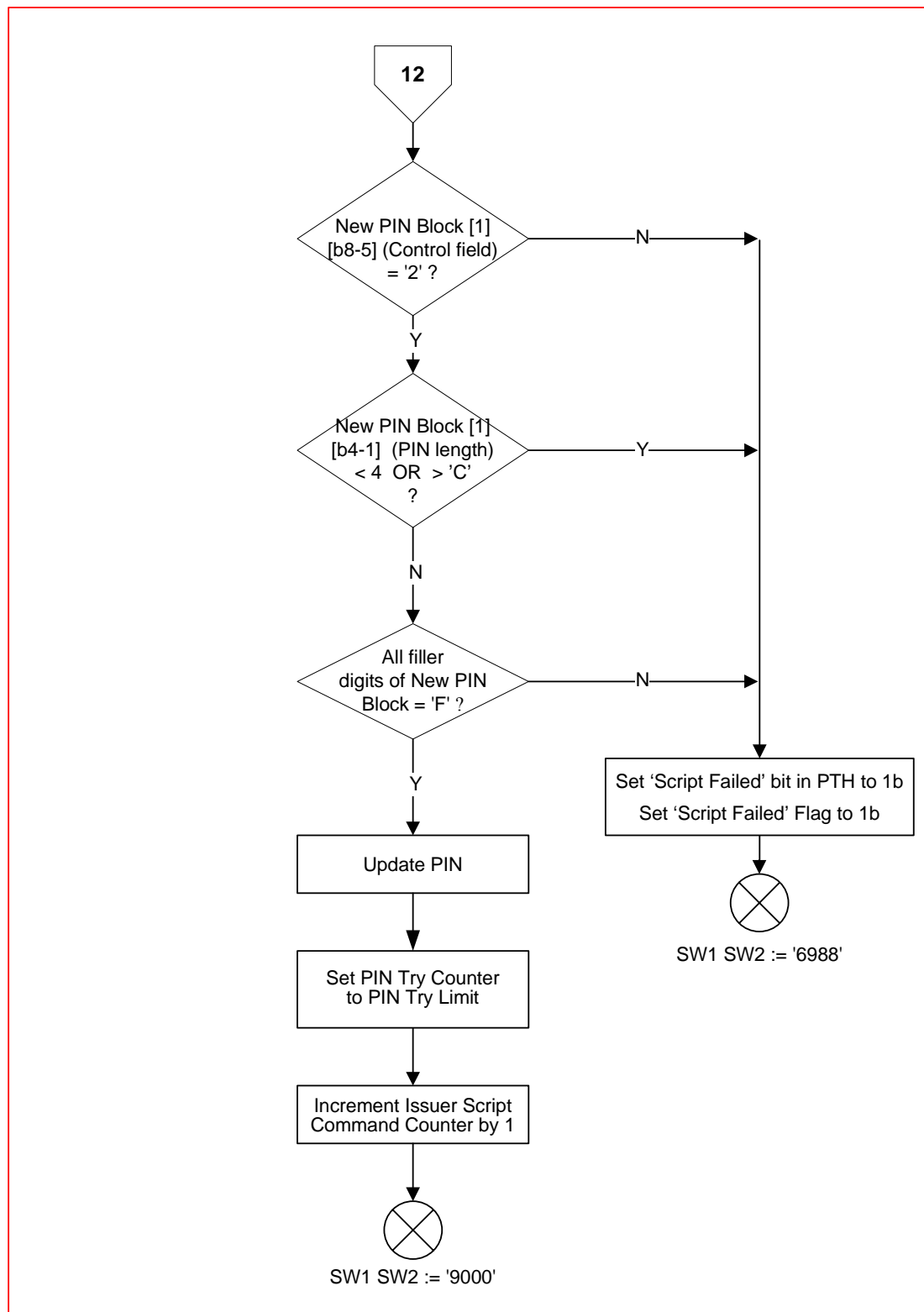
Figure 18-4: PIN CHANGE/UNBLOCK Flow











18.8 PUT DATA Command

The PUT DATA command allows specific data elements and templates in the card to be updated. A data element or template can be updated with this command only if it has a tag associated with it.

Req 18.30 (Data elements supported by PUT DATA):

Annex J, Table J-1 shows the only EMV-defined and CPA-defined application data elements and templates that may be updated using the PUT DATA command.

Req 18.31 (PUT DATA does not require filler bytes):

For template Tags, the PUT DATA command shall accept templates without filler bytes.

The PUT DATA command to a template Tag is allowed to contain filler bytes of value '00' in the command data.

See Annex G for an explanation of managing application resource data in templates for the PUT DATA command using a single template tag for each type of resource.

18.8.1 PUT DATA Command Coding

The P1 and P2 parameters of the PUT DATA command indicate the tag of the data which is to be updated.

Code	Value
CLA	'0C'
INS	'DA'
P1/P2	Tag
New Lc	Var.
Data	Secured Command Data Field
Le	not present

Table 18-6: PUT DATA Command Message

Req 18.32 (PUT DATA script command received):

The application shall set the 'Script received' bit in the PTH to the value 1b.

NOTE: If a PUT DATA command received before the second GENERATE AC command updates data elements that are used during processing of the second GENERATE AC command, then the updated values are used during processing of the second GENERATE AC command.

18.8.1.1 PUT DATA Command Format Validation

Req 18.33 (Check P1/P2 value for PUT DATA):

If P1/P2 does not contain a tag or template tag that can be updated using PUT DATA, then the card:

- *shall set the 'Script failed' bit in PTH to the value 1b,*
- *shall discontinue processing the PUT DATA command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6A86' (Incorrect Parameters P1 P2).*

Single byte tags are preceded with a leading '00' byte to fill P1/P2.

18.8.2 PUT DATA Processing

The format of the command data (Secured Command Data Field) for PUT DATA is shown in Figure 18-5.

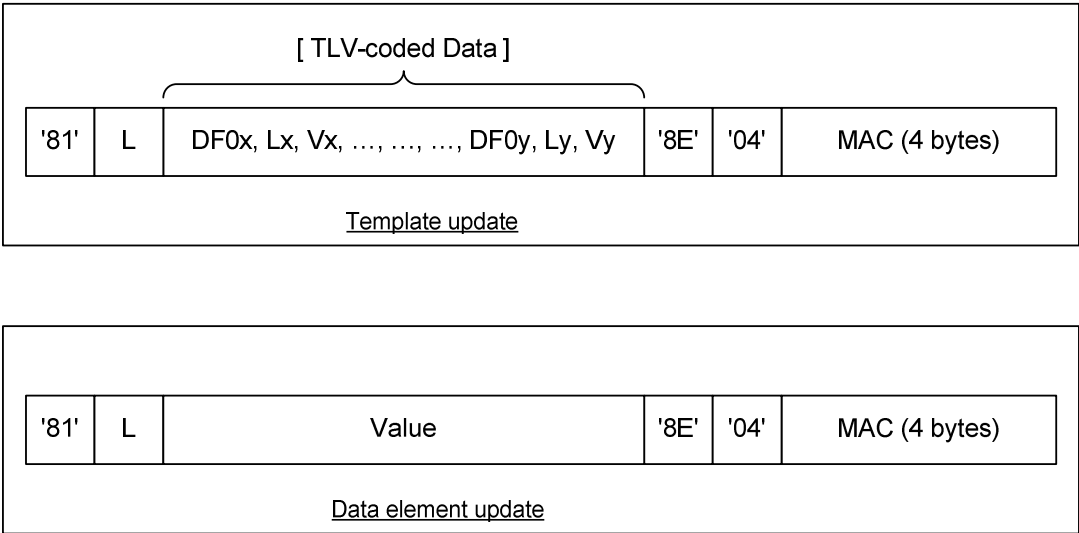


Figure 18-5: Command Data Format for PUT DATA

If P1/P2 for the PUT DATA command contains a template tag, the command data encapsulates a combination of TLV-coded data understood in the context of the template.

The PUT DATA command to a template tag updates individual TLV-coded data elements within the template. The command data field is not required to contain all the data elements in the template, only those that are added or modified. See Annex G for more information about updating a template using the PUT DATA command.

Req 18.34 (Check secure messaging template for PUT DATA):

If the first byte of Secured Command Data Field does not equal '81', then the application:

- shall set the 'Script failed' bit in the PTH to 1b,
- shall discontinue processing the Put Data command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6987' (Expected secure messaging data objects missing).

The application verifies the secure messaging format for the command data.

Req 18.35 (Check command data length and command data for one-byte length):

If L is coded on one byte, then:

- *If New Lc does not equal 8+L, then the card:*
 - *shall set the 'Script failed' bit in the PTH to the value 1b*
 - *shall discontinue processing the PUT DATA command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6700' (Wrong Length).*
- *If the value of byte (L + 3) of Secured Command Data Field has a value other than '8E' (MAC tag), then the card:*
 - *shall set the 'Script failed' bit in the PTH to the value 1b*
 - *shall discontinue processing the PUT DATA command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6987' (Expected secure messaging data objects missing).*
- *If the value of byte (L + 4) of Secured Command Data Field has a value other than '04' (MAC length), then the card:*
 - *shall set the 'Script failed' bit in the PTH to the value 1b*
 - *shall discontinue processing the PUT DATA command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6988' (Incorrect secure messaging data objects).*

Req 18.36 (Check command data length and command data for two-byte length):

If L is coded on two bytes, then:

- *If New Lc does not equal 9+L, then the card:*
 - *shall set the 'Script failed' bit in the PTH to the value 1b*
 - *shall discontinue processing the PUT DATA command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6700' (Wrong Length).*
- *If the value of byte (L + 4) of Secured Command Data Field has a value other than '8E' (MAC tag), then the card:*
 - *shall set the 'Script failed' bit in the PTH to the value 1b*
 - *shall discontinue processing the PUT DATA command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6987' (Expected secure messaging data objects missing).*
- *If the value of byte (L + 5) of Secured Command Data Field has a value other than '04' (MAC length), then the card:*
 - *shall set the 'Script failed' bit in the PTH to the value 1b*
 - *shall discontinue processing the PUT DATA command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6988' (Incorrect secure messaging data objects).*

Otherwise, the application shall continue processing with verifying the MAC.

The application verifies the MAC.

Req 18.37 (MAC verification failed for PUT DATA):

If the MAC verification is not successful, then the card:

- *shall set the 'Script failed' bit in the PTH to the value 1b,*
- *shall discontinue processing the PUT DATA command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6982' (Security status not satisfied)*

If the MAC verification is successful, then the card validates that the command data has a valid tag and length for updating application data.

The application allocates space for data elements that can be configured using the PUT DATA command, and cannot process the request to update the data element if the update would exceed the space allocated for the data element.

Req 18.38 (PUT DATA has a template tag in P1/P2):

If P1/P2 contains a template tag, then:

- *The card shall extract each TLV-coded data element within the command data.*
- *For each extracted TLV-coded data element:*
 - *If the tag is not supported within the template, then the card:*
 - *shall set the 'Script failed' bit in the PTH to the value 1b*
 - *shall discontinue processing the PUT DATA command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6A88' (Referenced data not found).*
 - *If the length in the template data is greater than the reserved length for the data element, then the card:*
 - *shall set the 'Script failed' bit in the PTH to the value 1b*
 - *shall discontinue processing the PUT DATA command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6700' (Wrong Length).*
- *If the tag and length are supported for every data element within the template, then the card shall:*
 - *update all the data elements*
 - *increment by one the Issuer Script Command Counter*
 - *respond with SW1 SW2 = '9000'*

NOTE: The card may check the length of fixed length data before updating the data element.

Req 18.39 (PUT DATA has a data element tag in P1/P2):

If P1/P2 contains a data element tag:

- *If L is less than or equal to the reserved length for the data element, then the card shall:*
 - *update the data element*
 - *increment by one the Issuer Script Command Counter*
 - *respond with SW1 SW2 = '9000'*
- *Otherwise the card:*
 - *shall set the 'Script failed' bit in the PTH to the value 1b*
 - *shall discontinue processing the PUT DATA command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6700' (Wrong Length).*

NOTE: The card may check the length of fixed length data before updating the data element.

The PUT DATA command does not require filler bytes in the template data. However, because the command has a length for the command data in addition to the length for each TLV-coded data element, the issuer is allowed to include filler bytes in the command data.

NOTE: EMV uses the value '00' for filler bytes.

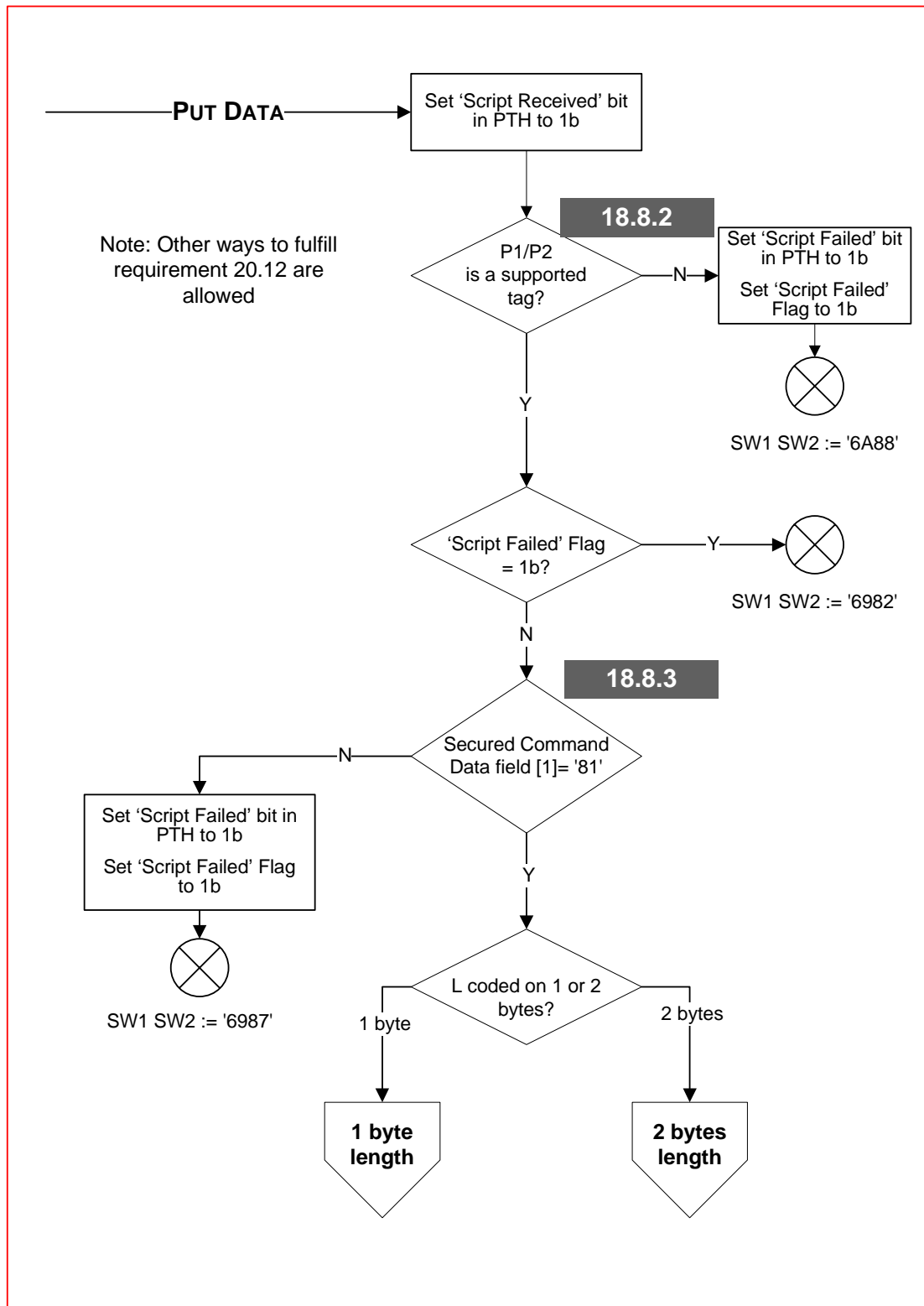
Req 18.40 (Filler bytes not required in PUT DATA command):

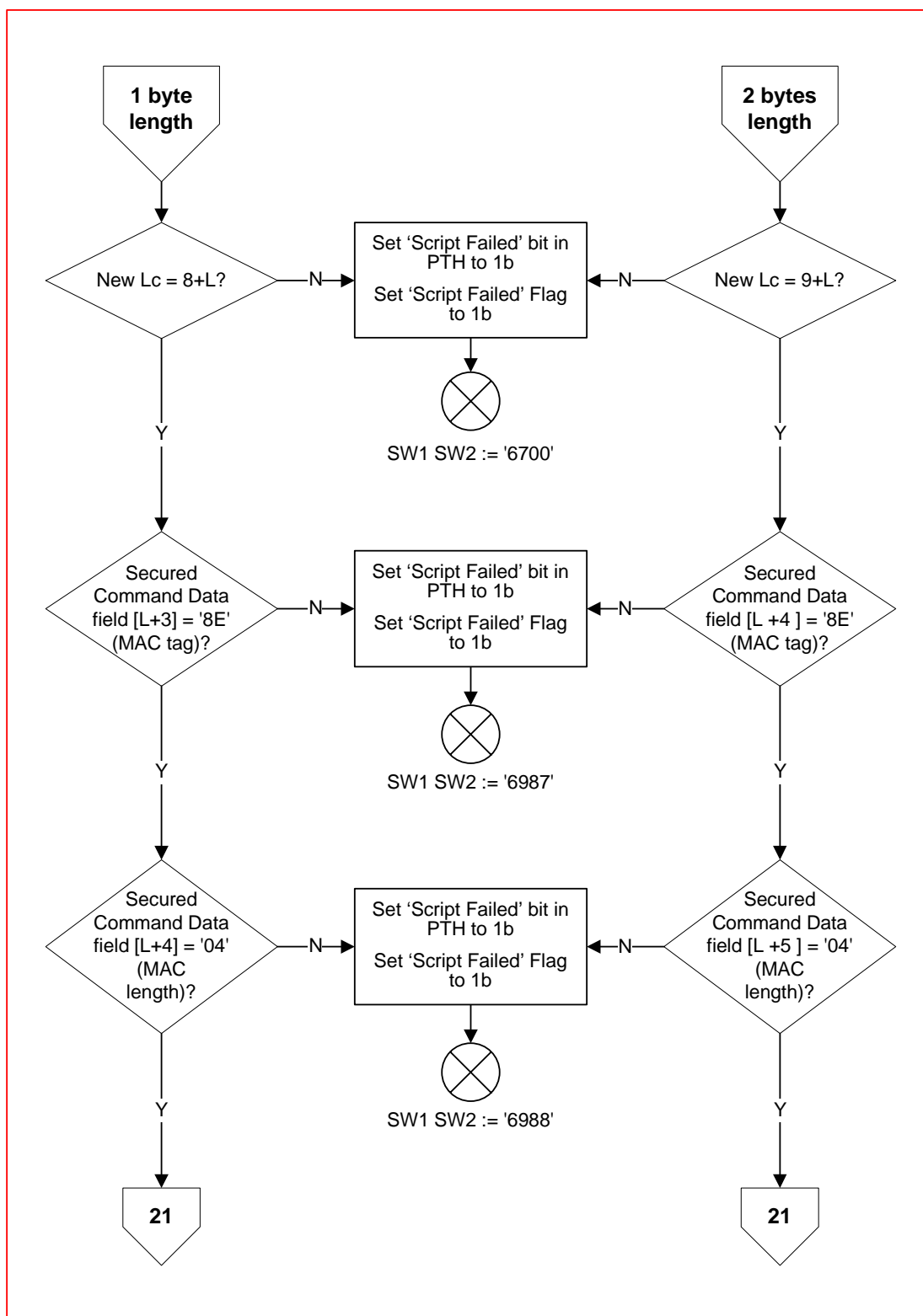
Filler bytes shall not be required to be sent in the template for a PUT DATA command.

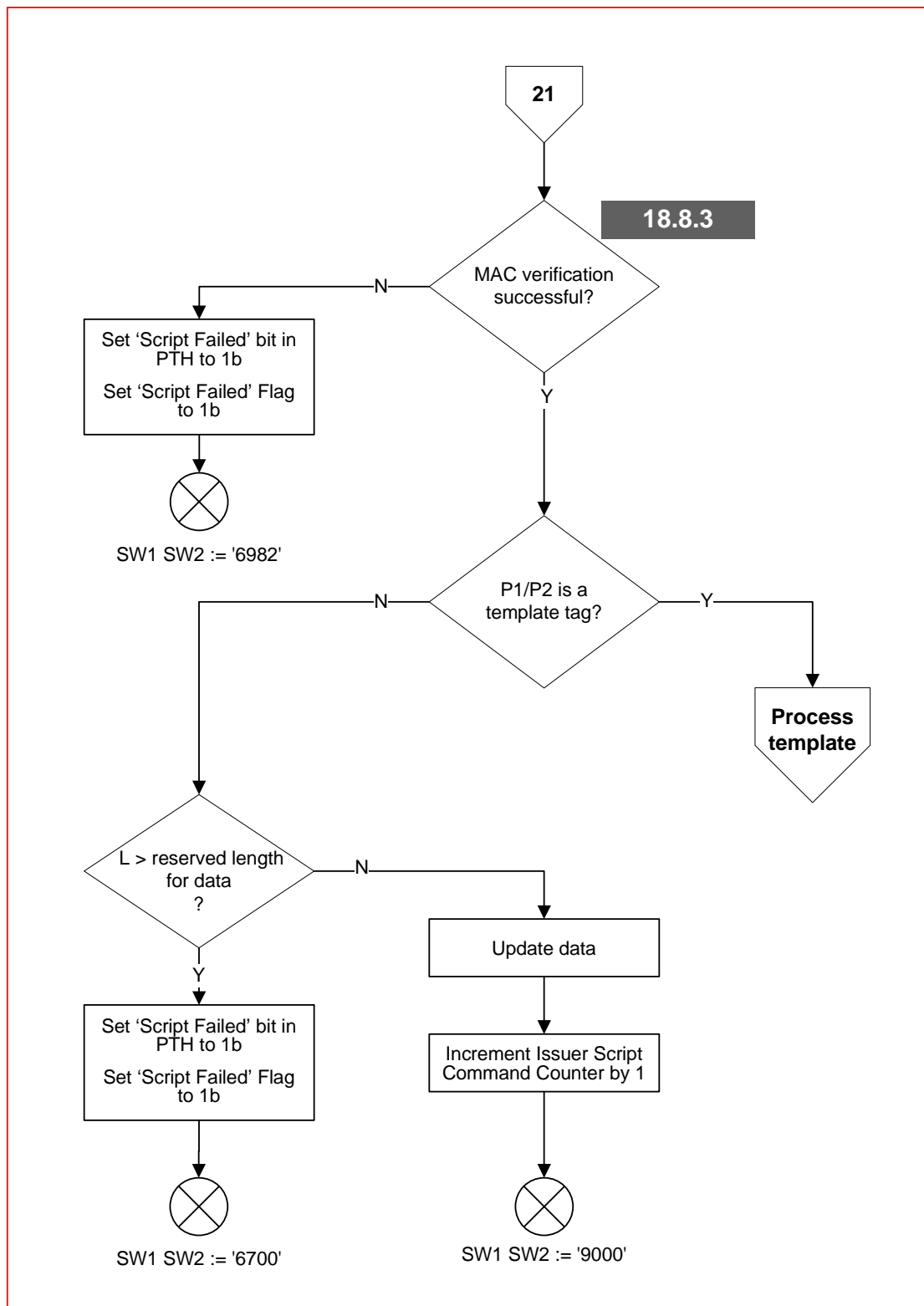
18.8.3 PUT DATA Flow

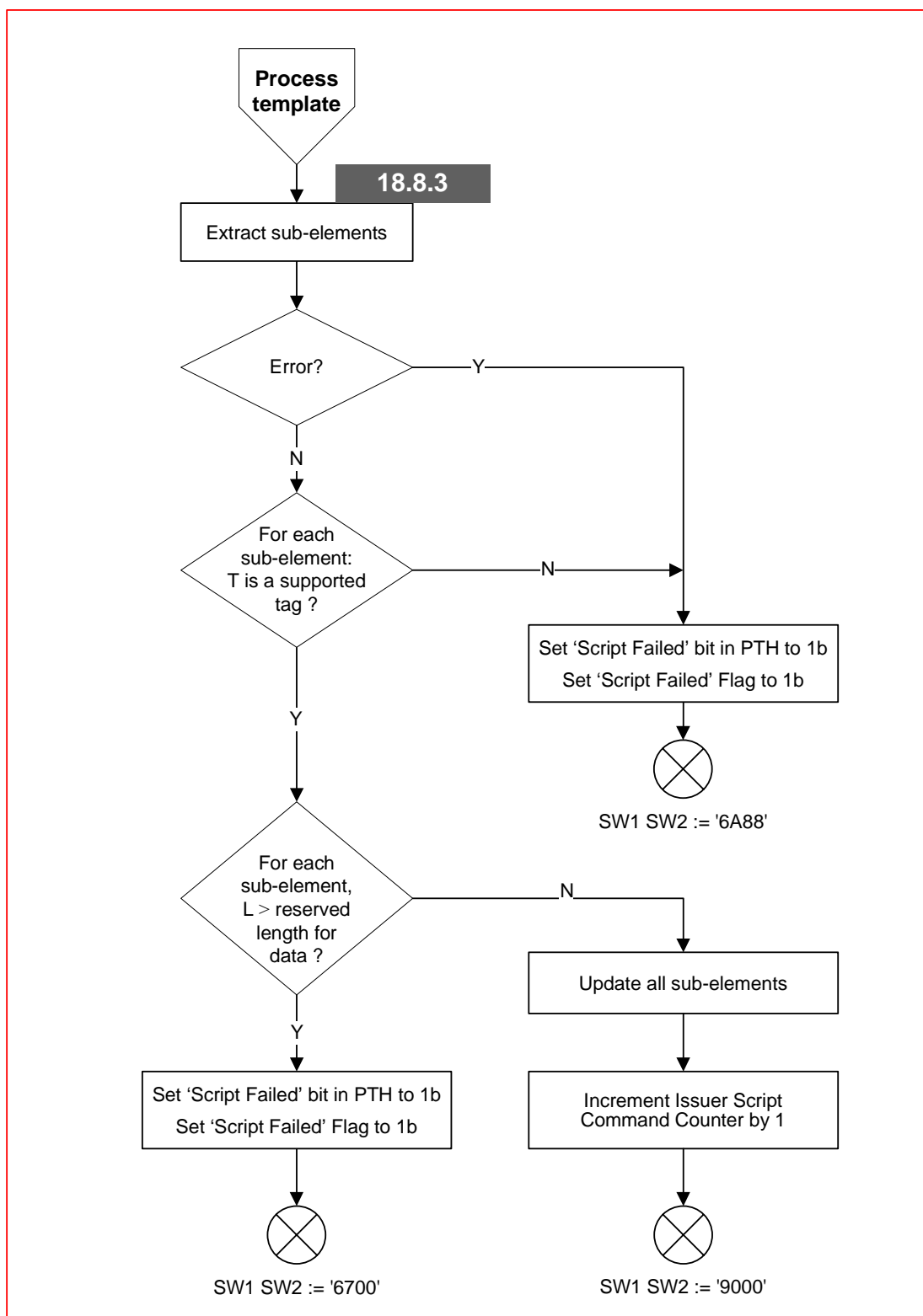
Figure 18-6 illustrates the flow of the PUT DATA command.

Figure 18-6: PUT DATA Flow









18.9 UPDATE RECORD Command

The UPDATE RECORD command is used to update a record in a file with the data provided in the command data field.

18.9.1 UPDATE RECORD Command Coding

The P1 parameter of the UPDATE RECORD command indicates which record number within the SFI referenced in P2 is to be updated. The P2 parameter identifies the SFI of the file containing the record to be updated, and indicates that P1 contains the record number. The data portion of the command contains the data for the record.

Code	Value
CLA	'0C'
INS	'DC'
P1	Record Number
P2	Reference Control Parameter
New Lc	Var.
Data	Record Related Data
Le	not present

Table 18-7: UPDATE RECORD Command Message

Table 18-8 shows the coding of the Reference Control Parameter in P2:

b8	b7	b6	b5	b4	b3	b2	b1	meaning
x	x	x	x	x				SFI
					x	x	x	
					1	0	0	P1 is a record number.

Table 18-8: Reference Control Parameter coding for UPDATE RECORD

Req 18.41 (UPDATE RECORD script command received):

The application shall set the 'Script received' bit in the PTH to the value 1b.

18.9.1.1 UPDATE RECORD Command Format Validation

Req 18.42 (Check P2 value for UPDATE RECORD):

If bits b3 through b1 of the P2 parameter have a value other than 100b, then the card:

- *shall set the 'Script failed' bit in PTH to the value 1b,*
- *shall discontinue processing the UPDATE RECORD command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6A86' (Incorrect Parameters P1 P2).*

The files (and the records within those files) that are supported by the application are defined at personalisation/pre-personalisation. The application checks whether the file referenced in P2 is supported for the UPDATE RECORD command.

All supported files with SFI in the range 1 to 10 are EMV files and may be updated with the UPDATE RECORD command.

The Record Related Data for updates to SFI in the range 1 to 10 contains the template tag '70'.

NOTE: This is a requirement on the issuer formatting the UPDATE RECORD command. The card does not verify that this requirement is met.

All supported files with SFI in the range 11 to 20 are payment system-specific files.

If the card supports VLP, the VLP Data referenced in SFI 11 is updateable with the UPDATE RECORD command.

Req 18.43 (UPDATE RECORD supported for VLP record):

If the card supports VLP, then SFI 11, record 1, shall be updateable with the UPDATE RECORD.

Other payment system-specific files may be used for additional functionality, but such use is beyond the scope of this specification. Support for these files by the UPDATE RECORD command is also beyond the scope of this specification.

The Transaction Log referenced in an SFI in the range 21 to 30 is not updateable with the UPDATE RECORD command.

Req 18.44 (UPDATE RECORD not allowed to Transaction Log):

If bits b8 through b4 of the P2 parameter identify the Transaction Log, then the card:

- *shall set the 'Script failed' bit in PTH to the value 1b,*
- *shall discontinue processing the UPDATE RECORD command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6985' (Command not allowed; conditions of use not satisfied).*

Req 18.45 (UPDATE RECORD supported for Profile Selection Entries):

The file containing the Profile Selection File shall be updateable with the UPDATE RECORD command.

The use of files that are not EMV files, payment system-specific files, the Transaction Log file, nor the file containing the Profile Selection File is permitted as additional functionality (for instance, issuer-specific files), but is beyond the scope of this specification. Support for these files by the UPDATE RECORD command is also beyond the scope of this specification.

Req 18.46 (UPDATE RECORD to unknown SFI):

If bits b8 through b4 of the P2 parameter identify an SFI unknown to the application, then the card:

- shall set the 'Script failed' bit in PTH to the value 1b,
- shall discontinue processing the command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6A82' (Incorrect Parameters P1 P2; file not found).

Req 18.47 (UPDATE RECORD to unknown record):

If the P1 parameter indicates a record number not supported by the application in the file identified in P2, then the card:

- shall set the 'Script failed' bit in PTH to the value 1b,
- shall discontinue processing the command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6A83' (Incorrect Parameters P1 P2; record not found).

18.9.2 UPDATE RECORD Processing

The format of the command data (Record Related Data) for UPDATE RECORD is shown in Figure 18-7.

'81'	L	Record Value	'8E'	'04'	MAC (4 bytes)
------	---	--------------	------	------	---------------

Figure 18-7: Command Data Format for UPDATE RECORD

Req 18.48 (Check secure messaging template for UPDATE RECORD):

If first byte of Secured Command Data Field does not equal '81', then the application:

- shall set the 'Script failed' bit in the PTH to 1b,
- shall discontinue processing the UPDATE RECORD command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6987' (Expected secure messaging data objects missing).

The application verifies the secure messaging format for the command data.

Req 18.49 (Check command data length for UPDATE RECORD):

If **either** of the following is true:

- *L is coded on one byte and New Lc does not equal 8+L*
- **or** *L is coded on two bytes and New Lc does not equal 9+L*

then the card:

- *shall set the 'Script failed' bit in the PTH to the value 1b*
- *shall discontinue processing the UPDATE RECORD command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6700' (Wrong Length).*

Req 18.50 (Check MAC tag for UPDATE RECORD):

If **either** of the following is true:

- *L is coded on one byte and the value of byte (L + 3) of Record Related Data has a value other than '8E' (MAC tag)*
- **or** *L is coded on two bytes and the value of byte (L + 4) of Record Related Data has a value other than '8E' (MAC tag)*

then the card:

- *shall set the 'Script failed' bit in the PTH to the value 1b*
- *shall discontinue processing the UPDATE RECORD command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6987' (Expected secure messaging data objects missing).*

Req 18.51 (Check MAC length for UPDATE RECORD):

If **either** of the following is true:

- *L is coded on one byte and the value of byte (L + 4) of Record Related Data has a value other than '04' (MAC length)*
- **or** *L is coded on two bytes and the value of byte (L + 5) of Record Related Data has a value other than '04' (MAC length)*

then the card:

- *shall set the 'Script failed' bit in the PTH to the value 1b*
- *shall discontinue processing the UPDATE RECORD command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6988' (Incorrect secure messaging data objects).*

Otherwise, the application shall continue processing with verifying the MAC.

When the card is issued, an area of memory is reserved for each record. The reserved length for a record represents the size allocated to a record and must be greater than or equal to the size needed for personalisation of the record. The reserved length for a record may vary between records. The reserved length for each record is a record attribute that is not modified by the UPDATE RECORD command. The memory allocated to a record remains available to store a new value using the UPDATE RECORD command. The method for reserving the length for a record is beyond the scope of this specification and is left to the implementation.

If the length of the new value for a record is less than or equal to the reserved length for a record, the UPDATE RECORD replaces the current record with a new record, even if the actual size of those records differ. Partial update of a record is not supported. The length of the record data is updated with the length of the new data, but the reserved length for the record has not changed.

The new record value is the value that would be returned in response to the READ RECORD command.

Records in files with an SFI in the range 1 to 10 must follow the '70' template. As a consequence, the new record value must follow the '70' template. However, the application does not interpret this value. It is the responsibility of the issuer to correctly format the new record value when generating the data for the issuer script command.

The UPDATE RECORD command does not require filler bytes in the command data.

For records containing the Profile Selection Entries; because the UPDATE RECORD command has a length for the command data in addition to the length of the Profile Selection Entry contained in the record, the issuer is allowed to add filler bytes to the end of the Profile Selection Entry in a record. To ensure that the Profile Selection Entry can be correctly processed by the application, if filler bytes are added, they should be added to the end of the Profile Selection Entry.

NOTE: EMV uses the value '00' for filler bytes.

Req 18.52 (Filler bytes not required in UPDATE RECORD to Profile Selection Entry):

The UPDATE RECORD command shall accept Profile Selection Entry records without filler bytes.

The UPDATE RECORD command to a Profile Selection Entry is allowed to contain trailing filler bytes of value '00'.

Req 18.53 (Update data is too long for the record):

If the length of the update data for the record is greater than the reserved length for the record, then the application shall discontinue processing the Update Record command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6700' (Wrong length).

The application verifies the MAC.

Req 18.54 (MAC verification for UPDATE RECORD):

If the MAC verification is not successful, then the card:

- *shall set the 'Script failed' bit in the PTH to the value 1b*
- *shall discontinue processing the UPDATE RECORD command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6982' (Security status not satisfied)*

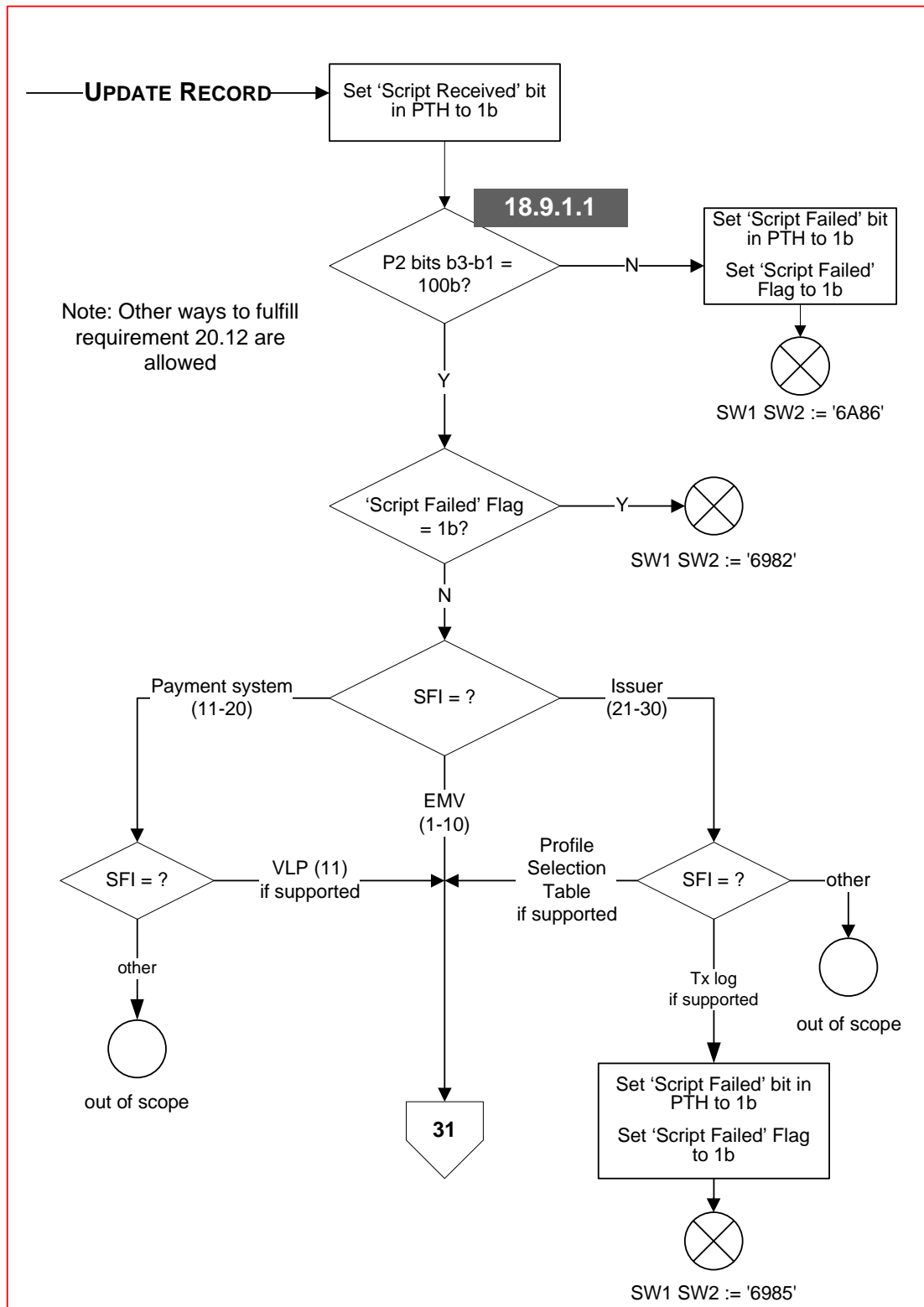
If the MAC verification is successful, then the card shall:

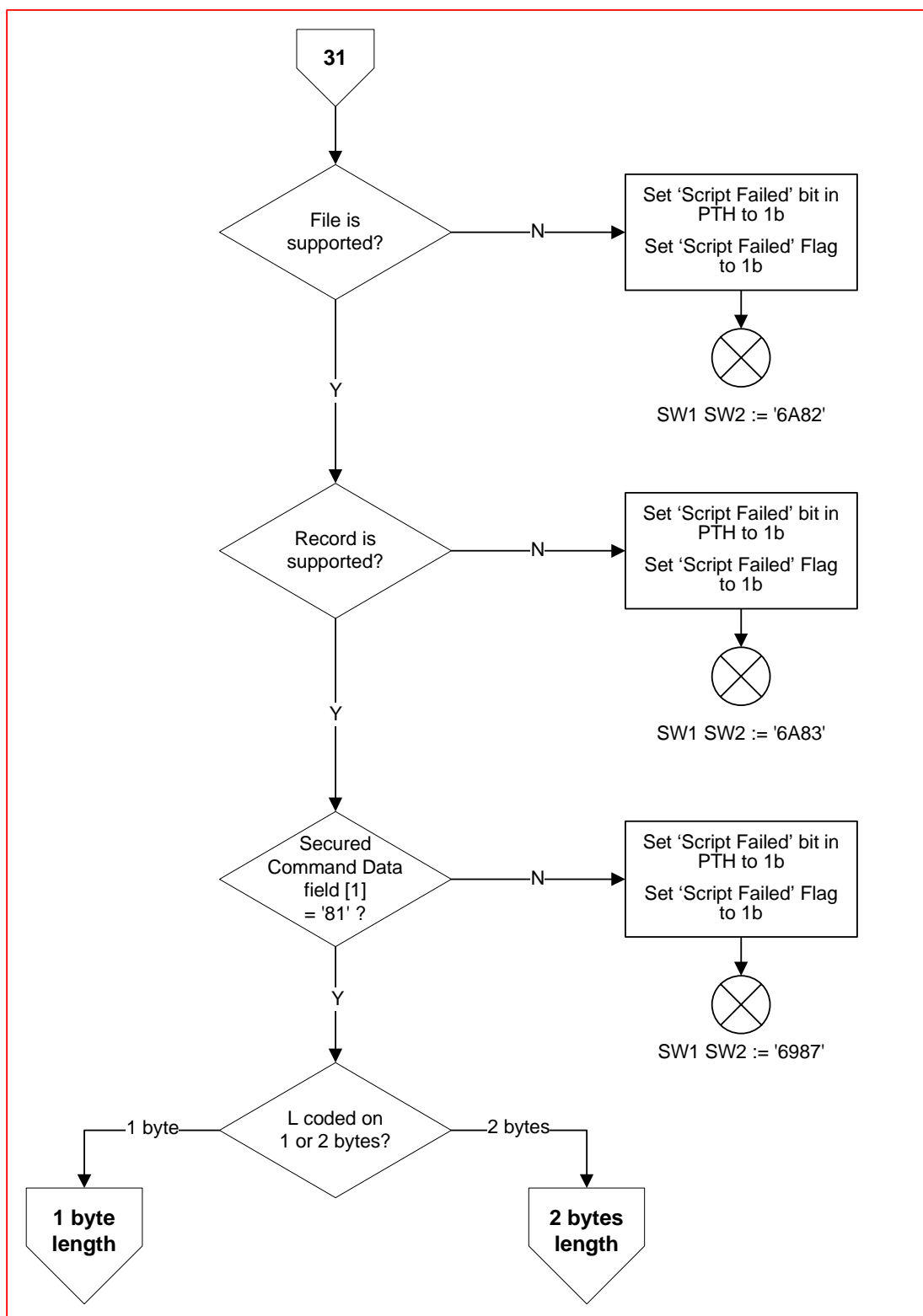
- *update the record*
- *increment by one the Issuer Script Command Counter*
- *respond with SW1 SW2 = '9000'.*

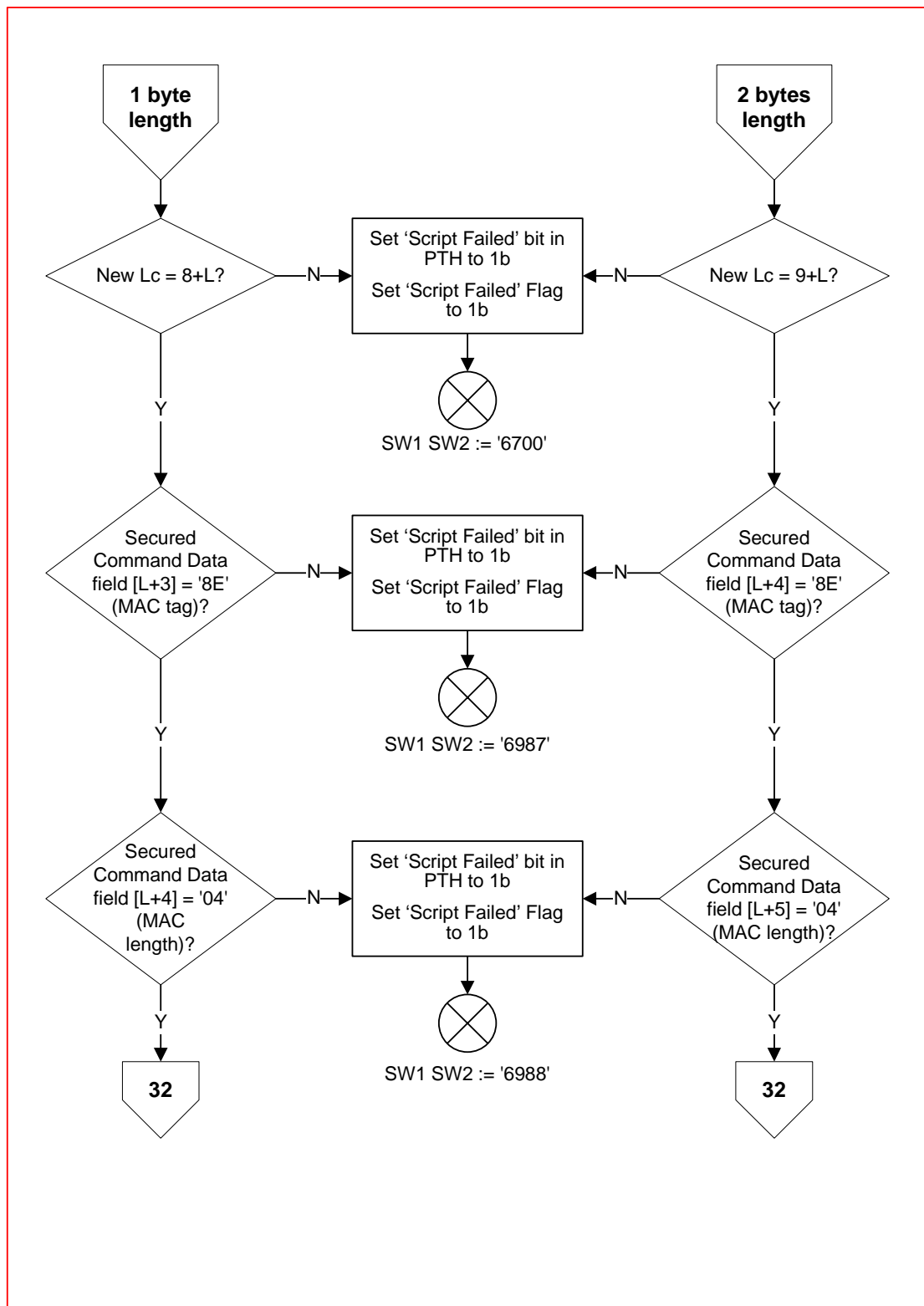
18.9.3 UPDATE RECORD Flow

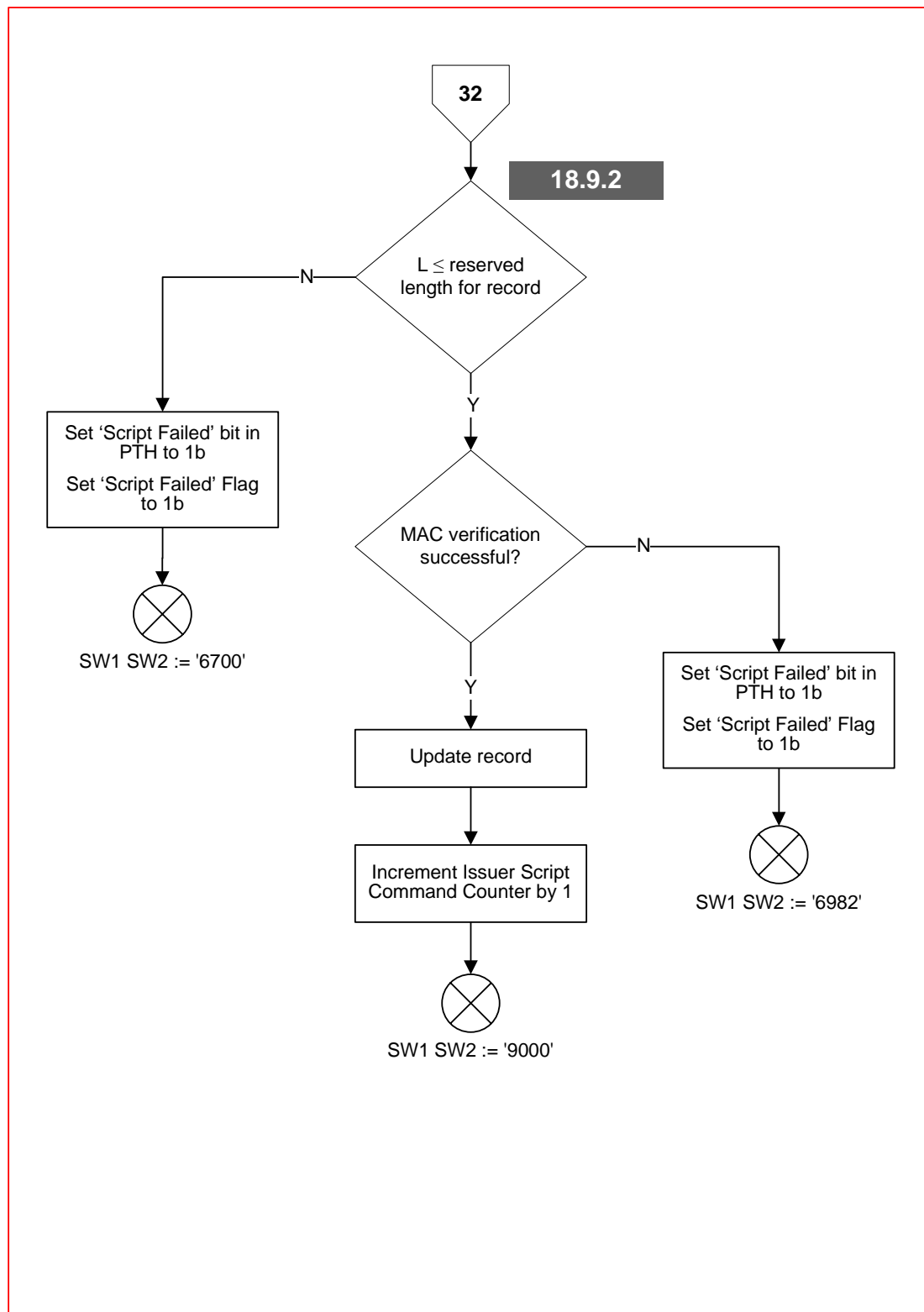
Figure 18-8 illustrates the flow of the UPDATE RECORD command.

Figure 18-8: UPDATE RECORD Flow









Part IV

Additional Topics

19 Additional Functions

This section is organised as follows:

- 19.1 Overview
- 19.2 Requirements for Additional Functionality
- 19.3 Examples of Additional Functionality
 - 19.3.1 Additional Accumulators, Counters, and Cyclic Accumulators
 - 19.3.2 Issuer-Discretionary Bits – CVR, ADR, and CIACs
 - 19.3.3 Issuer-Discretionary Bits – Internal Application Data
 - 19.3.4 Issuer-Discretionary Bytes – Issuer Application Data
 - 19.3.5 Issuer-Discretionary Bytes – Issuer Authentication Data
 - 19.3.6 Additional Issuer Script Commands
 - 19.3.7 Support for Other MAC Lengths

19.1 Overview

CPA specifies a core set of functionalities that issuers can rely on having available in every implementation of CPA “off-the-shelf”. EMVCo will specify the type approval requirements for this core functionality.

It is at the discretion of issuers to choose an implementation of CPA that may also include additional functionality. This section describes the requirements associated with additional functionality, and examples of how such additional functionality could be added to the core functionality described in this specification.

CPA includes issuer-discretionary portions in select data elements that may be used to support additional functionalities (for example, in the ADR, Application Control, CIACs, and CVR). CPA also allows extension of select data elements (for example, the Profile Control) to support additional functionalities.

19.2 Requirements for Additional Functionality

In order to enable type approval of an implementation of CPA that includes additional functionality, enhancements to the CPA are allowed under the following conditions:

Req 19.1 (configuring additional functionality to conform to CPA):

All requirements in the CPA specification (except items that are implementer-options) shall be included in the application, and it shall be possible to configure the application to conform to the CPA specification.

Req 19.2 (CPA compliance related to additional functionality):

An application is considered non-CPA compliant when additional features are configured in a way that causes the application to no longer conform to the CPA specifications.

Additional functionality that does not impact CPA behaviour is permitted.
Additional functionality is not tested as part of CPA type approval.

NOTE: Bits shown in Annex L: Data Dictionary as RFU are reserved for future use by EMV and are not to be used for additional functionality.

19.3 Examples of Additional Functionality

19.3.1 Additional Accumulators, Counters, and Cyclic Accumulators

CPA requires a minimum set of two accumulators (and the optional VLP Available Funds), three counters and two cyclic accumulators that must be implemented in CPA for issuers to use at their discretion. The following extensions to CPA would support additional accumulators, counters and cyclic accumulators.

- Extend the length of each Profile Control x data element to n bytes (n>8). Each nibble in bytes 9 through n would be assigned to designate a Profile Control ID for each additional accumulator, counter, or cyclic accumulator. For example, with one additional accumulator and one additional counter:
 - byte 9, bits b8-5: Accumulator Profile Control ID for Accumulator 3
 - byte 9, bits b4-1: Counter Profile Control ID for Counter 4
- For each additional Accumulator x, add:
 - an Accumulator x Control (template 'BF32')
 - an Accumulator x Data (template 'BF30')
- For each additional Counter x, add:
 - a Counter x Control (template 'BF37')
 - a Counter x Data (template 'BF35')
- For each additional Cyclic Accumulator x, add a Cyclic Accumulator x Control data element (template 'BF3A').
- Use the following bits in the ADR and CIACs for card risk management associated with the additional accumulators, counters and cyclic accumulators:
 - Additional Accumulator Lower Limit Exceeded
 - Additional Accumulator Upper Limit Exceeded
 - Additional Counter Lower Limit Exceeded
 - Additional Counter Upper Limit Exceeded
 - Additional Cyclic Accumulator Limit Exceeded

19.3.2 Issuer-Discretionary Bits – CVR, ADR, and CIACs

The issuer-discretionary bits in the CVR, ADR, and CIACs could be used for additional functionality that might be used in Card Risk Management. The CVR and ADR bits could be set as a result of the additional functionality in an implementation. The CVR bits would be used to indicate results of the additional functionality to the issuer. The ADR bits would be set so that the corresponding bits in the CIACs could be personalised to allow the application to include the results of the additional functionality in the decision whether to decline the transaction offline or to send the transaction online.

19.3.3 Issuer-Discretionary Bits – Internal Application Data

The issuer-discretionary bits in the Application Control and Issuer-Options Profile Controls could be used for additional functionality. In order to ensure CPA card type approval is possible with cards containing additional functionality, setting the Issuer-Discretionary bits to the value zero should disable any additional functionality that changes card behaviour.

19.3.4 Issuer-Discretionary Bytes – Issuer Application Data

The issuer-discretionary bytes in the Issuer Application Data may be set to the personalised default value, may carry the information described in the Annex L: Data Dictionary for bytes 19-32, or could be used to send the results of additional functionality for any Profile other than Profile '7E' (which requires these bytes to be set to zero).

19.3.5 Issuer-Discretionary Bytes – Issuer Authentication Data

The CPA card may support up to eight bytes of Proprietary Authentication Data in the Issuer Authentication Data. The additional functionality that uses the Proprietary Authentication Data is beyond the scope of CPA.

CPA requires the card to be able to accept 8-bytes of Issuer Authentication Data. The application may also support up to 16 bytes of Issuer Authentication Data (which includes up to 8 bytes of Proprietary Authentication Data). The specification of Issuer Authentication processing in section 17 describes CPA applications that support only 8 bytes of Issuer Authentication Data. When the CPA application also supports Proprietary Authentication Data:

- If the 'Proprietary Authentication Data Included' bit in the CSU has the value 0b, then the application validates the Issuer Authentication Data with a Proprietary Authentication Data of length zero bytes.
- If the 'Proprietary Authentication Data Included' bit in the CSU has the value 1b, then:
 - the validation of the ARPC should be modified to include the Proprietary Authentication Data
 - the length for Issuer Authentication Data should be changed to include the length of Proprietary Authentication Data
- Additional functionality using the PAD is beyond the scope of CPA.

19.3.6 Additional Issuer Script Commands

CPA requires a minimum set of issuer-to-card script commands that must be implemented either in the card or application for issuers to use at their discretion. Additional commands may be supported.

If a card supports additional script commands for use with CPA, the commands must be implemented such that the CVR and ADR bits for script indicators will be set for the current and subsequent transactions as if the requirements of section 18.5.4 were implemented for the command

The application should remain in the current state (see section 6.2) after successful processing of an additional issuer script command.

19.3.7 Support for Other MAC Lengths

CPA requires the card to be able to accept 4-byte MACs. The application may also support longer MACs (5-byte to 8-byte MACs) as specified for CCD-compliant applications in *EMV Book 2*, section 9. The specification of script commands in sections 18.6 through 18.9 describes CPA applications that support only 4-byte MACs. When the CPA application also supports longer MACs, the following requirement modifications apply:

- Checks on New Lc are modified to allow for longer command data
- Checks on length of data element '8E' are modified to allow for longer MACs.

20 Security and Key Management

This section is organised as follows:

- 20.1 Reference PIN Protection
- 20.2 Key Protection
- 20.3 Secure Messaging
- 20.4 Security Counters
- 20.5 Other Data Requirements

The security requirements in this section must be met for all implementations of CPA, but the method for ensuring the requirement (for example, whether the requirement is addressed by the application or by the platform, and whether or not counters are used to provide the security) is at the discretion of the implementer.

NOTE: The security requirements in this section apply regardless of whether the application supports the Application Security Counters implementation-option.

20.1 Reference PIN Protection

Comparison of the Transaction PIN with the Reference PIN needs to be done in a secure manner that prevents compromise of the Reference PIN through attacks such as, but not limited to, fault insertion, tearing, and timing or power analysis.

In the case of an incorrect PIN, it needs to be impossible to detect by timing or power measurements which digit or digits are incorrect and which (if any) are correct.

Req 20.1 (Decrement PIN Try Counter when PIN is accessed):

Whenever the Reference PIN is accessed, the PIN Try Counter shall be decremented by one.

Req 20.2 (Resetting PIN Try Counter):

The PIN Try Counter shall be reset only on successful verification of the PIN, or a successful processing of a PIN CHANGE/UNBLOCK command. The PIN Try Counter shall be set to the value specified in the CSU as a result of successful processing of a CSU where the 'Update PIN Try Counter' bit has the value 1b.

Req 20.3 (Reference PIN not externally accessible):

The Reference PIN used for offline PIN verification shall not be accessible external to the card.

Req 20.4 (Storing the Reference PIN):

The Reference PIN shall be stored in a manner that ensures that failures in PIN processing (including PIN decipherment errors or incorrect PIN entry) and transaction processing (such as tearing) would not cause a loss of or change in the Reference PIN.

Req 20.5 (Not revealing the Reference PIN):

The processing of the PIN comparison, the VERIFY command, and the PIN CHANGE/UNBLOCK command shall not reveal any information about the value of the Reference PIN.

20.2 Key Protection

It needs to be infeasible to determine the values of any application key in the card by misuse of the logical interface (for example, by forcing the application to process an excessive number of commands, or commands that contain cryptograms that are incorrectly formed).

Cryptographic keys can be protected by ensuring that keys can only be used (or misused) a limited number of times.

- For symmetric cryptography, the use of session keys naturally results in the limited use of an individual session key. However care needs to be taken to limit the risk of master key leakage during session key derivation. If the card platform is insufficiently protected against certain kinds of attack, then such leakage might occur through persistent misuse of the card and its session key derivation process.
- For offline data authentication asymmetric keys (DDA/CDA), the application should prevent an attacker from obtaining an excessive number of digital signatures from the card. Furthermore the application should prevent an attacker from obtaining an excessive number of signatures produced on the same data input.
- For offline PIN decipherment asymmetric keys, the RSA private key should not be used to decipher an excessive number of badly formed enciphered PINs.

Req 20.6 (Security of cryptographic keys):

Implementations of CPA shall ensure that it is not feasible for an attacker to determine the values of secret cryptographic keys used by the application.

20.3 Secure Messaging

Req 20.7 (States that support processing secure messaging):

The card shall not process secure messages unless in the ONLINE or SCRIPT state.

Req 20.8 (Stop script processing after MAC error):

Once a MAC error has occurred, the card shall discontinue processing subsequent script commands received in the same transaction, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6982' (Security status not satisfied).

20.4 Security Counters

The use of security counters is one method of ensuring that the requirements in section 20.2 are met. If a specific implementation uses security counters for such a purpose, the counters and their associated limits may be implemented either by the application or by the platform. Annex F specifies a method for implementing security counters in the application.

There could essentially be four types of counters in CPA:

- The Application Transaction Counter (ATC), which is application wide, and is incremented once per transaction. Key protection is achieved by applying controls to limit the relevant cryptographic operations per transaction.
 - With the ATC included in the cryptographic data, data input will be different for each transaction.
 - With the ATC included in the session key derivation, session keys will be different for each transaction.
- Key counters that control the number of times that a key is used in a cryptographic process in order to limit the persistent misuse of a card (for example, the generation of far more application cryptograms than would be expected through normal usage over a given period of time)
- PIN Try Counter that can limit the number of times that a PIN is wrongly entered
- Command counters that can limit the number of times a specific command is executed during a transaction (for example, one INTERNAL AUTHENTICATE per transaction)

For the application cryptograms, a key counter (and associated limit) that increments prior to symmetric session key derivation and is reset on successful validation of an ARPC can protect both the master and session keys as follows:

- By limiting the number of consecutive session key derivations that can occur (without issuer approval) and hence, the number of times the master key can be exercised without the issuer being aware.
- For each session key so derived, the number of Application Cryptogram generations is limited by the application to at most two per transaction.
- For each session key so derived, the number of attempted ARPC verifications is limited by the application to one per transaction.

For Secure Messaging, a key counter (and associated limit) that increments prior to symmetric session key derivation and is decremented on successful validation of a Secure Messaging MAC can protect the keys as follows:

- By limiting the total number of unsuccessful script MAC verifications and associated session key derivations, because at most one MAC failure may occur per transaction.
- By prohibiting attempts to force unsuccessful decryption of confidential data, since each such message is protected by a MAC that should be verified (and will fail) first.

For DDA/CDA keys, the number of signing functions per transaction needs to be limited.

For the PIN decipherment key, a security counter could be applied to limit the number of times an incorrectly-enciphered PIN is processed.

Req 20.9 (ATC does not roll over):

The ATC shall not be allowed to roll over.

NOTE: If an issuer chooses to limit the maximum number of transactions that may be processed by the application over the lifetime of the card to less than 65,535, the ATC may be personalised to a starting value other than zero. This is an issuer-option.

Req 20.10 (ATC not updated by scripts):

The ATC shall not be updateable using any script command.

Req 20.11 (Only one INTERNAL AUTHENTICATE per transaction):

Implementations of CPA shall ensure that the application will perform at most one INTERNAL AUTHENTICATE command for each value of the ATC. If the application receives additional (beyond the first) INTERNAL AUTHENTICATE commands in the same transaction, then the application shall discontinue processing the additional INTERNAL AUTHENTICATE command, shall respond with an SW1 SW2 that indicates an error, and should respond with SW1 SW2 = '6985' (Conditions of use not satisfied).

Req 20.12 (Security counters don't roll over):

Security counters shall not be allowed to roll over.

20.5 Other Data Requirements

Smart card application developers must take into account the following aspects:

- The application must be resistant to tearing attacks. Loss of power at any stage of processing must not leave the application in an insecure state. With the exception of data items such as security counters which protect sensitive data, either all or none of the changes made during processing a command must take place.
- Data items must be suitably protected against accidental and/or malicious corruption (for example, protected by checksums). If data is found to be corrupt, the application must reject all commands with a suitable error code.

The ICC Dynamic Number is generated by the application for use in DDA and CDA. The ICC Unpredictable Number is generated in response to the GET CHALLENGE command.

Req 20.13 (Random numbers generated by application):

The ICC Unpredictable Number (the challenge) and the ICC Dynamic Number (as included in DDA/CDA) shall be 8-byte unpredictable numbers and so shall appear to be randomly generated.

This may be achieved using a Random Number Generator (RNG) provided by the IC manufacturer. Such an RNG should comply with ISO/IEC FDIS 18032, NIST SP 800-22A, or AIS 20.31.

The following describes the method for encipherment of the 8-byte 'Counters' portion of the Issuer Application Data before the Application Cryptogram is generated. 'Counters' is enciphered if the 'Encipher Counters portion of IAD' bit in the Issuer Options Profile Control has value 1b.

Req 20.14 (Enciphering counters in Issuer Application Data):

If the (8-byte) Counters portion of the Issuer Application Data is to be enciphered, it shall be enciphered as follows:

The eight-byte Counters block shall be enciphered in ECB Mode as defined in Appendix A1.1 of EMV Book 2, with no additional padding applied (thus the ciphertext is eight bytes long).

The encipherment key (ECK) used shall be a variant of the AC session key (SK) computed as follows:

$SK_L = \text{the left-most bytes of SKAC}$

$SK_R = \text{the right-most bytes of SKAC}$

$ECK_L := SK_L \oplus ('59' || '00' || '00' || '00' || '00' || '00' || '00' || '00')$

$ECK_R := SK_R \oplus ('95' || '00' || '00' || '00' || '00' || '00' || '00' || '00')$

$ECK = ECK_L || ECK_R$

21 Personalisation

This section is organised as follows:

- 21.1 Data Elements to be Personalised
 - 21.1.1 EMV Data in Records with SFI 1 through 10
 - 21.1.2 CPA Data Elements Requiring Personalisation
 - 21.1.3 CPA Default Implementation
- 21.2 EMV CPS
 - 21.2.1 Personalisation Commands
 - 21.2.2 Data Grouping Rules
 - 21.2.3 Data Grouping Order
 - 21.2.4 Grouped Data Groupings
 - 21.2.5 DGIs for Record Data
 - 21.2.6 Files with SFI between 1 and 10
 - 21.2.7 Files with SFI between 11 and 30
 - 21.2.8 CPA Recommended Data Group Indicators for Records
 - 21.2.9 DGIs for Internal Application Data
 - 21.2.10 DGIs for Command Response Data
 - 21.2.11 DGIs for PIN and Key Related Data
 - 21.2.12 RFU DGIs
- 21.3 Requirements for Data Element Values
 - 21.3.1 Profile Selection Entries
 - 21.3.2 AIP
 - 21.3.3 AFL
 - 21.3.4 VLP Profile Data
 - 21.3.5 Token Authentication Profile Data
 - 21.3.6 Offline Counters
 - 21.3.7 CIACs
 - 21.3.8 Signed Static Data
 - 21.3.9 EMV Record Data
 - 21.3.10 CDOLs
 - 21.3.11 Keys
 - 21.3.12 Previous Transaction History (PTH)
 - 21.3.13 Log Entry
- 21.4 Missing Data Elements

21.1 Data Elements to be Personalised

This section specifies the data elements that are personalised for the application. The requirements of this section apply regardless of whether the application is personalised using EMV CPS, or another personalisation method.

How data elements are personalised or pre-personalised is beyond the scope of CPA, except when EMV CPS is used.

NOTE: EMV mandatory data is specified in EMV Book 3, section 7.2.

21.1.1 EMV Data in Records with SFI 1 through 10

Either before or after the personalisation of CPA, the following are determined:

- The files (that is, values for SFI) used to store EMV data.
- The records (that is, values for the record number within a file) used and the length reserved for each record.

Some CPA implementations do not need to determine the organisation of data in records before personalisation, since some multi-application card platforms do not require a file system and the applications can then simulate the files and records themselves.

Other implementations will need to determine the organisation of data in records before personalisation. This is the case, for example, when a real file system is used to store the records and when the file structure cannot be created by the applications.

Req 21.1 (Organisation of EMV record data):

The following requirements apply to the organisation of EMV records into files:

- *At a minimum, the CPA shall support up to 2K (2048) bytes of memory to store EMV records if the application supports the Dynamic-RSA implementer-option.*
- *At a minimum, the CPA shall support up to 1.5K (1536) bytes of memory to store EMV records if the application does not support the Dynamic-RSA implementer-option.*
- *It shall be an issuer-option to store the EMV records in any file with an SFI between 1 and 10 (for example, records may be stored in SFI 1 and 2; or in SFI 1, 3, and 4; or in SFI 5, 6, 8, 9).*
- *At a minimum, the CPA shall support up to a total of 16 records for EMV data.*
- *It shall be an issuer-option to place up to 16 records in any file with SFI between 1 and 10, provided the total number of records is less than or equal to the maximum number of records supported by the application (for example, two records in file 1, three records in file 2 etc.).*
- *It shall be an issuer-option to request records with a record length of up to 254 bytes.*

In other words; in any implementation of CPA, allocation of the EMV data to files and records can be performed in any file with an SFI between 1 to 10 and any record, provided that:

- The total memory for records needed is less than or equal to:
 - 2048 (2K) bytes if the Dynamic-RSA implementer-option is supported
 - 1536 (1.5K) bytes if the Dynamic-RSA implementer-option is not supported
- The total number of records is less than or equal to 16
- The length of records is less than or equal to 254 bytes, including the tag '70' and the length byte(s).

Implementations may support:

- More than the minimum 2048 or 1536 bytes
- More than 16 records total in all files with SFI between 1 and 10

As previously implied, some implementations will support the above requirements without the need to prepare the card before personalisation to meet an issuer's data organisation needs whilst other implementations will need to be customised before personalisation.

21.1.2 CPA Data Elements Requiring Personalisation

Any value of AID and FCI allowed by EMV may be chosen by the issuer. The personalisation of the FCI when EMV CPS is used is described in section 21.2.

Req 21.2 (Personalisation of SELECT response):

The data elements listed in Table 21-1 may be set during either pre-personalisation or personalisation, to any value selected by the issuer that is allowed in EMV.

Tag	Data Element Name	Size (bytes)	Format
'84'	<i>AID (returned in SELECT command response)</i>	<i>var. 5 - 16</i>	<i>binary</i>
'6F'	<i>FCI</i>	<i>var. up to 240</i>	<i>binary</i>

Table 21-1: SELECT Command Response Data Elements – Mandatory

The data elements Command Parameters, PIN Try Limit, Previous Transaction History, and Application Issuer Life Cycle Data only require tags when the data elements are personalised using the EMV CPS implementer-option. If the EMV CPS implementer-option is not supported, this specification does not require that the data elements be tagged, and does not require the value to be used for the tag associated with each data element.

Req 21.3 (Personalisation of mandatory data):

The data elements listed in Table 21-2 shall be personalised for CPA.

Tag	Data Element Name	Size (bytes)	Format
'C1'	<i>Application Control</i>	4	binary
'C8'	<i>Application Issuer Life Cycle Data</i>	20	binary
'9F10'	<i>Issuer Application Data</i> ⁶	32	binary
'5F28'	<i>Issuer Country Code</i>	2	n3
—	<i>Master Key for SMC</i>	16	binary
—	<i>Master Key for SMI</i>	16	binary
—	<i>Master Key for AC</i>	16	binary

Table 21-2: Unique CPA Persistent Data Elements – Mandatory

⁶ The value personalised for Issuer Application Data is used by the application to determine the default values for Bytes 9-16 and 18-32 of the Issuer Application Data data element sent in the GENERATE AC response. The Length Indicators, Common Core Identifier, DKI, and CVR portions of the Issuer Application Data sent in the GENERATE AC response are generated by the application and not taken from this default value. The portions of bytes 9-16 and 18-32 that are not filled by application processing are filled with the personalised default value.

Req 21.4 (Personalisation of issuer-optional data):

The data elements listed in Table 21-3 shall be personalised for CPA if the condition is true.

Tag	Data Element Name	Condition	Size (bytes)	Format
–	<i>Reference Personal Identification Number (PIN)</i>	<i>If an issuer supports offline PIN.</i>	<i>8</i>	<i>binary</i>
'9F17'	<i>PIN Try Counter</i>	<i>If an issuer chooses to support offline PIN, and wants the PIN Try Counter to start at a value other than the PIN Try Limit.</i>	<i>1</i>	<i>binary</i>
'9F36'	<i>Application Transaction Counter (ATC)</i>	<i>If the issuer chooses to limit the number of transactions over the lifetime of the card to less than 65,535.</i>	<i>2</i>	<i>binary</i>
'C2'	<i>Profile Selection File Entry</i>	<i>If profile selection is active for the application.</i>	<i>2</i>	<i>binary</i>
'C3'	<i>Number of Days Offline Limit</i>	<i>If the Number of Days Offline Check is active for any profile personalised in the application.</i>	<i>2</i>	<i>n 4</i>
'C6'	<i>PIN Try Limit</i>	<i>If an issuer supports offline PIN.</i>	<i>1</i>	<i>binary</i>
'C7'	<i>Previous Transaction History</i>	<i>If an issuer chooses to send new cards online.</i>	<i>2</i>	<i>binary</i>

Table 21-3: Unique CPA Persistent Data Elements – Issuer-optional

Req 21.5 (Personalisation of key data for DDA/CDA):

If in any profile for the application, the issuer personalises the card to support DDA or CDA, then the ICC Private Key elements shall be personalised.

Req 21.6 (Personalisation of ICC private key data for enciphered PIN):

If in any profile for the application, the issuer personalises the card to support offline enciphered PIN using the ICC Public/Private Key pair, then the ICC Private Key elements shall be personalised.

Req 21.7 (Personalisation of ICC PIN encipherment key data for enciphered PIN):

If in any profile for the application, the issuer personalises the card to support offline enciphered PIN using the ICC PIN Encipherment Public/Private Key pair, then the ICC PIN Encipherment Private Key elements shall be personalised.

Template Tag	Tag #	Data Element Name	Size (bytes)	Format
–	–	<i>ICC Private Key elements</i>	<i>variable</i>	<i>binary</i>
–	–	<i>ICC PIN Encipherment Private Key elements</i>	<i>variable</i>	<i>binary</i>

Table 21-4: Unique CPA Persistent Data Elements – Dynamic RSA Option Elements

Req 21.8 (Personalisation of transaction logging data):

If in any profile for the application, the issuer personalises the card to support transaction logging; then the Log Entry Data element shall be personalised;

It shall be an issuer-option to personalise any of the Log Data Tables listed in Table 21-5.

Template Tag	Tag #	Data element name	Size (bytes)	Format
–	'9F4D'	<i>Log Entry</i>	<i>2</i>	<i>binary</i>
'BF40'	'DF01'	<i>First Gen AC Log Data Table</i>	<i>1 + (N * 2)</i>	<i>binary</i>
'BF40'	'DF03'	<i>First Gen AC Unchanging Log Data Table</i>	<i>1 + (N * 2)</i>	<i>binary</i>
'BF40'	'DF02'	<i>Second Gen AC Log Data Table</i>	<i>1 + (N * 2)</i>	<i>binary</i>

Table 21-5: Unique CPA Persistent Data Elements – Issuer-optional Transaction Logging Elements

Req 21.9 (Personalisation of optional security data):

If the optional Session Key Counter described in Annex F and in Annex L: Data Dictionary is implemented, then the Session Key Counter Limit in Table 21-6 shall be personalised.

Template Tag	Tag #	Data Element Name	Size (bytes)	Format
–	'C5'	<i>Security Limits</i>	<i>6</i>	<i>binary</i>

Table 21-6: Unique CPA Persistent Data Elements – Optional Security Limit Elements

Req 21.10 (Personalisation of conditional VLP data):

The data elements listed in Table 21-7 shall be personalised for CPA if the VLP implementer-option is supported and active.

Template Tag	Tag #	Data Element Name	Size (bytes)	Format
–	'9F77'	<i>VLP Funds Limit</i>	6	<i>n 12</i>
–	'9F42'	<i>Application Currency Code</i>	2	<i>n 3</i>

Table 21-7: Unique CPA Persistent Data Elements – Conditional VLP Elements**Req 21.11 (Personalisation of optional VLP data):**

The data element in Table 21-8 shall be personalised for CPA if the VLP implementer-option is supported and active, and the condition is true.

Tag #	Data Element Name	Condition	Size (bytes)	Format
'9F78'	<i>VLP Single Transaction Limit</i>	<i>If the issuer limits the value for each VLP transaction.</i>	6	<i>n 12</i>

Table 21-8: Unique CPA Persistent Data Elements – Issuer-Optional VLP Elements

Req 21.12 (Personalisation of CPA template data):

The data elements listed in Table 21-9 shall be personalised for CPA if the condition is true.

Template Tag	Tag #	Data element name	Condition	Size (bytes)	Format
–	–	<i>Profile Selection Entries</i>	<i>If profile selection is active for the application.</i>	<i>var.</i>	<i>records</i>
'BF30'	'DF01'– 'DF0n', 'DF11'– 'DF1n'	<i>Accumulator Data: value ('DF0x') and Limits ('DF1x')</i>	<i>Always (at least one)</i>	<i>N * (18 or 30)</i>	<i>numeric</i>
'BF31'	'DF01' – 'DF0n'	<i>Accumulator Profile Controls</i>	<i>Always (at least one)</i>	<i>N * 2</i>	<i>binary</i>
'BF32'	'DF01' – 'DF0n'	<i>Accumulator x Controls</i>	<i>Always (at least one)</i>	<i>N * 3</i>	<i>binary</i>
'BF33'	'DF01'– 'DF0n'	<i>Additional Check Tables</i>	<i>If any Additional Check Table Check is active for any profile personalised in the application.</i>	<i>N * var.</i>	<i>binary</i>
'BF34'	'DF01'– 'DF0n'	<i>CIACs Entries (CIAC-Decline, CIAC-Online, and CIAC-Default)</i>	<i>Always (at least one)</i>	<i>N * 18</i>	<i>binary</i>
'BF35'	'DF01' – 'DF0n', 'DF11'– 'DF1n'	<i>Counter Data: value ('DF0x') and Limits ('DF1x')</i>	<i>Always (at least one)</i>	<i>N * (3 or 5)</i>	<i>numeric</i>
'BF36'	'DF01' – 'DF0n'	<i>Counter Profile Controls</i>	<i>Always (at least one)</i>	<i>N * 1</i>	<i>binary</i>
'BF37'	'DF01' – 'DF0n'	<i>Counter X Controls</i>	<i>Always (at least one)</i>	<i>N * 1</i>	<i>binary</i>

Table 21-9: CPA Persistent Data Elements – Data Sets
(continues)

Template Tag	Tag #	Data element name	Condition	Size (bytes)	Format
'BF38'	'DF01' – 'DF0n'	Currency Conversion Tables	If currency conversion is allowed in a Maximum Transaction Amount Check, any Accumulator <i>x</i> Check, or any Cyclic Accumulator <i>x</i> Check that is active for any profile personalised in the application.	$N * (2 + (n * 5))$	binary
'BF39'	'DF01' – 'DF0n'	Cyclic Accumulator Profile Controls	If any Cyclic Accumulator <i>x</i> is active for any profile personalised in the application.	$N * 2$	binary
'BF3A'	'DF01' – 'DF0n'	Cyclic Accumulator <i>X</i> Controls	If any Cyclic Accumulator <i>x</i> is active for any profile personalised in the application.	$N * 3$	binary
'BF3B'	'DF01' – 'DF0n'	Issuer Options Profile Controls	Always (at least one)	$N * 7$	binary
'BF3C'	'DF01' – 'DF0n'	Limits Entries	If the Maximum Transaction Amount Check, or any Cyclic Accumulator <i>x</i> Check is active for any profile personalised in the application.	$N * 6$	<i>n</i> 12

Table 21-9: CPA Persistent Data Elements – Data Sets, continued

Template Tag	Tag #	Data element name	Condition	Size (bytes)	Format
'BF3D'	'DF01' – 'DF0n'	Maximum Transaction Amount Profile Controls	If the Maximum Transaction Amount Check is active for any profile personalised in the application.	$N * 4$	binary
'BF3E'	'DF01' – 'DF0n'	GPO Parameters	Always (at least one)	$N * 2$	binary
'BF3F'	'DF01' – 'DF0n'	Profile Controls	Always (at least one)	$N * 8$	binary
'BF41'	'DF01' – 'DF0n'	AIP/AFL Entries	Always (at least one)	$N * (2+1+61)$	binary

Table 21-9: CPA Persistent Data Elements – Data Sets, continued

Req 21.13:

Each Currency Conversion Parameter in a Currency Conversion Table shall contain the data elements listed in Table 21-11.

Data	Length	Description
<i>Source Currency Code</i>	<i>2</i>	<i>The Currency Code of the currency to be converted to the currency identified by the Target Currency Code for the accumulator using this Currency Conversion Parameter.</i>
<i>Conversion Rate</i>	<i>2</i>	<i>The rate (used with the Conversion Exponent) multiplied with the transaction amount value to approximate the transaction value in the accumulator currency.</i>
<i>Conversion Exponent</i>	<i>1</i>	<p><i>A signed number that indicates the power of 10 used to modify the Conversion Rate. Bit b8 indicates the sign of the exponent, and bits b7 through b1 indicate the value of the exponent.</i></p> <p><i>If the sign is positive (b8 = 0b):</i></p> $\text{Approximate Value} = \text{Transaction amount} * \text{Conversion Rate} * 10^{\text{Conversion Exponent (b7 to b1)}}$ <p><i>If the sign is negative (b8 = 1b):</i></p> $\text{Approximate Value} = (\text{Transaction amount} * \text{Conversion Rate}) / 10^{\text{Conversion Exponent (b7 to b1)}}$

Table 21-10: Currency Conversion Parameter

Filler bytes are allowed in the personalisation data for records that contain the Profile Selection Entries.

Filler bytes are allowed in the personalisation data for templates listed in Table 21-9.

NOTE: EMV uses the value '00' for filler bytes.

21.1.3 CPA Default Implementation

Table 21-11 indicates the minimum and maximum number of each type of profile resource that may be implemented in CPA. The column “Min #” indicates the minimum number of each type of profile resource that must be supported in any implementation of CPA. The column “Max #” indicates the maximum number of each type of profile resource that could be implemented, due to design constraints (such as limits on tag ranges).

It is the issuer’s choice how many of each type of profile resource are used in an application after personalisation, depending on the issuer’s design of profiles and profile behaviour. Issuers that design their profiles to use no more than the minimum number of any type of profile resource will be able to personalise their application on any CPA implementation. Issuers that design their profiles to use more than the minimum number of any type of profile resource will need to ensure that the CPA implementation they choose supports the additional number of the profile resource type.

Req 21.14 (Minimum/maximum numbers for support of profile resource data):

At a minimum, implementations of CPA shall support at least the minimum number of Accumulators, Counters, and Cyclic Accumulator Data data elements indicated in Table 21-11.

Req 21.15 (Minimum/maximum numbers for support of profile resource data):

At a minimum, implementations of CPA shall be capable of being personalised with the minimum number of each type of profile resource listed in Table 21-11.

Profile resource	Template tag	Length (bytes)	Min #	Max #
Accumulators⁷	–	6	2	14
<i>Accumulator Controls</i> <i>Tag 'DF0x' = Accumulator x Control</i>	'BF32'	3	2	14
<i>Accumulator Profile Controls</i> <i>Tag 'DF0x' = Accumulator Profile Control x</i>	'BF31'	2	6	14
<i>Accumulators Data</i> <i>Tag 'DF0x' = Accumulator x value</i> <i>Tag 'DF1x' = Accumulator x Limits</i>	'BF30'	18 or 30	2	14
<i>Additional Check Tables</i> <i>Tag 'DF0x' = Additional Check Table x</i>	'BF33'	Var.	2	14
<i>AIP/AFL Entries</i> <i>Tag 'DF0x' = AIP/AFL Entry x</i>	'BF41'	Var.	6	14
<i>CIACs Entries</i> <i>Tag 'DF0x' = CIACs Entry x</i>	'BF34'	18	6	14

Table 21-11: Number of Each Type of Profile Resource in CPA
(continues)

⁷ This does not include the VLP Amount. If VLP is supported, there is an additional accumulator for VLP Amount.

Profile resource	Template tag	Length (bytes)	Min #	Max #
Counters	–	1	3	14
<i>Counter Controls</i> <i>Tag 'DF0x' = Counter x Control</i>	'BF37'	1	3	14
<i>Counter Profile Controls</i> <i>Tag 'DF0x' = Counter Profile Control x</i>	'BF36'	1	9	14
<i>Counters Data</i> <i>Tag 'DF0x' = Counter x value</i> <i>Tag 'DF1x' = Counter x Limits</i>	'BF35'	3 or 5	3	14
<i>Currency Conversion Tables</i> <i>Tag 'DF0x' = Currency Conversion Table x</i>	'BF38'	Var	3	14
Cyclic Accumulator x Data	–	11 ⁸	2	14
<i>Cyclic Accumulator Controls</i> <i>Tag 'DF0x' = Cyclic Accumulator x Control</i>	'BF3A'	3	2	14
<i>Cyclic Accumulator Profile Controls</i> <i>Tag 'DF0x' = Cyclic Accumulator Profile Control x</i>	'BF39'	2	6	14
<i>GPO Parameters</i> <i>Tag 'DF0x' = GPO Parameters x</i>	'BF3E'	2	3	14
<i>Issuer Options Profile Controls</i> <i>Tag 'DF0x' = Issuer Options Profile Control x</i>	'BF3B'	3	7	14
<i>Limits Entries</i> <i>Tag 'DF0x' = Limit Entry x</i>	'BF3C'	6	6	14

Table 21-11: Number of Each Type of Profile Resource in CPA, continued

⁸ For each Cyclic Accumulator x, this includes: Cyclic Accumulator x (6 bytes), Cyclic Accumulator x Reference Date (3 bytes), and Cyclic Accumulator x Reference Day (2 bytes)

Profile resource	Template tag	Length (bytes)	Min #	Max #
<i>Log Data Tables</i>	<i>'BF40'</i>			
<i>Tag 'DF01' = First GEN AC Log Data Table</i>		<i>Var.</i>	<i>1</i>	<i>1</i>
<i>Tag 'DF02' = Second GEN AC Log Data Table</i>		<i>Var.</i>	<i>1</i>	<i>1</i>
<i>Tag 'DF03' = First GEN AC Unchanging Log Data Table</i>		<i>Var.</i>	<i>1</i>	<i>1</i>
<i>MTA Profile Controls</i>	<i>'BF3D'</i>	<i>4</i>	<i>6</i>	<i>14</i>
<i>Tag 'DF0x' = MTA Profile Control x</i>				
<i>Profile Controls</i>	<i>'BF3F'</i>	<i>Min. 8</i>	<i>8</i>	<i>126</i>
<i>Tag 'DFx' = Profile Control x</i>				

Table 21-11: Number of Each Type of Profile Resource in CPA, continued

21.2 EMV CPS

Support of the EMVCo Card Personalisation Specification (CPS) is an implementer-option for CPA. For cards that support the CPS implementer-option, the format of the personalisation data must be consistent across implementations for an issuer to be able to personalise different implementations with the same personalisation data. The requirements of this section apply when the application is personalised using EMV CPS as the personalisation method.

21.2.1 Personalisation Commands

21.2.1.1 Overview

Command APDUs are defined in the EMV Card Personalisation Specification (EMV CPS), Chapter 3. A typical personalisation consists of the following commands in the sequence shown:

- SELECT
- INITIALIZE UPDATE
- EXTERNAL AUTHENTICATE
- STORE DATA

21.2.1.2 SELECT Command

The SELECT command is used to select each IC card application to be personalised. Application selection is described in EMV Book 1.

The SELECT command will be issued once for each IC card application to be personalised. The data in the FCI (the response to the SELECT command) will be changed during the personalisation process. There is no specific requirement for it prior to personalisation, other than the requirement to return the AID Tag (84) with length and value.

21.2.1.3 INITIALIZE UPDATE Command

The INITIALIZE UPDATE command is the first command issued after the personalisation device selects the application. INITIALIZE UPDATE will begin setting up the Secure Channel Session to be used during personalisation. The data to perform mutual authentication is exchanged.

Refer to EMV CPS for a detailed definition of the INITIALIZE UPDATE command.

21.2.1.4 EXTERNAL AUTHENTICATE Command

The EXTERNAL AUTHENTICATE command follows the INITIALIZE UPDATE command and is used to authenticate the personalisation device to the IC card application. EXTERNAL AUTHENTICATE will be issued once for each secure channel initiation and shall be issued at least once for each application to be personalised.

Refer to EMV CPS for a detailed definition of the EXTERNAL AUTHENTICATE command.

CPA applications shall support the three security levels allowed in EMV CPS:

- **No security** – The secure channel is established for authentication purposes only
- **MAC** – The secure channel is established for integrity as well as authentication purposes
- **Encryption and MAC** – The secure channel is established for confidentiality and integrity as well as authentication purposes

21.2.1.5 STORE DATA Command

The STORE DATA command is used to send personalisation data to the card application. The data preparation process organises the personalisation data to be sent into data groupings. A Data Grouping Identifier (DGI) identifies each data grouping. The IC card application then uses the DGI to determine how the data grouping is to be processed.

Refer to EMV CPS for a definition of the STORE DATA command.

Req 21.16 (Ignore P1 value in STORE DATA command):

The P1 value in the STORE DATA command header shall be ignored except for the high-order bit, used to indicate the last STORE DATA command.

Req 21.17 (Ignore P2 value in STORE DATA command):

The P2 value in the STORE DATA command header shall be ignored.

Support for a single DGI spanning two STORE DATA commands is needed to personalise a record containing an Issuer Certificate with a CA key size of 1984-bits, and template tags that may overflow the length of the command data field in a single STORE DATA command. Support of multiple DGIs in a single STORE DATA command may reduce the number of commands needed, and perhaps the personalisation time. A STORE DATA command that supports multiple DGIs is not required to also be able to span the last DGI over a second STORE DATA command.

Req 21.18 (Support multiple DGIs in STORE DATA command):

A single STORE DATA command shall support multiple DGIs.

Req 21.19 (Support DGIs spanning multiple STORE DATA commands):

The application shall support any single DGI spanning two STORE DATA commands.

Req 21.20 (DGIs lengths in STORE DATA command):

The application shall support single-byte DGI lengths only.

Req 21.21 (TLV format for data in DGIs):

All tagged data elements shall be entered in the DGI in TLV format, and the application shall accept tagged data elements in any order within the specified DGI.

21.2.2 Data Grouping Rules

Rules for creation of data groupings are defined in EMV CPS Section 2.2. Refer to chapter 3 for a definition of the Data Groupings requiring personalisation for CPA.

21.2.3 Data Grouping Order

It is recommended that application developers allow Data Groupings to be sent to the CPA application in any order. However, in some implementations there may be constraints on the way in which the Data Groupings are ordered.

The application developer and Data Preparation must ensure that any such implementation-specific constraints are respected.

21.2.4 Grouped Data Groupings

It is recommended that application developers support any grouping of Data Groupings, with the exception of Data Groupings identified in the version control field (VERCNTL) in the IC Card Application Data. However, in some implementations there may be constraints on how Data Groupings are grouped.

The application developer and Data Preparation must ensure that any such implementation-specific constraints are respected.

The requirement column titled “Req.”, in the following tables of data elements for each DGI, lists the requirements for each data element:

- M (Mandatory) indicates that the data element must be present.
- C (Conditional) indicates that the data element is necessary under certain conditions.
- O (Optional) indicates that the data element is optional

21.2.5 DGIs for Record Data

For EMV applications, the persistent data elements stored in files with an SFI between 1 to 30, are stored in records and are retrievable with the EMV Read Record command. A record is always the value of a Data Grouping.

During personalisation, the application receives a series of STORE DATA commands corresponding to the record values and then stores the record values in records. The application must have the permanent memory available to store such records, using one of the following methods:

- The pre-allocation of the memory and file structure
- The allocation of the memory and file structure during personalisation

Data Groupings are reserved for record values for Data Grouping Identifiers in the range 'XXYY', where 'XX' indicates the SFI and 'YY' indicates the record number as follows:

- '01' <= 'XX' <= '1E' and
- '01' <= 'YY' <= 'FF'

'XX' represents the *SFI* where the record is stored. 'YY' represents the record number. This specification does not mandate the file and record structure for the personalisation of these files.

As defined by EMV, the persistent data elements stored in files with a Short File Indicator (SFI) between 1 to 10, are stored in records following the '70' template and are retrievable with the Read Record command. Note that the P1 value used in the Read Record command corresponds to the 'YY' value in the DGI.

Also as specified by EMV, CPA applications, in both non-personalised and personalised states, do not interpret the data elements stored in these records but instead process the Read Record command so that the appropriate personalised record is returned in the response message.

21.2.6 Files with SFI between 1 and 10

Table 21-12 illustrates the recommended organisation of data elements for CPA. The issuer defines how the data elements are organised and must be able to add proprietary data elements, in addition to the data elements shown in the table.

Data Groupings are reserved for EMV SFI and record values in the range 'XXYY' where:

- SFI '01' <= 'XX' <= '0A'
- Record '01' <= 'YY' <= 'FF'

There are therefore ten files in which EMV records can be stored. Each file may contain up to 255 records. However, the CPA applications do not approach these limits.

21.2.7 Files with SFI between 11 and 30

Data Groupings are reserved for EMV record values for Data Grouping Identifiers with a value of 'XX', where:

- 'SFI '0B' <= 'XX' <= '14'
- 'SFI '15' <= 'XX' <= '1E'

Support of these Data Grouping Identifiers is optional. Among those Data Grouping Identifiers, only 'XX' = '0B' is defined for CPA. DGI 0Bnn is used for the optional VLP feature. CPA applications may support Data Grouping Identifiers for records in other files in with a Short File Identifier between 11 and 30.

21.2.8 CPA Recommended Data Group Indicators for Records

DGI	Description	Table	Encrypt	Defined
'0101'	SFI 1 Record 1: Track Data and Cardholder Name	Table 21-13	No	CPA
'0102'	SFI 1 Record 2: Track Data without Cardholder Name	Table 21-14	No	CPA
'0201'	SFI 2 Record 1: Data Authentication Data	Table 21-15	No	CPA
'0202'	SFI 2 Record 2: Data Authentication Data	Table 21-16	No	CPA
'0203'	SFI 2 Record 3: Signed Static Application Data	Table 21-17	No	CPA
'0204'	SFI 2 Record 4: ICC Dynamic Authentication Data	Table 21-18	No	CPA
'0205'	SFI 2 Record 5: PIN Encipherment Data	Table 21-19	No	CPA
'020n'	SFI 2 Record n: Duplicate Data Authentication Data	Table 21-20	No	CPA
'0301'	SFI 3 Record 1: Card Risk Management Data	Table 21-21	No	CPA
'0302'	SFI 3 Record 2: Card Risk Management Data	Table 21-22	No	CPA
'0303'	SFI 3 Record 3: Cardholder Verification Method List (only)	Table 21-23	No	CPA
'0B01'	SFI 11 Record 1: VLP Data	Table 21-24	No	CPA
'ssrr'	SFI 'ss' Record 'rr': Profile Selection Entry 'bb'	Table 21-25	No	CPA

Table 21-12: DGI Summary for Record Data

For the DGIs with the first byte equal to '01' through '1E', the first byte indicates the SFI in which the data is to be stored and the second byte indicates the record number within the SFI. Other DGIs in this range are also supported, but the ones listed above reflect the default record layout.

Req.	Tag	Data Element	Length	Encrypt
M	'57'	Track 2 Equivalent Data *	To 19	N/A
M	'5F20'	Cardholder Name	2-26	N/A
M	'9F1F'	Track 1 Discretionary Data	Var.	N/A

* This field may be padded at the end with a single hex 'F' to ensure whole bytes

Table 21-13: Data Content for DGI '0101'

Req.	Tag	Data Element	Length	Encrypt
M	'57'	Track 2 Equivalent Data *	To 19	N/A
M	'9F1F'	Track 1 Discretionary Data	Var.	N/A

* This field may be padded at the end with a single hex 'F' to ensure whole bytes.

Table 21-14: Data Content for DGI '0102'

Req.	Tag	Data Element	Condition	Length	Encrypt
C	'90'	Issuer Public Key (IPK) Certificate	If SDA, DDA, CDA, or Offline Enciphered PIN is supported	Var.	N/A

Table 21-15: Data Content for DGI '0201'

Req.	Tag	Data Element	Condition	Length	Encrypt
C	'9F32'	IPK Exponent	If SDA, DDA, CDA, or Offline Enciphered PIN is supported	1 or 3	N/A
C	'92'	IPK Remainder	If SDA, DDA, CDA, or Offline Enciphered PIN is supported and entire IPK does not fit in IPK Certificate	Var	N/A
C	'8F'	Certificate Authority Public Key Index	If SDA, DDA, CDA, or Offline Enciphered PIN is supported	1	N/A
C	'9F47'	ICC Public Key Exponent	If DDA, CDA, or Offline Enciphered PIN using ICC Public Key is supported	1 or 3	N/A
C	'9F48'	ICC Public Key Remainder	If DDA, CDA, or Offline Enciphered PIN using ICC Public Key is supported and entire ICC PK does not fit in ICC PK Certificate	Var	N/A
C	'9F49'	DDOL	If DDA or CDA is supported	3	N/A
C	'9F2E'	ICC PIN Encipherment Public Key Exponent	If Offline Enciphered PIN using ICC PIN Encipherment Public Key is supported	Var	N/A
C	'9F2F'	ICC PIN Encipherment Public Key Remainder	If Offline Enciphered PIN using ICC PIN Encipherment Public Key is supported and entire ICC PIN Encipherment PK does not fit in ICC PIN Encipherment PK Certificate	Var	N/A

Table 21-16: Data Content for DGI '0202'

Req.	Tag	Data Element	Condition	Length	Encrypt
C	'93'	Signed Static Application Data	If SDA is supported	Var.	N/A

Table 21-17: Data Content for DGI '0203'

Req.	Tag	Data Element	Condition	Length	Encrypt
C	'9F46'	ICC Public Key Certificate	If DDA, CDA, or Offline Enciphered PIN using ICC Public Key is supported	Var.	N/A

Table 21-18: Data Content for DGI '0204'

Req.	Tag	Data Element	Condition	Length	Encrypt
C	'9F2D'	ICC PIN Encipherment Public Key Certificate	If Offline Enciphered PIN using ICC PIN Encipherment Public Key is supported	Var.	N/A

Table 21-19: Data Content for DGI '0205'

Req.	Tag	Data Element	Condition	Length	Encrypt
C	'90'	Issuer Public Key (IPK) Certificate	If DDA, CDA, or Offline Enciphered PIN using ICC Public Key is supported and multiple certificates are to be personalised.	Var.	N/A

Req.	Tag	Data Element	Condition	Length	Encrypt
C	'8F'	Certificate Authority Public Key Index	If DDA, CDA, or Offline Enciphered PIN using ICC Public Key is supported and multiple certificates are to be personalised.	1	N/A
C	'9F32'	IPK Exponent	If DDA, CDA, or Offline Enciphered PIN using ICC Public Key is supported and multiple certificates are to be personalised.	1 or 3	N/A
C	'92'	IPK Remainder	If DDA, CDA, or Offline Enciphered PIN using ICC Public Key is supported and entire IPK does not fit in IPK Certificate and multiple certificates are to be personalised.	Var	N/A

Table 21-20: Data Content for DGI '020n'
(continues)

Req.	Tag	Data Element	Condition	Length	Encrypt
C	'93'	Signed Static Application Data	If SDA is supported and multiple certificates are to be personalised.	Var.	N/A

Table 21-20: Data Content for DGI '020n', continued

Req.	Tag	Data Element	Condition	Length	Encrypt
M	'5F24'	Application Expiration Date	Always	3	N/A
O	'5F25'	Application Effective Date	Optional	3	N/A
M	'5A'	Application Primary Account Number (PAN)	Always	Var	N/A
O	'5F34'	Application PAN Sequence Number	Optional	1	N/A
O	'9F07'	Application Usage Control	Optional	2	N/A
M	'8C'	Card Risk Management Data Object List 1 (CDOL1)	Always	Var	N/A
M	'8D'	Card Risk Management Data Object List 2 (CDOL2)	Always	Var	N/A
M	'8E'	Cardholder Verification Method (CVM) List	If Cardholder Verification is to be performed for any profile.	Var	N/A
M	'9F0D'	Issuer Action Code (IAC) Default	Always	5	N/A
M	'9F0E'	Issuer Action Code (IAC) Denial	Always	5	N/A
M	'9F0F'	Issuer Action Code (IAC) Online	Always	5	N/A
C	'5F28'	Issuer Country Code	If the 'Include Only if International' bit is personalised to 1b in any Counter Profile Control active for the application.	2	N/A
C	'9F4A'	SDA Tag List	If the AIP is signed	1	N/A

Table 21-21: Data Content for DGI '0301'

NOTE: The data elements in this DGI '0301' are included in the Signed Application Data (SAD). If updates to the Cardholder Verification Methods (CVM) List are to be made by the Issuer using Issuer Script Processing or if multiple CVM Lists are used and a single SAD is used, the CVM List should be included in DGI 0303 rather than in DGI 0301.

NOTE: The AIP included in the Signed Application Data (SAD) is not included in the record data because the value may change with the profile selected for the transaction.

Req.	Tag	Data Element	Length	Encrypt
O	'9F05'	Application Discretionary Data	Var	N/A
O	'9F0B'	Cardholder Name Extended (27-45)	Var	N/A
C	'9F44'	Application Currency Exponent	1	N/A
C	'9F42'	Application Currency Code	2	N/A
O	'5F30'	Service Code	2	N/A
M	'9F08'	Application Version Number	2	N/A

Table 21-22: Data Content for DGI '0302'

Req.	Tag	Data Element	Length	Encrypt
M	'8E'	Cardholder Verification Method (CVM) List	Var	N/A

Table 21-23: Data Content for DGI '0303'

DGI '0B01' is used only when VLP is being personalised. Note that many of the data elements in this record are also included in other records. Duplicating elements here is not required but it does allow an AFL list for VLP to contain fewer element and records.

Req.	Tag	Data Element	Condition	Length	Encrypt
M	'9F74'	VLP Issuer Authorisation Code	Always	6	N/A
M	'9F79'	VLP Available Funds	Always	6	N/A
C	'5A'	Application Primary Account Number (PAN)	If a different PAN is used for VLP.	Var	N/A
C	'5F34'	Application PAN Sequence Number	If a different PAN Sequence Number is used for VLP.	1	N/A
C	'8C'	CDOL1	If a different CDOL1 is used for VLP.	Var	N/A
C	'8D'	CDOL2	If a different CDOL2 is used for VLP.	Var	N/A
C	'8E'	Cardholder Verification Method (CVM) List	If a different CVM List is used for VLP.	Var	N/A
C	'9F0D'	Issuer Action Code (IAC) Default	If a different IAC Default is used for VLP.	5	N/A
C	'9F0E'	IAC Denial	If a different IAC Denial is used for VLP.	5	N/A
C	'9F0F'	IAC Online	If a different IAC Online is used for VLP.	5	N/A

Table 21-24: Data Content for DGI '0B01'
(continues)

Req.	Tag	Data Element	Condition	Length	Encrypt
C	'5F24'	Application Expiration Date	If a different Application Expiration date is used for VLP.	3	N/A
C	'5F28'	Issuer Country Code	If a different Issuer Country Code is used for VLP.	2	N/A
C	'9F07'	Application Usage Control (AUC)	If a different AUC is used for VLP.	2	N/A
O	'5F25'	Application Effective Date	Optional	3	N/A
C	'9F42'	Application Currency Code	If a different Application Currency Code is used for VLP.	2	N/A
O	'9F08'	Application Version Number	Optional	2	N/A
O	'5F20'	Cardholder Name	Optional	2-26	N/A

Table 21-24: Data Content for DGI '0B01', continued

For DGIs with the first byte equal to 'ss', where 'ss' is the value of the first byte of Profile Selection File Entry; the first byte indicates the SFI in which the data is to be stored, and the second byte indicates the record number within the SFI. 'ss' ranges in value from '15' to '1E', and 'rr' ranges in value from '01' to '0F'.

Req.	Tag	Data Element	Length	Encrypt
C	-	Profile Selection Entry 'rr'	var.	No

Table 21-25: Data Content for DGI 'ssrr'

21.2.9 DGIs for Internal Application Data

DGI	Description	Table	Encrypt	Defined
'3000'	Card Internal Data Elements	Table 21-27	No	CPA
'3F30'	Template 'BF30', Accumulators Data	—	No	CPA
'3F31'	Template 'BF31', Accumulator Profile Controls	—	No	CPA
'3F32'	Template 'BF32', Accumulator Controls	—	No	CPA
'3F33'	Template 'BF33', Additional Check Tables	—	No	CPA
'3F34'	Template 'BF34', CIACs Entries	—	No	CPA
'3F35'	Template 'BF35', Counters Data	—	No	CPA
'3F36'	Template 'BF36', Counter Profile Controls	—	No	CPA
'3F37'	Template 'BF37', Counter Controls	—	No	CPA
'3F38'	Template 'BF38', Currency Conversion Tables	—	No	CPA
'3F39'	Template 'BF39', Cyclic Accumulator Profile Controls	—	No	CPA
'3F3A'	Template 'BF3A', Cyclic Accumulator Controls	—	No	CPA
'3F3B'	Template 'BF3B', Issuer Options Profile Controls	—	No	CPA
'3F3C'	Template 'BF3C', Limits Entries	—	No	CPA
'3F3D'	Template 'BF3D', MTA Profile Controls	—	No	CPA
'3F3E'	Template 'BF3E', GPO Parameters	—	No	CPA
'3F3F'	Template 'BF3F', Profile Controls	—	No	CPA
'3F40'	Template 'BF40', Log Data Tables	—	No	CPA
'3F41'	Template 'BF41', AIP/AFL Entries	—	No	CPA

Table 21-26: DGI Summary for Internal Application Data

Req.	Tag	Data Element	Length	Encrypt
M	'5F28'	Issuer Country Code	2	N/A
O	'9F36'	Application Transaction Counter (ATC)	2	N/A
M	'9F10'	Issuer Application Data	32	N/A
C	'9F42'	Application Currency Code (This tag, required for VLP, is also contained in DGI 0302)	2	N/A
C	'9F77'	VLP Funds Limit	6	N/A
O	'9F78'	VLP Single Transaction Limit	6	N/A
C	'9F4F'	Log Format	Var	N/A
M	'C1'	Application Control	4	N/A
O	'C2'	Profile Selection File Entry	2	N/A
M	'C3'	Number of Days Offline Limit	2	N/A
C	'C5'	Security Limits	6	N/A
O	'C7'	Previous Transaction History	2	N/A
M	'C8'	Application Issuer Life Cycle Data	20	N/A

Table 21-27: Data Content for DGI '3000'

For details on DGIs '3F31' through '3F41' see template tags 'BF31' through 'BF41' respectively in Table 21-9.

21.2.10 DGIs for Command Response Data

The only command response data needing a DGI is SELECT, as shown in Table 21-28. This DGI is used for personalising the 'A5' template tag. Note that transaction logging requires tag '9F4D' in template tag 'BF0C' which is personalised as part of DGI '9102'.

DGI	Description	Table	Encrypt	Defined
'9102'	SELECT Command Response	EMV CPS V1.0 Table 14	No	EMV CPS

Table 21-28: DGI Summary for Command Response Data

Note that GET PROCESSING OPTIONS (GPO) response data for CPA uses DGI 'BF41' rather than DGI '9104' as defined in EMV CPS. The values needed by the application to build the Issuer Application Data contained in the GENERATE AC command response is personalised using tag '9F10' in DGI '0300'.

21.2.11 DGIs for PIN and Key Related Data

The DGIs for PIN and Key Related Data are specified in EMV CPS, with the following additions:

The tag for the PIN Try Limit data element in DGI '9010' is 'C6'.

The $q^{-1} \bmod p$ is the default convention to be used to generate the values for DGIs containing the CRT components for a CPA application (see Table 21-29).

Req.	Tag	Data Element	Length	Encrypt
C	N/A	CRT constant $q^{-1} \bmod p$	var.	SKU _{DEK}
C	N/A	CRT constant $d \bmod (q-1)$	var.	SKU _{DEK}
C	N/A	CRT constant $d \bmod (p-1)$	var.	SKU _{DEK}
C	N/A	CRT constant prime factor q	var.	SKU _{DEK}
C	N/A	CRT constant prime factor p	var.	SKU _{DEK}

Table 21-29: Data Content for DGI '8301' through '8305'

21.2.12 RFU DGIs

A range of DGIs is reserved for possible future use by this specification, and a range of DGIs is allocated for use with add-on functionality and proprietary use.

Req 21.22 (RFU DGIs):

DGIs of value '3xxx' shall be RFU for this specification.

Req 21.23 (Proprietary DGIs):

DGIs of value '4xxx' shall be reserved for proprietary use.

21.3 Requirements for Data Element Values

This section provides requirements on the personalisation values for application data elements.

21.3.1 Profile Selection Entries

Req 21.24 (No template tag in Profile Selection Entries):

The records containing Profile Selection Entries shall not contain a template tag.

21.3.2 AIP

The 'Issuer Authentication is supported' bit in the AIP is set to 0b to indicate that the card supports Issuer Authentication as part of processing the Second GENERATE AC command

Req 21.25 (Personalise at least one AIP/AFL):

An application shall be personalised with at least one AIP, AFL Length, and AFL to be used by the application.

21.3.3 AFL

Req 21.26 (Personalisation of VLP record data in VLP profile AFL):

If VLP is supported by the implementation, then the AFL used for Profile '7D' (that is, the AFL portion of AIP/AFL Entry x, where x is the value of the AIP/AFL ID in Profile Control '7D') shall include record 1 of SFI 11.

Req 21.27 (VLP record data not included in offline data authentication):

If VLP is supported by the implementation, then the AFL used for Profile '7D' (that is, the AFL portion of AIP/AFL Entry x, where x is the value of the AIP/AFL ID in Profile Control '7D') shall not include record 1 of SFI 11 in the list of records to be included in offline data authentication.

Req 21.28 (Profile Selection Entry records not personalised in AFL):

The records containing Profile Selection Entries shall not be listed in any AIP/AFL Entries used by the application.

21.3.4 VLP Profile Data

Req 21.29 (Personalisation of Optional CRM for VLP):

If VLP is supported by the implementation, then:

- *in the Issuer Options Profile Control x used for VLP (that is, where x is the value of the Issuer Options Profile Control ID in Profile Control '7D'):*
 - *for each Additional Check Table x supported by the application, the 'Activate Additional Check Table x Check' bit shall be personalised to the value 0b*
 - *the 'Activate Maximum Number of Days Offline Check' bit shall be personalised to the value 0b*
- *in Profile Control '7D', the MTA Profile Control ID shall have the value 'F' (that is, the Maximum Transaction Amount check is not active).*

Req 21.30 (SFI 11 format for VLP):

If VLP is supported by the implementation, then the file with SFI 11 shall be reserved for use by VLP Data. Each record shall begin with the template tag '70' and a single length byte, followed by TLV coded data.

Req 21.31 (SFI 11 data contents for VLP):

If the 'Allow VLP Processing' bit in the Application Control has the value 1b, (that is, the VLP Profile may be selected by the application), then the application shall be personalised with record 1 in SFI 11 beginning with the data elements listed in Table 21-30, in the order shown.

Data Element	Tag	Length
<i>VLP Issuer Authorisation Code</i>	<i>'9F74'</i>	<i>6</i>
<i>VLP Available Funds</i>	<i>'9F79'</i>	<i>6</i>

Table 21-30: VLP Record Data (SFI 11)

NOTE: Filler bytes are allowed in the personalisation data for the VLP Record Data.

Req 21.32 (Personalisation of other data for VLP):

If the 'Allow VLP Processing' bit in the Application Control has the value 1b, (that is, the VLP Profile may be selected by the application), then the application shall be personalised with:

- *Profile Control '7D'*
- *A PDOL that begins with the tag and length of the following data elements in the order shown in Table 21-31.*

Data Element	Tag	Length
<i>VLP Terminal Support Indicator</i>	<i>'9F7A'</i>	<i>1</i>
<i>Transaction Currency Code</i>	<i>'5F2A'</i>	<i>2</i>
<i>Amount Authorised</i>	<i>'9F02'</i>	<i>6</i>

Table 21-31: PDOL Data to Support VLP Transaction

21.3.5 Token Authentication Profile Data

Req 21.33 (Personalisation of Token Authentication profile selection):

*The Profile Selection Entries shall be personalised to ensure that the Token Authentication Profile (Profile '7E') shall only be selected for the transaction if **all** of the following are true:*

- *the issuer chooses to support the functionality,*
- ***and** the Terminal Type received in the GET PROCESSING OPTIONS command data has value '34',*
- ***and** the Additional Terminal Capabilities received in the first GET PROCESSING OPTIONS command data has value '0000'.*

Req 21.34 (Personalisation of PDOL for Token Authentication profile selection):

If the issuer chooses to support the Token Authentication Profile, then the PDOL shall contain the data elements listed in Table 21-32.

Data Element	Tag	Length
<i>Terminal Type</i>	<i>'9F35'</i>	<i>1</i>
<i>Additional Terminal Capabilities</i>	<i>'9F40'</i>	<i>2 to 5 ⁹</i>

Table 21-32: PDOL Data to Support Token Authentication

⁹ Additional Terminal Capabilities is a 5-byte data element in EMV, but only the first two bytes are required to support Token Authentication. To support Token Authentication, the length in the PDOL may be any value greater than or equal to 2 bytes.

21.3.6 Offline Counters

The following requirements will ensure CCD compliance for offline counters and accumulators.

Req 21.35 (Personalization of counters and accumulators for CCD-compliant profiles):

For each Profile Control personalised for the application, except Profile Control '7D' and Profile Control '7E', for which the profile is to comply with CCD:

- *If the Profile processes domestic transactions, at least one Accumulator x Control ID shall be personalised to a value y, where **all** of the following are true:*
 - *y does not equal 'F',*
 - ***and** the 'Allow Accumulation' bit in Accumulator Control y has the value 1b.*
 - ***and** the 'Include Offline Approvals' bit in Accumulator Control y has the value 1b.*
- *If the Profile processes transactions which could be approved offline without being accumulated in at least one Accumulator x, then the transactions that would not be accumulated must be counted in at least one Counter y.*

21.3.7 CIACs

The following requirements identify values that must be personalised in application data elements in order to ensure CPA is compliant with CCD.

Req 21.36 (Personalization of CIAC-Online for CCD-compliant profiles):

The following bits in the CIAC-Online shall be personalised to the value 1b for a profile to be CCD-compliant:

- *Go Online On Next Transaction Was Set*
- *Last Online Transaction Not Completed*
- *Accumulator 1 Lower Limit Exceeded*
- *Accumulator 2 Lower Limit Exceeded*
- *Counter 1 Lower Limit Exceeded*
- *Counter 2 Lower Limit Exceeded*
- *Counter 3 Lower Limit Exceeded*
- *Additional Accumulator Lower Limit Exceeded*
- *Additional Counter Lower Limit Exceeded*

Req 21.37 (Personalization of CIAC-Default for CCD-compliant profiles):

The following bits in the CIAC-Default shall be personalised to the value 1b (for a profile to be CCD-compliant:

- *Accumulator 1 Upper Limit Exceeded*
- *Accumulator 2 Upper Limit Exceeded*
- *Counter 1 Upper Limit Exceeded*
- *Counter 2 Upper Limit Exceeded*
- *Counter 3 Upper Limit Exceeded*
- *Additional Accumulator Upper Limit Exceeded*
- *Additional Counter Upper Limit Exceeded*

21.3.8 Signed Static Data

The issuer chooses which data items are included in the static signed data. This specification does not mandate a list. However, Table 21-33 indicates a minimum set that are recommended. Excluding items from this list may constitute a risk and the Issuer needs to carefully consider all such exclusions and be satisfied that any residual risks are acceptable.

Data Items	Tag	Reason for Inclusion
AIP	'82'	Omission allows attacker to avoid or alter offline data authentication and may allow alteration to terminal risk management and cardholder verification
Application Effective Date	'5F24'	Omission allows use of card beyond its authorised time window
Application Expiry Date	'5F25'	Omission allows use of card beyond its authorised time window
Application Usage Control	'9F07'	Omission may allow attacker to avoid usage restrictions of card
CDOL1 if card supports CDA	'8C'	Omission for CDA-capable applications may allow attacker to mislead the interpretation of the CDOL data
CDOL2 if card supports CDA	'8D'	Omission for CDA-capable applications may allow attacker to mislead the interpretation of the CDOL data
CVM list	'8E'	Omission may allow attacker to avoid cardholder verification
Issuer Action Code-Default	'9F0D'	Omission allows attacker to alter terminal action analysis (and thereby affect risk management)
Issuer Action Code-Denial	'9F0E'	Omission allows attacker to alter terminal action analysis (and thereby affect risk management)
Issuer Action Code-Online	'9F0F'	Omission allows attacker to alter terminal action analysis (and thereby affect risk management)
Issuer Country Code (tag 5F28)	'5F28'	Omission may allow attacker to avoid international restrictions
PAN	'5A'	Omission allows attacker to claim a bogus card Identity in offline transactions
PAN Sequence Number	'5F34'	Omission allows attacker to claim a bogus card Identity in offline transactions
SDA Tag List	'9F4A'	Omission allows attacker to avoid or alter offline data authentication and may allow alteration to terminal risk management and cardholder verification

Table 21-33: Static Data to be Authenticated

21.3.9 EMV Record Data

Req 21.38 (Personalization of EMV record data):

The following requirements apply to the organisation of those EMV records into files:¹⁰

- *At a minimum, the issuer shall be able to request up to 2K (2048) bytes of memory to store EMV records for CPA*
- *At a minimum, the issuer shall be able to store these bytes in any file with SFI 1-10 (for example in SFI 1 and 2, or in SFI 1, 3, and 4; or in SFI 5, 6, 8, and 9)*
- *At a minimum, the issuer shall be able to request up to a total of 16 records; with any number of records in each file, provided the total number of records is less than or equal to 16 (for example 2 records in file 1, 3 records in file 2,...)*
- *An issuer shall be able to request records of up to 254 bytes for each record.*

Each record in SFI 1 through 10 begins with template tag '70' and one or two length bytes, followed by the TLV-coded record data.

¹⁰ These are the minimum requirements that must be supported by any implementation of CPA. It is allowed for an implementation to support more than this minimum, but issuers should note that not all implementations may support more than the minimum.

21.3.10 CDOLs

Req 21.39 (Personalization of CDOL1):

CDOL1 shall begin with the tags and lengths of the data elements listed in Table 15-6 (except First Generate AC Extension Data), in the order shown. First Generate AC Extension Data is issuer-optional.

If First Generate AC Extension Data is requested to be included in the First GENERATE AC command data, then the tag and length of one or more additional data elements are included in CDOL1 after the tag and length for CVM Results.

Req 21.40 (Personalization of CDOL2 with Amounts included):

If the 'Amounts Included in CDOL2' bit in the Application Control has the value 1b (amounts are included), then CDOL2 shall begin with the tags and lengths of the data elements listed in Table 17-5 (except Second Generate AC Extension Data), in the order shown. Second Generate AC Extension Data is issuer-optional.

If Second Generate AC Extension Data is requested to be included in the Second GENERATE AC command data, then the tag and length of one or more additional data elements are included in CDOL2 after the tag and length for Amount, Other.

Req 21.41 (Personalization of CDOL2 without Amounts included):

If the 'Amounts Included in CDOL2' bit in the Application Control has the value 0b (amounts are not included), then CDOL2 shall begin with the tags and lengths of the data elements listed in Table 17-6 (except Second Generate AC Extension Data), in the order shown. Second Generate AC Extension Data is issuer-optional.

If Second Generate AC Extension Data is requested to be included in the Second GENERATE AC command data, then the tag and length of one or more additional data elements are included in CDOL2 after the tag and length for Unpredictable Number.

21.3.11 Keys

Req 21.42 (Expiration date for application):

An expiration date is assigned to each Issuer Public Key certificate. The application's expiration data shall not be greater than the expiration date of the certificate.

Req 21.43 (Personalization of Public Key data for SDA/DDA/CDA):

All cards which support SDA, DDA or CDA shall be personalised with an Issuer Public Key Certificate and a CA Public Key Index to identify the CA Public Key to use to decipher the certificate.

In EMV, the same Issuer Public/Private Keys and Issuer PK Certificates are used for SDA, DDA and CDA.

Req 21.44 (Use of public/private key data for Offline Enciphered PIN):

In all cards which support DDA or CDA, the ICC PK Certificate is personalised on the card. The ICC public/private key data may also be used to support the Offline Enciphered PIN method of cardholder verification described in Chapter 12, Cardholder Verification.

21.3.12 Previous Transaction History (PTH)

The issuer has the option to personalise the 'Go Online on Next Transaction' bit in the PTH to 1b to cause a new card to attempt to go online. The other bits are set to zero because applications are not required to use the value personalised for any other bit in the PTH as the initial value for a new card.

An implementation may support the PTH functionality in another way. However, an implementation must accept the personalisation data for the PTH, and must behave according to the value personalised for the 'Go Online on Next Transaction' bit in the PTH as if the PTH were implemented.

Req 21.45 (Support for personalization of PTH):

An application shall support personalisation of the PTH.

Req 21.46 (Personalization of Go Online on Next Transaction bit in PTH):

The value personalised for the 'Go Online on Next Transaction' bit in the PTH shall be used by the application as the initial value for a new card.

Req 21.47 (Personalization of other PTH bits):

All bits in the PTH other than the 'Go Online on Next Transaction' bit shall be personalised to the value 0b.

21.3.13 Log Entry

The Log Entry data element indicates the SFI to be used for logging transactions, and the maximum number of records to be supported in the transaction log file. If transactions are to be logged, then the application needs to be able to read the contents of the Log Entry data element.

Req 21.48 (Personalization of LOG Entry for application use):

If transactions are to be logged, then the Log Entry data element shall also be personalised for the application in addition to the personalisation of the FCI.

21.4 Missing Data Elements

If application data elements are not present in the application (for example, the data was not personalised, or is incorrectly formatted such that the application does not recognise the data), then the application behaviour cannot be correctly configured. This section specifies how the application is to behave when the necessary data element is missing.

Req 21.49 (Missing GPO Command Parameter):

If GPO Parameters x (that is, entry x in the GPO Parameters template that is to be used for the transaction) is not present in the application (where x identifies the entry in the GPO Parameters template that is to be used for the transaction); then the application shall discontinue processing the GET PROCESSING OPTIONS command, and shall respond with SW1SW2 = '6985' (Conditions of use not satisfied).

Req 21.50 (Missing Profile Control):

If Profile Control x is not present in the application (where x is the value of the Profile Control ID selected for the transaction); then the application shall discontinue processing the GET PROCESSING OPTIONS command, and shall respond with SW1SW2 = '6985' (Conditions of use not satisfied).

Req 21.51: (Missing Application Control)

If the Application Control is not present in the application, then the application shall discontinue processing the GET PROCESSING OPTIONS command, and shall respond with SW1SW2 = '6985' (Conditions of use not satisfied).

Req 21.52 (Missing CIACs):

*For any Profile Control x (where x has a value other than '7D' or '7E'), if **either** of the following is true:*

- *CIACs Entry y is not present (where y is the value of the CIACs ID in Profile Control x)*
- *or the CIACs ID in Profile Control x has the value 'F'*

then the application shall discontinue processing the GENERATE AC command, and shall respond with SW1 SW2 = '6985' (Conditions of use not satisfied).

Req 21.53 (Missing Accumulator Control):

If the Accumulator x Control is not present for any Accumulator x in the application, then the application shall:

- *use the value 'F' for the Accumulator Profile Control ID for Accumulator x in the transaction (that is, Accumulator x is not active for the transaction)*
- *set the 'Check Failed' bit in the ADR to the value 1b*
- *set the 'Check Failed' bit in the CVR to the value 1b*

Req 21.54 (Missing Accumulator Profile Control):

If Accumulator Profile Control y is not present for any Accumulator x in the application (where y is the value of the Accumulator Profile Control ID for Accumulator x in the Profile Control selected for the transaction), then the application shall:

- *use the value 'F' for the Accumulator Profile Control ID for Accumulator x in the transaction (that is, Accumulator x is not active for the transaction)*
- *set the 'Check Failed' bit in the ADR to the value 1b*
- *set the 'Check Failed' bit in the CVR to the value 1b*

Req 21.55 (Missing Counter Control):

If the Counter x Control is not present for any Counter x in the application, then the application shall:

- *use the value 'F' for the Counter Profile Control ID for Counter x in the transaction (that is, Counter x is not active for the transaction)*
- *set the 'Check Failed' bit in the ADR to the value 1b*
- *set the 'Check Failed' bit in the CVR to the value 1b*

Req 21.56 (Missing Counter Profile Control):

If the Counter Profile Control y is not present for any Counter x in the application (where y is the value of the Counter Profile Control ID for Counter x in the Profile Control selected for the transaction), then the application shall:

- *use the value 'F' for the Counter Profile Control ID for Counter x in the transaction (that is, Counter x is not active for the transaction)*
- *set the 'Check Failed' bit in the ADR to the value 1b*
- *set the 'Check Failed' bit in the CVR to the value 1b*

Req 21.57 (Missing Cyclic Accumulator Control):

If the Cyclic Accumulator x Control is not present for any Cyclic Accumulator x in the application, then the application shall:

- *use the value 'F' for the Cyclic Accumulator Profile Control ID for Cyclic Accumulator x in the transaction (that is, Cyclic Accumulator x is not active for the transaction)*
- *set the 'Check Failed' bit in the ADR to the value 1b*
- *set the 'Check Failed' bit in the CVR to the value 1b*

Req 21.58 (Missing Cyclic Accumulator Profile Control):

If Cyclic Accumulator Profile Control y is not present for any Cyclic Accumulator x in the application (where y is the value of the Cyclic Accumulator Profile Control ID for Cyclic Accumulator x in the Profile Control selected for the transaction), then the application shall:

- *use the value 'F' for the Cyclic Accumulator Profile Control ID for Cyclic Accumulator x in the transaction (that is, Cyclic Accumulator x is not active for the transaction)*
- *set the 'Check Failed' bit in the ADR to the value 1b*
- *set the 'Check Failed' bit in the CVR to the value 1b*

Req 21.59 (Missing Issuer Country Code):

If the Issuer Country Code is not present in the application, then for any Counter x that is active for the transaction and has the 'Include Only if International' bit in the Counter x Control set to the value 1b; the application shall:

- *use the value 'F' for the Counter Profile Control ID for Counter x in the transaction (that is, Counter x is not active for the transaction)*
- *set the 'Check Failed' bit in the ADR to the value 1b*
- *set the 'Check Failed' bit in the CVR to the value 1b*

Req 21.60 (Missing Application Currency Code):

If the Application Currency Code is not present in the application, then the application shall not select Profile ID '7D' (VLP Profile) to process the transaction.

Req 21.61 (Missing Additional Check Table):

If Additional Check Table x is not present in the application, and the 'Activate Additional Check Table x' bit in the Issuer Options Profile Control selected for the transaction has the value 1b; then the application shall:

- *process the transaction as if the 'Activate Additional Check Table x' bit in the Issuer Options Profile Control selected for the transaction has the value 0b (that is, Additional Check Table x is not active for the transaction)*
- *set the 'Check Failed' bit in the ADR to the value 1b*
- *set the 'Check Failed' bit in the CVR to the value 1b*

Req 21.62 (Missing Number of Days Offline Limit):

If Number of Days Offline Limit is not present in the application, and the 'Activate Maximum Number of Days Offline Check' bit in the Issuer Options Profile Control selected for the transaction has the value 1b; then the application shall:

- *process the transaction as if the 'Maximum Number of Days Offline Check' bit in the Issuer Options Profile Control selected for the transaction has the value 0b (that is, the Maximum Number of Days Offline Check is not active for the transaction)*
- *set the 'Check Failed' bit in the ADR to the value 1b*
- *set the 'Check Failed' bit in the CVR to the value 1b*

Req 21.63 (Missing AIP/AFL):

If AIP/AFL Entry x is not present in the application (where x is the value of the AIP/AFL ID in the Profile Control selected for the transaction); then the application shall discontinue processing the GET PROCESSING OPTIONS command, and shall respond with SW1SW2 = '6985' (Conditions of use not satisfied).

Req 21.64 (Missing Issuer Options Profile Control):

If Issuer Options Profile Control x is not present in the application (where x is the value of the Issuer Options Profile Control ID in the Profile Control selected for the transaction), and the Profile ID selected for the transaction does not have the value '7E'; then the application shall discontinue processing the GENERATE AC command, and shall respond with SW1SW2 = '6985' (Conditions of use not satisfied)

Req 21.65 (Missing MTA Profile Control):

If MTA Profile Control x is not present in the application (where x is the value of the MTA Profile Control ID in the Profile Control selected for the transaction), then the application shall:

- *use the value 'F' for the MTA Profile Control ID in the transaction (that is, the MTA Check is not active for the transaction)*
- *set the 'Check Failed' bit in the ADR to the value 1b*
- *set the 'Check Failed' bit in the CVR to the value 1b*

Req 21.66 (Missing VLP Profile Control):

If VLP Profile Control x is not present in the application (where x is the value of the VLP Profile Control ID in the Profile Control selected for the transaction), then the application shall use the default value '40 00' for VLP Profile Control x.

Req 21.67 (Missing Accumulator Limit):

If Limit Set y is not present for any Accumulator x in the application (where y is the value of the Limit Set ID in the Accumulator Profile Control for Accumulator x in the transaction), then the application shall:

- *use the value 'F' for the Accumulator Profile Control ID for Accumulator x in the transaction (that is, Accumulator x is not active for the transaction)*
- *set the 'Check Failed' bit in the ADR to the value 1b*
- *set the 'Check Failed' bit in the CVR to the value 1b*

Req 21.68 (Missing Counter Limit):

If Limit Set y is not present for any Counter x in the application (where y is the value of the Limit Set ID in the Counter Profile Control for Accumulator x in the transaction), then the application shall:

- *use the value 'F' for the Counter Profile Control ID for Counter x in the transaction (that is, Counter x is not active for the transaction)*
- *set the 'Check Failed' bit in the ADR to the value 1b*
- *set the 'Check Failed' bit in the CVR to the value 1b*

Req 21.69 (Missing Cyclic Accumulator Limit):

If Limit Entry y is not present for any Cyclic Accumulator x in the application (where y is the value of the Limit Entry ID in the Cyclic Accumulator Profile Control for Accumulator x in the transaction), then the application shall:

- *use the value 'F' for the Cyclic Accumulator Profile Control ID for Cyclic Accumulator x in the transaction (that is, Cyclic Accumulator x is not active for the transaction)*
- *set the 'Check Failed' bit in the ADR to the value 1b*
- *set the 'Check Failed' bit in the CVR to the value 1b*

Req 21.70 (Missing MTA Limit):

If Limit Entry x is not present in the application (where x is the value of the Limit Entry ID in the MTA Profile Control for the transaction), then the application shall:

- *use the value 'F' for the MTA Profile Control ID for the transaction (that is, the MTA Check is not active for the transaction)*
- *set the 'Check Failed' bit in the ADR to the value 1b*
- *set the 'Check Failed' bit in the CVR to the value 1b*

Req 21.71 (Missing Accumulator Currency Conversion Table):

If Currency Conversion Table y is not present for any Accumulator x in the application (where y is the value of the Currency Conversion Table ID in the Accumulator Profile Control for Accumulator x in the transaction), then the application shall use the value 'F' for the Currency Conversion Table ID for Accumulator x in the transaction (that is, currency conversion is not active for Accumulator x in the transaction).

Req 21.72 (Missing Cyclic Accumulator Currency Conversion Table):

If Currency Conversion Table y is not present for any Cyclic Accumulator x in the application (where y is the value of the Currency Conversion Table ID in the Cyclic Accumulator Profile Control for Cyclic Accumulator x in the transaction), then the application shall use the value 'F' for the Currency Conversion Table ID for Cyclic Accumulator x in the transaction (that is, currency conversion is not active for Cyclic Accumulator x in the transaction).

Req 21.73 (Missing MTA Currency Conversion Table):

If Currency Conversion Table x is not present in the application (where x is the value of the Currency Conversion Table ID in the MTA Profile Control for the transaction), then the application shall use the value 'F' for the Currency Conversion Table ID for the Maximum Transaction Amount Check in the transaction (that is, currency conversion is not active for the MTA Check in the transaction).

Part V

Annexes

Annex A Profile Selection File Processing

The application optionally uses the Processing Options Data Object List (PDOL) to request that the terminal send data in the command data of the GET PROCESSING OPTIONS (GPO) command. When the Profile Selection File functionality is activated, the application uses the content of the GPO command data and the Profile Selection File to determine the Profile ID of the Profile to be used in processing the transaction. A Profile ID may also indicate that the application should discontinue processing the GPO command and respond with an error status.

If the Profile Selection Using Card Data implementer-option is supported, the application also uses the Profile Selection Diversifier (PSD) contained in the GPO Parameter x used for the transaction as described in section A2.1, when processing the Profile Selection File.

A1 Profile Selection Entry

The Profile Selection File is a variable length file. It is the concatenation of a variable number of records (Profile Selection Entries). Figure A-1 illustrates the Profile Selection Entry format.

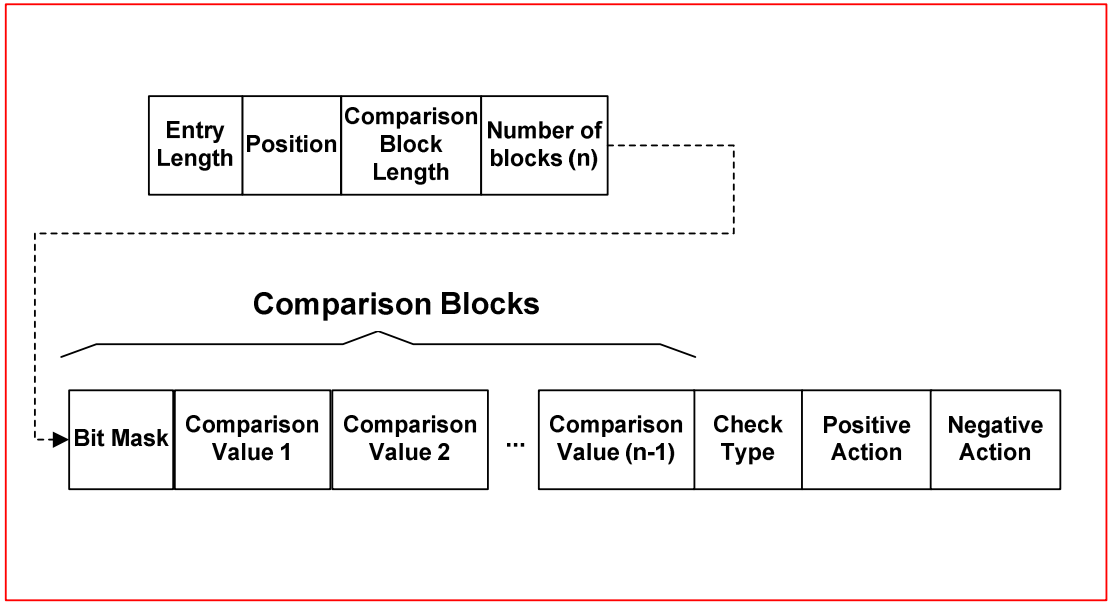


Figure A-1: Profile Selection Entry Format

The format of each Profile Selection Entry is as shown in Table A-1.

Data Element	Length	Description	
Entry Length	1	Indicates the length of the Profile Selection Entry (not including Entry Length).	
Position in GPO Command Data	1	Indicates the starting position (in bytes) of the portion of GPO command data that is compared to the Comparison Value(s) listed in the Profile Selection Entry. If the first byte of GPO command data is checked against the Comparison Value(s), then the value of Position in GPO Command Data is '01'.	
Comparison Block Length	1	Contains the length of the portion of GPO Command Data that is compared to the Comparison Value(s).	
Number of Comparison Blocks	1	Indicates the number of Comparison Blocks in the Profile Selection Entry. The first Comparison Block is a Bit Mask. The second and subsequent Comparison Blocks are Comparison Value(s) that are compared to the data extracted from the GPO Command Data.	
Comparison Blocks	var.	Contains the concatenation of the Bit Mask and one or more comparison value	
	Comparison Block Length	Bit Mask	Used to mask the comparison data to allow only selected portions of the extracted data to be compared with the Comparison Value(s) to use (for example, the comparison may be made with only a few bits of a byte). Each bit that is to be used in the comparison is set to 1b. Each bit that is not to be used in the comparison is set to 0b.
	Comparison Block Length	Comparison Value	Each Comparison Value data element contains a value to be compared to the masked data extracted from the GPO Command Data.

Table A-1: Data Elements in Profile Selection Entry
(continues)

Data Element	Length	Description
Check Type	1	<p>Identifies the type of test to be performed using the masked data extracted from the GPO command data and the Comparison Value(s). The Check Type is identified as follows:</p> <ul style="list-style-type: none">• Match (Check Type = '00') Tests whether the masked value extracted from the GPO command data is equal to the value of any of the Comparison Values in this Profile Selection Entry.• Less Than (Check Type = '01') Tests whether the masked value extracted from the GPO command data is less than the value of Comparison Value 1.• Greater Than (Check Type = '02') Tests whether the masked value extracted from the GPO command data is greater than the value of Comparison Value 1.

Table A-1: Data Elements in Profile Selection Entry, continued

Data Element	Length	Description
Positive Action	1	Indicates the action to be taken by the application if the Check Type test with the masked data extracted from the GPO Command Data and the Comparison Value(s) is true . (See format notes in Negative Action description.)
Negative Action	1	<p>Indicates the action to be taken by the application if the Check Type test with the masked data extracted from the GPO Command Data and the Comparison Value(s) is false.</p> <p>The Positive Action and Negative Action bytes indicate one of the following:</p> <ul style="list-style-type: none"> Profile ID Identifies the Profile to be used for the transaction: If b8 has the value 0b, then the Positive or Negative Action byte contains the Profile ID. NOTE: If the Profile ID has the value '7F', then the application will discontinue processing the GPO command and respond with SW1 SW2 = '6985'. Number of Profile Selection Entries to Skip Identifies the number of Profile Selection Entries to skip down in the Profile Selection File for the next Profile Selection Entry to process: If b8 has the value 1b, then the Profile Selection algorithm skips down x number of Profile Selection Entries, where x is the value indicated in bits b7 – b1 of the Positive or Negative Action byte.

Table A-1: Data Elements in Profile Selection Entry, continued

A2 Processing

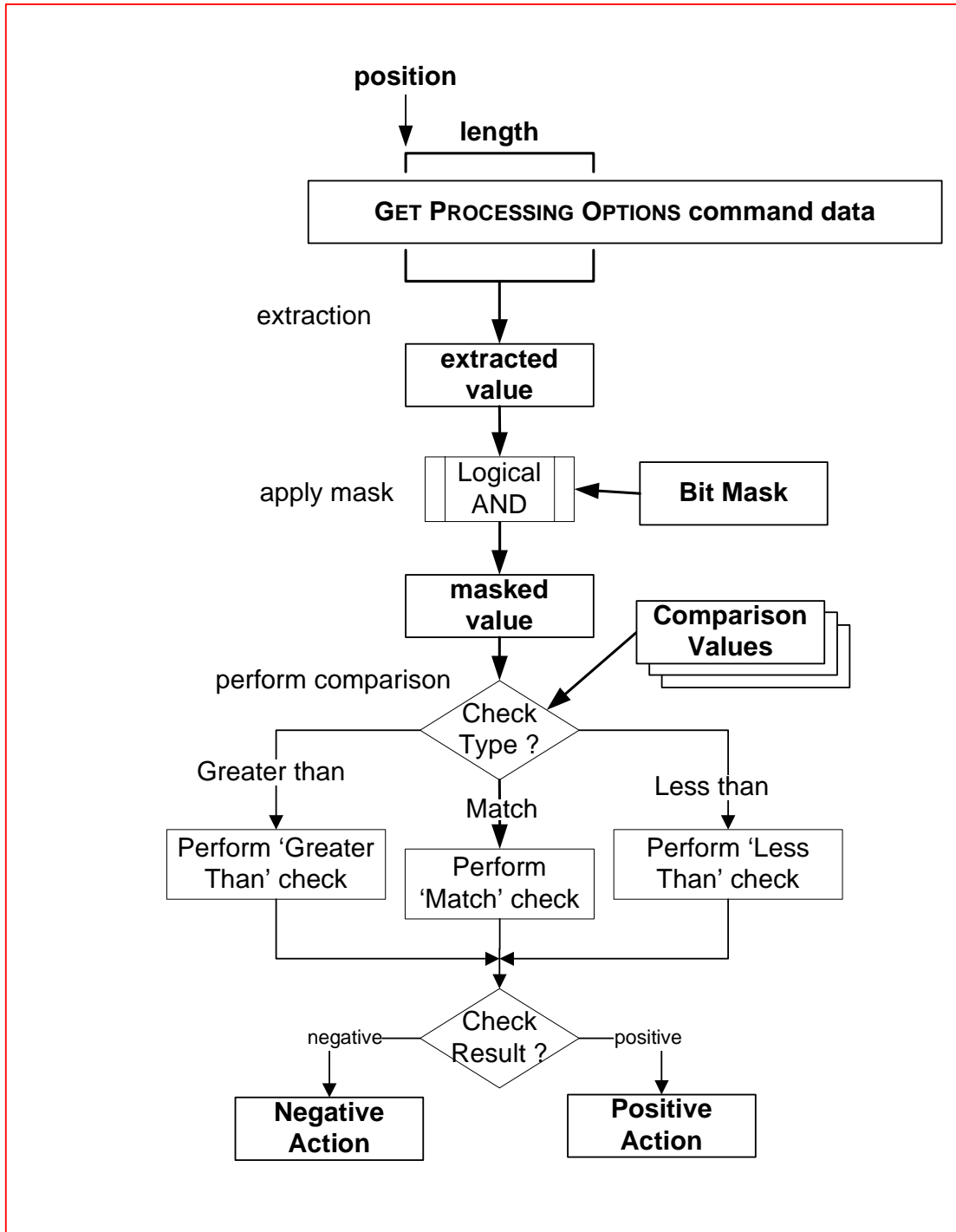


Figure A-2: Profile Selection Algorithm

The application processes each Profile Selection Entry in the order in which it appears in the Profile Selection File, starting with record 1, according to the following rules:

1. The application extracts a value from the GPO command data. The part to be extracted is defined at personalisation using two parameters in the Profile Selection Entry: Position in GPO Command Data and Comparison Block Length.
2. The application masks the extracted value with the Bit Mask (to force some bits to zero) and then compares the masked value with the values stored in the Profile Selection Entry. This allows for comparison of only a portion of a data element, such as a single bit in Terminal Capabilities.
3. The application performs the test indicated by the Check Type as follows:

Match (Check Type = '00')

The application tests whether the value extracted from the GPO command data is **equal to** any of the Comparison Value(s) in this Profile Selection Entry.

If a match is found, the Positive Action shall be performed.

If no match is found, the Negative Action shall be performed.

Less Than (Check Type = '01')

The application tests whether the value extracted from the GPO command data is **less than** Comparison Value 1.

If the value of the masked extracted data is less than the value of Comparison Value 1, the Positive Action shall be performed.

If the value of the masked extracted data is greater than or equal to the value of Comparison Value 1, the Negative Action shall be performed.

Greater Than (Check Type = '02')

The application tests whether the value extracted from the GPO command data is **greater than** Comparison Value 1.

If the value of the masked extracted data is greater than the value of Comparison Value 1, the Positive Action shall be performed.

If the value of the masked extracted data is less than or equal to the value of Comparison Value 1, the Negative Action shall be performed.

4. The Positive and Negative Action are each coded as follows:

If b8 has the value 0b, then the Profile ID used for the transaction shall have the value indicated by the (Positive or Negative) Action byte.

If b8 has the value 1b, then the profile selection algorithm shall skip down x number of Profile Selection Entries, where x is the value indicated in bits b7–b1 of the (Positive or Negative) Action byte.

If processing of the Profile Selection Entries does not result in selection of a Profile ID for which a Profile Control is present, then the Profile ID used for the transaction shall be '7F' (used to indicate an error in the GET PROCESSING OPTIONS response).

NOTE: If the Profile ID has the value '7F', then the application will discontinue processing the GPO command and respond with SW1 SW2 = '6985'.

A2.1 Profile Selection Using Card Data

If the Profile Selection Using Card Data implementer-option is supported, then the application inserts the one-byte Profile Selection Diversifier at the beginning of the GPO Command Data. The application performs Profile Selection File processing as described in section A2 by extracting data from the concatenation of the Profile Selection Diversifier and the GPO Command Data as shown in Figure A-3 (that is, as if the Profile Selection Diversifier was included in the GPO Command Data).

NOTE: Depending on the values for Position in GPO Command Data and Comparison Block Length, the Profile Selection Diversifier might not be included in the extracted data.

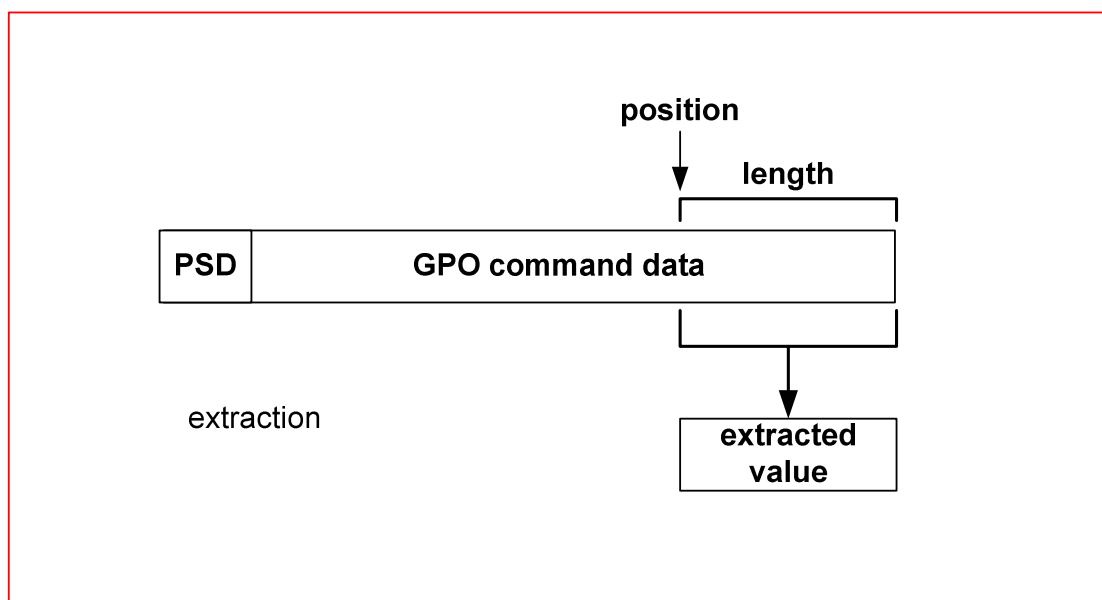


Figure A-3: Profile Selection Using Card Data

A3 Examples

NOTE: The following examples are coded for an application that supports the Profile Selection Using Card Data implementer-option. This implementer-option adds the Profile Selection Diversifier byte to the beginning of GPO Command Data before processing the Profile Selection Entries.

Simple

This example shows how the Profile Selection File could be configured to select:

- Profile '04' for domestic country transactions
- Default Profile (Profile ID = '01') for international transactions

NOTE: This would allow an issuer the capability to (for example) specify different CVM lists for domestic and international environments.

The simplified Profile Selection Entry logic is illustrated in Table A-2.

	Extracted Data	Comparative Value	Check Type	Positive Action	Negative Action
1	Terminal Country Code	Issuer Country Code ('0056')	Match	Select Profile '04'	Select Profile '01'

Table A-2: Profile Selection File – Simple Example

The PDOL and Profile Selection Entry data elements would be coded as follows:

- The value of the PDOL for this example is '9F1A02', consisting of the tag and length of the Terminal Country Code.
- The Profile Selection Entry for this example is '0A 02 02 02 FFFF 0056 00 04 01'.

Complex

This example shows how the Profile Selection File could be configured to select one of several Profiles based on the transaction environments specified. Table A-3 lists the profiles.

Transaction Environment	Selection Criteria	Profile ID
AID associated with Profile Selection Diversifier '02' is used to select the application	Profile Selection Diversifier = '02'	'02'
Unattended, online-only, Cardholder-controlled terminal with no Transaction Type capabilities in Additional Terminal Capabilities	(Terminal Type = '34') AND (Additional Terminal Capabilities bytes 1 and 2 = '0000')	'7E'
ATMs and attended cash disbursement terminals	(Terminal Type = '1X') AND (cash bit = 1b in Additional Terminal Capabilities)	'04'
Domestic online- and offline-capable POS terminals supporting DDA	(Terminal Country Code = Issuer Country Code) AND (DDA bit = 1b in Terminal Capabilities) AND ((Terminal Type = '22' or '25' or '35') OR ((Terminal Type = '12' or '15') AND (cash bit ≠ 1b in Additional Terminal Capabilities)))	'05'
POS terminals that are not online capable	(Terminal Type = '23' or '26' or '36') OR ((Terminal Type = '13' or '16') AND (cash bit ≠ 1b in Additional Terminal Capabilities))	'06'
All other terminals	(online-capable POS terminals)	'07'

Table A-3: Profiles for Complex Profile Selection File

The simplified list of Profile Selection Entries in Table A-4 shows that the five profiles can be selected with the profile selection algorithm:

	Extracted Data	Comparative Value	Check Type	Positive Action	Negative Action
1	Profile Selection Diversifier	'02'	Match	Select profile '02'	Skip down 1
2	Terminal Type	'34'	Match	Skip down 1	Skip down 2
3	Additional Terminal Capabilities (bytes 1 and 2)	'0000'	Match	Select profile '7E'	Skip down 1
4	Terminal Type	'1X' (using masking)	Match	Skip down 1	Skip down 2
5	Additional Terminal Capabilities	cash bit = 1b (using masking)	Match	Select profile '04'	Skip down 1
6	Terminal Country Code	Issuer Country Code	Match	Skip down 1	Skip down 5
7	Terminal Capabilities	DDA bit = 1b	Match	Skip down 1	Skip down 4
8	Terminal Type	'22' '25' '35'	Match	Select profile '05'	Skip down 1
9	Terminal Type	'12' '15'	Match	Skip down 1	Skip down 2
10	Additional Terminal Capabilities	Cash bit = 1b (using masking)	Match	Skip down 1	Select Profile '05'
11	Terminal Type	'23' '26' '36'	Match	Select profile '06'	Skip down 1
12	Terminal Type	'13' '16'	Match	Skip down 1	Select Profile '07'
13	Additional Terminal Capabilities	Cash bit = 1b (using masking)	Match	Select profile '07'	Select profile '06'

Table A-4: Profile Selection File – Complex Example

The PDOL and Profile Selection Entries data elements would be coded as follows:

- The value of the PDOL for this example is '9F35019F40029F1A029F3303', consisting of the tag and length of the data elements shown in Table A-5.

Data element	Tag	Length
Terminal Type	'9F35'	1
Additional Terminal Capabilities	'9F40'	2
Terminal Country Code	'9F1A'	2
Terminal Capabilities	'9F33'	3

Table A-5: PDOL Contents – Complex Example

- The coding of the Profile Selection Entries for this example is shown in Table A-6.

Profile Selection Entry	Value
1	'08 01 01 02 FF 02 00 02 81'
2	'08 02 01 02 FF 34 00 81 82'
3	'0A 03 02 02 FFFF 0000 00 7E 81'
4	'08 02 01 02 F0 10 00 81 82'
5	'08 03 01 02 80 80 00 04 81'
6	'0A 05 02 02 FFFF iii ¹ 00 81 85'
7	'0C 07 03 02 000040 000040 00 81 84'
8	'0A 02 01 04 FF 22 25 35 00 05 81'
9	'09 02 01 03 FF 12 15 00 81 82'
10	'08 03 01 02 80 80 00 81 05'
11	'0A 02 01 04 FF 23 26 36 00 06 81'
12	'09 02 01 03 FF 13 16 00 81 07'
13	'08 03 01 02 80 80 00 07 06'

Table A-6: Profile Selection Entries – Complex Example

¹ The term 'iii' represents the Issuer country code stored in the CPA application.

Annex B Additional Check Table Functionality

The Additional Check Table x is used to enable issuers to specify optional additional card risk management tests. When Additional Check Table functionality is activated, the application uses the content of the GENERATE AC command data and the Additional Check Table to perform optional card risk management. Checks can be applied to the value of any data element that can be requested from the terminal. CPA considers any matches found with Additional Check Table x when deciding whether to approve or decline the transaction offline, or to send the transaction online.

B1 Additional Check Table Content

The Additional Check Table x is the concatenation (without TLV coding) of the data elements identified in this section.

Figure B-1 illustrates the contents of the Additional Check Table x.

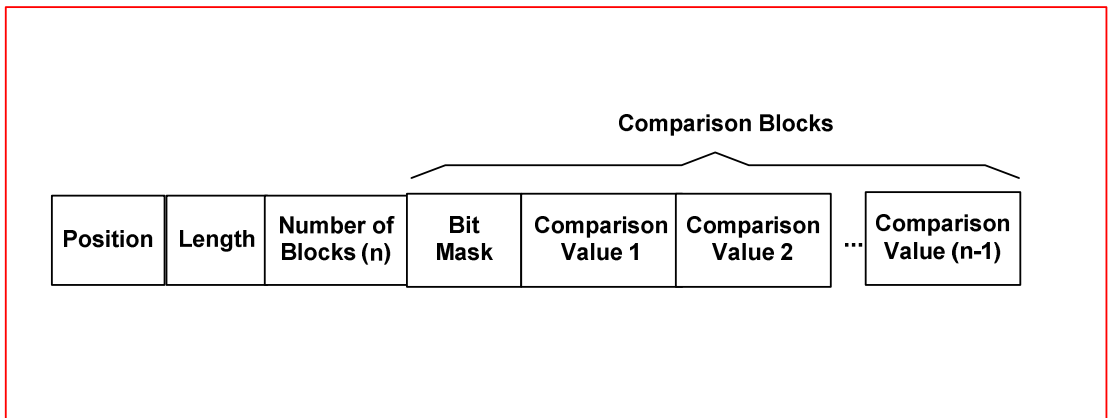


Figure B-1: Additional Check Table x

The format of each Additional Check Table x is as shown in Table B-1.

Data Element	Length	Description	
Position in First GENERATE AC Command Data	1	Indicates the starting position (in bytes) of the portion of First GENERATE AC command data that is compared to the Comparison Values listed in the Additional Check Table. If the first byte of First GENERATE AC command data is checked against the Comparison Values, then the value of Position in First GENERATE AC Command Data is '01'.	
Comparison Data Length	1	Contains the length of the portion of First GENERATE AC command data that is compared to the Comparison Values.	
Number Of Comparison Blocks (n)	1	Represents the number of Comparison Blocks in the Additional Check Table x. The first Comparison Block is a Bit Mask. The second and subsequent Comparison Blocks are Comparison Value(s) that are compared to the data extracted from the GENERATE AC command data.	
Comparison Blocks	var.	Contains the concatenation of the Bit Mask and the Comparison Values.	
	Comparison Data Length	Bit Mask	Used to mask the comparison data to allow only selected portions of the extracted data to be compared with the Comparison Value(s). (For example, the comparison may be made with only a few bits of a byte). Each bit that is to be used in the comparison is set to 1b. Each bit that is not to be used in the comparison is set to 0b.
	Comparison Data Length	Comparison Value	Each Comparison Value data element contains a value to be compared to the masked data extracted from the First GENERATE AC command data.

Table B-1: Additional Check Table x

B2 Processing Additional Check Table x

If the associated 'Activate Additional Check Table x' bit in the Issuer Options Profile Control for the transaction is personalised to the value 1b, then the application processes the Additional Check Table x by performing the following steps, illustrated in Figure B-2.

1. If there is a format error in the Additional Check Table, this test is skipped. Errors are:
 - The Position in First Generate AC Command Data is 0.
 - The Comparison Data Length is 0.
 - Position in First Generate AC Command Data + Comparison Data Length - 1 is greater than First Generate AC Command Data Length.
2. The application extracts a value from the first GENERATE APPLICATION CRYPTOGRAM command data field (First GENERATE AC Command Data). The part to be extracted from First GENERATE AC Command Data is defined at personalisation by setting the following parameters in the Additional Check Table x data element:
 - Position in First Generate AC Command Data
 - Comparison Data Length.
3. The application masks the extracted value with a Bit Mask to force some of the bits to 0b. That is, for each bit in the Bit Mask that has the value 0b, the corresponding bit in extracted value is set to 0b.
4. The application compares the masked value with each of the Values listed in the Additional Check Table x.
5. If the masked value matches any of the Values in the Additional Check Table, the application sets:
 - a 'Match Found in Additional Check Table x' bit in the ADR to 1b
 - a 'No Match Found in Additional Check Table x' bit in the ADR to 0b
 - the 'Match Found in any Additional Check Table' bit in the CVR to 1bOtherwise, the application sets:
 - a 'Match Found in Additional Check Table x' bit in the ADR to 0b
 - a 'No Match Found in Additional Check Table x' bit in the ADR to 1b

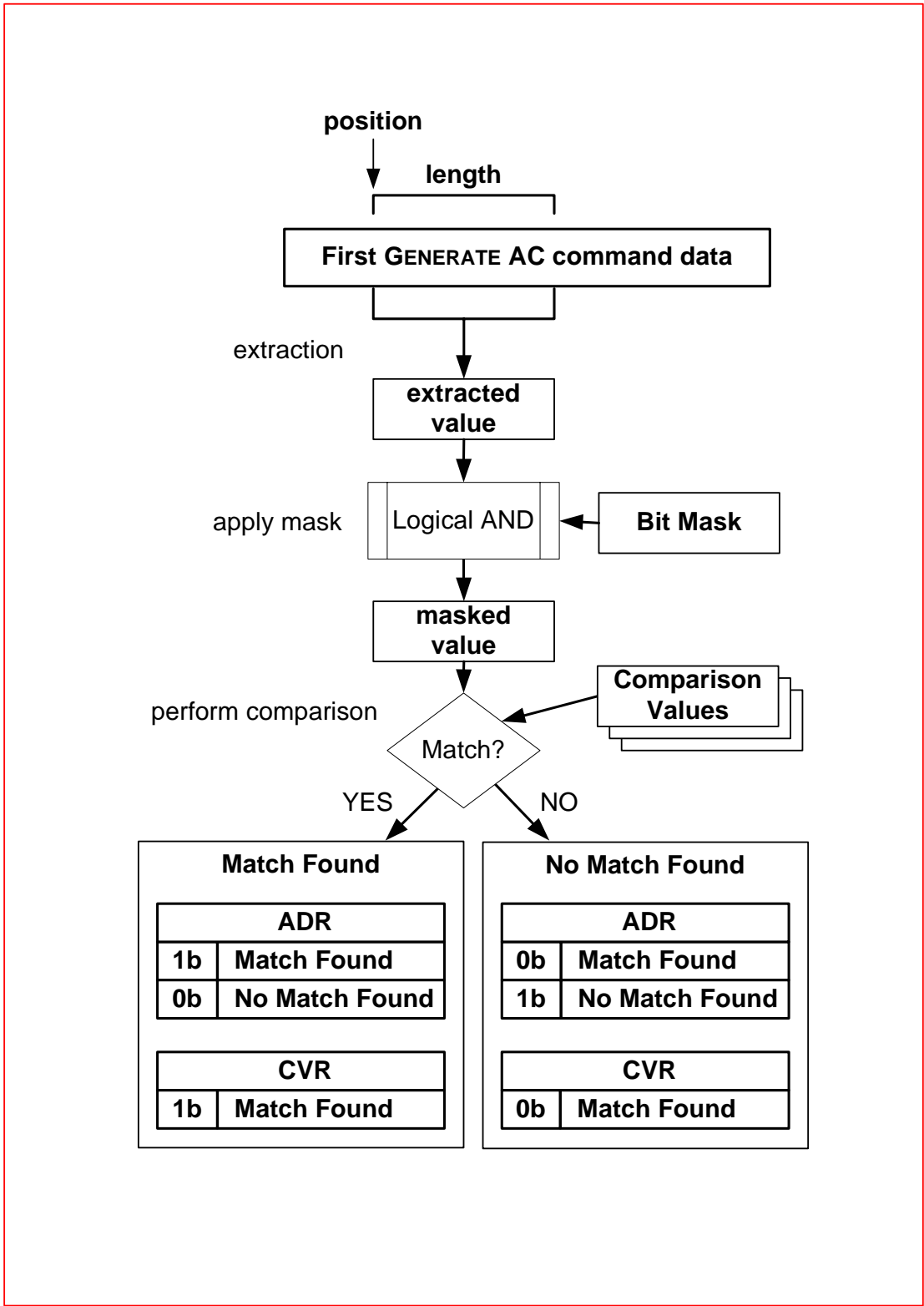


Figure B-2: Additional Check Table Usage

B3 Examples of Additional Check Table x Value

B3.1 Example 1: Country Code Check

In this example, the CPA is personalised to determine whether the value of the Terminal Country Code indicates that the transaction did not take place in either of the following countries:

- Belgium ('0056')
- France ('0250')

The tag and length for the Terminal Country Code data element are included in CDOL1 because it is required for generation of the Application Cryptogram.

The value of the Additional Check Table x is '0D0203FFFF00560250'. Table B-2 describes each of the sub-components of this value.

Data Element	Value	Description
Position in First GENERATE AC Command Data	'0D'	Terminal Country Code is located in the thirteenth byte of the First GENERATE AC Command Data; that is, '0D' in hexadecimal.
Comparison Data Length	'02'	The length of the Terminal Country Code is two bytes.
Number Of Comparison Blocks	'03'	The mask and the two values in the table used for the comparison (the Terminal Country Codes for Belgium and France) mean three comparison blocks are present.
Comparison Blocks		
Bit Mask	'FFFF'	The comparison is performed on the complete value of the Terminal Country Code. The Bit Mask is therefore equal to 'FFFF'.
Value 1	'0056'	The value of the country code for Belgium.
Value 2	'0250'	The value of the country code for France.

Table B-2: Example of Additional Check Table x Value

B3.2 Example 2: Force Decline

In this example, the CPA is personalised to enable the card to decline offline all transactions processed in a Profile where this check is active.

This example will use:

- Additional Check Table 2
- the first byte of the first data element in the command data

The application uses the Mask to force the Masked Value to '00', and compares it with the personalised Comparison Value 'FF'. This check will always result in the 'No Match Found in Additional Check Table 2' bit in the ADR being set to the value 1b. If the 'No Match Found in Additional Check Table 2' bit in the CIAC - Decline for this profile is also set to the value 1b, all transactions processed using this Profile will be declined offline.

The value of the Additional Check Table 2 is '01010200FF'. Table B-3 describes each of the sub-components of this value.

Data Element	Value	Description
Position in First GENERATE AC Command Data	'01'	The first byte of the First GENERATE AC Command Data.
Comparison Data Length	'01'	Use only one byte of the data for the comparison.
Number Of Comparison Blocks	'02'	Use only the Mask, and the single Comparison Value.
Comparison Blocks		
Bit Mask	'00'	The Mask is used to force the Masked Value to '00'.
Value 1	'FF'	The comparison value 'FF' will not match the Masked Value.

Table B-3: Example of Additional Check Table x Value

Annex C Currency Conversion Functionality

The CPA application supports accumulation of transactions conducted in multiple currencies in an accumulator based on a single currency, if the Issuer personalises the Currency Conversion Table data element in the application as described below. The accumulator accumulates in the single currency associated with the accumulator by converting amounts from other specified currencies. This applies to transactions:

- performed in the accumulator currency
- performed in any of the currencies personalised in the Currency Conversion Table

NOTE: If the Currency Conversion function is used in the application, the amount converted to the accumulator currency is only an approximation of the amount of the transaction in the accumulator currency based on the conversion rate in the Currency Conversion Parameter.

NOTE: Currency Conversion may also be used to convert Amount, Authorised to the currency associated with the Maximum Transaction Amount.

C1 Currency Conversion Table Data Element

The Currency Conversion Table data element is the concatenation of the Target Currency Code and one or more Currency Conversion Parameters. Each Currency Conversion Parameter is coded as shown in Annex L: Data Dictionary.

Data	Length	Description
Source Currency Code	2	The Currency Code of the currency to be converted to the currency identified by the Target Currency Code for the accumulator using this Currency Conversion Parameter.
Conversion Rate	2	The rate (used with the Conversion Exponent) multiplied with the transaction amount value to approximate the transaction value in the accumulator currency.
Conversion Exponent	1	<p>A signed number that indicates the power of 10 used to modify the Conversion Rate. Bit b8 indicates the sign of the exponent, and bits b7 through b1 indicate the value of the exponent.</p> <p>If the sign is positive (b8 = 0b):</p> $\text{Approximate Value} = \text{Transaction amount} * \text{Conversion Rate} * 10^{\text{Conversion Exponent (b7 to b1)}}$ <p>If the sign is negative (b8 = 1b):</p> $\text{Approximate Value} = (\text{Transaction amount} * \text{Conversion Rate}) / 10^{\text{Conversion Exponent (b7 to b1)}}$

Table C-1: Currency Conversion Parameter

C2 Example

These examples assume the Currency Conversion Parameter values listed in Table C-2. The target currency in this example is the USD (U.S. Dollar).

Conversion Parameter 1		Conversion Parameter 2	
Data	Value	Data	Value
JPY (Japanese Yen)	'0392'	GBP (British Pounds)	'0826'
Rate: 1 JPY = 0.85 USD ²	'0085'	Rate: 1 GBP = 1.8 USD	'0018'
Conversion Exponent	'82'	Conversion Exponent	'81'

Table C-2: Example Currency Conversion Parameters

For Conversion Parameter 1, the Conversion Exponent value of '82' is the equivalent of 1000 0010b in binary representation. 1b in bit 8 indicates the sign is negative, 000 0010b in bits 7 through 1 indicates the power of 10 is two. Thus, the conversion amount is divided by 10^2 (that is, 10 to the power of 2).

To convert a transaction amount of '00 00 00 05 55 55' in JPY (55555 Japanese Yen) to the accumulator currency (USD) using Conversion Parameter 1:

Transaction amount in JPY: '000000055555'

Transaction currency code '0392'

amount converted to accumulator currency =

$$(000000055555 \times 0085) / 10^2 = '000000047222' (\$472.22)$$

For Conversion Parameter 2, the Conversion Exponent value of '81' is the equivalent of 1000 0001b in binary representation. 1b in bit 8 indicates the sign is negative, 000 0001b in bits 7 through 1 indicates the power of 10 is one. Thus, the conversion amount is divided by 10^1 (that is, 10 to the power of 1).

To convert a transaction amount of '00 00 00 00 01 25' in GBP (£ 1.25) to the accumulator currency (USD) using Conversion Parameter 2:

Transaction amount in GBP: '000000000125'

Transaction currency code '0826'

$$\begin{aligned} \text{amount converted to accumulator currency} &= (000000000125 \times 0018) / 10^1 \\ &= '000000000225' (\$2.25) \end{aligned}$$

² Note that the Yen has no implied decimal point, so an amount of “300” with the JPY currency code is 300 Yen. The US Dollar has an implied decimal point before the last two digits, so an amount of “200” with the USD currency code is equivalent to \$2.00. So if 1 Yen = \$0.0085, then 1 JPY = 0.85 USD.

Annex D Transaction Logging

CPA supports logging as described in EMV Book 3, Annex D and in this annex.

The Transaction Log file is a file having a cyclic structure with records of fixed length. Record #1 corresponds to the most recent transaction. Record #2 is the next prior transaction, etc.

Logging occurs at the end of the first GENERATE AC processing for transactions approved or declined offline in the first GENERATE AC and at the end of the second GENERATE AC processing for transactions sent online for authorisation.

D1 Transaction Log Entry Description

Each entry of the transaction log file is the concatenation of values identified in the Log Format data element. These values are obtained from card and terminal data as described in Table D-1.

Source	Data Element	Description
Mandatory terminal data (received as part of First or Second GENERATE AC command data) Note: If a transaction is to be logged, this data is always included in the Transaction Log record for the transaction.	Amount, Authorised	Current transaction amount received from the terminal as part of First/Second GENERATE AC command data.
	Transaction Currency Code	Current transaction currency code received from the terminal as part of First GENERATE AC command data.
	Transaction Date	Transaction date received from the terminal as part of First GENERATE AC command data.
Optional Card data Note: Logging of optional card data is conditional on the Application Control settings related to Transaction Logging.	CVR	CVR resulting from the offline Card Risk Management that is returned to the terminal in the GENERATE AC command response.
	ATC	Current ATC value
	CID	Application CID returned to the terminal in the GENERATE AC command response.
Optional terminal data (received as part of First or Second GENERATE AC command data) Note: If the Log Data Tables are either not present or empty, then no optional terminal data is logged for the transaction.	First GEN AC Unchanging Log Data Table Extracted Data	Data extracted from First GENERATE AC command data (that is, data associated with the tags and lengths listed in CDOL1) that are logged in both First GENERATE AC and Second GENERATE AC
	First GEN AC Log Data Table Extracted Data	Data extracted from First GENERATE AC command data (that is, data associated with the tags and lengths listed in CDOL1) that is logged in first GENERATE AC command processing.
	Second GEN AC Log Data Table Extracted Data	Data extracted from second GENERATE AC command data (that is, data associated with the tags and lengths listed in CDOL2) that is logged in second GENERATE AC command processing.

Table D-1: Structure of Log Format data element

D2 Issuer Options for Transaction Logging

At personalization, the issuer specifies how the card will support transaction logging. Byte 3 of the Application Control data element controls the following options:

Option	Description
Log Declined Transactions	This option indicates whether or not a transaction is logged if the CPA application responds with an AAC to the GENERATE AC command. This option applies to logging during both the first and second GENERATE AC command processing.
Log Approved Transactions	This option indicates whether or not a transaction is logged if the CPA application responds with a TC to the GENERATE AC command. Whether an approved transaction is logged during the first GENERATE AC command depends on the 'Log Offline Only' option. Whether an approved transaction is logged during the second GENERATE AC command depends on the 'Log Offline Only' option and whether the terminal indicated it was unable to go online.
Log Offline Only	This option applies to the Log Approved Transactions option, and indicates whether the CPA application logs only offline approved transactions or both offline and online approved transactions.
Log the ATC	This option indicates whether or not the ATC is logged.
Log the CID	This option indicates whether or not the CID is logged.
Log the CVR	This option indicates whether or not the CVR is logged.

Table D-2: Transaction Logging Options in Application Control byte 3

Apart from the general transaction logging options specified in the Application Control data element, the 'Log Transactions' bit in the Issuer Options Profile Control indicates whether transaction logging is enabled or disabled in the Profile used for the current transaction. If transaction logging is enabled, the options specified in Application Control (that is, Table D-2) will apply.

D3 Log Data Tables

CPA supports flexible logging of transaction data using the First GEN AC Log Data Table, First GEN AC Unchanging Log Data Table, and Second GEN AC Log Data Table data elements. Each of these tables has the format described in Table D-3.

Data Element	Length	Description
Count of Data Entries, n	1	The number of the following data entries that are logged after ICC-sourced data in the Log Entry.
Data Entry 1	2	The position (first byte) and length (second byte) of the 1st block of additional terminal-sourced log data in the GENERATE AC command data.
...	var.	
Data Entry n	2	The position and length of the nth block of additional terminal-sourced log data in the GENERATE AC command data.

Table D-3: Log Data Table Format

Data extracted from First GENERATE AC Command Data (using First GEN AC Unchanging Log Data Table) is data that will not change during a transaction. The value given by the terminal in First GENERATE AC Command Data is then used for logging in both First GENERATE AC and Second GENERATE AC command.

For data that might change during the transaction (that is, the value provided by the terminal might be different between the First GENERATE AC and Second GENERATE AC command) two additional tables are used; the First GEN AC Log Data Table and the Second GEN AC Log Data Table. Both tables are necessary because the data may be in different positions in the First GENERATE AC Command Data than in the Second GENERATE AC Command Data

Data extracted from First GENERATE AC Command Data (using First GEN AC Log Data Table) are only logged in First GENERATE AC command. For the Second GENERATE AC command the same data elements have to be requested again from the terminal to obtain the new value. This new value is extracted from Second GENERATE AC Command Data (using Second GEN AC Log Data Table) to be logged in Second GENERATE AC Command.

NOTE: The positions of data elements in the Second GEN AC Log Data Table may differ from the positions of the same data elements in the First GEN AC Log Data Table.

D3.1 Log Data Table processing

In this section, First GEN AC Log Data Table is used to illustrate the processing of each Log Data Table. In this example, the values of the Terminal Verification Results (TVR) and Additional Terminal Capabilities data elements are logged by the CPA application.

Table D-4 shows that First GEN AC Log Data Table is personalized with the value '02XX05YY05'. This value indicates how Terminal Verification Results (TVR) and Additional Terminal Capabilities data elements values are extracted from First Generate AC command data.

Value	Description
2	The number of the following data entries that are logged after ICC-sourced data in the Log Entry.
XX	Position of the first byte of TVR in the First GENERATE AC command data.
5	Length of the TVR data to be logged from the First GENERATE AC command data.
YY	Position of first byte of Additional Terminal Capabilities in the First GENERATE AC command data.
5	Length of Additional Terminal Capabilities data to be logged from the First GENERATE AC command data.

Table D-4: Example of GEN AC Log Data Table

As indicated in Figure D-1, the CPA application processes each First GEN AC Log Data Table entry in order to extract a data element value to be logged.

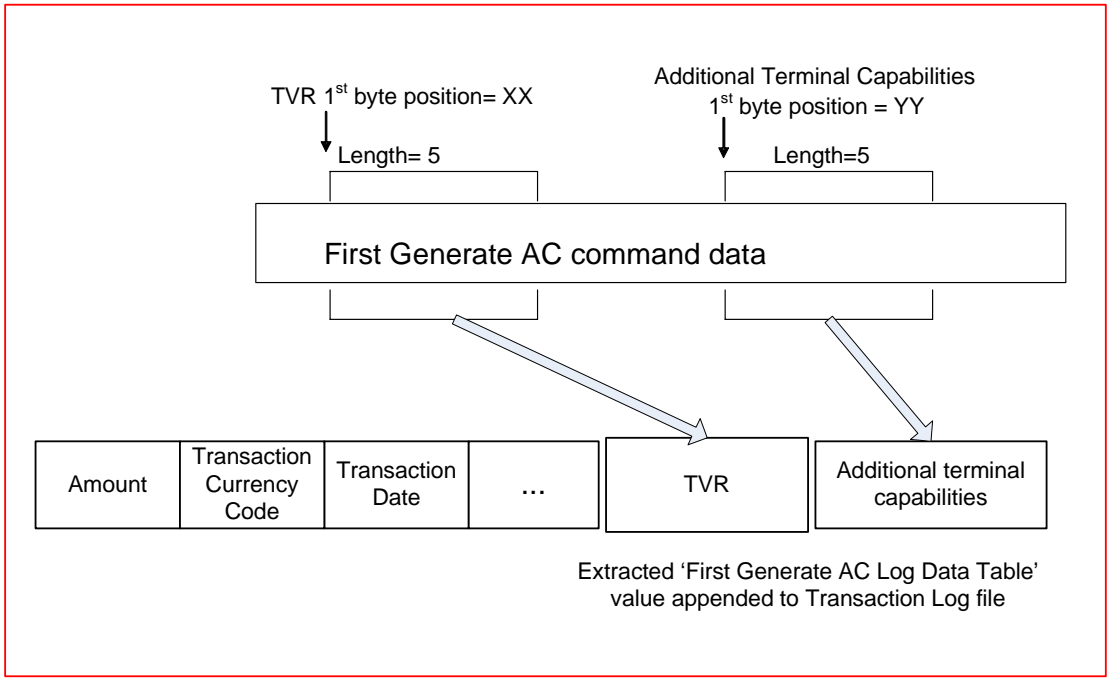


Figure D-1: Processing First GEN AC Log Data Table

D4 Processing Transaction Logging

Transaction logging occurs as follows:

- when the first GENERATE AC response is a TC or an AAC, prior to responding to the first GENERATE AC command
- when the first GENERATE AC response is an ARQC, prior to responding to the second GENERATE AC command

D4.1 First GENERATE AC Transaction Logging

If the application responds with a TC/AAC and the issuer chooses to log such transactions, a record with the information listed in Table D-5 is appended to the Transaction Log.

Data to Log	Condition
Amount, Authorised	always
Transaction Currency Code	always
Transaction Date	always
CVR	if 'Log the CVR' in Application Control=1b
ATC	if 'Log the ATC' in Application Control=1b
CID	if 'Log the CID' in Application Control=1b
First GEN AC Unchanging Log Data Table Extracted Data	if any
First GEN AC Log Data Table Extracted Data	if any

Table D-5: Data Logged at First GENERATE AC for a TC or ACC

If the application responds with an ARQC, the CPA application temporarily saves the data listed in Table D-6 so that it can be logged during second GENERATE AC transaction logging.

Data to Log	Condition
Amount, Authorised	if 'Amounts included in CDOL2' of Application Control= 0b
Transaction Currency Code	always
Transaction Date	always
First GEN AC Unchanging Log Data Table Extracted Data	if any

Table D-6: Data Saved for Second GENERATE AC after an ARQC

D4.2 Second GENERATE AC Transaction Logging

Prior to responding to the second Generate AC command, the CPA application appends the data listed in Table D-7 to the Transaction Log file.

Data to Log	Condition
Amount, Authorised	always
Transaction Currency Code	always
Transaction Date	always
CVR	if 'Log the CVR' in Application Control = 1b
ATC	if 'Log the ATC' in Application Control = 1b
CID	if 'Log the CID' in Application Control = 1b
First GEN AC Unchanging Log Data Table Extracted Data	if any
Second GEN AC Log Data Table Extracted Data	if any

Table D-7: Data Logged at Second GENERATE AC

D5 Example

The following example illustrates the use of the data elements and options associated with transaction logging.

Transaction Logging Settings

The Application Control byte 3 indicates the transaction logging settings illustrated in Table D-8.

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
1								Log Declined Transactions
	1							Log Approved Transactions
		0						Log Offline Only
			1					Log the ATC
				1				Log the CID
					0			Log the CVR
						x		RFU
							x	RFU

Table D-8: Example – Transaction Logging Settings in Application Control

Transaction Parameters

- The 'Amounts included in CDOL2' bit of the Application Control data element has the value 0b.
- First GENERATE AC Command Data has the value '0000000100000000000000008400000010000840051101001122334411010002FF80F0F3FF', and contains the data elements listed in Table D-9.

Data Element	Value
Amount, Authorised	'000000010000'
Amount Other	'0000000000000'
Terminal Country Code	'0840' (USA)
TVR	'0000001000'
Transaction Currency Code	'0840'
Transaction Date	'051101' (1 November 2005)
Transaction Type	'00'
Unpredictable Number	'11223344'
Terminal Type	'11'
CVM Results	'010002'
Additional Terminal Capabilities	'FF80F0F3FF'

Table D-9: Example – First GENERATE AC Command Data

- Second GENERATE AC Command Data has the value '9C77066103FF00003030000000000044444444', and contains the data elements listed in Table D-10.

Data Element	Value
Issuer Authentication Data	'9C77066103FF0000'
Authorisation Response Code	'3030'
TVR	'0000000000'
Unpredictable Number	'44444444'

Table D-10: Example – Second GENERATE AC Command Data

- First GEN AC Unchanging Log Data Table has the value '020D022205', which indicates the positions of the following data elements in First GENERATE AC Command Data:
 - Terminal Country Code
 - Additional Terminal Capabilities
- First GEN AC Log Data Table has the value '010F05', which indicates the position of the TVR data element in First GENERATE AC Command Data
- Second GEN AC Log Data Table has the value '010B05', which indicates the position of the TVR data element in Second GENERATE AC Command Data

First Generate AC Transaction Logging

During First GENERATE AC command processing, the CPA application extracts from First GENERATE AC Command Data the data elements referenced in First GEN AC Unchanging Log Data Table and First GEN AC Log Data Table as indicated in tables D-11 and D-12.

Data Element	Value
Terminal Country Code	'0840'
Additional Terminal Capabilities	'FF80F0F3FF'

Table D-11: Example – Data Referenced in First GEN AC Unchanging Log Data Table

Data Element	Value
TVR	'0000001000'

Table D-12: Example – Data Referenced in First GEN AC Log Data Table

If the CPA application returns an ARQC to the first GENERATE AC command, this results in the temporary storage of the data elements indicated in Table D-13 so that it can be used for transaction logging during processing of the second GENERATE AC command.

Source	Data Element	Value
Terminal Data (received as part of the first GENERATE AC command)	Amount, Authorised	'000000010000'
	Transaction Currency Code	'0840'
	Transaction Date	'051101'
First GEN AC Unchanging Log Data Table extracted data (as indicated in Table D-11)	Terminal Country Code	'0840'
	Additional Terminal Capabilities	'FF80F0F3FF'

Table D-13: Example – Data Saved if First GENERATE AC Response is an ARQC

If the CPA returns a TC/AAC to the first GENERATE AC command, the CPA application appends the following data to the final Transaction Log file entry.

Source	Data Element	Value
Terminal Data (received as part of the first GENERATE AC command)	Amount, Authorised	'000000010000'
	Transaction Currency Code	'0840'
	Transaction Date	'051101'
	ATC	'001C'
	CID	'40' or '00'
First GEN AC Unchanging Log Data Table extracted data (as indicated in Table D-11)	Terminal Country Code	'0840'
	Additional Terminal Capabilities	'FF80F0F3FF'
First GEN AC Log Data Table extracted data (as indicated in Table D-12)	TVR	'0000001000'

Table D-14: Example – Data Logged at First GENERATE AC for a TC or AAC

Second Generate AC Transaction Logging

During second GENERATE AC command processing, the CPA application extracts from Second GENERATE AC Command Data the data elements referenced in Second GEN AC Log Data Table as indicated in Table D-15.

Data Element	Value
TVR	'0000000000'

**Table D-15: Example – Data Extracted from
Second GENERATE AC Command Data**

Prior to responding to the second GENERATE AC command, the CPA application appends the data listed in Table D-16 to the final Transaction Log file entry.

Source	Data Element	Value
Terminal Data (received as part of the second GENERATE AC command)	Amount, Authorised	'000000010000'
	Transaction Currency Code	'0840'
	Transaction Date	'051101'
	ATC	'001C'
	CID	'40'
First GEN AC Unchanging Log Data Table extracted data (as indicated in Table D-11)	Terminal Country Code	'0840'
	Additional Terminal Capabilities	'FF80F0F3FF'
Second GEN AC Log Data Table extracted data (as indicated in Table D-12)	TVR	'0000000000'

Table D-16: Example – Transaction Data Logged at Second GENERATE AC

Annex E Management of Dates in Days

E1 Date Conversion

The following algorithm could be used in the number of days offline computation to convert a transaction date in the EMV format (YYMMDD) into a transaction date in days (the number of days elapsed since a reference date, Day 0). It could also be used to determine a start date for a Cyclic Accumulator's weekly cycle.

NOTE: Throughout this annex, whole number calculations are described. The result of any division shall be truncated to a whole number.

NOTE: The leap year conversion routine is limited and will not work in other centuries (that is, where the first two digits for the year are not 20). The algorithms associated with converting dates to days assume the dates are within the same century (years of the format 20xx).

NOTE: The default 'Day 0' is the 31st of December 1999.

NOTE: Selection of a different reference day 0 is permitted (for example to keep the value of the date in days small, allowing for more efficient calculations). However, the algorithms described in this section would need to be modified to adjust for the different reference day 0.

To compute the number of days elapsed in the **previous years**:

$$\text{Number of days in previous years} = 365 * YY + (YY + 3) / 4$$

(The second term counts the extra days of leap years.)

To compute the number of days elapsed in the **previous months in the current year**:

If MM > 2 and YY is a multiple of 4, then Number of days in previous months = (month table entry MM) + 1

Else Number of days in previous months = month table entry MM

The month table is then:

{0, 31, 59, 90, 120, 151, 181, 212, 243, 273, 304, 334}

The number of days elapsed since the 31st December 1999, is then:

$$\begin{aligned} \text{transaction date in days} = & [\text{Number of days in previous years}] \\ & + [\text{Number of days in previous months}] \\ & + [\text{Number of days in current month}] \end{aligned}$$

If a different (later) date is used as 'Day 0' the equation becomes:

$$\begin{aligned} \text{transaction date in days} = & [\text{Number of days in previous years}] \\ & + [\text{Number of days in previous months}] \\ & + [\text{Number of days in current month}] \\ & - [\text{Adjustment for Day 0 after 31 December 1999}] \end{aligned}$$

Examples:

1 st January, 2006 (YYMMDD)	= 060101)
transaction date in days	= [365*6 + (6 +3)/4] + [0] + 1
	= 2193
27th March, 2012 (YYMMDD)	= 120327)
transaction date in days	= [365*12 + (12+3)/4] + [59 + 1] + 27
	= 4470

E2 Computation of Reference Day

When a Cyclic Accumulator uses a weekly cycle, the beginning of a cycle will always be on the same day of the week, as specified in the First Day of Cycle element in the Cyclic Accumulator x Control (see Annex L). If the Transaction Date falls on the day of the week specified in First Day of Cycle, then the Cyclic Accumulator Reference Day is the same as the Transaction Date in Days. If the Transaction Date falls on any other day of the week, then the Cyclic Accumulator Reference Day is equivalent to the day of the week specified by First Day of Cycle immediately preceding the Transaction Date in Days previously calculated.

The general equation for calculating a Reference Day from a Transaction Date is the following:

$$(\text{Transaction Date in Days} - \text{First Day in Cycle}) / 7 * 7$$

Subtracting the First Day in Cycle value from the Transaction Date in Days shifts the reference Day 0 to the following day of the week specified by First Day of Cycle. The integer division and multiplication operations will then produce the weekday corresponding to First Day of Cycle which immediately precedes the Transaction Date.

E2.1 Example - Week Starting on Monday

This example assumes each weekly cycle begins on a Monday. To compute the Reference Day (a Monday) from a date in days, the following calculations are performed:

$$\text{Reference Day 0 is} = (\text{transaction date in days} - 3) / 7 * 7$$

New 'day 0' becomes Monday, 3rd January, 2000. Because the reference date is a Monday, any date in days (adjusted for the First Day of Week) that can be divided by 7 with no remainder is also a Monday. The new Reference Day will thus be the Monday at the beginning of the week in which the Transaction Date falls.

Examples:

Transaction Date: Sunday, 1st January 2006 (YYMMDD = 060101)

Transaction Date in days = 2193

$$\text{Transaction Date Reference Day} = (2193-3) / 7 * 7 = 2184$$

Transaction Date: Tuesday, 27th March 2012 (YYMMDD = 120327)

Transaction Date in days = 4470

$$\text{Transaction Date Reference Day} = (4470-3) / 7 * 7 = 4466$$

Transaction Date: Monday, 11th February 2008 (YYMMDD = 080211)

Transaction Date in days = 2964

$$\text{Transaction Date Reference Day} = (2964-3) / 7 * 7 = 2961$$

E2.2 Example - Week Starting on Sunday

This example assumes each weekly cycle begins on a Sunday. To compute the Reference Day (a Sunday) from a date in days, the following calculations are performed:

$$\text{New Reference Day is} = (\text{transaction date in days} - 2) / 7 * 7$$

New Day 0 becomes Sunday, 2nd January, 2000. Any date in days (adjusted for First Day of Week) that can be divided by 7 with no remainder is also a Sunday. The new Reference Day will thus be the Sunday at the beginning of the week in which the Transaction Date falls.

Examples:

Transaction Date: Sunday, 1st January 2006 (YYMMDD = 060101)

Transaction Date in days = 2193

$$\text{Transaction Date Reference Day} = (2193 - 2) / 7 * 7 = 2191$$

Transaction Date: Monday, 11th February 2008 (YYMMDD = 080211)

Transaction Date in days = 2964

$$\text{Transaction Date Reference Day} = (2964 - 2) / 7 * 7 = 2961$$

Annex F Security Counters

This annex specifies an option of how to implement the use of security counters within the CPA. This is an implementer-option for CPA.

Security counters and limits are specified for the keys that secure the following functions:

- Application Cryptogram generation and ARPC verification
- Secure Messaging for Integrity (SMI)
- PIN Decipherment

For forensic purposes it will be possible to determine if a security counter has reached its limit by interrogating the Security Limits Status data element described in Annex L: Data Dictionary.

F1 Symmetric Keys

The CPA application is required to maintain Session Key Counters and associated limits as specified in this section.

To support EMV common session key derivation, the ICC uses two 2-byte counters, AC Session Key Counter and SMI Session Key Counter, each with an associated limit:

- The AC Session Key Counter is a two-byte counter initialized to zero that counts AC session key derivations since successful validation of an ARPC.
- The SMI Session Key Counter is a two-byte counter initialized to zero that counts SMI session key derivations not followed by successful validation of a Secure Messaging MAC.

Initiation of AC session key derivation is controlled as follows:

1. If the AC Session Key Counter is less than the AC Session Key Counter Limit, then increment the AC Session Key Counter.
2. If the value of the AC Session Key Counter (after step 1) is greater than the value of the AC Session Key Counter Limit or has reached the value 'FF FF', then the AC master key is not used, AC session key derivation is discontinued, and the application responds to the GENERATE AC command with SW1 SW2 = '6985'.

Otherwise (the counter value is not 'FF FF' and has not reached the counter limit), then the application continues with the AC session key derivation.

The AC Session Key Counter is reset to zero only when an ARPC is successfully validated.

SMI session key derivation is only performed if the card receives a script command. Initiation of the SMI session key derivation is controlled as follows:

1. If the SMI Session Key Counter is less than the SMI Session Key Counter Limit, then increment the SMI Session Key Counter.
2. If the value of the SMI Session Key Counter (after step 1) is greater than the value of the SMI Session Key Counter Limit, or has reached the value 'FF FF'; then the SMI master key is not used, SMI session key derivation is discontinued, and the application responds to the script command with SW1 SW2 = '6985'.

Otherwise (the counter value is not 'FF FF' and has not reached the counter limit), then the application continues with the SMI session key derivation.

The SMI Session Key Counter is decremented if the first MAC in the script is successfully validated.

Usage notes:

- In order to prevent the AC Session Key Counter reaching its limit, the card must successfully validate an ARPC. If a transaction is sent online, then the issuer will have the ability to reset this counter by sending the card a valid ARPC. However, if the issuer decides that the card is being misused (for example, there is an implausible increase in the ATC value that could be the result of an attacker causing the card to derive session keys repeatedly) then the issuer can choose not to send an ARPC and thereby not reset the counter.
- For Secure Messaging, the counter controls the total number of times that a card will attempt to process scripts for which the MAC fails. If the limit is reached, then the card will no longer attempt to process scripts, but the earlier stages of the transaction are unaffected.
- The initial values of the Session Key Counter Limits are dictated by security considerations. In setting the value, the issuer should consider whether Issuer Authentication Data (including the ARPC) is normally sent to the card and whether the card is likely to be used in markets where acquirers do not always forward chip data. Issuers may also give consideration to the possibility of updating the limit by secure messaging during the life of the card.

F2 PIN Encipherment Key

The CPA application is required to maintain a PIN Decipherments Error Counter and associated limit as specified in this section.

The PIN Decipherments Error Counter is a two byte counter initialised to zero.

The PIN Decipherments Error Counter is used in the following way:

- If the PIN Decipherments Error Counter equals the PIN Decipherments Error Counter Limit or it has reached 'FF FF', then the application discontinues processing the VERIFY command and responds with SW1 SW2 = '6984' (Command not allowed; referenced data invalidated).
- The PIN Decipherments Error Counter is incremented by one when processing the VERIFY command for the purposes of verifying an enciphered PIN and prior to accessing the PIN decipherment private key.
- If the PIN is successfully deciphered in accordance with section 7 of *EMV Book 2*, then the PIN Decipherments Error Counter is decremented by one.

Annex G Management of Profile Data

This annex describes the management of profile resources (such as Counter Profile Controls or Issuer Options Profile Controls) for PUT DATA and GET DATA commands using a single template tag for each type of resource.

G1 Profile Resources

Data elements that can be retrieved using the GET DATA command and updated using the PUT DATA command are part of two data object categories: Primitive BER-TLV data objects and constructed BER-TLV data objects, as defined in EMV Book 3, Annex B.

- A primitive BER-TLV data object has a data element in its value field.
- A constructed BER-TLV data object has a value field consisting of one or more primitive data objects. The value field of a constructed data object is called a template and defines a logical grouping of application resources of the same type.

Such application resources are data elements to be used by the card during the processing of a transaction when a specific profile is selected. They are referred to in this specification as *profile resources*.

Examples of profile resources are:

- AIP/AFL Entries used to configure the GPO response
- CIAC triplets (CIAC-Denial, CIAC-Online, and CIAC-Default)
- Accumulator controls

Templates containing profile resources of the same type are referred to hereafter as *profile resource templates*.

The constructed data object encapsulating a profile resource template in its value field is referred to hereafter as a *template data object*.

G2 Structure of Profile Resource Templates

Profile resources are identified and managed in the context of the profile resource template in which they appear according to the following rules:

- Each profile resource appears in the value field of a template data object (the profile resource template) and is identified by the tag of the template (referred to hereafter as `Template_TAG`). The length of the profile resource template (referred to hereafter as `Template_Length`) corresponds to the length of the constructed data object value field.
- Within a profile resource template, each profile resource corresponds to the value field of a BER-TLV primitive data object. This means that the profile resource has a tag (denoted hereafter as `Resource_TAG`) that uniquely identifies the resource within the template, and a length `L`.

Figure G-1 illustrates how the rules described above apply to the description of the format of a profile resource template encapsulating `n` profile resources of the same type.

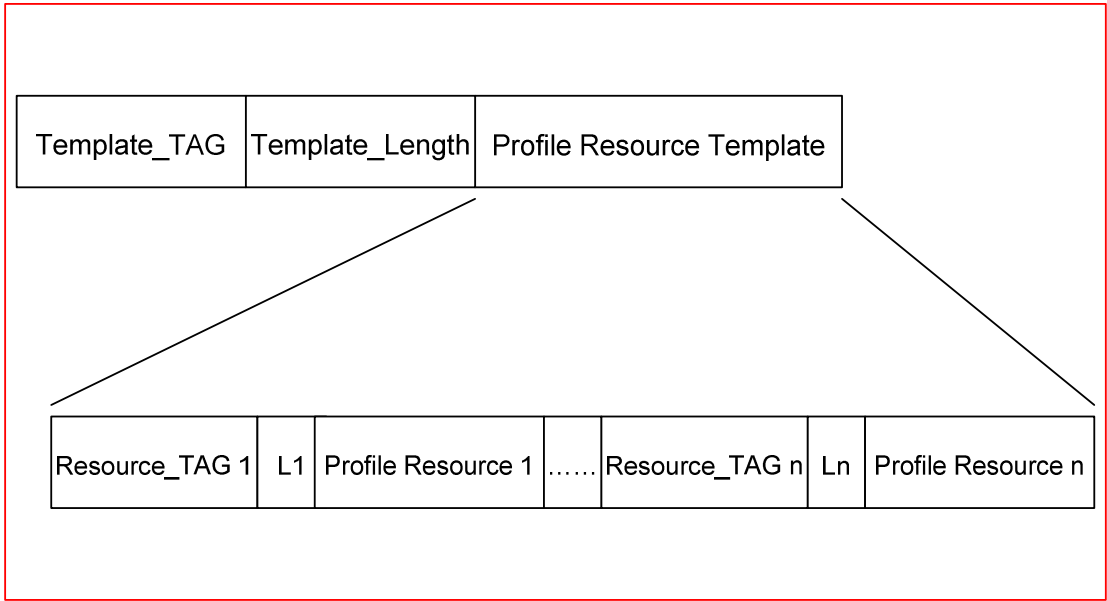


Figure G-1: Profile Resource Template

G3 Profile Resource Templates for PUT DATA and GET DATA

Table G-1 lists the profile resource templates introduced in this specification that can be accessed using the command GET DATA and updated using the command PUT DATA.

Profile Resource	Template Tag	PUT DATA	GET DATA
Accumulator Controls Tag 'DF0x' = Accumulator x Control	'BF32'	Y	Y
Accumulator Profile Controls Tag 'DF0x' = Accumulator Profile Control x	'BF31'	Y	Y
Accumulators Data Tag 'DF0x' = Accumulator x value Tag 'DF1x' = Accumulator x Limits	'BF30'	Y	Issuer option ³
Additional Check Tables Tag 'DF01' = Additional Check Table 1 Tag 'DF02' = Additional Check Table 2	'BF33'	Y	Y
AIP/AFL Entries Tag 'DF0x' = AIP/AFL Entry x	'BF41'	Y	Y
CIACs Entries Tag 'DF0x' = CIACs Entry x	'BF34'	Y	Y
Counter Controls Tag 'DF0x' = Counter x Control	'BF37'	Y	Y
Counter Profile Controls Tag 'DF0x' = Counter Profile Control x	'BF36'	Y	Y
Counters Data Tag 'DF0x' = Counter x value Tag 'DF1x' = Counter x Limits	'BF35'	Y	Issuer option ⁴

Table G-1: Profile Resource Templates for PUT DATA and GET DATA

³ See section 12.5.2 for further explanation of the issuer-option.

⁴ See section 12.5.2 for further explanation of the issuer-option.

Profile Resource	Template Tag	PUT DATA	GET DATA
Currency Conversion Tables Tag 'DF0x' = Currency Conversion Table x	'BF38'	Y	Y
Cyclic Accumulator Controls Tag 'DF0x' = Cyclic Accumulator x Control	'BF3A'	Y	Y
Cyclic Accumulator Profile Controls Tag 'DF0x' = Cyclic Accumulator Profile Control x	'BF39'	Y	Y
Cyclic Accumulator x Data Tag 'DF0x' = Cyclic Accumulator x Tag 'DF1x' = Cyclic Accumulator x Reference Date Tag 'DF2x' = Cyclic Accumulator x Reference Day	'BF42'	Y	Y
Issuer Options Profile Controls Tag 'DF0x' = Issuer Options Profile Control x	'BF3B'	Y	Y
Limits Entries Tag 'DF0x' = Limit Entry x	'BF3C'	Y	Y
Log Data Tables Tag 'DF01' = First GEN AC Log Data Table Tag 'DF02' = Second GEN AC Log Data Table Tag 'DF03' = First GEN AC Unchanging Log Data Table	'BF40'	N	Y
MTA Profile Controls Tag 'DF0x' = MTA Profile Control x	'BF3D'	Y	Y
Profile Controls Tag 'DF0x' = Profile Control x	'BF3F'	Y	Y

Table G-1: Profile Resource Templates for Put Data and Get Data, continued

NOTE: Using the same tag for data elements in different templates is unambiguous for the application because the tag is only interpreted within the context of the template. For example, a PUT DATA command for Template 'BF32', tag 'DF01' would update Accumulator 1 Control. A PUT DATA command for Template 'BF38', tag 'DF01' would update Currency Conversion Table 1.

G4 PUT DATA Command Applied to Templates

G4.1 Updating a Profile Resource Template

A PUT DATA command on a template data object allows updating of all, or part of, the template resource values. The command is tagged (that is, has a P1/P2 value) with the tag of the constructed data object having the template in its value field.

The data field of the PUT DATA command encapsulates BER-TLV coded data that is interpreted by the card as any combination of resources (data elements) that are understood within the context of the template.

Example 1 illustrates the processing of a PUT DATA command when applied to the Counter Controls template (tag 'BF37').

Example 1: Updating a template with the PUT DATA command

Figure G-2 shows the contents of a Counter Controls template as it exists in the card before the PUT DATA command is processed.

'BF37'	'0C'	'DF0101F8DF0201E8DF0301E0'
--------	------	----------------------------

Figure G-2: Counter Controls before PUT DATA Command

Figure G-2 indicates that the resources shown in Table G-2 are encapsulated in the Counter Controls template.

Data Element	Value
Counter 1 Control	'F8'
Counter 2 Control	'E8'
Counter 3 Control	'E0'

Table G-2: Counter Controls Contents before PUT DATA Command

The coding of the PUT DATA command received by the card is provided in Table G-3.

Code	Value
CLA	'0C'
INS	'DA'
P1/P2	'BF37'
Lc	'10'
Data	'8108DF0101E8DF0301F08E04675B0CCA'

Table G-3: Coding of Received PUT DATA Command

The card interprets the data field as follows:

- Counter 1 Control (tag 'DF01') is to be updated to value 'E8'
- Counter 3 Control (tag 'DF03') is to be updated to value 'F0'

Figure G-3 shows the contents of the Counter Controls template after successful processing of the PUT DATA command. It also highlights that only Counter 1 Control and Counter 3 Control have been updated.

'BF37'	'0C'	'DF0101E8DF0201E8DF0301F0'
--------	------	----------------------------

Figure G-3: Counter Controls Coding after PUT DATA command

Figure G-3 indicates that the resources shown in Table G-4 are encapsulated in the Counter Controls template.

Data Element	Value
Counter 1 Control	'E8'
Counter 2 Control	'E8'
Counter 3 Control	'F0'

Table G-4: Counter Controls Contents after PUT DATA Command

G4.2 Adding a Resource to a Profile Resource Template

Prior to personalization, an area of card memory is allocated and a default value (that is, 'FF...FF') is set for each profile resource.

At personalisation for each template, the issuer decides on the profile resources that are to be used for transaction processing with the profiles defined for the application. Once the card is issued, each template is coded as described in Figure G-4.

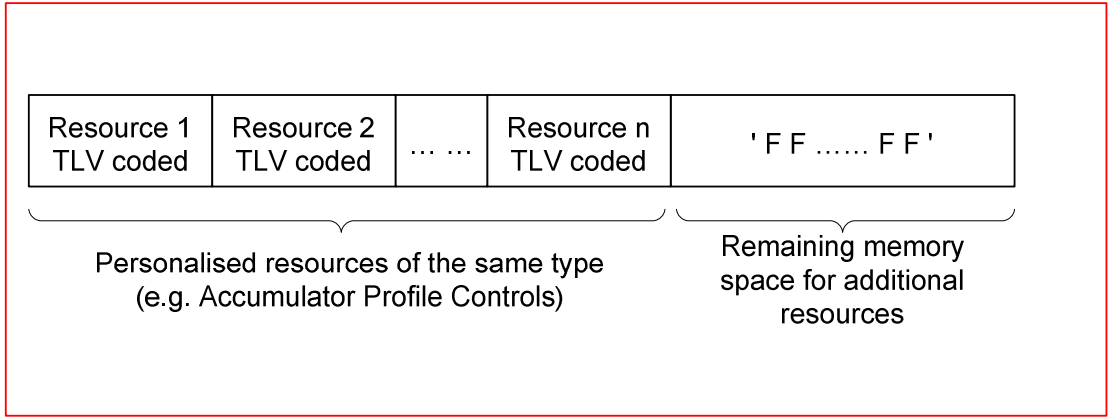


Figure G-4: Profile Resource Template Coding after Personalisation

Figure G-4 shows that a memory space might be reserved so that additional resources could be added to a template after personalising the card. Such additions could be performed using the PUT DATA command, which can add new resources to the template as well as update existing template resources as described in section G4.1.

The following Example 2 illustrates how a PUT DATA command can be used both to update existing template resources and to add new ones. The Accumulator Profile Controls template (tag 'BF31') will be used for Example 2.

Example 2: Updating and adding new resources to a template with the PUT DATA command

Figure G-5 describes the contents of an Accumulator Profile Controls template as it exists in the card before the PUT DATA command is processed:

'BF31'	'2E'	'DF0102EC01DF0202FC02FFFFFFFFFFFFFFFFFFFFFFFF'
--------	------	--

Figure G-5: Accumulator Profile Controls before PUT DATA Command

Figure G-5 indicates that the resources shown in Table G-5 have been created at personalization.

Data Element	Value
Accumulator 1 Control	'EC01'
Accumulator 2 Control	'FC02'

Table G-5: Accumulator Profile Controls Contents before PUT DATA Command

The coding of the PUT DATA command received by the card is provided in Table G-6.

Code	Value
CLA	'0C'
INS	'DA'
P1/P2	'BF31'
Lc	'12'
Data	'810ADF0102EC02DF0302FC028E04675B0CCA'

Table G-6: Coding of Received PUT DATA Command

The card interprets the data field as follows:

- Accumulator Profile Control 1 (tag 'DF01') is to be updated to the value 'EC02'.
- Accumulator Profile Control 3 (tag 'DF03') is to be added with value 'FC02'.

Figure G-6 indicates the contents of the Accumulator Profile Controls template after successful processing of the PUT DATA command.

'B F 3 1'	'2 E'	'DF0102EC02DF0202FC02DF0302FC02FFFFFFFFF...'
-----------	-------	--

Figure G-6: Accumulator Profile Controls Template Contents after PUT DATA Command

Figure G-6 indicates that the resources shown in Table G-7 are encapsulated in the Accumulator Controls template.

Data Element	Value
Accumulator 1 Control	'EC02'
Accumulator 2 Control	'FC02'
Accumulator 3 Control	'FC02'

Table G-7: Accumulator Controls Contents after the PUT DATA Command

G4.3 (Im)possibility of Deleting a Profile Resource Template

When a card is issued, a memory area has been allocated for each template. This memory area cannot be erased using the PUT DATA command or by any other command supported by the CPA application. Thus, the PUT DATA command cannot be used to partially or entirely delete a template for which space has previously been allocated.

G5 GET DATA Command Applied to Templates

The GET DATA command allows the retrieval of all values contained in a profile resource template. This means it is not possible to retrieve only selected portions of a profile resource template.

NOTE: If filler bytes are included in the memory area that contains a template, the filler bytes may also be returned in response to a GET DATA command performed on the template.

Example 3 illustrates how the card responds to a GET DATA command. Example 3 uses a Counter Controls template.

Example 3: GET DATA command

The contents of the Counter Controls template before the GET DATA command is received are as illustrated in Table G-4: Counter Controls Contents after PUT DATA Command above.

The coding of the GET DATA command received by the card is provided in Table G-8.

Code	Value
CLA	'80'
INS	'CA'
P1/P2	'BF37'
Le	'00'

Table G-8: Coding of the Received PUT DATA Command

The card response to the GET DATA command is provided in Figure G-7. It shows that all the template resources (that is, Counter 1 Control, Counter 2 Control, and Counter 3 Control) are returned to the terminal after successful processing of the GET DATA command.

'BF37'	'0C'	'DF0101E8DF0201E8DF0301F0'	SW1-SW2 '9000'
--------	------	----------------------------	-------------------

Figure G-7: Card Response to the GET DATA Command

Annex H Issuer Profile Options Specification and Processing

The use of profile-specific options and data allows the Issuer to tailor the behaviour of the application to correspond to transaction-specific requirements. This annex explains how the application processes Issuer-defined profile options and configures card behaviour based on these options.

H1 Profile Selection

The Profile Selection process enables the application to evaluate transaction characteristics (such as Terminal Type, Transaction Currency Code, or Amount), and determine which Issuer-defined profile to use in that specific transaction context. If the Profile selection using Card Data implementer-option is supported, the application can also use the Profile Selection Diversifier byte to distinguish between multiple issuer defined profiles.

Processing begins with receipt of GET PROCESSING OPTIONS command data (as requested in the card-supplied PDOL). Assuming that the Application Control is configured to activate Profile Selection File processing, the application uses GPO command data to iterate through the Profile Selection Entries, identify the specific transaction context, and determine the Profile ID which will be used for this transaction. (See Annex A for details of Profile Selection process.)

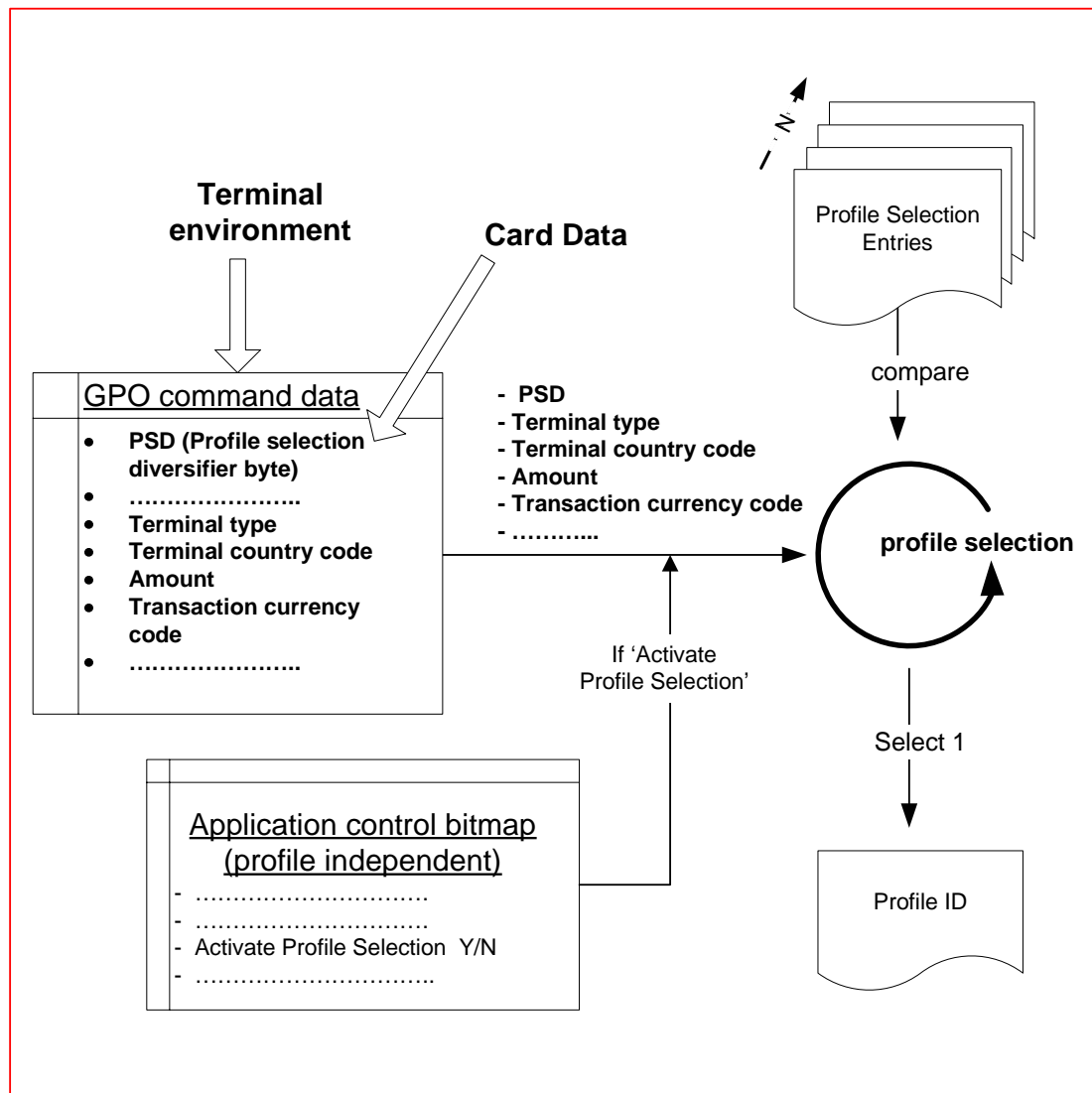


Figure H-1: Profile Selection Process

H2 Configuring Profile Resources

Using the Profile ID selected for the transaction, the application selects one of the Profile Control resources. Each Profile ID corresponds to a single Profile Control object. For Profile x (that is, the profile where the Profile ID has the value x), the application uses Profile Control x. This Profile Control x contains a number of references to profile-specific resources.

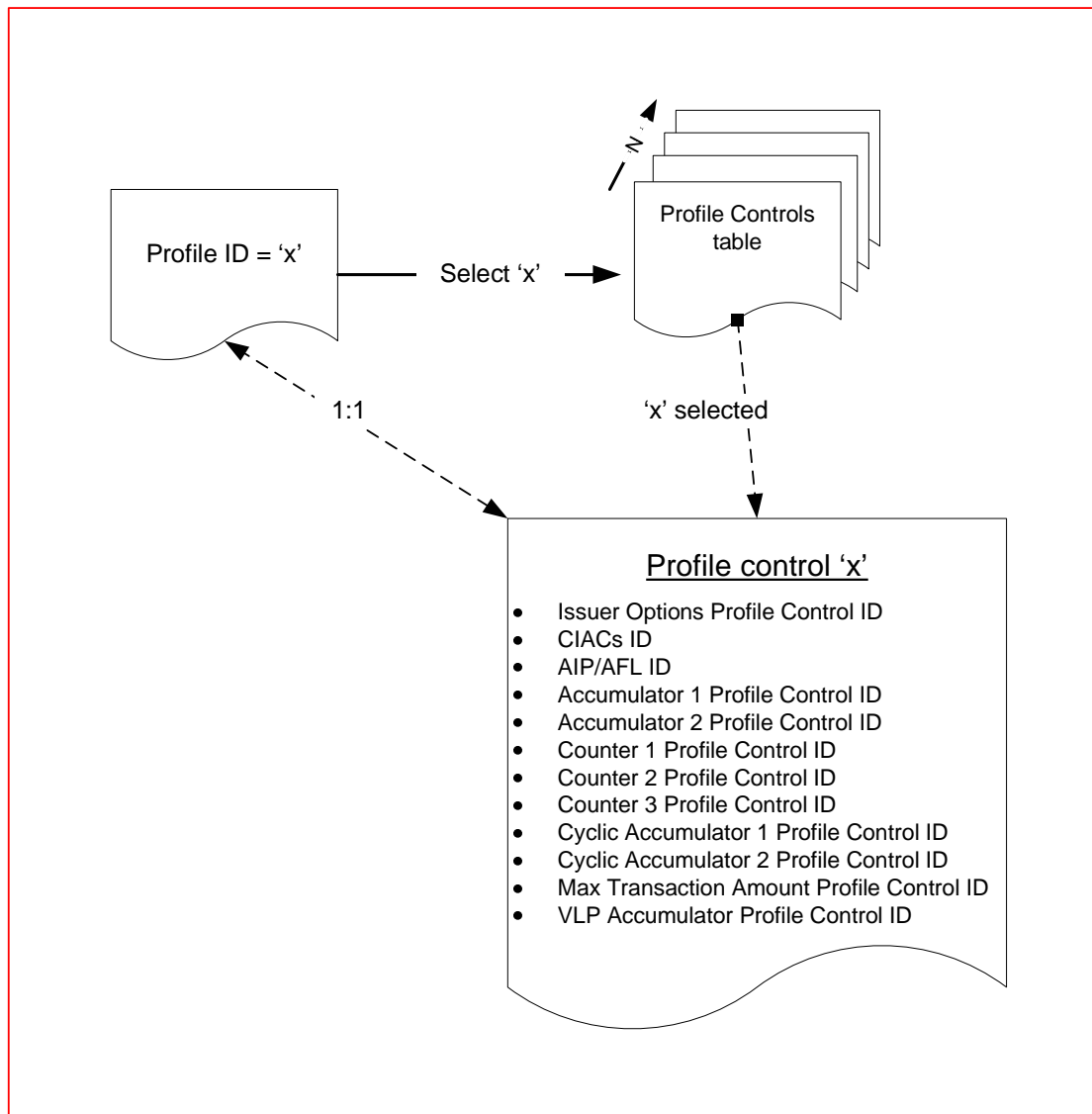


Figure H-2: Profile Control

Resources

The application contains sets of resources that can be used to configure application behaviour. These resources are organised in templates (as explained in Annex G).

The application contains several types of application counters:

- Accumulators (Accumulator 1 and Accumulator 2)
- Counters (Counter 1, Counter 2, Counter 3)
- Cyclic Accumulators (Cyclic Accumulator 1, Cyclic Accumulator 2)
- VLP Available Funds (if VLP is implemented)

Each of these application counters is not necessarily used for all transactions, and when used, they may not be controlled using the same parameters. However, to ensure that the meaning of each application counter is consistent for all profiles in which the application counter is used, the following profile-independent controls apply to application counters regardless of the profile used for the transaction:

- Accumulator Controls (Accumulator 1 Control and Accumulator 2 Control)
- Counter Controls (Counter 1 Control, Counter 2 Control, Counter 3 Control)
- Cyclic Accumulator Controls (Cyclic Accumulator 1 Control and Cyclic Accumulator 2 Control)

Which application counters to use, and how to use the counters may be profile dependent. The list of possible ways to use application counters when a profile is selected are defined in Profile Controls:

- Accumulator Profile Controls for accumulators
- Counter Profile Controls for counters
- Cyclic Accumulator Profile Controls for cyclic accumulators
- VLP Profile Controls for the VLP Available Funds

NOTE: A VLP Profile Control is a special type of Accumulator Profile Control. Some of the bits that are used for an Accumulator do not apply for VLP Available Funds.

Furthermore several optional transaction processing functions can also be enabled or disabled per profile:

- Log Transactions
- Additional Check Table 1 Check
- Additional Check Table 2 Check
- Maximum Number of Days Offline Check
- Reset Maximum Number of Days offline with an online response
- Override CIAC-Default for Transactions at Terminal Type 26
- Encipher Counters portion of IAD

The Issuer Options Profile Control contains the options to apply during a transaction. In order to use a different set of options in different profiles, several Issuer Options Profile Controls are available in the Issuer Options Profile Control template.

Other application parameters may be profile-dependent. The possible values these parameters may have when a profile is selected are defined in the following tables:

- AIP/AFL Entries (used to configure the GPO response)
- CIAC Entries (each containing a CIAC triplets: CIAC-Denial, CIAC-Online, and CIAC-Default)
- MTA (Maximum Transaction Amount) Profile Controls

Therefore, when defining a profile, the issuer has to decide:

- for each application counter (for example, the 2 accumulators, the 3 counters, the 2 cyclic accumulators, and the VLP accumulator if VLP is implemented) which Profile Control to use
- the optional transaction processing to enable
- the values to use for the other parameters specific to the profile

The issuer has to define a Profile Control x associated with Profile x (that is, the profile that has Profile ID x) specifying the profile-specific entry to use from each application profile control when processing a transaction using Profile x.

If an ID of 'F' is used for any one of these controls, the specified resource is not active in the profile, and will not be taken into account during Card Risk Management.

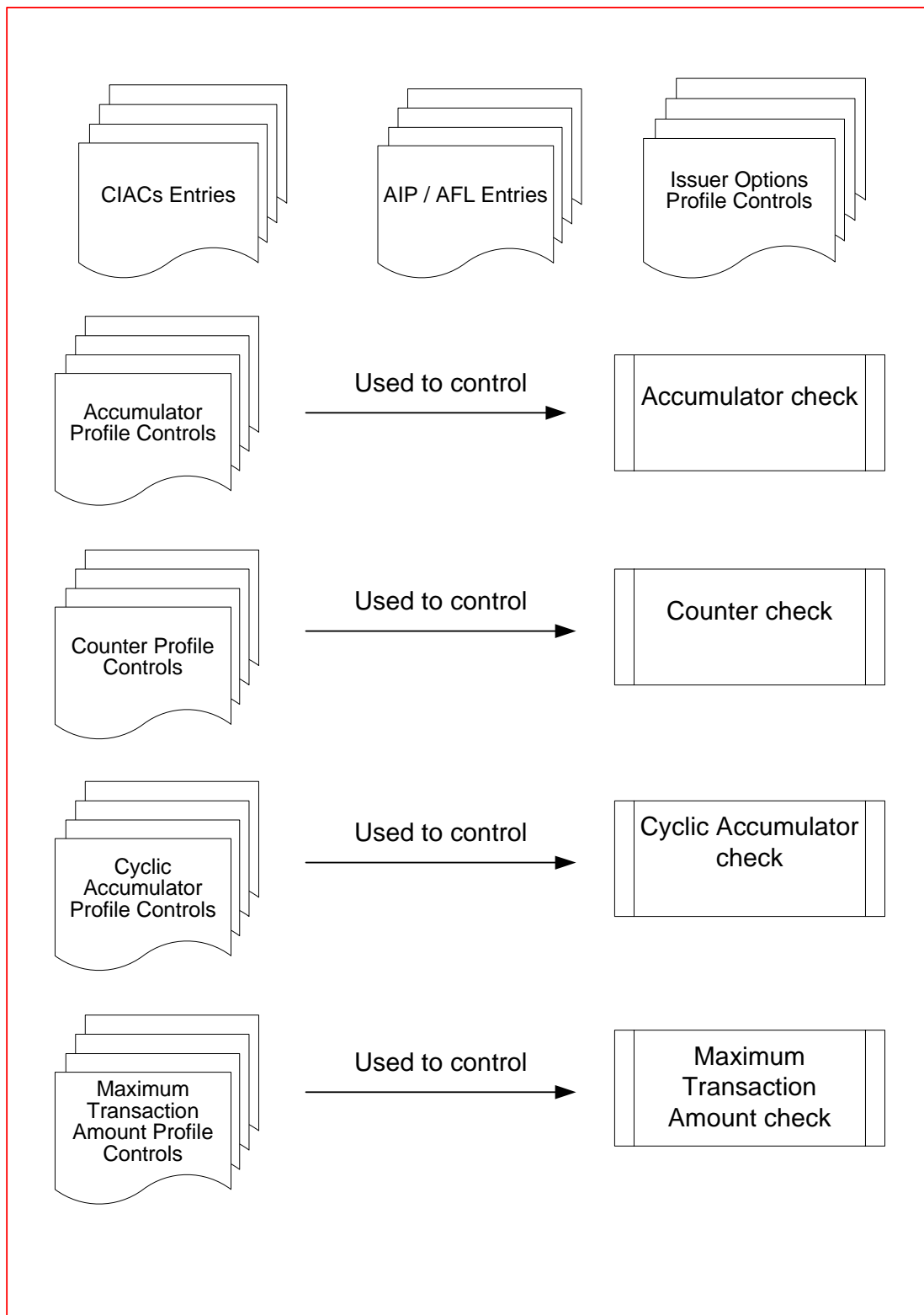


Figure H-3: Profile-specific Controls

The following information is collected in Profile Control x:

- Issuer Options Profile Control ID that identifies the Issuer Options Profile Control (template tag 'BF3B') containing the optional transaction processing to enable for the transaction
- AIP/AFL ID that identifies the AIP/AFL Entry (template tag 'BF41') containing the AIP and AFL values to use in the GET PROCESSING OPTIONS response.
- CIACs ID that identifies the CIACs Entry (template tag 'BF34') containing the CIAC values to use in the GENERATE AC command processing
- Accumulator Profile Control ID for Accumulator 1 that identifies the entry in Accumulator Profile Controls (template tag 'BF31') to apply to Accumulator 1 for the profile— 'F' means Accumulator 1 is not active for the profile
- Accumulator Profile Control ID for Accumulator 2 that identifies the entry in Accumulator Profile Controls (template tag 'BF31') to apply to Accumulator 2 for the profile— 'F' means Accumulator 2 is not active for the profile
- Counter Profile Control ID for Counter 1 that identifies the entry in Counter Profile Controls (template tag 'BF36') to apply to Counter 1 for the profile— 'F' means Counter 1 is not active for the profile
- Counter Profile Control ID for Counter 2 that identifies the entry in Counter Profile Controls (template tag 'BF36') to apply to Counter 2 for the profile— 'F' means Counter 2 is not active for the profile
- Counter Profile Control ID for Counter 3 that identifies the entry in Counter Profile Controls (template tag 'BF36') to apply to Counter 3 for the profile— 'F' means Counter 3 is not active for the profile
- Cyclic Accumulator Profile Control ID for Cyclic Accumulator 1 that identifies the entry in Cyclic Accumulator Profile Controls (template tag 'BF39') to apply to Cyclic Accumulator 1 for the profile— 'F' means Cyclic Accumulator 1 is not active for the profile
- Cyclic Accumulator Profile Control ID for Cyclic Accumulator 2 that identifies the entry in Cyclic Accumulator Profile Controls (template tag 'BF39') to apply to Cyclic Accumulator 2 for the profile— 'F' means Cyclic Accumulator 2 is not active for the profile
- MTA Profile Control ID that identifies the MTA Profile Control (template tag 'BF3D') containing the parameters to use during MTA checking – 'F' means the MTA check is not active for the profile
- VLP Profile Control ID that identifies the Accumulator Profile Control (template tag 'BF31') to apply to VLP Available Funds for the profile— 'F' means VLP Available Funds is not active for the profile

Several profiles may use the same profile-specific resources.

H3 Profile-specific Issuer Options Profile Control

The Profile Control *x* selected for the transaction contains the Issuer Options Profile Control ID which is the reference to a specific Issuer Options Profile Control to be used for the transaction. If the Issuer Options Profile Control ID has the value *y*, then Issuer Options Profile Control *y* is used for the transaction. This process is illustrated in Figure H-4.

Issuer Options Profile Control *y* includes indicators to enable or disable the following functionality:

- Log Transactions
- Additional Check Table 1 Check
- Additional Check Table 2 Check
- Maximum Number of Days Offline Counter
- Reset Maximum Number of Days offline with an online response
- Override CIAC-Default for Transactions at Terminal Type 26
- Encipher Counters Portion of Issuer Application Data (IAD)

These indicators configure part of the application behaviour for processing a transaction within a specific profile.

The Issuer Options Profile Control also includes a number of Issuer-defined configuration items which can vary from profile to profile. These include:

- Length of First Generate AC command data
- Length of Second Generate AC command data
- Common Core Indicator (CCI), specifying key derivation algorithm
- and Derivation Key Index (DKI)

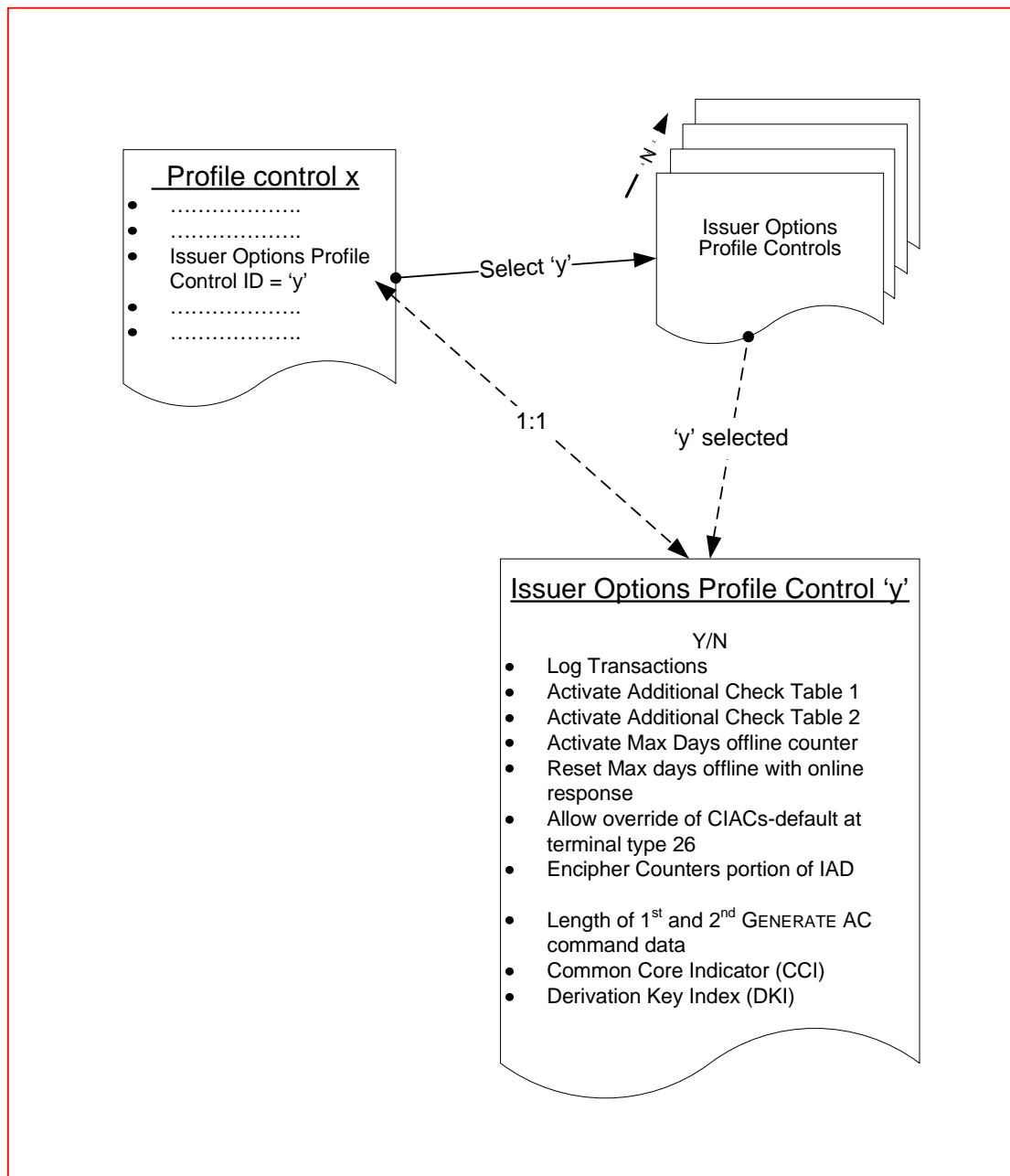


Figure H-4: Issuer Options Profile Control

H4 Profile-specific AIP and AFL

Depending on the transaction context, an Issuer may wish to return different AIP and AFL records in response to the Get Processing Options command. This allows the issuer to specify different offline authentication profiles and/or application data, specifically tailored to the current transaction context.

The Profile Control x selected for the transaction contains the AIP/AFL ID which is a reference to a specific AIP/AFL Entry to be used for the transaction. If the AIP/AFL ID has the value y, then AIP/AFL Entry y is used for the transaction.

AIP/AFL Entry y contain the AIP, AFL Length, and AFL that are returned in the GET PROCESSING OPTIONS response. This process is illustrated in Figure H-5.

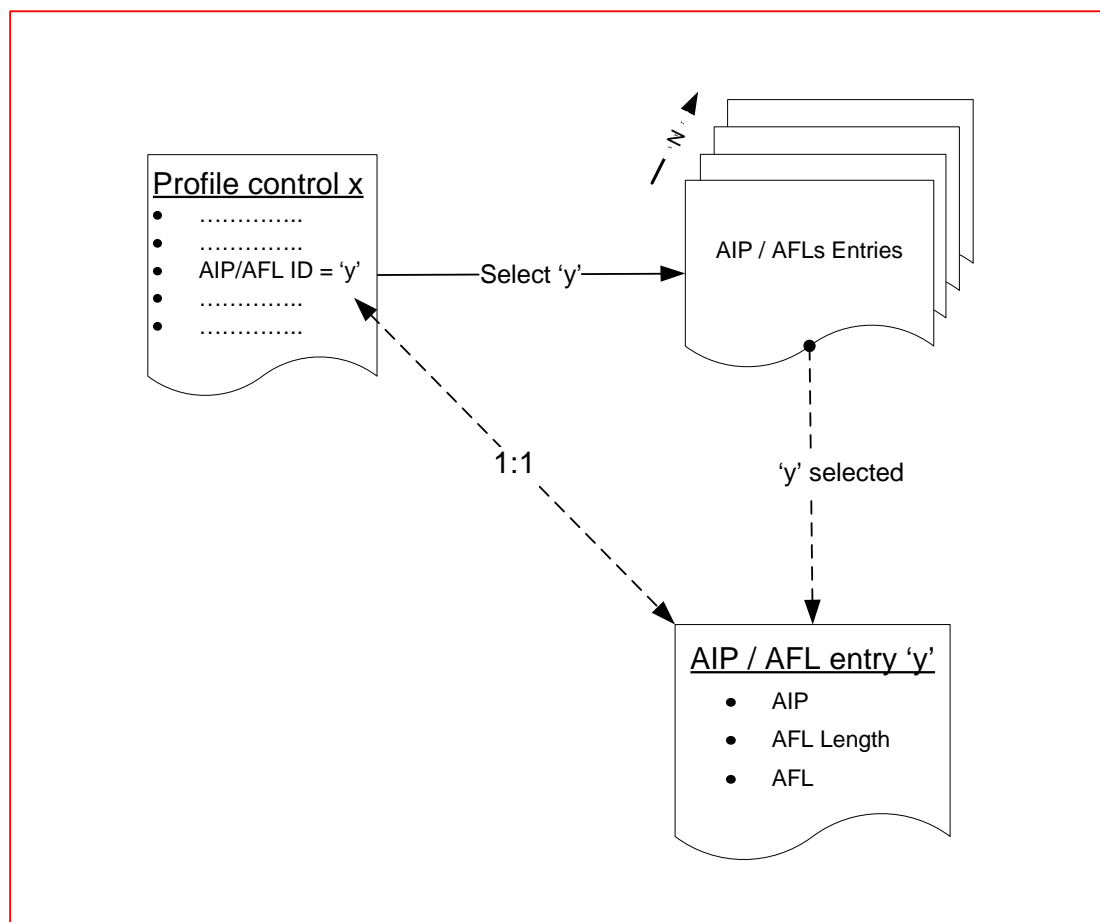


Figure H-5: AIP/AFL Entry

H5 Profile-specific CIACs

In the same manner, an Issuer may choose to tailor the application's behaviour in a given transaction context, by using a customised set of CIACs (Card Issuer Action Codes) during subsequent Card Risk Management process. The Profile Control x selected for the transaction contains the CIACs ID which is a reference to a specific CIACs Entry. If the CIACs ID has the value 'y', then CIACs Entry 'y' is used for the transaction. CIACs Entry 'y' contains a CIACs triplet (CIAC-Denial, CIAC-Online, and CIAC-Default) specifically adapted to the transaction context that will be used in subsequent Card Risk Management. This process is illustrated in Figure H-6.

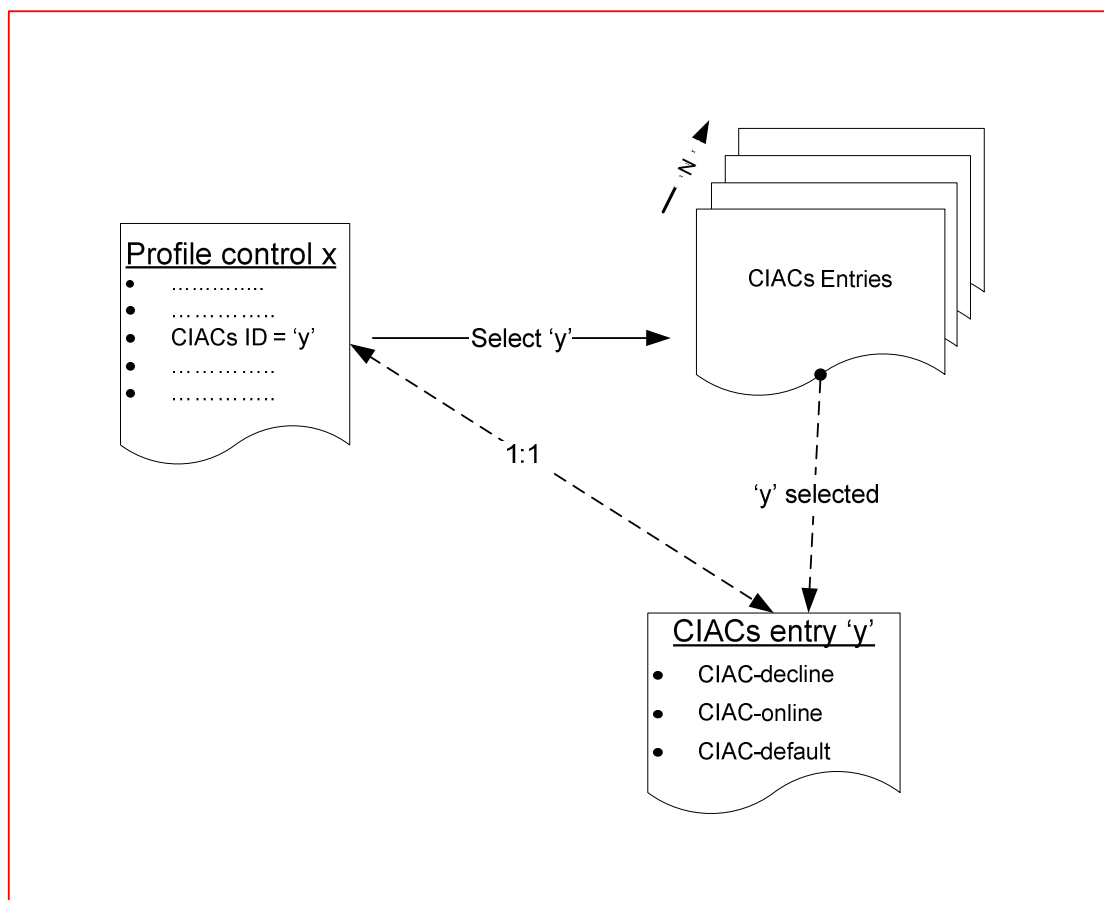


Figure H-6: CIACs Entry

H6 Profile-specific Configuration of MTA Check

Using profile-specific controls, it is also possible to configure certain parameters of the Maximum Transaction Amount (MTA) verification process. In particular, it is possible to configure the maximum transaction amount permissible, the different currency types which will be subject to MTA checking, or even to disable MTA checking, in a specific transaction context.

As previously mentioned in section H1, the Profile Selection mechanism allows the application to identify a specific Profile Control *x* to be used in a given transaction context. This Profile Control *x* contains the MTA Profile Control ID which is the reference to a specific MTA Profile Control (MTA Profile Control *x*) to be used in customising MTA behaviour. If the MTA Profile Control ID has the value 'F', then the MTA Check is not performed for the profile. MTA Profile Control *x* contains the MTA Currency Code, a Limit Entry ID, and a Currency Conversion Table ID. If the Limit ID has the value 'y', then Limit Entry 'y' will provide the maximum amount permissible for this transaction. If the Currency Conversion Table ID contains the value 'z', then Currency Conversion Table 'z' will be used in this transaction. If the Currency Conversion Table ID in the MTA Profile Control has the value 'F', then currency conversion is not performed for the profile. Currency Conversion Table 'z' contains conversion rates and exponents for all currencies which will be subject to MTA checking in this transaction context, as well as the Target Currency Code used with MTA Limit 'y'.

The Process used to identify Issuer-specified, profile-specific MTA options is illustrated in Figure H-7.

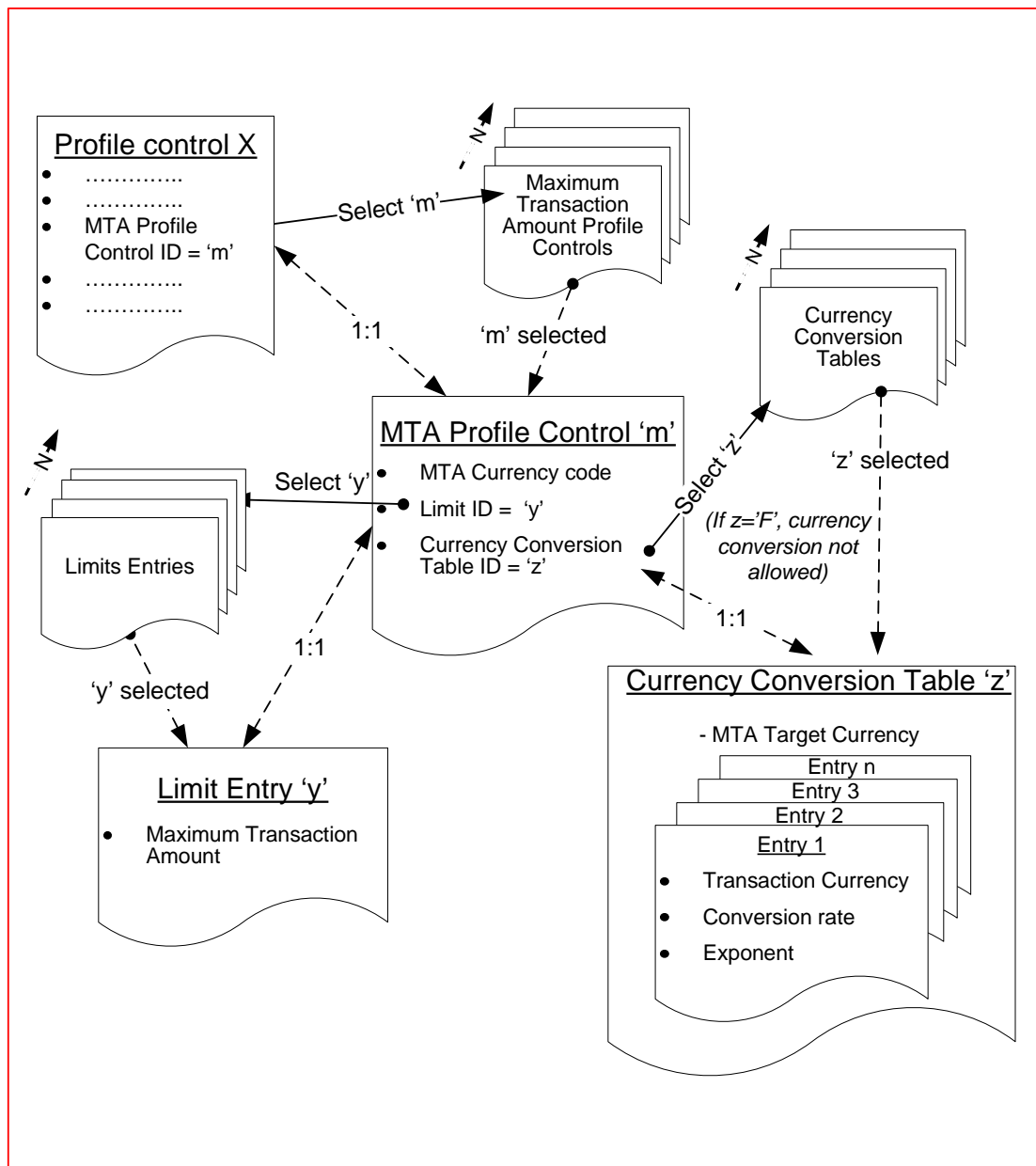


Figure H-7: MTA Profile Control

H7 Profile-specific Configuration of Offline Counters

Profile-specific controls can also be used to configure the behaviour of offline counters, in order to correspond to a specific transaction context. Choice of context-specific counter options is accomplished in the following way:

As previously mentioned in section H1, the Profile Selection mechanism allows the application to identify a specific Profile Control *x* which corresponds to a given transaction context. This Profile Control *x* contains the references for three offline Counters: Counter Profile Control ID for Counter 1, Counter Profile Control ID for Counter 2, and Counter Profile Control ID for Counter 3.

If Counter Profile Control ID for Counter 1 has the value 'x', Counter Profile Control ID for Counter 2 has the value 'y', and Counter Profile Control ID for Counter 3 has the value 'z', then Counter Profile Controls 'x', 'y', and 'z' will be used to configure the respective behaviours of the three offline counters. If a Counter Profile Control ID for any Counter *n* has the value 'F', then Counter *n* is not active for the profile (that is, the Counter *n* Lower Limit Exceeded Check and the Counter *n* Upper Limit Exceeded Check are not performed, Counter *n* is not incremented or reset, and the Counter *n* value is not sent online in the Issuer Application Data).

Each of the Counter Profile Controls contains profile-specific configuration options ('Allow Counting', 'Reset counter with online response', 'Send counter in IAD'), as well as an indication of which limit set to use ('Limit Set 0' or 'Limit Set 1') when performing card risk management. The two potential Limit Sets, as well as the current value of Counter *x* are specified in the Counter *x* Data object associated with Counter *x*.

Additional counter configuration options are specified in the 'Counter *x* Control' data object. These last configuration options ('Include ARQC Transaction in CRM', 'Include Offline declines', 'Include Offline approvals', 'Include only if not-Accumulated', and 'Include only if International') are not profile-specific, but would apply to all transaction types.

The process used to identify which issuer-specified and profile-specific behaviours to apply during a given transaction context is illustrated in Figure H-8.

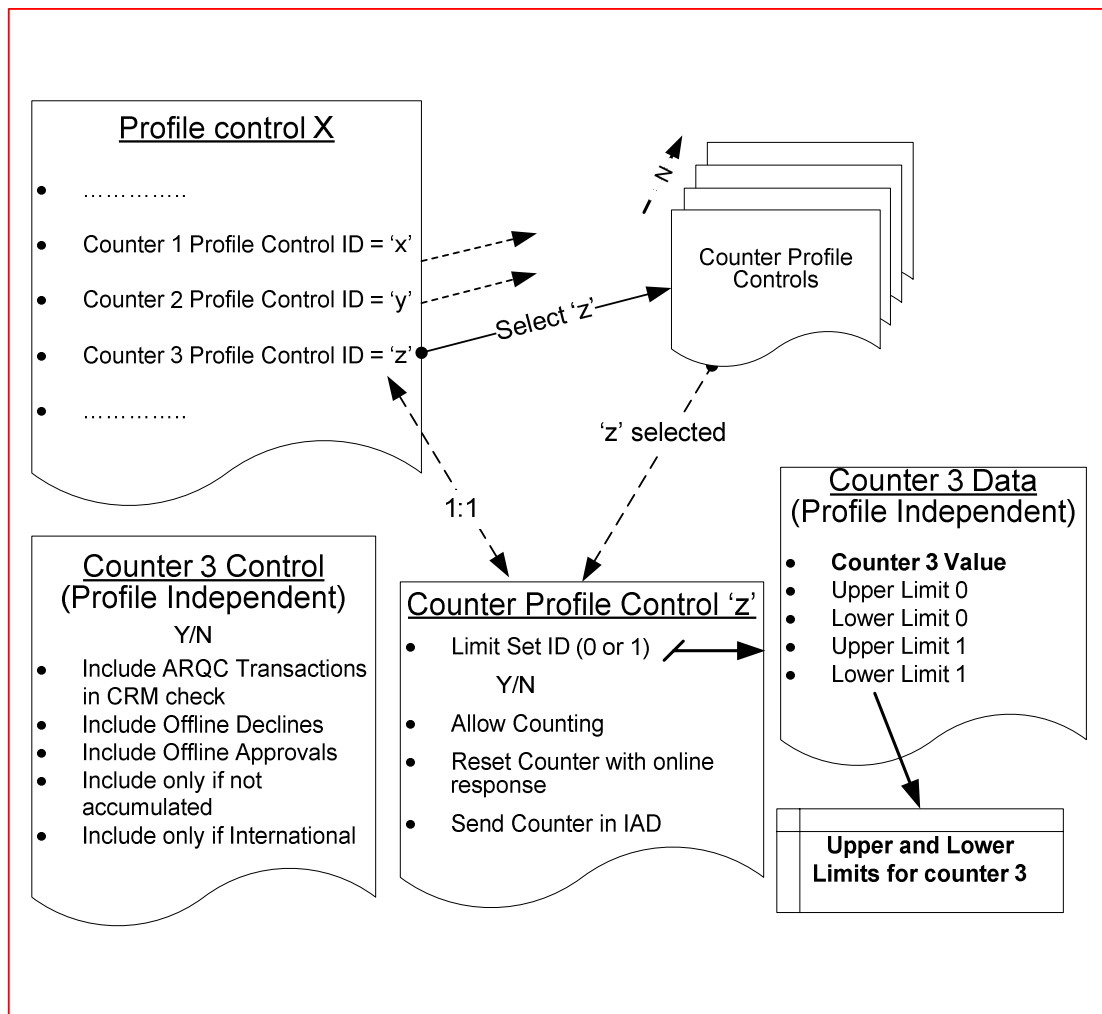


Figure H-8: Counter Profile Control for Counter n - Example (n=3)

H8 Profile-specific Configuration of Accumulators

In a like manner, profile-specific controls can be used to customise certain aspects of accumulator behaviour, depending on specific transaction context. This is accomplished in the following way:

As previously mentioned in section H1, the Profile Selection mechanism allows the application to identify a specific Profile Control *x* which corresponds to a given transaction context. This Profile Control *x* contains references for two accumulators: Accumulator Profile Control ID for Accumulator 1, and Accumulator Profile Control ID for Accumulator 2.

If Accumulator Profile Control ID for Accumulator 1 has the value '*x*' and Accumulator Profile Control ID for Accumulator 2 has the value '*y*', then Accumulator Profile Controls '*x*' and '*y*' will be used to configure the respective behaviours of the two Accumulators. If an Accumulator Profile Control ID for any Accumulator *n* has the value '*F*', then Accumulator *n* is not active for the profile (that is, the Accumulator *n* Lower Limit Exceeded Check and the Accumulator *n* Upper Limit Exceeded Check are not performed, Accumulator *n* is not incremented or reset, and the value and balance of Accumulator *n* are not sent online in the Issuer Application Data).

Each Accumulator Profile Control entry contains the following configuration options: 'Allow Accumulation', 'Reset Accumulator with online response', 'Send Accumulator Balance in authorisation request', as well as a 'Limit set ID' and a 'Currency conversion Table ID'. If the Limit Set ID field has the value '*0*', then Limit Set '*0*' will be used. Conversely, if Limit Set ID contains the value '*1*', then Limit Set '*1*' will be used in this transaction. The Currency conversion table ID specifies which of multiple Currency Conversion tables should be used when performing card risk management. If Currency Conversion Table ID contains the value '*x*', then Currency Conversion Table '*x*' will be used. If the Currency Conversion Table ID in the Accumulator Profile Control has the value '*F*', then currency conversion is not performed for the associated Accumulator in the profile. The currency conversion table consists of a Target Currency Code and a number of Currency Conversion Parameters, each parameter providing a Source Currency Code, a Currency Conversion Factor, and a Currency Conversion Exponent to be used in converting a specific (acceptable) currency.

Additional Issuer-specified but profile-independent Accumulator options are specified in the Accumulator *x* Control (Accumulator Currency Code, 'Include ARQC transaction in CRM test', and 'Include Offline Approvals'). The Accumulator *x* Data, which is also profile-independent, is used to contain the actual accumulated value of Accumulator *x*, as well as up to two Limit Sets (Limit Set '*0*' and Limit Set '*1*'), either of which can be specified using the Profile Selection mechanism.

The process used to identify profile-specific Accumulator configuration options is illustrated in Figure H-9.

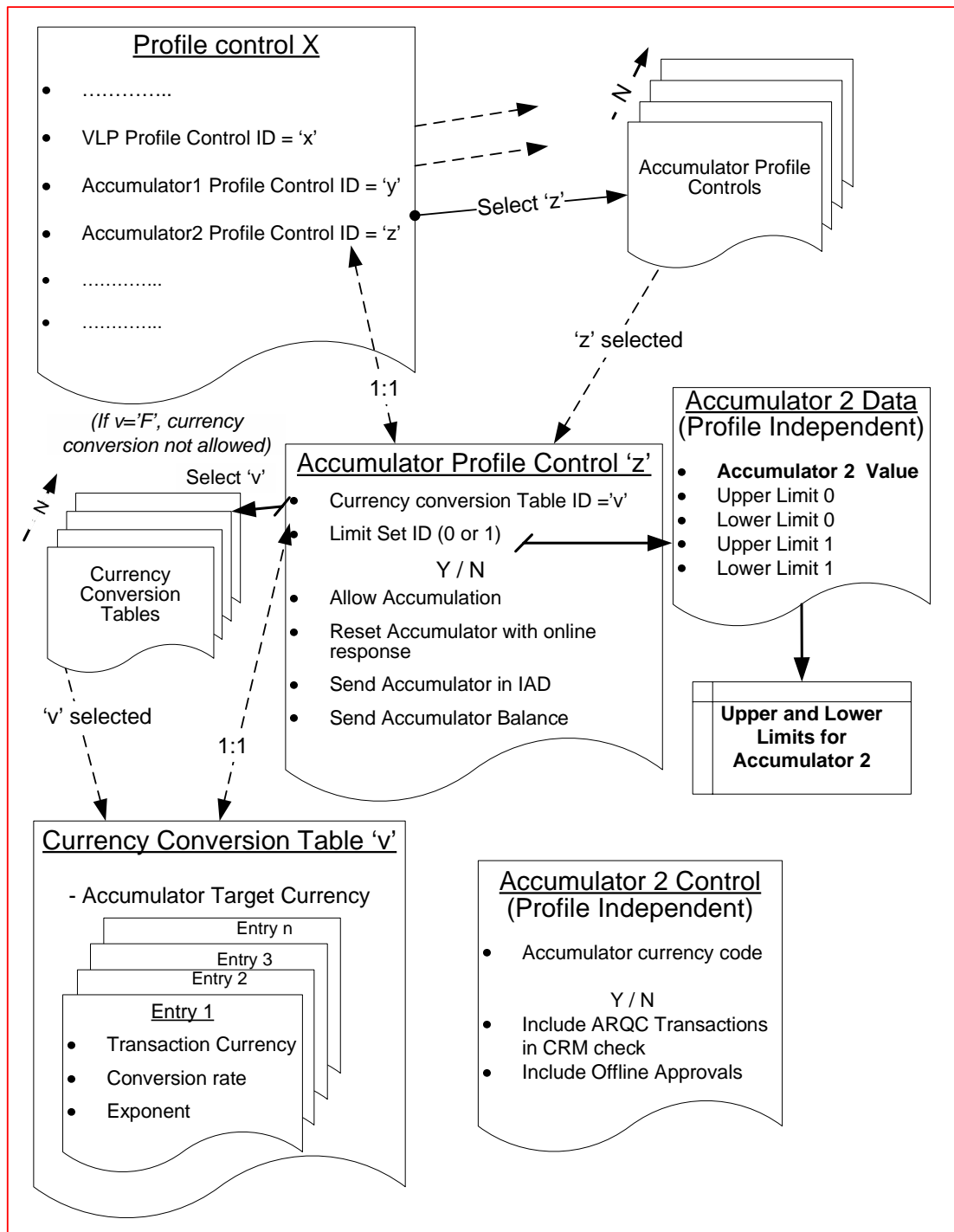


Figure H-9: Accumulator Profile Control for Accumulator n - Example (n=2)

H8.1 Profile-specific Configuration of VLP Available Funds

The VLP Profile Control is a special type of Accumulator Profile Control used in applications that support the VLP implementer-option. If the application supports VLP, then Profile Control x contains a VLP Profile Control ID which is a reference to the Accumulator Profile Control to be used as the VLP Profile Control that configures profile-specific behaviour for the VLP Available Funds. If the VLP Profile Control ID in a Profile Control has the value 'F', then the VLP Available Funds is not active, is not reset, and is not sent online in the Issuer Application Data. If the VLP implementer-option is not supported in the application, then the VLP Profile Control ID field of Profile Control x is RFU.

H9 Profile-specific Configuration of Cyclic Accumulators

Certain parameters and behavioural options for the two cyclic accumulators implemented in the application can also be specified using profile-specific controls. This is accomplished in the following way:

As previously mentioned in section H1, the Profile Selection mechanism allows the application to identify a specific Profile Control *x* which corresponds to a given transaction context. This Profile Control contains references for two Cyclic Accumulators: Cyclic Accumulator Profile Control ID for Cyclic Accumulator 1, and Cyclic Accumulator Profile Control ID for Cyclic Accumulator 2.

If Cyclic Accumulator Profile Control ID for Cyclic Accumulator 1 contains the value 'x', and Cyclic Accumulator Profile Control ID for Cyclic Accumulator 2 contains the value 'y', then Cyclic Accumulator Profile Controls 'x' and 'y' will be used to configure the respective behaviours of the two Cyclic Accumulators. If a Cyclic Accumulator Profile Control ID for any Cyclic Accumulator *n* has the value 'F', then Cyclic Accumulator *n* is not active for the profile (that is, the Cyclic Accumulator *n* Lower Limit Exceeded Check and the Cyclic Accumulator *n* Upper Limit Exceeded Check are not performed, and Cyclic Accumulator *n* is not incremented or reset).

Each Cyclic Accumulator Profile Control contains a reference to a Limit Entry ID, and a Currency Conversion Table ID. If Limit Entry ID contains the value 'w', then Limit Entry 'w' will be used as the limit for the Cyclic Accumulator in question. If the Currency Conversion Table ID contains the value 'z', then Currency Conversion Table 'z' will be used to accumulate transaction amounts in currencies other than that of the Cyclic Accumulator. If the Currency Conversion Table ID in the Cyclic Accumulator Profile Control has the value 'F', then currency conversion is not performed for the associated Cyclic Accumulator in the profile. Each Currency Conversion Table includes a Target Currency Code and a number of Currency Conversion Parameters. Each parameter contains a Source Currency Code, a Currency Conversion Factor, and a Currency Conversion Exponent used in conversion of a specific foreign currency (the source currency) into the Target Currency used by the Cyclic Accumulator. A currency Conversion Table can only be used if the Target Currency Code for a Currency Conversion Table matches the Accumulator Currency Code for the Cyclic Accumulator. The Accumulator Currency Code is specified in the profile-independent Cyclic Accumulator *x* Control object, which also specifies the Cycle Type of the Cyclic Accumulator (Daily, Weekly, or Monthly), the First Day of a weekly cycle, and also includes a flag which indicates whether online approved transactions should also be accumulated. The accumulated value as well as Reference Date and Day used by the cyclic accumulator are contained in an additional profile-independent data object called Cyclic Accumulator *x* Data.

The procedures used to identify profile-specific and issuer-specified options for the two Cyclic Accumulators are illustrated in Figure H-10.

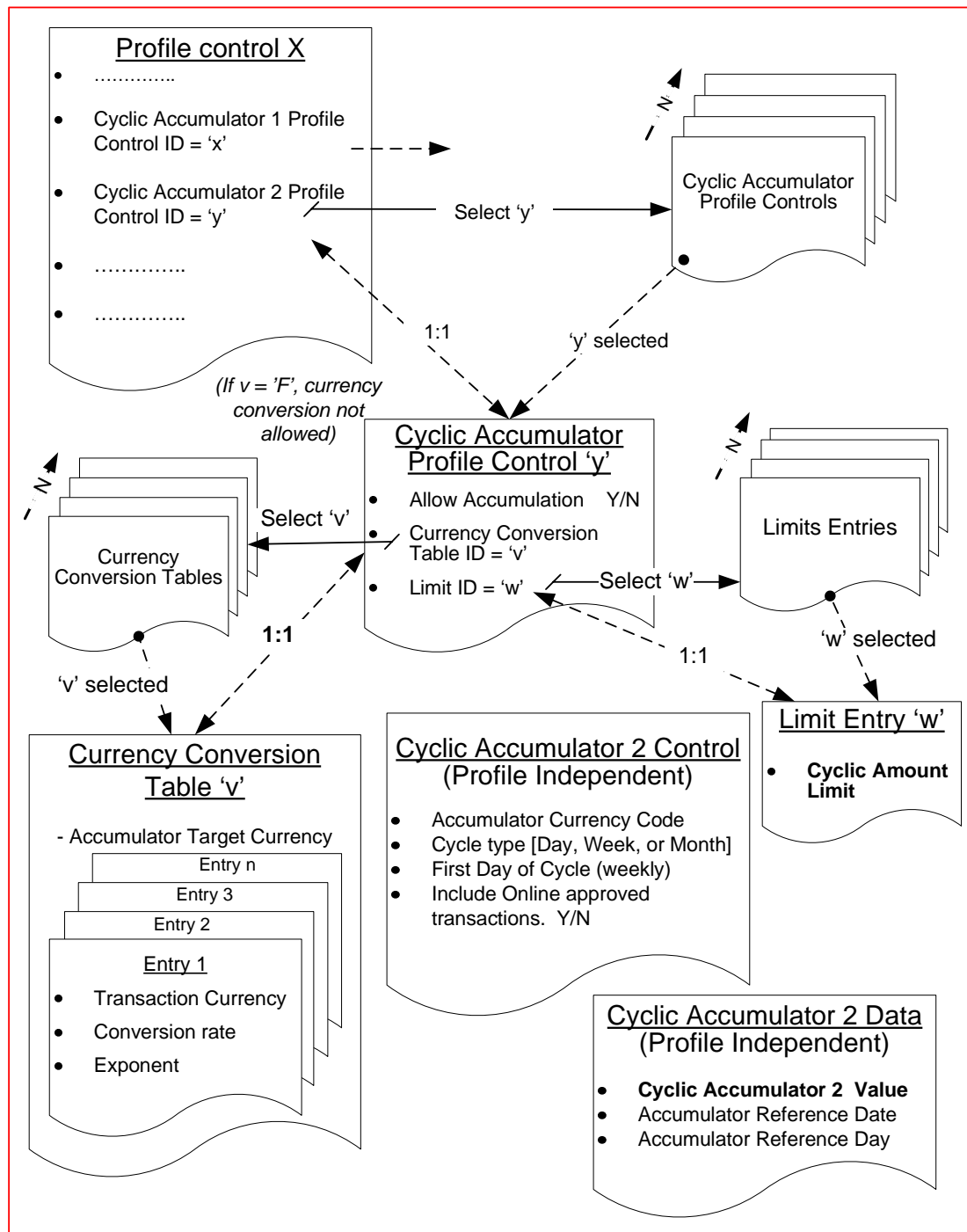


Figure H-10: Cyclic Accumulator Control

H10 Pre-defined Issuer profiles

There are three pre-defined Issuer profiles, for which specific Profile IDs have been reserved:

- Default Profile Profile ID '01'
- Authentication Token Profile Profile ID '7E'
- VLP Profile Profile ID '7D'

H10.1 Default Profile

The Default Profile (Profile ID '01') is used when the Profile Selection mechanism has not been activated (that is, the 'Activate Profile Selection File' bit in Application control has the value 0b). The Profile Control for the Default Profile Control object (Profile Control ID '01') at least contains a valid Issuer Options Profile Control ID, a valid AIP/AFL ID, and a valid CIACs ID. All other resource ID fields may be either initialized with a valid ID, or with an 'F' if the resource is not used in this profile (issuer-optional).

H10.2 Authentication Token Profile

The Authentication Token Profile (Profile ID '7E') is used in cases when the only desired functionality is to authenticate the card-holder or the card. This profile cannot be used to process an EMV transaction. The Profile Control for the Authentication Token Profile Control object (Profile Control ID '7E') at least contains a valid AIP/AFL ID and Issuer Options Profile Control ID. All other fields in the Profile Control object will not be interpreted by the application in this profile.

H10.3 VLP Profile

Implementation of the VLP Profile (Profile ID '7D') is optional, and can only be used when the application provider has chosen to implement the optional VLP functionality. The purpose of VLP is to have a fast offline transaction for very low value transactions. If the application supports VLP, and the transaction meets all the qualifications for a VLP transaction (which includes that the terminal requests VLP processing for the transaction – see section 8.5.3), then the VLP Profile will be used. Otherwise, the application will either choose the default profile (if profile selection is not activated by the issuer) or will process the Profile Selection Entries to select the profile.

The Profile Control object for the VLP Profile (Profile Control ID '7D') at least contains a valid Issuer Options Profile Control ID, a valid AIP/AFL ID, and a valid VLP Profile Control ID. All other resource ID fields may be either initialized with an 'F' if the resource is not used in this profile, or with a valid ID if the intent is to combine VLP and other functionalities (issuer-optional).

For transactions processed using the VLP Profile, the Issuer Options Profile Control is only used to specify the First GENERATE AC Command Data Length, and to determine whether to log the transaction and whether to encipher counters in the Issuer Application Data.

A transaction processed using the VLP Profile skips card risk management processing. If the terminal requests an approval, then the card approves the transaction at the first GENERATE AC. If the terminal requests a decline, then the card declines the transaction at the first GENERATE AC. CIACs are not used for processing transactions in the VLP Profile, so the CIACs ID should be set to 'F'.

The MTA Check is not performed for transactions processed using the VLP Profile (the optional VLP Single Transaction Limit, checked at the time of Profile Selection, is used instead), so the MTA Profile Control ID should be set to 'F'.

For transactions processed using the VLP Profile, Counter x is neither incremented nor reset, and no Counter x card risk management is performed. For VLP transactions, the Counter Profile Control for Counter x is only used to determine whether to include the Counter x value in the Issuer Application Data. If the issuer chooses to not send the value of Counter x in the Issuer Application Data, then the Counter Profile Control ID for Counter x should be 'F'.

For transactions processed using the VLP Profile, Accumulator x is neither incremented nor reset, and no Accumulator x card risk management is performed. For VLP transactions, the Accumulator Profile Control for Accumulator x is only used to determine whether to include the Accumulator x value or balance in the Issuer Application Data. If the issuer chooses to not send the value or balance of Accumulator x in the Issuer Application Data, then the Accumulator Profile Control ID for Accumulator x should be 'F'.

H11 Example – Simple CCD-Compliant Profile

A basic CCD-compliant profile can be configured by personalising the following data elements for the profile:

- one Profile Control – contains identifiers that specify which application resources are to be used when processing a transaction in the profile.
- one AIP/AFL Entry – contains the AIP (identifying the capability of the application to support specific functions in the application), the AFL Length, and the AFL (indicating the files and records which contain card data to be read by the terminal). Different AIP/AFL Entries for different profiles allow the issuer to specify different EMV record data (such as CDOLs and CVM lists) for different profiles.
- one CIACs Entry – contains indicators specifying the situations when the issuer wants the card to decline a transaction offline, send the transaction online, or decline a transaction that cannot go online (see section 21.3.7 for CIAC requirements for a CCD-compliant profile)
- one Issuer Options Profile Control – contains profile-specific options such as whether to log transactions, whether to encipher counters sent to the issuer, and the expected length for the first and second GENERATE AC command data (associated with CDOL1 and CDOL2)
- Accumulator and Counter Controls

In order to illustrate configuration of counters and accumulators, this example shows how to configure support of a simple CCD-compliant profile (Profile '01'), with the following accumulator and counter behaviour activated for the profile:

- one accumulator (accumulating offline approved transactions, including those supported with currency conversion), which uses the following accumulator controls:
 - one accumulator-specific control, Accumulator 1 Control
 - one accumulator profile-specific control, Accumulator Profile Control 1
 - one accumulator-specific data to specify the limits, Accumulator 1 Data
 - one currency conversion table, Currency Conversion Table 1
- one counter (counting offline approved transactions that are not accumulated), which uses the following counter controls:
 - one counter-specific control, Counter 1 Control
 - one counter- and profile-specific control, Counter Profile Control 1
 - one counter-specific data to specify the limits, Counter 1 Data
- a second counter (counting offline approved international transactions), which uses the following counter controls:
 - one counter-specific control, Counter 2 Control
 - one counter- and profile-specific control, Counter Profile Control 2
 - one counter-specific data to specify the limits, Counter 2 Data

In addition, the issuer could configure the card to support the following functionalities which are not activated in the profile for this example:

- Cyclic Accumulators
- Maximum Transaction Amount
- Number of Days Offline
- Additional Check Tables
- VLP Available Funds

H11.1 Coding of Profile-Related Data Elements

Profile Control

Profile '01' is controlled by Profile Control 1 = '11 11 F1 2F FF FF 00 00'

- Profile Control 1 activates the following:
 - Issuer Options Profile Control 1
 - AIP/AFL Entry 1
 - CIACs Entry 1
 - Accumulator 1 using Accumulator Profile Control 1
 - Counter 1 using Counter Profile Control 1
 - Counter 2 (international) using Counter Profile Control 2
- Profile Control 1 also indicates that the following are not activated:
 - Accumulator 2
 - Counter 3
 - Cyclic Accumulators 1 and 2
 - Maximum Transaction Amount
 - VLP Available Funds.

Issuer Options Profile Control

Issuer Options Profile Control 1 = '00 21 1F A5 01 00 00'

This indicates that:

- Number of Days Offline check is not activated
- Additional Check Table checks are not activated
- no transactions are logged
- bytes 9-16 of the Issuer Application Data are not enciphered
- the CIAC-Default is not overridden at type 26 terminals.
- the First Generate AC command data is 33 bytes long
- the Second Generate AC command data is 31 bytes long (includes Amount, Authorised)
- the Common Core Identifier is 'A5' (that is, Profile '01' is CCD-compliant)
- The Derivation Key Index is '01'

Accumulator 1 (Transactions in Euros or converted to Euros)

Accumulator 1 accumulates transactions in Euros, and includes transactions conducted in Euros, and transactions conducted in British Pounds Sterling (which are converted to Euros for accumulation).

Accumulator 1 is controlled by:

- Accumulator 1 Control = '09 78 C0':
 - the domestic currency is the Euro ('0978')
 - ARQC transaction is included in CRM test
 - offline approvals are included
- and Accumulator Profile Control 1 = 'E0 01':
 - allow Accumulation
 - the accumulator is reset with an online response
 - the accumulator value is sent online in the IAD
 - Limit Set 0 is used
 - Currency Conversion Table 1 is used
- and Accumulator 1 Data = '00 00 00 00 00 00 00 00 00 00 20 00 00 00 00 01 00 00':
 - the initial value of the accumulator is zero
 - Lower Limit 0 for the accumulator is 20 Euros
 - Upper Limit 0 for the accumulator is 100 Euros
- and Currency Conversion Table 1 = '0978 0826 0146 82'
 - Target Currency Code is Euros ('0978')
 - British Pounds Sterling are converted at the rate of 1 GBP = 1.46 Euro
 - Source Currency Code '0826'
 - Rate '0146'
 - Exponent '82'

Counter 1 (Non-cumulated Approved Transactions)

Counter 1 counts all approved transactions that are not cumulated in Accumulator 1.

Counter 1 is controlled by:

- Counter 1 Control = 'B0':
 - ARQC is included in CRM test
 - Declines are not counted
 - Approvals are included
 - if the transaction is not accumulated
- and Counter Profile Control 1 = '0E':
 - Limit Set 0 is used
 - counting is allowed in the profile
 - the counter is reset with an online response
 - the counter is sent online in the IAD
- and Counter 1 Data = '00 03 06':
 - the initial value of the counter is zero
 - Lower Limit 0 for the counter is 3
 - Upper Limit 0 for the counter is 6

Counter 2 (International Approved Transactions)

Counter 2 is controlled by:

- Counter 2 Control = 'A8':
 - ARQC is included in CRM test
 - Declines are not counted
 - Approvals are included
 - if the transaction is international
- and Counter Profile Control 2 = '0C':
 - Limit Set 0 is used
 - counting is allowed in the profile
 - the counter is reset with an online response
 - the counter is not sent online in the IAD
- and Counter 2 Data = '00 02 05':
 - the initial value of the counter is zero
 - Lower Limit 0 for the counter is 2
 - Upper Limit 0 for the counter is 5

Annex I Understanding Cyclic Accumulators

This annex describes the behaviour and management of Cyclic Accumulators in the CPA application. A minimum of two Cyclic Accumulators are implemented in the CPA application. Use of Cyclic Accumulators is optional for the issuer. Activation of the Cyclic Accumulators in a given profile (Profile x) is accomplished by setting either or both of the two fields Cyclic Accumulator 1 Profile Control ID and Cyclic Accumulator 2 Profile Control ID in Profile Control x to a value other than 'F' (Invalid ID).

NOTE: Implementation of additional Cyclic Accumulators is an implementation option. If additional Cyclic Accumulators are added, the Profile Control x data element is extended beyond 8 bytes length for the implementation. Activation of each additional Cyclic Accumulator is controlled by one of the nibbles in the extended portion (beyond the initial 8 bytes) of Profile Control x.

I1 Cyclic Accumulators

A Cyclic Accumulator is a data element which accumulates the amount of transactions approved during a given period of time (cycle). The Cyclic Accumulator can be configured for a daily, weekly, or monthly cycle. Each Cyclic Accumulator is composed of three data elements, as illustrated in Figure I-1:

- Cyclic Accumulator x Data
- Cyclic Accumulator x Control
- Cyclic Accumulator Profile Control for Cyclic Accumulator x

I1.1 Cyclic Accumulator x Data

Cyclic Accumulator x Data is specific to the Cyclic Accumulator in question (Cyclic Accumulator x), and does not vary across multiple profiles. Cyclic Accumulator x Data consists of the elements listed in Table I-1

Data	Description
Cyclic Accumulator x Value	Contains the sum of transaction amounts which have been accumulated during the current cycle. Cannot be modified by the counter reset mechanisms associated with the CSU in the authorisation response message.
Cyclic Accumulator Reference Date	Used for a Daily or Monthly Cycle Type. Contains the Transaction Date of the last transaction which reset the Cyclic Accumulator. Automatically updated when the CPA application receives a Transaction Date which is later than the last day of the current cycle (that is, different day or month, depending on Cycle Type).
Cyclic Accumulator Reference Day	Used when the Cyclic Accumulator is configured for a Weekly Cycle Type. Contains the number of days elapsed between a set Reference Date (day 0), and the first day of the current weekly cyclic period. See Annex E for further description of day 0, and the management of Transaction Dates as a count of days. Automatically updated when the CPA application receives a Transaction Date which is not contained in the current weekly cycle.

Table I-1: Cyclic Accumulator x Data

NOTE: See section I3 for a description of how erroneous Transaction Dates are managed.

I1.2 Cyclic Accumulator x Control

Cyclic Accumulator x Control is specific to a given Cyclic Accumulator (Cyclic Accumulator x). It is set at card personalisation (or using the PUT DATA command), and does not vary across multiple profiles. Cyclic Accumulator x Control contains the elements listed in Table I-2.

Data	Description
Accumulator Currency Code	Contains a currency code which defines the currency in which transaction amounts are accumulated. The Cyclic Accumulator will accumulate transaction amounts which are in the specified currency, or (optionally) the approximate converted amount of transactions in currencies which can be converted into the Accumulator Currency using the Currency Conversion Table specified for the Cyclic Accumulator Profile Control in this profile (see Cyclic Accumulator Profile Control y description).
Cycle Type	Contains a configuration option which defines the type of cycle used by the Cyclic Accumulator. Can be Daily, Weekly, or Monthly.
First Day of Cycle	Used only for Weekly Cycle Type. Contains an offset which defines which weekday is the first day of a weekly cycle. This offset is used during the computation of the Cyclic Accumulator Reference Day (see Annex E for a description of new Reference Day computation).
Include Online Approved Transactions	<p>By default, offline approved transactions (conducted in either the Accumulator Currency or a currency that can be converted to the Accumulator Currency) are accumulated in the Cyclic Accumulator. This configuration option indicates whether the issuer also wants online-approved transaction amounts to be added to the Cyclic Accumulator total. The values are as follows:</p> <p>1b Any eligible online approved transaction amount (or the approximate converted amount) will be added to the Cyclic Accumulator value, and will thus be included in subsequent card risk management checks, until the counter is next reset.</p> <p>0b An online approved transaction amount will not be accumulated, and the Cyclic Accumulator value will accumulate only offline approved transactions.</p>

Table I-2: Cyclic Accumulator x Control

I1.3 Cyclic Accumulator Profile Control y for Cyclic Accumulator x

Cyclic Accumulator Profile Control for Cyclic Accumulator x is profile-specific. It is set at card personalisation (or using the PUT DATA command), and can vary if different Issuer profiles are personalised for the application. Multiple instances of Cyclic Accumulator Profile Controls might be personalised if profile-specific Cyclic Accumulator behaviour is desired. Cyclic Accumulator Profile Control y for Cyclic Accumulator x includes the configuration options listed in Table I-3.

Data	Description
Limit Entry ID	Contains the ID associated with one of the transaction amount Limits in Limit Entries. When the accumulated value of transactions approved during the current cycle exceeds the value of the indicated Limit, then the application may be configured to go online or to decline transactions offline.
Currency Conversion Table ID	<p>Contains a reference to one of the Currency Conversion Tables. The Currency Conversion Table identifies the transaction currencies that can be accumulated in the Cyclic Accumulator, and provides the currency conversion parameters necessary to convert the transaction amount into the accumulator currency.</p> <p>If the Currency Conversion Table ID has the value 'F'; then currency conversion is not allowed for the Cyclic Accumulator in the profile selected for the transaction, and only the amounts of transactions conducted in the accumulator currency may be accumulated.</p> <p>If the Currency Conversion Table ID has a value different from 'F'; then currency conversion is allowed for the Cyclic Accumulator in the profile selected for the transaction, and the approximate converted amount of transactions in a currency which can be found in the Currency Conversion Table specified for this Cyclic Accumulator may be accumulated.</p>

Table I-3: Cyclic Accumulator Profile Control y for Cyclic Accumulator x
(continues)

Data	Description
Allow Accumulation	A configuration option which indicates whether the issuer allows accumulator values to be modified by transactions processed using the current profile. If the 'Allow accumulation' bit has the value 1b, then the accumulator will function in the usual manner. However, in certain profiles, the issuer may wish to perform card risk management on the value of transactions previously accumulated in the Cyclic Accumulator, while preventing the current transaction amount being accumulated in the Cyclic Accumulator. This can be accomplished by setting the 'Allow accumulation' bit in the Accumulator Profile Control used for the Cyclic Accumulator in the profile to 0b. In this case, transaction amounts will not be accumulated, and card risk management will only take into account transaction amounts previously accumulated.

Table I-3: Cyclic Accumulator Profile Control y for Cyclic Accumulator x, continued

I1.4 Cyclic Accumulator Illustration

Figure I-1 illustrates the various components which define the contents and behaviour of the Cyclic Accumulator.

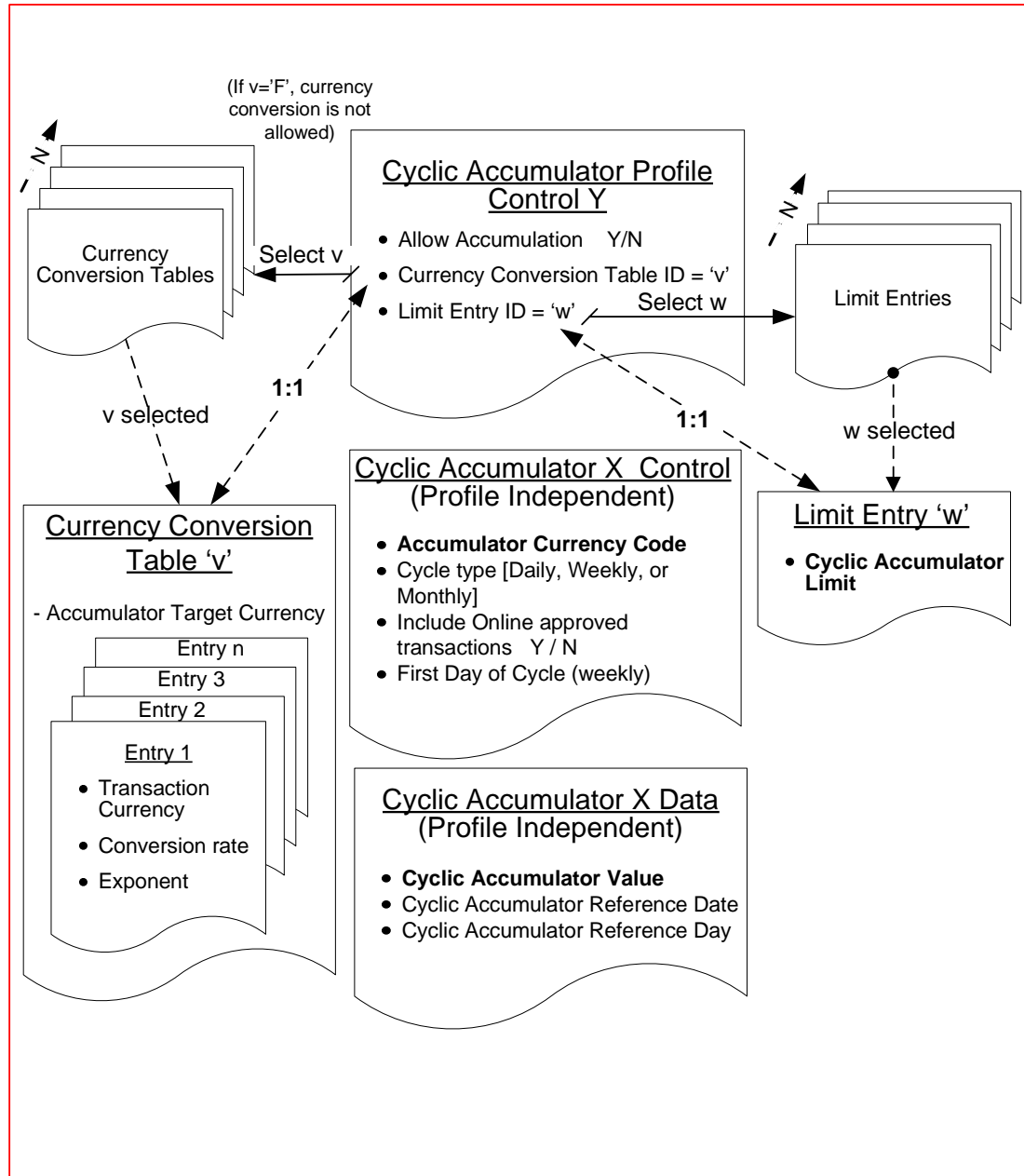


Figure I-1: Cyclic Accumulator

I2 Management of Cyclic Accumulators

The state of Cyclic Accumulators is evaluated during the Card Risk Management process of the First and Second GENERATE AC commands. If the Transaction Date is in a new cycle (that is, the date is outside the range of dates contained in the current cycle specified for the Cyclic Accumulator) and the Allow accumulation configuration option of Cyclic Accumulator Profile Control y for Cyclic Accumulator x has the value 1b, then the Cyclic Accumulator Reference Date is automatically reset during the Card Risk Management process. For a Daily or Monthly cycle, the Cyclic Accumulator Reference Date will be reset to the Transaction Date of the current transaction. For a Weekly cycle, the Cyclic Accumulator Reference Day will be reset to the date beginning the week containing the Transaction Date, as specified in First Day of Cycle in Cyclic Accumulator x Control (see Annex E for a description of date conversion mechanisms). In all cases, if either Cyclic Accumulator Reference Date or Cyclic Accumulator Reference Day is changed to a later date because a new cycle has begun, the cumulative amount stored in the Cyclic Accumulator will be reset to 0. (See section I3 for a description of the way in which erroneous Transaction Dates are managed.)

If the value stored in the Cyclic Accumulator (plus, optionally, the current transaction amount) is equal to or greater than the cumulative transaction amount limit configured for the accumulator (as specified by the Limit Entry ID stored in Cyclic Accumulator Profile Control y for Cyclic Accumulator x), then the CPA application could request an online authorisation (if possible), or decline the transaction. The actual behaviour occurring as a result of exceeding the limit is specified by the contents of the three CIACs (CIAC-Denial, CIAC-Default, and CIAC-Online) associated with the profile selected for the transaction, and customised by the Issuer (for example, the 'Cyclic Accumulator 1 Limit Exceeded' and 'Cyclic Accumulator 2 Limit Exceeded' bits of the CIACs).

NOTE: The Cyclic Accumulator value, Cyclic Accumulator Reference Date, and Cyclic Accumulator Reference Day in Cyclic Accumulators cannot be reset by either the CSU contained in the Issuer authorisation response, or through a PUT DATA command. Reset is performed automatically when a new cycle begins.

Figure I-2 illustrates how Cyclic Accumulators are managed during the Card Risk Management process. (For simplicity, standard error checking steps are not shown.)

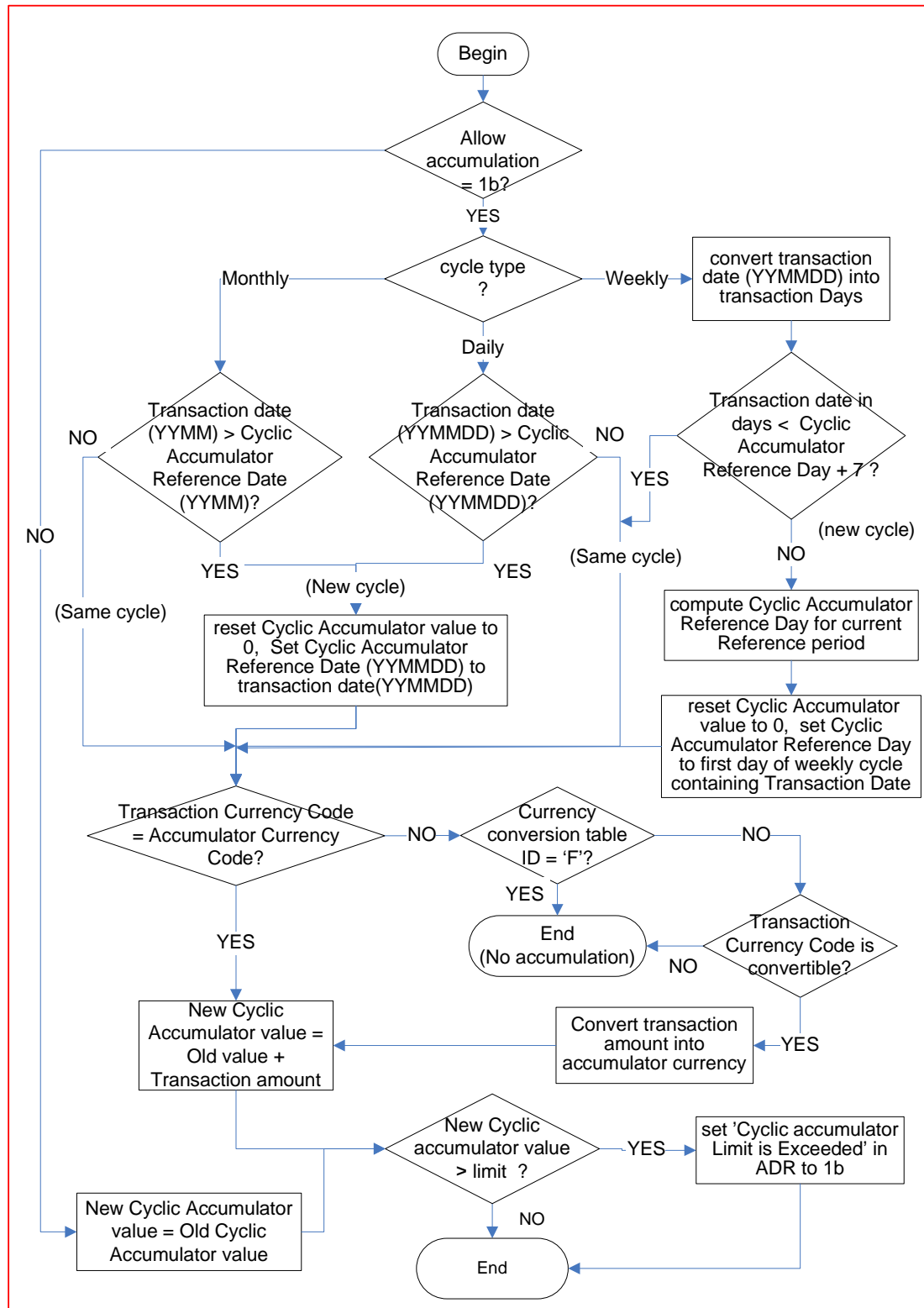


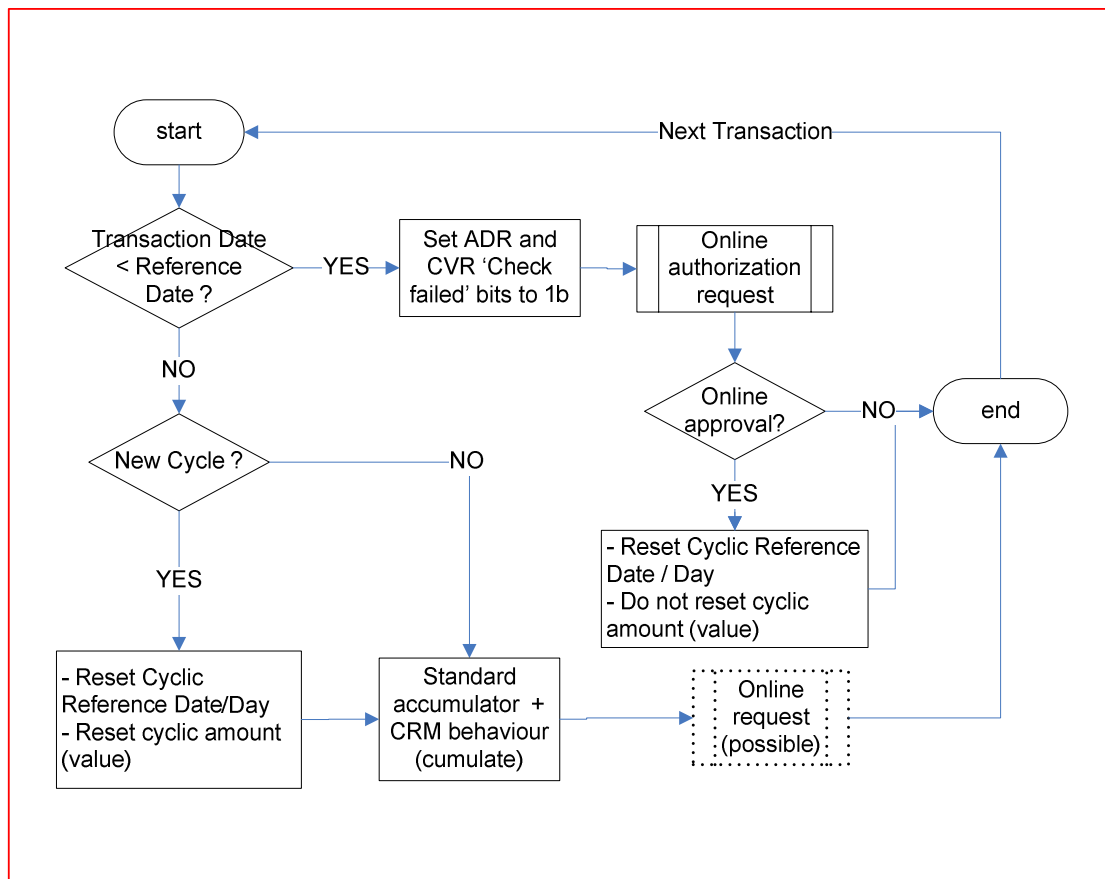
Figure I-2: Cyclic Accumulator Behaviour

I3 Management of Erroneous Transaction Dates

In certain cases, an improper Transaction Date may be supplied by the terminal. It is important that the CPA card be able to recover in such a situation without requiring an issuer script to correct the problem, so that the Cyclic Accumulator may continue to provide its card risk management function. An invalid date could be the result of a bad terminal configuration or, possibly, a fraudulent attempt to circumvent Cyclic Accumulator limits. Table I-4 describes two types of erroneous dates and describes the recovery process for each. Figure I-3 illustrates the recovery process.

Invalid Date Type	Description	Recovery
Past Date	Transaction Date is earlier than the actual date, and thus, most likely, earlier than the Reference Date/Day stored in the Cyclic Accumulator x Data object.	When the CPA card detects such an event, the desired behaviour is that the application sets the 'Check Failed' bit in the ADR to 1b, so that the issuer could force the transaction online by setting the associated CIAC bit to 1b. The application also sets the 'Check Failed' bit in the CVR to 1b to indicate to the issuer that an error occurred. Assuming that the CIACS are configured in such a way as to force the transaction online and that the transaction is approved online, the application will set the Cyclic Accumulator Reference Date/Day to the earlier date, but will not reset the Cyclic Accumulator value. Subsequent card risk management checks will thus take into account previously accumulated amounts, even though the Cycle has just been reset. During the next transaction containing a valid date, both the Cyclic Accumulator Reference Date/Day and Cyclic Accumulator value will probably be reset in the usual fashion.
Future Date	Transaction Date is later than the actual date, and thus (most likely) later than the Cyclic Accumulator Reference Date/Day.	As long as the Day, Month, and Year values are valid, this type of error will not be detected by the card. The standard behaviour would thus be to reset Cyclic Accumulator value and Reference Date to the faulty date contained in terminal data. This situation would be corrected during the following transaction, when the CPA card would evoke the rules above for Past Date, causing the application to attempt to send the transaction online, which would reset the Reference Date/Day to the proper value. As explained above, in this case the Cyclic Accumulator value will not be modified.

Table I-4: Management of Erroneous Transaction Dates

**Figure I-3: Management of Erroneous Transaction Dates**

I4 Cyclic Accumulator Usage Scenarios

This section presents five usage scenarios, illustrating the main features of Cyclic Accumulator behaviour and management.

I4.1 Example 1

This example illustrates the behaviour for a transaction conducted in a convertible currency, that occurs during the current Daily Cycle; and that, if approved offline, would exceed the Cyclic Accumulator limit.

Data Element	Value	Meaning
Cyclic Accumulator Profile Control y for Cyclic Accumulator x	'80 21'	Allow accumulation Limit Entry ID = 2 Currency Conversion Table ID = 1)
Cyclic Accumulator x Control	'09 78 40'	Accumulator Currency code = Euros Daily cycle Do not include online approved transactions
Cyclic Accumulator x Data	'00 00 00 08 54 00 05 07 15 00 00'	Accumulated value = 854.00 euros Reference Date = 15 July 2005
Limit Entry 2	'100000'	1000 euros
Currency Conversion Table 1	'09 78 08 40 00 82 82 01 24 00 67 82 08 10 00 28 83'	Target currency = Euros 1 US Dollar converts to 0.82 Euros 1 Canadian Dollar converts to 0.67 Euros 1 Russian Ruble converts to 0.028 Euros

Table I-5: Parameters for Cyclic Accumulator Usage Example 1

Transaction Date = '05 07 15' (July 15th 2005),

Transaction amount = '03 50 00'

Transaction Currency Code = '08 40' (350.00 US\$)

The Cyclic Accumulator Cycle Type is Daily, and Reference Date is equal to Transaction Date, so the transaction is in the current Cycle. The Cyclic Accumulator Profile Control options indicate that accumulation is allowed and Currency Conversion Table 1 is active. The Transaction Currency Code is different from the Cyclic Accumulator Currency Code, but the Transaction Currency Code is found in the chosen Currency Conversion Table (Currency Conversion Table 1), so the transaction can be converted using Currency Conversion Table 1.

Using the currency conversion factor 82 and exponent 10^{-2} (corresponding to US \$), the Transaction Amount converts to 287.00 Euros. When added to the current value stored in the Cyclic Accumulator, the new (temporary) Cyclic Accumulator value becomes 1141.00 (854.00 + 287.00), which is over the amount of 1000.00 stored in Limit Entry 2.

This transaction should thus go online, since the daily limit would be surpassed if this transaction were approved offline⁵. If this transaction is approved online, the state of the 'Include online approved transactions' bit will determine whether the actual Cyclic Accumulator value will be modified. Since, in this case, the relevant bit is not set, the Cyclic Accumulator value will remain at the original 854.00 euro balance. If however the terminal is unable to go online and the transaction is approved offline, then the Cyclic Accumulator value will be incremented.

⁵ Note: The actual decision of whether or not to go online depends on the settings in the CIACS Entry used for the current transaction.

14.2 Example 2

This example illustrates the behaviour for a transaction conducted in a convertible currency, that occurs during a new Weekly Cyclic period; and that, if approved offline, would not exceed the Cyclic Accumulator limit.

Data Element	Value	Meaning
Cyclic Accumulator Profile Control y for Cyclic Accumulator x	'80 31'	Allow accumulation Limit Entry ID = 3 Currency Conversion Table ID = 1)
Cyclic Accumulator x Control	'09 78 83'	Accumulator Currency code = Euros Weekly cycle Do not include online approved transactions First Day of cycle = 03 (Monday)
Cyclic Accumulator x Data	'00 00 00 28 69 50 05 07 11 07 E0'	Accumulated value = 2869.50 Euros Reference Date = 11 July 2005 Reference Day = 2016
Limit Entry ID 3	'300000'	3000 Euros
Currency Conversion Table 1	'09 78 08 40 00 82 82 01 24 00 67 82 08 10 00 28 83'	Target currency = Euros 1 US Dollar converts to 0.82 Euros 1 Canadian Dollar converts to 0.67 Euros 1 Russian Ruble converts to 0.028 Euros

Table I-6: Parameters for Cyclic Accumulator Usage Example 2

Transaction Date = '05 07 21' (July 21st 2005),

Transaction amount = '44 55 00'

Transaction Currency Code = '01 24' (4455.00 Can\$)

In this case, the Cyclic Accumulator Cycle Type is Weekly, and Transaction Date is more than 7 days past reference date (thus new cycle). By applying Transaction Date to Days conversion algorithm, a new reference Day of 2023 (that is, '07E7') is obtained, which is the equivalent of Monday July 18th 2005. New reference day must be stored in the Cyclic Accumulator x Data, and new cyclic amount becomes 0.00.

Cyclic Accumulator Profile Control options allow Accumulation, and Currency Conversion Table 1 is active. The Transaction Currency Code is found in current Currency Conversion Table 1. Using currency conversion factor 67 and exponent 10^{-2} (corresponding to Can \$), transaction amount converts to 2984.85 euros. Since current value stored in accumulator was just reset to 0, new (temporary) Cyclic Accumulator value becomes 2984.85 which is just below the amount of 3000.00 stored in Limit Entry 3.

This transaction could thus be accepted offline (barring any other card risk management factors), since the weekly limit would not be surpassed if this transaction was approved. If the transaction is accepted offline, the Cyclic Accumulator value will then be modified to the new total of 2984.85 euros.

14.3 Example 3

This example illustrates the behaviour for a transaction conducted in a non-convertible currency that occurs during the current Monthly Cycle of a Cyclic Accumulator which is already over limit.

Data Element	Value	Meaning
Cyclic Accumulator Profile Control y for Cyclic Accumulator x	'80 41'	Allow accumulation Limit Entry ID = 4 Currency Conversion Table ID = 1)
Cyclic Accumulator x Control	'09 78 C0'	Accumulator Currency code = Euros Monthly cycle Do not include online approved transactions
Cyclic Accumulator x Data	'00 00 01 51 09 45 05 07 15 00 00'	Accumulated value = 15109.45 Euros Reference Date = 15 July 2005
Limit Entry ID 4	'1500000'	15000 Euros
Currency Conversion Table 1	'09 78 08 40 00 82 82 01 24 00 67 82 08 10 00 28 83'	Target currency = Euros 1 US Dollar converts to 0.82 Euros 1 Canadian Dollar converts to 0.67 Euros 1 Russian Ruble converts to 0.028 Euros

Table I-7: Parameters for Cyclic Accumulator Usage Example 3

Transaction Date = '05 07 25' (July 25th 2005),

Transaction amount = '00 03 50 00'

Transaction Currency Code = '04 14' (350.00 Kuwaiti Dinars)

In this case, the Cyclic Accumulator Cycle Type is Monthly, and the month (as well as year) of the Transaction Date are equal to the month and year of the Reference date (thus same cycle).

The Cyclic Accumulator Profile Control options are such that Accumulation is allowed. The Transaction Currency Code is different from the Cyclic Accumulator currency code, and the Transaction Currency Code is not found in Currency Conversion Table 1. Therefore the transaction amount is not convertible, and cannot be added to the Cyclic Accumulator total.

However, regardless of the fact that current transaction is not accumulated, Cyclic Accumulator total is still over limit specified in Limit Entry 4 (Because of previous transactions during the current cycle). This transaction should therefore (once again) result in an online authorisation request, as will all other transactions performed during the current cycle (until reference date is reset at end of cycle).⁶

⁶ Note: in a case such as this (Currency not convertible), an offline counter will often be incremented by 1, and the state of this counter will also be evaluated during card risk management.

I4.4 Example 4

This example illustrates the behaviour for a transaction conducted in a convertible currency, that occurs during the current Monthly Cycle; and that, if approved offline, would exceed the Cyclic Accumulator limit. Because the 'Include online approved transactions' bit is set, the transaction amount is added to the Cyclic Accumulator.

Data Element	Value	Meaning
Cyclic Accumulator Profile Control y for Cyclic Accumulator x	'80 41'	Allow accumulation Limit Entry ID = 4 Currency Conversion Table ID = 1)
Cyclic Accumulator x Control	'09 78 E0'	Accumulator Currency code = Euros Monthly cycle Include online approved transactions
Cyclic Accumulator x Data	'00 00 01 49 89 45 05 07 15 00 00'	Accumulated value = 14989.45 Euros Reference Date = 15 July 2005
Limit Entry ID 4	'1500000'	15000 Euros
Currency Conversion Table 1	'09 78 08 40 00 82 82 01 24 00 67 82 08 10 00 28 83'	Target currency = Euros 1 US Dollar converts to 0.82 Euros 1 Canadian Dollar converts to 0.67 Euros 1 Russian Ruble converts to 0.028 Euros

Table I-8: Parameters for Cyclic Accumulator Usage Example 4

Transaction Date = '05 07 25' (July 25th 2005),
Transaction amount = '00 06 50 00'
Transaction Currency Code = '04 14' (650.00 Rubles)

In this case, the Cyclic Accumulator Cycle Type is Monthly, and the month (as well as year) of the Transaction Date are equal to the month and year of the Reference date (thus same cycle). The Cyclic Accumulator Profile Control options are such that Accumulation is allowed and Currency Conversion Table 1 is active. The Transaction Currency Code is different from the Cyclic Accumulator currency code, but the Transaction Currency Code is found in Currency Conversion Table 1.

Using currency conversion factor 28 and exponent 10^{-3} (corresponding to Russian Rubles), the transaction amount converts to 18.20 euros. When added to the current value stored in the Cyclic Accumulator, the new (temporary) Cyclic Accumulator value becomes 15007.65 ($14989.45 + 18.20$), which is over the amount of 15000.00 stored in Limit Entry 4.

This transaction should thus go online, since the Monthly limit would be surpassed if this transaction was approved offline. If this transaction is approved online, the state of the 'Include online approved transactions' bit dictates whether the actual Cyclic Accumulator value will be modified. Since, in this case, the relevant bit is set, the Cyclic Accumulator value will be incremented by the transaction amount of 18.20, and the new Cyclic Accumulator total will become 15007.65. (In consequence, all subsequent transactions occurring during this monthly cycle will probably also go online for an authorisation request).

If the online authorisation request is declined, the Cyclic Accumulator will keep its original value of 14989.45, since only approved transactions (offline and/or offline) are accumulated.

14.5 Example 5

This example illustrates the behaviour for a transaction conducted in the Accumulator currency that occurs during the current Daily Cycle of an Accumulator which does not allow accumulation in the currently selected Issuer profile.

Data Element	Value	Meaning
Cyclic Accumulator Profile Control y for Cyclic Accumulator x	'00 51'	Do not allow accumulation Limit Entry ID = 5 Currency Conversion Table ID = 1
Cyclic Accumulator x Control	'09 78 60'	Accumulator Currency code = Euros Daily cycle Include online approved transactions
Cyclic Accumulator x Data	'00 00 00 08 54 00 05 07 15 00 00'	Accumulated value = 854.00 euros Reference Date = 15 July 2005
Limit Entry ID 5	'100000'	1000 euros
Currency Conversion Table 1	'09 78 08 40 00 82 82 01 24 00 67 82 08 10 00 28 83'	Target currency = Euros 1 US Dollar converts to 0.82 Euros 1 Canadian Dollar converts to 0.67 Euros 1 Russian Ruble converts to 0.028 Euros

Table I-9: Parameters for Cyclic Accumulator Usage Example 5

Transaction Date = '05 07 25' (July 25th 2005),

Transaction amount = '00 01 50 00'

Transaction Currency Code = '09 78' (150.00 euros)

In this case, the Cyclic Accumulator Cycle Type is Daily, and the Reference date is equal to the Transaction Date (same cycle). Cyclic Accumulator Profile Control options are such that Currency Conversion Table 1 is active, but the 'Allow accumulation bit' is not set (the value is 0b). The Transaction Currency Code is the same as the Accumulator Currency Code, but since accumulation is not allowed in this profile, the current transaction amount is not added to the temporary Cyclic Accumulator total before performing card risk management.

If, for whatever reason (that is, some other Card Risk Management factor), this transaction should go online and be accepted by the Issuer, the Cyclic Accumulator value will not be modified regardless of the setting of the 'Include online approved transactions' bit. Likewise, if the transaction is accepted offline (that is, no issues requiring online authorization are discovered during Card Risk Management), modification of the Cyclic Accumulator value will again be prevented by the fact that the 'Allow accumulation' bit is not set. In both cases, the value of the Cyclic Accumulator will remain at 854.00 euros.

Annex J GET DATA and PUT DATA Data Elements

Table J-1 lists the data elements that are supported for the GET DATA and PUT DATA commands.

Data Element	Tag or Template Tag	GET DATA	PUT DATA
Accumulators Data	'BF30'	Y (if allowed by option ⁷ in Application Control)	Y
Accumulators Profile Controls	'BF31'	Y	Y
Accumulator x Controls	'BF32'	Y	Y
Additional Check Tables	'BF33'	Y	Y
AIP/AFL Entries	'BF41'	Y	Y
Application Control	'C1'	Y	Y
Application Life Cycle Data	'9F7E'	Y	N
Application Transaction Counter (ATC)	'9F36'	Y	N
CIACs Entries	'BF34'	Y	Y
Counters Data	'BF35'	Y (if allowed by option ⁸ in Application Control)	Y
Counters Profile Controls	'BF36'	Y	Y
Counter x Controls	'BF37'	Y	Y
Currency Conversion Tables	'BF38'	Y	Y

Table J-1: GET DATA Command Data Elements
(continues)

⁷ That is, if the 'Allow Retrieval of Values and Limits of Accumulators and Counters' bit in the Application Control has the value 1b.

Data Element	Tag or Template Tag	GET DATA	PUT DATA
Cyclic Accumulators Profile Controls	'BF39'	Y	Y
Cyclic Accumulator x Controls	'BF3A'	Y	Y
Cyclic Accumulator x Data	'BF42'	Y	Y
GPO Parameters	'BF3E'	Y	Y
Issuer Options Profile Controls	'BF3B'	Y	Y
Limits Entries	'BF3C'	Y	Y
Log Data Tables	'BF40'	Y	N
Log Format	'9F4F'	Y	N
MTA Profile Controls	'BF3D'	Y	Y
Number of Days Offline Limit	'C3'	Y	Y
Offline Balance	'9F50'	Y	N
Offline Balance Currency Code	'C9'	Y	N
PIN Try Counter	'9F17'	Y	N
Profile Controls	'BF3F'	Y	Y
Profile Selection File Entry	'C2'	Y	Y
Security Limits Status	'C4'	Y (if 'Security Limits' is supported by the card)	N
Security Limits	'C5'	N	Y (if 'Security Limits' is supported by the card)
VLP Funds Limit	'9F77'	Y (if VLP is supported by the card)	Y (if VLP is supported by the card)
VLP Single Transaction Limit	'9F78'	Y (if VLP is supported by the card and data element is present)	Y (if VLP is supported by the card)

Table J-1: Get Data Command Data Elements, continued

Annex K Data Element Tags

Table K-1 lists the Tags and Template Tags used by CPA, including the tags of EMV data elements used by CPA.

Data Element	Template or Tag	Source
Application Identifier (AID)	'4F'	Card
Track 2 Equivalent Data	'57'	Card
Application Primary Account Number (PAN)	'5A'	Card
Cardholder Name	'5F20'	Card
Application Expiration Date	'5F24'	Card
Application Effective Date	'5F25'	Card
Issuer Country Code	'5F28'	Card
Transaction Currency Code	'5F2A'	Terminal
Service Code	'5F30'	Card
Application PAN Sequence Number	'5F34'	Card
DF Name	'6F'	Card
Application Interchange Profile (AIP)	'82'	Card
Dedicated File (DF) Name	'84'	Card
Short File Identifier (SFI)	'88'	Card
Authorisation Response Code	'8A'	Terminal
Card Risk Management Data Object List 1 (CDOL 1)	'8C'	Card
Card Risk Management Data Object List 2 (CDOL 2)	'8D'	Card
Cardholder Verification Method (CVM) List	'8E'	Card
Certificate Authority (CA) Public Key Index	'8F'	Card
Issuer Public Key Certificate	'90'	Card
Issuer Authentication Data	'91'	Terminal

Table K-1: Data Element Tags
(continues)

Data Element	Template or Tag	Source
Issuer Public Key Remainder	'92'	Card
Signed Static Application Data (SSAD)	'93'	Card
Application File Locator (AFL)	'94'	Card
Terminal Verification Results (TVR)	'95'	Terminal
Transaction Date	'9A'	Terminal
Transaction Type	'9C'	Terminal
Amount, Authorised (numeric)	'9F02'	Terminal
Application Discretionary Data	'9F05'	Card
Application Usage Control (AUC)	'9F07'	Card
Application Version Number	'9F08'	Card
Cardholder Name Extended	'9F0B'	Card
IAC-Default	'9F0D'	Card
IAC-Denial	'9F0E'	Card
IAC-Online	'9F0F'	Card
Issuer Application Data	'9F10'	Card
PIN Try Counter	'9F17'	Card
Terminal Country Code	'9F1A'	Terminal
Track 1 Discretionary Data	'9F1F'	Card
Application Cryptogram	'9F26'	Card
Cryptogram Information Data	'9F27'	Card
ICC PIN Encipherment Public Key Certificate	'9F2D'	Card
ICC PIN Encipherment Public Key Exponent	'9F2E'	Card
ICC PIN Encipherment Public Key Remainder	'9F2F'	Card
Issuer Public Key Exponent	'9F32'	Card
Terminal Capabilities	'9F33'	Terminal
Cardholder Verification Method (CVM) Results	'9F34'	Terminal
Terminal Type	'9F35'	Terminal
Application Transaction Counter (ATC)	'9F36'	Card

Table K-1: Data Element Tags, continued

Data Element	Template or Tag	Source
Unpredictable Number	'9F37'	Terminal
Processing Options Data Object List (PDOL)	'9F38'	Card
Additional Terminal Capabilities	'9F40'	Terminal
Application Currency Code	'9F42'	Card
Application Currency Exponent	'9F44'	Card
ICC Public Key Certificate	'9F46'	Card
ICC Public Key Exponent	'9F47'	Card
ICC Public Key Remainder	'9F48'	Card
Dynamic Data Authentication Data Object List (DDOL)	'9F49'	Card
SDA Tag List	'9F4A'	Card
Signed Dynamic Application Data	'9F4B'	Card
ICC Dynamic Number	'9F4C'	Card
Log Entry	'9F4D'	Card
Log Format	'9F4F'	Card
Offline Balance	'9F50'	Card
Card Verification Results (CVR)	'9F52'	Card
VLP Issuer Authorisation Code ⁸	'9F74'	Card
VLP Funds Limit ⁸	'9F77'	Card
VLP Single Transaction Limit ⁸	'9F78'	Card
VLP Available Funds ⁸	'9F79'	Card
VLP Terminal Support Indicator ⁸	'9F7A'	Terminal
Application Life Cycle Data	'9F7E'	Card
Accumulators Data	'BF30'	Card
Accumulator Profile Controls	'BF31'	Card
Accumulator Controls	'BF32'	Card

Table K-1: Data Element Tags, continued

⁸ If VLP is supported by the card

Data Element	Template or Tag	Source
Additional Check Tables	'BF33'	Card
Card Issuer Action Codes (CIACs) Entries	'BF34'	Card
Counter x Data	'BF35'	Card
Counter Profile Controls	'BF36'	Card
Counter x Controls	'BF37'	Card
Currency Conversion Tables	'BF38'	Card
Cyclic Accumulator Profile Controls	'BF39'	Card
Cyclic Accumulator x Controls	'BF3A'	Card
Issuer Options Profile Controls	'BF3B'	Card
Limits Entries	'BF3C'	Card
MTA Profile Controls	'BF3D'	Card
GPO Parameters	'BF3E'	Card
Profile Controls	'BF3F'	Card
Log Data Tables	'BF40'	Card
AIP/AFL Entries	'BF41'	Card
Cyclic Accumulator x Data	'BF42'	Card
Application Control	'C1'	Card
Profile Selection File Entry	'C2'	Card
Number of Days Offline Limit	'C3'	Card
Security Limits Status	'C4'	Card
Security Limits	'C5'	Card
PIN Try Limit	'C6'	Card
Previous Transaction History (PTH)	'C7'	Card
Application Issuer Life Cycle Data	'C8'	Card
Offline Balance Currency Code	'C9'	Card

Table K-1: Data Element Tags, continued

NOTE: Tags in the range 'CA' to 'CF' are RFU for this specification.

NOTE: Template tags in the range 'BF43' to 'BF49' are RFU for this specification.

Annex L Data Dictionary

This annex defines:

- new data elements defined for the Common Payment Application,
- and EMV data elements for which the coding/use of the data element needs further definition for CPA.

For convenience, it also includes CCD- and EMV-defined data that is used in the Common Payment Application.

Note: Bits and bytes identified as RFU should be set to zero to avoid conflict with potential future use of the bits or bytes. Application behaviour should not depend on the setting of these bits because they may be used for future functionality.

Elements Described

The following data elements are described in this annex:

AC Session Key Counter	L-6
AC Session Key Counter Limit	L-6
Accumulator Currency Code	L-6
Accumulator Profile Control for Accumulator x	L-6
Accumulator Profile Control x	L-7
Accumulator x	L-8
Accumulator x Balance	L-9
Accumulator x Control	L-10
Accumulator x Data	L-11
Accumulator x Limits	L-11
Additional Check Table x	L-12
Additional Terminal Capabilities	L-14
AIP/AFL Entry x	L-14
Amount, Authorised	L-15
Amount, Other	L-15
Application Control	L-16
Application Cryptogram	L-18
Application Currency Code	L-19
Application Currency Exponent	L-19
Application Decisional Results (ADR)	L-20

Application Discretionary Data	L-22
Application Effective Date	L-22
Application Expiration Date	L-23
Application Elementary File (AEF)	L-23
Application File Locator (AFL)	L-23
Application Identifier (AID)	L-23
Application Interchange Profile (AIP)	L-23
Application Issuer Life Cycle Data	L-24
Application Life Cycle Data	L-25
Application Primary Account Number (PAN)	L-26
Application Transaction Counter (ATC)	L-26
Application Usage Control (AUC)	L-26
Application Version Number	L-27
Authorisation Response Code	L-27
Card Issuer Action Codes Entry x (CIACs Entry x)	L-28
Card Risk Management Data Object List 1 (CDOL1)	L-31
Card Risk Management Data Object List 2 (CDOL2)	L-32
Card Status Update (CSU)	L-33
Card Verification Results (CVR)	L-34
Cardholder Name	L-36
Cardholder Name Extended	L-36
Cardholder Verification Method (CVM) List	L-36
Cardholder Verification Method (CVM) Results	L-36
Certificate Authority (CA) Public Key	L-37
Certificate Authority (CA) Public Key Index	L-37
Common Core Identifier (CCI)	L-37
Counter Profile Control for Counter x	L-37
Counter Profile Control x	L-38
Counter x	L-39
Counter x Control	L-40
Counter x Data	L-41
Counter x Limits	L-41
Cryptogram Information Data (CID)	L-42
Currency Conversion Table x	L-42
CVM Results	L-43
Cyclic Accumulator Profile Control for Cyclic Accumulator x	L-43
Cyclic Accumulator Profile Control x	L-44
Cyclic Accumulator x	L-45

Cyclic Accumulator x Control	L-45
Cyclic Accumulator x Data	L-47
Cyclic Accumulator x Reference Date	L-47
Cyclic Accumulator x Reference Day	L-48
Derivation Key Index (DKI)	L-48
DF Name	L-48
Dynamic Data Authentication Data Object List (DDOL)	L-48
File Control Information (FCI) Proprietary Template	L-48
File Control Information (FCI) Template	L-49
First GEN AC Log Data Table	L-50
First GEN AC Unchanging Log Data Table	L-51
First GENERATE AC Command Data Length	L-52
GPO Command Data	L-53
GPO Command Data Length	L-53
GPO Parameters x	L-53
IAC - Default	L-54
IAC - Denial	L-54
IAC - Online	L-54
ICC Dynamic Data	L-55
ICC Dynamic Number	L-55
ICC PIN Encipherment Private Key	L-55
ICC PIN Encipherment Public Key Certificate	L-56
ICC PIN Encipherment Public Key Exponent	L-56
ICC PIN Encipherment Public Key Remainder	L-56
ICC Private Key	L-56
ICC Public Key (PK) Certificate	L-56
ICC Public Key Exponent	L-57
ICC Public Key Remainder	L-57
ICC Unpredictable Number	L-57
Internal Flags	L-57
Issuer Action Codes (IACs)	L-58
Issuer Application Data	L-58
Issuer Authentication Data	L-59
Issuer Country Code	L-59
Issuer Options Profile Control x	L-60
Issuer Public Key (PK) Certificate	L-61
Issuer Public Key Exponent	L-61
Issuer Public Key Remainder	L-62

Issuer Script Command Counter	L-62
Issuer Script Results	L-62
Last Online Transaction Date in days	L-63
Limits Entry x	L-63
Log Entry	L-64
Log Format	L-65
Master Key for AC	L-67
Master Key for SMC	L-67
Master Key for SMI	L-67
Maximum Transaction Amount (MTA) Profile Control x	L-68
Number of Days Offline Limit	L-69
Offline Balance Currency Code	L-69
Offline Balance	L-70
PDOL	L-70
PDOL Related Data	L-70
PIN Decipherments Error Counter	L-71
PIN Decipherments Error Counter Limit	L-71
PIN Try Counter	L-71
PIN Try Limit	L-71
Previous Transaction History (PTH)	L-72
Processing Options Data Object List (PDOL)	L-73
Profile Control x	L-74
Profile ID	L-75
Profile Selection Diversifier	L-76
Profile Selection Entries	L-76
Profile Selection File Entry	L-79
Reference PIN	L-79
Registered Application Identifier (RID)	L-79
SDA Tag List	L-80
Second GEN AC Log Data Table	L-81
Second GENERATE AC Command Data Length	L-82
Security Limits	L-82
Security Limits Status	L-83
Service Code	L-83
Session Key for AC	L-84
Session Key for SMC	L-84
Session Key for SMI	L-84
Short File Identifier (SFI)	L-84

Signed Dynamic Application Data	L-84
Signed Static Application Data (SSAD)	L-85
SMI Session Key Counter	L-85
SMI Session Key Counter Limit	L-85
Source Currency Code	L-86
Static Data Authentication Tag List	L-86
Target Currency Code	L-86
Terminal Action Code – Default	L-87
Terminal Action Code – Denial	L-87
Terminal Action Code – Online	L-87
Terminal Action Codes (TACs)	L-87
Terminal Capabilities	L-87
Terminal Country Code	L-88
Terminal Type	L-88
Terminal Verification Results (TVR)	L-88
Track 1 Discretionary Data	L-88
Track 2 Equivalent Data	L-88
Transaction Currency Code	L-89
Transaction Date	L-89
Transaction Date in Days	L-89
Transaction PIN	L-89
Transaction Type	L-89
Unpredictable Number	L-90
VLP Available Funds	L-90
VLP Funds Limit	L-90
VLP Issuer Authorisation Code	L-90
VLP Profile Control	L-91
VLP Single Transaction Limit	L-91
VLP Terminal Support Indicator	L-92

AC Session Key Counter

Tag: — The AC Session Key Counter counts the number of
Length: 2 Application Cryptogram session key derivations since the
Format: b application last successfully validated an ARPC.

Note: This is an implementer-optional counter that could be
used to implement security requirements in the
application instead of in the platform. See Annex F.

AC Session Key Counter Limit

Tag: — The AC Session Key Counter Limit limits the number of
Length: 2 Application Cryptogram session key derivations since the
Format: b application last successfully validated an ARPC.

Note: This is an implementer-optional limit that could be used
to implement security requirements in the application
instead of in the platform. See Annex F.

Accumulator Currency Code

Tag: — Indicates the currency in which the accumulator is
Length: 2 managed. The value is coded according to ISO 4217.

Format: n 3 This parameter is part of the Accumulator x Control data
element.

Accumulator Profile Control (x = 1 to 14) for Accumulator x

Tag: - The Accumulator Profile Control y that is used to control
Length: 2 Accumulator x in the Profile selected for the transaction.
Format: b y is the value of the Accumulator Profile Control ID for
Accumulator x in the Profile Control selected for the
transaction.

Accumulator Profile Control x

(x = 1 to 14)

Template: 'BF31' Indicates the issuer's choice of data and behaviour to
 Tag: 'DF0x' configure an accumulator within a Profile.
 Length: 2 The Accumulator Profile Controls template may be obtained
 Format: b from the application using the GET DATA command for
 template tag 'BF31', and may be updated in the application
 using the PUT DATA command.

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
x								Allow Accumulation
0								Do Not Allow Accumulation ¹
1								Allow Accumulation
	1							Reset Accumulator with online response
		1						Send Accumulator in IAD
			x					Send Accumulator Balance
			0					Send Accumulator x value
			1					Send Accumulator x Balance
				x				RFU
					x			RFU
						x		RFU
							x	RFU

Table L-1: Accumulator Profile Control x, Byte 1

¹ This setting allows the Velocity-Checking for Accumulator x Check to include accumulators that cannot be increased in the profile selected for the transaction.

b8	b7	b6	b5	b4	b3	b2	b1	Meaning	
x	x	x						RFU	
								x	Limit Set ID
								0	Use Limit Set 0
								1	Use Limit Set 1
			x	x	x	x	Currency Conversion Table ID		
			1	1	1	1	Currency Conversion Not Allowed		

Table L-2: Accumulator Profile Control x, Byte 2

Accumulator x

(x = 1 to 14)

Template: 'BF30'

Tag: 'DF0x'

Length: 6

Format: n

Represents a cumulative amount of transactions. May include offline approved transactions, and may also include online approved transactions. Transactions can be accumulated if they are in the accumulator currency, or (if currency conversion is allowed for the accumulator) in a currency that can be converted to the accumulator currency using the Currency Conversion Table for Accumulator x.

The value of Accumulator x may be sent to the issuer as part of the Issuer Application Data.

The value of Accumulator x may be obtained from the application using the GET DATA command for template tag 'BF30' (Accumulators Data), if allowed by the Application Control; and may be updated using the PUT DATA command for template tag 'BF30', data element tag 'DF0x'.

The value of Accumulator x may be obtained from the application as part of the Accumulators Data template, if allowed by the 'Allow Retrieval of Values and Limits of Accumulators and Counters' bit in the Application Control.

Accumulator x Balance**(x = 1 to 14)**

Tag: — Represents the amount of offline spending available for
Length: 6 Accumulator x.
Format: 12 n Accumulator x Balance may be sent to the issuer as part of
the Issuer Application Data.

For any Accumulator x, the Accumulator x Balance is computed as follows:

Accumulator x Balance = Accumulator x Upper Limit minus
Accumulator x value.

If Accumulator x Upper Limit < Accumulator x value, then the value sent
in the Issuer Application Data for Accumulator x Balance is zero
(‘00 00 00 00 00 00’).

Note: The Accumulator x Upper Limit used in the computation of the Accumulator x
Balance is the Upper Limit (identified by the Limit Set ID in the Accumulator
Profile Control for Accumulator x) used for Accumulator x in the profile.

Accumulator x Control

(x = 1 to 14)

Template: 'BF32' Indicates the issuer's choice of data and behaviour to
 Tag: 'DF0x' configure Accumulator x independent of a Profile.
 Length: 3 The Accumulator Controls template may be obtained from
 Format: b the application using the GET DATA command for template
 tag 'BF32', and may be updated in the application using the
 PUT DATA command.

Position	Data	Length	Value
byte 1-2	Accumulator Currency Code	2	Currency Code in which the accumulator is managed, coded according to ISO 4217
byte 3	Accumulator Parameters	1	See Table L-4

Table L-3: Accumulator x Control

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
1								Include ARQC Transaction in CRM Test
	1							Include Offline Approvals ²
		x						RFU
			x					RFU
				x				RFU
					x			RFU
						x		RFU
							x	RFU

Table L-4: Accumulator Parameters

² If this bit is set to the value 0b, then Accumulator x will accumulate only online transactions (which may be accumulated when the Accumulator is updated during processing of the second GENERATE AC).

Accumulator x Data

(x = 1 to 14)

Template: 'BF30' Represents both the value of Accumulator x and
 Tag: 'DF0x', 'DF1x' Accumulator x Limits.
 Length: 18 or 30 The Accumulators Data template may be obtained from the
 Format: n application using the GET DATA command with template tag
 'BF30', if allowed by the Application Control; and may be
 updated using the PUT DATA command.

The value and Limits of Accumulators may be obtained from the application, if allowed by the 'Allow Retrieval of Values and Limits of Accumulators and Counters' bit in the Application Control.

Accumulator x Limits

(x = 1 to 14)

Template: 'BF30' Contains one or two sets of lower and upper limits for
 Tag: 'DF1x' Accumulator x.
 Length: 12 or 24 The Accumulator x Limits may be obtained from the
 Format: n application using the GET DATA command with template tag
 'BF30' (Accumulators Data), if allowed by the Application
 Control; and may be updated using the PUT DATA command
 for template tag 'BF30', data element tag 'DF1x'.

The value of Accumulator x Limits may be obtained from the application as part of the Accumulators Data template, if allowed by the 'Allow Retrieval of Values and Limits of Accumulators and Counters' bit in the Application Control.

Data Element	Length
Accumulator x Lower Limit 0	6 Bytes
Accumulator x Upper Limit 0	6 Bytes
Accumulator x Lower Limit 1	6 Bytes
Accumulator x Upper Limit 1	6 Bytes

Table L-5: Accumulator x Data

Additional Check Table x

(x = 1 or 2)

Template: 'BF33'

Tag: 'DF0x'

Length: var.

Format: b

Additional Check Table x contains values that are compared during Card Risk Management to values sent by the terminal in the First GENERATE AC Command Data. The result of the comparison is reflected in the Application Decisional Results and the Card Verification Results as Match Found in Additional Check Table x, or No Match Found in Additional Check Table x.

The check with the Additional Check Table x is performed only if the 'Activate Additional Check Table x' bit in the Issuer Options Profile Control is set to 1b.

The Additional Check Tables template may be obtained from the application using the GET DATA command for template tag 'BF33', and may be updated in the application using the PUT DATA command.

Each Additional Check Table is coded as the concatenation (without TLV coding) of the parameters shown in Table L-6.

Data Element	Length	Format	Description
Position in First GENERATE AC Command Data	1	b	The starting position (in bytes) of the portion of the data extracted from the First GENERATE AC Command Data that is compared to the Comparison Values listed in Additional Check Table x. If the first byte in the First GENERATE AC Command Data is extracted for comparison, the value of Position in First GENERATE AC Command Data is '01'.
Comparison Data Length	1	b	The length in bytes of the data extracted from the First GENERATE AC Command Data that is compared to the Comparison Values.

Table L-6: Additional Check Table x
(continues)

Data Element	Length	Format	Description	
Number Of Comparison Blocks (n)	1	b	The number of Comparison Blocks in Additional Check Table x. The Comparison Blocks include the Bit Mask and the Comparison Values.	
Comparison Values	var.	b	Contains the concatenation of the Bit Mask and the Comparison Values used for the comparison. The first Comparison Block is used as a Bit Mask.	
	Comparison Data Length	b	Bit Mask	Used to mask the comparison data to allow only selected portions of the extracted data to be compared with the Comparison Value(s). (For example, the comparison may be made with only a few bits of a byte). Each bit that is to be used in the comparison is set to 1b. Each bit that is not to be used in the comparison is set to 0b.
	Comparison Data Length	b	Comparison Value	Each Comparison Value data element contains a value to be compared to the masked data extracted from the First GENERATE AC command data.

Table L-6: Additional Check Table x, continued

Additional Terminal Capabilities

Tag: '9F40'	Indicates the data input and output capabilities of the terminal.
Length: 5	
Format: b	For details, see <i>EMV Book 4</i> , Annex A, Table 27 through Table 31.

AIP/AFL Entry x (x = 1 to 14)

Template: 'BF41'	Indicates the issuer's choice of AIP and AFL used in Initiate Application Processing to generate the response to the GET PROCESSING OPTIONS command.
Tag: 'DF0x'	
Length: var.	
Format: b	The AIP/AFL Entries template may be obtained from the application using the GET DATA command for template tag 'BF41', and may be updated in the application using the PUT DATA command.

At a minimum, the Application shall be able to support up to six AIP/AFL Entries. The AIP/AFL Entry used for the transaction is identified in the Profile Control for the transaction. If the AIP/AFL ID in the Profile Control = y, then AIP/AFL Entry y will be used for the transaction.

Bytes 1-2	Byte 3	Bytes 4-64
AIP x	AFL x Length	AFL x

Table L-7: AIP/AFL Entry x

Amount, Authorised

Tag: '9F02'	Authorised amount of the transaction (excluding
Length: 6	adjustments).
Format: n 12	

Amount, Other

Tag: '9F03'	Secondary amount associated with the transaction
Length: 6	representing a cashback amount.
Format: n 12	

Application Control

Tag: 'C1'
Length: 4
Format: b

The Application Control activates or de-activates functions in the application. The Application Control value is the same for all transactions, regardless of the selected profile.

Byte 4 is reserved for Payment Systems and Issuers to use for control of add-on functionality not specified in CCD.

The value of the Application Control may be obtained from the application using the GET DATA command, and may be updated in the application using the PUT DATA command.

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
1								Issuer Authentication Required to be performed
	1							Issuer Authentication Required to Pass when Performed
		1						Issuer Authentication Requirements apply to Resetting of Non-Velocity-Checking Indicators and Counters ³
			1					Issuer Authentication Requirements apply to Resetting of Velocity-Checking Counters ⁴
				x				Key pair used for Offline Enciphered PIN Verification (RFU for SDA-only application) ⁵
				0				Use ICC Public/Private key pair
				1				Use ICC PIN Encipherment Public/Private key pair
					1			Offline Enciphered PIN Verification Supported (RFU for SDA-only application)
						1		Offline Plaintext PIN Verification Supported
							1	Allow Retrieval of Values and Limits of Accumulators and Counters

Table L-8: Application Control, Byte 1

³ If Issuer Authentication is required to be performed, Issuer Authentication is mandatory for resetting of non-velocity-checking indicators and counters.

⁴ If Issuer Authentication is required to be performed, Issuer Authentication is mandatory for resetting of velocity-checking indicators and counters.

⁵ If the VERIFY command is implemented at the application level, this bit indicates which key is be used for Offline Enciphered PIN verification. If the VERIFY command is implemented by the platform instead of the application, the method for determining which key is used is beyond the scope of this specification, and this bit is RFU.

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
x								Use Default Update Counters to Control Offline Counters if CSU is Generated by Issuer Proxy
0								Use Update Counters Received in CSU to Control Offline Counters if CSU is Generated by Issuer Proxy
1								Use Default Update Counters in Application Control to Control Offline Counters if CSU is Generated by Issuer Proxy
		x	x					Default Update Counters
		0	0					Do Not Update Offline Counters
		0	1					Set Offline Counters to Upper Offline Limits
		1	0					Reset Offline Counters to Zero
		1	1					Add Transaction to Offline Counters
			1					Activate VLP
				1				Activate Profile Selection File
					1			Amounts included in CDOL2
						x		RFU
							x	RFU

Table L-9: Application Control, Byte 2

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
1								Log Declined Transactions
	1							Log Approved Transactions
		x						Log Offline Only ⁶
		0						Log Both Offline and Online
		1						Log Offline Only
			1					Log the ATC
				1				Log the CID
					1			Log the CVR
						x		RFU
							x	RFU

Table L-10: Application Control, Byte 3

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
x	x	x	x					Reserved for use by Payment Systems for optional extensions
0	0	0	0					Disables Payment Systems optional extensions functionality
				x	x	x	x	Reserved for use by Issuer
				0	0	0	0	Disables add-on Functionality so that application applies the standard CPA functionality

Table L-11: Application Control, Byte 4

Application Cryptogram

Tag: '9F26'

Length: 8

Format: b

Cryptogram returned by the ICC in response to the GENERATE AC command.

⁶ This option applies only to approved transactions. If declined transactions are logged; both offline declined and online declined transactions are logged, regardless of the setting of this bit.

Application Currency Code

Tag: '9F42'	Indicates the currency in which the account is managed
Length: 2	according to ISO 4217.
Format: n 3	

Application Currency Exponent

Tag: '9F44'	Indicates the implied position of the decimal point from the
Length: 1	right of the amount represented according to ISO 4217.
Format: n 1	

Application Decisional Results (ADR)

Tag: —
 Length: 6
 Format: b

Application Decisional Results is used internal to the application to indicate exception conditions that occurred during the current and previous transactions. The Card Issuer Action Codes (CIAC - Decline, CIAC - Online, and CIAC - Default) are each compared to the Application Decisional Results to determine whether the transaction should be declined offline or go online. The format and coding of Application Decisional Results is the same as for each CIAC, described on page L-28.

The Application Decisional Results data element is coded as follows:

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
1								Last Online Transaction Not Completed
	1							Go Online On Next Transaction Was Set
		1						Issuer Script Processing Failed
			1					Issuer Authentication Failed
				1				Issuer Authentication Data Not Received in Previous Online Transaction
					1			PIN Try Limit Exceeded
						1		Offline PIN Verification Not Performed
							1	Offline PIN Verification Failed

Table L-12: Application Decisional Results, Byte 1

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
1								Unable To Go Online
	1							Terminal Erroneously Considers Offline PIN OK
		1						Script Received
			1					Offline Data Authentication Failed on Previous Transaction
				1				Match Found In Additional Check Table 1
					1			No Match Found In Additional Check Table 1
						1		Match Found In Additional Check Table 2
							1	No Match Found In Additional Check Table 2

Table L-13: Application Decisional Results, Byte 2

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
1								Accumulator 1 Lower Limit Exceeded
	1							Accumulator 2 Lower Limit Exceeded
		1						Counter 1 Lower Limit Exceeded
			1					Counter 2 Lower Limit Exceeded
				1				Counter 3 Lower Limit Exceeded
					1			Additional Accumulator Lower Limit Exceeded
						1		Additional Counter Lower Limit Exceeded
							1	Number of Days Offline Limit Exceeded

Table L-14: Application Decisional Results, Byte 3

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
1								Accumulator 1 Upper Limit Exceeded
	1							Accumulator 2 Upper Limit Exceeded
		1						Counter 1 Upper Limit Exceeded
			1					Counter 2 Upper Limit Exceeded
				1				Counter 3 Upper Limit Exceeded
					1			Additional Accumulator Upper Limit Exceeded
						1		Additional Counter Upper Limit Exceeded
							1	MTA exceeded

Table L-15: Application Decisional Results, Byte 4

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
1								Cyclic Accumulator 1 Limit Exceeded
	1							Cyclic Accumulator 2 Limit Exceeded
		1						Additional Cyclic Accumulator Limit Exceeded
			1					Check Failed ⁷
				x				RFU
					x			RFU
						x		RFU
							x	RFU

Table L-16: Application Decisional Results, Byte 5

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
x	x	x	x	x	x	x	x	Reserved for use by issuer

Table L-17: Application Decisional Results, Byte 6

Application Discretionary Data

Tag: '9F05' Issuer or payment system specified data relating to the application.
Length: 1-32
Format: b

Application Effective Date

Tag: '5F25' Date from which the application may be used.
Length: 3
Format: n 6
(Yymmdd)

⁷ This bit is used to identify that at least one of many possible error conditions has occurred (for example, data is missing or has the wrong format), preventing the card from successful processing of card risk management checks.

Application Expiration Date

Tag: '5F24' Date after which application expires.
Length: 3
Format: n 6
 (YYMMDD)

Application Elementary File (AEF)

Tag: —	An Application Elementary File (AEF) in the range 1-10,
Length: var.	contains one or more primitive BER-TLV data objects
Format: var.	grouped into constructed BER-TLV data objects (records)
	according to <i>EMV Book 3</i> , Annex B.

Application File Locator (AFL)

Tag: '94'	Indicates the location (SFI, range of records) of the AEFs related to a given application. For details, see <i>EMV Book 3</i> , section 10.2.
Length: var. up to 252	
Format: var.	

Application Identifier (AID)

Tag: '4F'	Identifies the application as described in ISO/IEC 7816-5.
Length: 5-16	For details, see <i>EMV Book 1</i> , section 12.2.1.
Format: b	

Application Interchange Profile (AIP)

Tag: '82'	Indicates the capabilities of the card to support specific
Length: 2	functions in the application.
Format: b	For details, see <i>EMV Book 3</i> , Annex C, Table 37.

Application Issuer Life Cycle Data

Tag⁸: 'C8'
Length: 20
Format: b

Application Issuer-specified data element that is the portion of the Application Life Cycle Data element that is personalised on the card.

The coding of the contents of this field will not be specified by EMVCo but is at the discretion of the issuer, and may be set during data preparation.

The Application Life Cycle Data may be obtained from the application only as part of the Application Life Cycle Data (tag '9F7E') using the GET DATA command.

⁸ This tag is required for this data element only when the EMV CPS implementer-option is supported. If EMV CPS is not supported, this specification does not require that the data element be tagged, and does not require this value for the tag associated with this data element.

Application Life Cycle Data

Tag: '9F7E'

Length: 48

Format: b

The purpose of the Application Life Cycle Data is to uniquely identify the application version and card approval session. It also allocates a portion of Life-Cycle Data for use by the application issuer and application provider. The coding of this data element is shown below.

The Application Life Cycle Data may be obtained from the application using the GET DATA command.

Data Element	Length	Format	Description
Version Number	1	b	Identifies the version of the CPA implemented in the application. This data element is not personalised on the card, but may be set during pre-personalisation or coded into the application. CPA Version 1 shall assign two values: <ul style="list-style-type: none"> '00' = CPA SDA-only implementation '01' = CPA RSA-capable implementation
Card Approval ID	7	b	Identifier assigned by EMVCo before the application is submitted for Card Approval. This data element may not be modified after the application passes Card Approval. It is not personalised on the card, but may be set during pre-personalisation or coded into the application.
Application Issuer Life Cycle Data	20	b	Application Issuer-specified data element. The contents of this data element are personalised on the card, and may be set during data preparation. The coding of the contents of this field will not be specified by EMVCo but is at the discretion of the issuer.

Table L-18: Application Life Cycle Data
(continues)

Data Element	Length	Format	Description
Application Code ID	20	b	<p>Application provider-specified data element that differentiates between different application behaviours.</p> <p>This data element is not personalised on the card, but may be set during pre-personalisation or coded into the application.</p> <p>The coding of the contents of this field will not be specified by EMVCo but is at the discretion of the application provider.</p>

Table L-18: Application Life Cycle Data, continued

Application Primary Account Number (PAN)

Tag: '5A' Valid cardholder account number.

Length: var. up
 to 10

Format: cn var. up
 to 19

Application Transaction Counter (ATC)

Tag: '9F36' Counter maintained by the application in the ICC
 (incrementing the ATC is managed by the ICC).

Length: 2

Format: b The value of the Application Transaction Counter may be
 obtained from the application using the GET DATA
 command.

Application Usage Control (AUC)

Tag: '9F07' Indicates issuer's specified restrictions on the geographic
 usage and services allowed for the application.

Length: 2

Format: b For details, see *EMV Book 3*, Annex C, Table 38.

Application Version Number

Tag: '9F08'	Version number assigned by the payment system for the
Length: 2	application.
Format: b	

Authorisation Response Code

Tag: '8A'	Code that defines the disposition of a message.
Length: 2	For details, see <i>EMV Book 4</i> , Annex A, Table 34.
Format: an 2	

Card Issuer Action Codes Entry x (CIACs Entry x)

(x = 1 to 14)

Template: 'BF34'	Each Card Issuer Action Codes Entry contains the CIACs for Decline, Default, and Online. Each CIAC is compared to the Application Decisional Results to take transaction decisions.	
Tag: 'DF0x'		
Length: 18		
Format: b	CIAC - Decline	Used by the issuer to set the situations when a transaction is always declined at the first GENERATE AC.
	CIAC - Online	Used by the issuer to set the situations when a transaction goes online if the terminal is online-capable.
	CIAC - Default	Used by the issuer to set the situations when a transaction is declined if the terminal is not online-capable or if connection to the issuer is not possible.
The CIACs Entries template may be obtained from the application using the GET DATA command with template tag 'BF34', and may be updated in the application using the PUT DATA command.		

Card Issuer Action Codes is the concatenation of the data elements listed in Table L-19.

Position	Data	Length
Bytes 1 - 6	CIAC - Decline	6 Byte
Bytes 7 - 12	CIAC - Default	6 Byte
Bytes 13 - 18	CIAC - Online	6 Byte

Table L-19: Card Issuer Action Codes

Each CIAC is coded as follows:

b8	b7	b6	b5	b4	b3	b2	b1	Take Action if ADR Bit is Set For
1								Last Online Transaction Not Completed
	1							Go Online On Next Transaction Was Set
		1						Issuer Script Processing Failed
			1					Issuer Authentication Failed
				1				Issuer Authentication Data Not Received in Previous Online Transaction
					1			PIN Try Limit Exceeded
						1		Offline PIN Verification Not Performed
							1	Offline PIN Verification Failed

Table L-20: Card Issuer Action Code, Byte 1

b8	b7	b6	b5	b4	b3	b2	b1	Take Action if ADR Bit is Set For
1								Unable To Go Online
	1							Terminal Erroneously Considers Offline PIN OK
		1						Script Received
			1					Offline Data Authentication Failed on Previous Transaction
				1				Match Found In Additional Check Table 1
					1			No Match Found In Additional Check Table 1
						1		Match Found In Additional Check Table 2
							1	No Match Found In Additional Check Table 2

Table L-21: Card Issuer Action Code, Byte 2

b8	b7	b6	b5	b4	b3	b2	b1	Take Action if ADR Bit is Set For
1								Accumulator 1 Lower Limit Exceeded
	1							Accumulator 2 Lower Limit Exceeded
		1						Counter 1 Lower Limit Exceeded
			1					Counter 2 Lower Limit Exceeded
				1				Counter 3 Lower Limit Exceeded
					1			Additional Accumulator Lower Limit Exceeded
						1		Additional Counter Lower Limit Exceeded
							1	Number of Days Offline Limit Exceeded

Table L-22: Card Issuer Action Code, Byte 3

b8	b7	b6	b5	b4	b3	b2	b1	Take Action if ADR Bit is Set For
1								Accumulator 1 Upper Limit Exceeded
	1							Accumulator 2 Upper Limit Exceeded
		1						Counter 1 Upper Limit Exceeded
			1					Counter 2 Upper Limit Exceeded
				1				Counter 3 Upper Limit Exceeded
					1			Additional Accumulator Upper Limit Exceeded
						1		Additional Counter Upper Limit Exceeded
							1	MTA exceeded

Table L-23: Card Issuer Action Code, Byte 4

b8	b7	b6	b5	b4	b3	b2	b1	Take Action if ADR Bit is Set For
1								Cyclic Accumulator 1 Limit Exceeded
	1							Cyclic Accumulator 2 Limit Exceeded
		1						Additional Cyclic Accumulator Limit Exceeded
			1					Check Failed
				x				RFU
					x			RFU
						x		RFU
							x	RFU

Table L-24: Card Issuer Action Code, Byte 5

b8	b7	b6	b5	b4	b3	b2	b1	Take Action if ADR Bit is Set For
x	x	x	x	x	x	x	x	Reserved for use by issuer

Table L-25: Card Issuer Action Code, Byte 6

Card Risk Management Data Object List 1 (CDOL1)

Tag: '8C'

Length: var.

Format: b

CDOL1 is a list of the tags and lengths of data elements that must be passed to the ICC in the first GENERATE AC command. CDOL1 may be extended with tags and lengths of additional terminal-sourced data to be included in the Transaction Log or to be used with the Additional Check Table, and other data at the discretion of the issuer.

CDOL1 contains the tags and lengths of the data elements listed in Table L-26. Mandatory data elements shall be listed in CDOL1 in the order shown.

Data Element	Tag	Length	Presence
Amount, Authorised	'9F02'	6	Mandatory
Amount, Other	'9F03'	6	Mandatory
Terminal Country Code	'9F1A'	2	Mandatory
TVR	'95'	5	Mandatory
Transaction Currency Code	'5F2A'	2	Mandatory
Transaction Date	'9A'	3	Mandatory
Transaction Type	'9C'	1	Mandatory
Unpredictable Number	'9F37'	4	Mandatory
Terminal Type	'9F35'	1	Mandatory
CVM Results	'9F34'	3	Mandatory
First GENERATE AC Extension Data (includes terminal-sourced data that may be used in the Additional Check Table Card Risk Management Check or appended to the transaction log)	var.	var.	Optional

Table L-26: CDOL1

Card Risk Management Data Object List 2 (CDOL2)

Tag: '8D'
 Length: var.
 Format: b

CDOL2 is a list of the tags and lengths of data elements that must be passed to the ICC in the second GENERATE AC command. CDOL2 may be extended with tags and lengths of additional terminal-sourced data to be included in the Transaction Log, and other data at the discretion of the issuer.

CDOL2 shall contain the tags and lengths of at least the data elements listed in Table L-27. Mandatory data elements shall be listed in CDOL2 in the order shown.

Data Element	Tag	Length	Presence
Issuer Authentication Data	'91'	8 to 16	Mandatory
Authorisation Response Code	'8A'	2	Mandatory
TVR	'95'	5	Mandatory
Unpredictable Number	'9F37'	4	Mandatory
Amount, Authorised	'9F02'	6	Optional
Amount, Other	'9F03'	6	Optional
Second GENERATE AC Extension Data (includes terminal-sourced data that may be appended to the transaction log)	varied	var.	Optional

Table L-27: CDOL2

Note: The personalisation of Amount, Authorised and Amount, Other in CDOL2 must be consistent with personalisation value for the 'Amounts Included in CDOL2' bit in Application Control.

Card Status Update (CSU)

Tag: — Contains data sent to the ICC to indicate whether the issuer approves or declines the transaction, and to initiate actions specified by the issuer. Transmitted to the card in Issuer Authentication Data.

Length: 4

Format: b

For details, see *EMV Book 3*, Common Core Definitions, Table CCD-11.

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
1								Proprietary Authentication Data Included
	x	x	x					RFU
				x	x	x	x	PIN Try Counter

Table L-28: CSU (Card Status Update), Byte 1

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
1								Issuer Approves Online Transaction
	1							Card Block
		1						Application Block
			1					Update PIN Try Counter
				1				Set Go Online on Next Transaction
					1			CSU Generated by Proxy for the Issuer
						x	x	Update Counters
						0	0	Do Not Update Offline Counters
						0	1	Set Offline Counters to Upper Offline Limits
						1	0	Reset Offline Counters to Zero
						1	1	Add Transaction to Offline Counters

Table L-29: CSU (Card Status Update), Byte 2

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

Table L-30: CSU (Card Status Update), Byte 3

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
x	x	x	x	x	x	x	x	Issuer-Discretionary

Table L-31: CSU (Card Status Update), Byte 4

Card Verification Results (CVR)

Tag: '9F52'

Length: 5

Format: b

The Card Verification Results are used to inform the issuer about exception conditions that occurred during the current and previous transactions. It is transmitted to the terminal in Issuer Application Data.

Note: The CVR matches the EMV-specified CVR for CCD, with the exception of bits that are issuer-defined in CCD.

b8	b7	b6	b5	b4	b3	b2	b1	Meaning	
x	x							AC Type Returned in Second GENERATE AC	
0	0							AAC	
0	1							TC	
1	0							Second GENERATE AC Not requested	
1	1							RFU	
		x	x					AC Type Returned in First GENERATE AC	
		0	0					AAC	
		0	1					TC	
		1	0					ARQC	
		1	1					RFU	
				1				CDA performed	
					1				Offline DDA Performed
						1			Issuer Authentication Not Performed
							1	Issuer Authentication Failed	

Table L-32: CVR (Card Verification Results), Byte 1

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
x	x	x	x					Low Order Nibble of PIN Try Counter
				1				Offline PIN Verification Performed
					1			Offline PIN Verification Performed and PIN Not Successfully Verified
						1		PIN Try Limit Exceeded
							1	Last Online Transaction Not Completed

Table L-33: CVR (Card Verification Results), Byte 2

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
1								Lower Offline Transaction Count Limit Exceeded
	1							Upper Offline Transaction Count Limit Exceeded
		1						Lower Cumulative Offline Amount Limit Exceeded
			1					Upper Cumulative Offline Amount Limit Exceeded
				x				Issuer-Discretionary Bit 1
					x			Issuer-Discretionary Bit 2
						1		Check Failed ⁹
							1	Match Found In any Additional Check Table

Table L-34: CVR (Card Verification Results), Byte 3

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
x	x	x	x					Number of Issuer Script Commands Containing Secure Messaging Processed
				1				Issuer Script Processing Failed
					1			Offline Data Authentication Failed on Previous Transaction
						1		Go Online on Next Transaction Was Set
							1	Unable to Go Online

Table L-35: CVR (Card Verification Results), Byte 4

⁹ This bit is used to indicate to the issuer that at least one of many possible error conditions has occurred (for example, data is missing or has the wrong format), preventing the card from successful processing of card risk management checks.

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

Table L-36: CVR (Card Verification Results), Byte 5

Cardholder Name

Tag: '5F20' Indicates cardholder name according to ISO 7813.
 Length: 2-26
 Format: ans 2-26

Cardholder Name Extended

Tag: '5F20' Indicates the whole cardholder name when greater than 26
 Length: 2-26 characters, using the same coding convention as in ISO
 Format: ans 2-26 7813.

Cardholder Verification Method (CVM) List

Tag: '8E' Identifies a prioritised list of methods of verification of the
 Length: var. up to cardholder supported by the application.
 252
 Format: b For details, see *EMV Book 3*, Section 10.5 and Annex C,
 Table 39 and Table 40.

Cardholder Verification Method (CVM) Results

Tag: '9F34' Indicates the results of the last CVM performed.
 Length: 3 For details, see *EMV Book 4*, Annex A, Table 32.
 Format: b

Certificate Authority (CA) Public Key

Tag: — The public key portion of the CA public/private key pair
Length: var. used in offline data authentication (SDA, DDA, and CDA)
Format: b

Certificate Authority (CA) Public Key Index

Tag: '8F' Identifies the certification authority's public key in
Length: 1 conjunction with the RID.
Format: b

Common Core Identifier (CCI)

Tag: — Part of the Issuer Application Data data element. The CCI
Length: 1 identifies the format of the Issuer Application Data and the
Format: b Cryptogram Version used to generate the Application
Cryptogram. Set to the value 'A5' for CCD-compliant
profiles.

Counter Profile Control for Counter x

(x = 1 to 14)

Tag: — The Counter Profile Control y that is used to control
Length: 1 Counter x in the Profile selected for the transaction.
Format: b

Counter Profile Control x

(x = 1 to 14)

Template: 'BF36'

Tag: 'DF0x'

Length: 1

Format: b

Indicates the issuer's choice of data and behaviour to configure a counter within a Profile.

The Counters Profile Controls template may be obtained from the application using the GET DATA command with template tag 'BF36', and may be updated using the PUT DATA command.

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
x	x	x						RFU
			x					Limit Set ID
			0					Use Limit Set 0
			1					Use Limit Set 1
				1				Allow Counting
				0				Do Not Allow Counting ¹⁰
				1				Allow Counting
					1			Reset Counter with online response
						1		Send Counter in IAD
							x	RFU

Table L-37: Counter Profile Control x

¹⁰ This setting allows the Velocity-Checking for Counter x Check to include counters that cannot be incremented in the profile selected for the transaction.

Counter x

(x = 1 to 14)

Template: 'BF35'	Represents a count of transactions.
Tag: 'DF0x'	The value of Counter x may be sent to the issuer as part of the Issuer Application Data.
Length: 1	
Format: b	The value of Counter x may be obtained from the application using the GET DATA command for template tag 'BF35' (Counters Data), if allowed by the Application Control; and may be updated using the PUT DATA command for template tag 'BF35', data element tag 'DF0x'.

The value of Counter x may be obtained from the application as part of the Counters Data template, if allowed by the 'Allow Retrieval of Values and Limits of Accumulators and Counters' bit in the Application Control.

Counter x Control

(x = 1 to 14)

Template: 'BF37'

Tag: 'DF0x'

Length: 1

Format: b

Indicates the issuer's choice of data and behaviour to configure Counter x independent of a Profile.

The Counter x Controls template may be obtained from the application using the GET DATA command with template tag 'BF37', and may be updated using the PUT DATA command.

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
1								Include ARQC Transaction in CRM Test
	1							Include Offline Declines
		1						Include Offline Approvals
			x					Include only if Not Accumulated (transaction is not Accumulated in any non-cyclic Accumulator) ¹¹
			0					include always
			1					include only if Not Accumulated
				x				Include only if International (Terminal Country Code does not match the Issuer Country Code) ¹²
				0				include always
				1				include only if International
					x			RFU
						x		RFU
							x	RFU

Table L-38: Counter x Control¹¹ This bit applies only if the 'Include Offline Approvals' bit is set to 1b.¹² This bit applies only if the 'Include Offline Approvals' bit is set to 1b.

Counter x Data

(x = 1 to 14)

Template: 'BF35' Represents both the value of Counter x and Counter x
Tag: 'DF0x', 'DF1x' Limits.
Length: 3 or 5 The Counters Data template may be obtained from the
Format: b application using the GET DATA command with template tag
'BF35', if allowed by the Application Control; and may be
updated using the PUT DATA command.

The Counters Data template may be obtained from the application, if allowed by the 'Allow Retrieval of Values and Limits of Accumulators and Counters' bit in the Application Control.

Counter x Limits

(x = 1 to 14)

Template: 'BF35' Contains one or two sets of lower and upper limits for
Tag: 'DF1x' Counter x.
Length: 2 or 4 The Counter x Limits may be obtained from the application
Format: b using the GET DATA command with template tag 'BF35'
(Counters Data), if allowed by the Application Control; and
may be updated using the PUT DATA command for template
tag 'BF35', data element tag 'DF1x'.

The value of Counter x Limits may be obtained from the application as part of the Counters Data template, if allowed by the 'Allow Retrieval of Values and Limits of Accumulators and Counters' bit in the Application Control.

Data Element	Length
Counter x Lower Limit 0	1 Byte
Counter x Upper Limit 0	1 Byte
Counter x Lower Limit 1	1 Byte
Counter x Upper Limit 1	1 Byte

Table L-39: Counter x Data

Cryptogram Information Data (CID)

Tag: '9F27' Indicates the type of cryptogram and the actions to be
Length: 1 performed by the terminal.
Format: b

Currency Conversion Table x (x = 1 to 14)

Template: 'BF38' The Currency Conversion Table is used to convert
Tag: 'DF0x' transactions amounts in recognised currencies into an
Length: var. amount in the target currency. The Target Currency must
Format: b match the Accumulator Currency associated with an
 accumulator in order for a converted amount to be
 accumulated.

The Currency Conversion Tables template may be obtained from the application using the GET DATA command with template tag 'BF38', and may be updated in the application using the PUT DATA command.

Data Element	Length
Target Currency Code	2
Currency Conversion Parameters 1 (see Table L-41)	5
...	
Currency Conversion Parameters n	5

Table L-40: Currency Conversion Table x

Position	Data	Length	Value
byte 1-2	Source Currency Code	2	Currency Code of Currency to be converted, coded according to ISO 4217
byte 3-4	Conversion Rate	2	Decimal, BCD coding of multiplication factor
byte 5	Conversion Exponent	1	Binary coding of 10-power (the most significant bit is the sign of the exponent). If the most significant bit is 0 (positive), multiply the conversion rate by 10 raised to the power of the Conversion Exponent (bits b7 through b1). If the most significant bit is 1, divide the conversion rate by 10 raised to the power of the Conversion Exponent (bits b7 through b1).

Table L-41: Currency Conversion Parameters

CVM Results

See **Cardholder Verification Method (CVM) Results**.

Cyclic Accumulator Profile Control for Cyclic Accumulator x

(x = 1 to 14)

Tag: —
Length: 2
Format: b

The Cyclic Accumulator Profile Control y that is used to
control Cyclic Accumulator x in the Profile selected for the
transaction.

Cyclic Accumulator Profile Control x

(x = 1 to 14)

Template: 'BF39'

Tag: 'DF0x'

Length: 2

Format: b

Indicates the issuer's choice of data and behaviour to configure a cyclic accumulator within a Profile.

The Cyclic Accumulators Profile Controls template may be obtained from the application using the GET DATA command with template tag 'BF39', and may be updated using the PUT DATA command.

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
x								Allow Accumulation
0								Do Not Allow Accumulation ¹³
1								Allow Accumulation
	x	x	x	x	x	x	x	RFU

Table L-42: Cyclic Accumulator Profile Control x, Byte 1

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
x	x	x	x					Limit Entry ID
				x	x	x	x	Currency Conversion Table ID
				1	1	1	1	Currency Conversion Not Allowed

Table L-43: Cyclic Accumulator Profile Control x, Byte 2

¹³ Setting this bit to 0b allows the issuer to activate a Cyclic Accumulator in a profile without incrementing it for any transactions using the profile. This enables the application to check whether the limit has been reached for a Cyclic Accumulator that is incremented in another profile (other than the profile for the current transaction), and to send the transaction online or decline offline if the limit has been exceeded.

Cyclic Accumulator x

(x = 1 to 14)

Template: 'BF42' Represents the value of Cyclic Accumulator x, a cumulative amount of transactions performed within a single cycle. The duration of the cycle may be configured to be a day, week, or month.
Tag: 'DF0x'
Length: 6
Format: n 12

Cyclic Accumulator x Control

(x = 1 to 14)

Template: 'BF3A' Indicates the issuer's choice of data and behaviour to configure Cyclic Accumulator x independent of a Profile.
Tag: 'DF0x'
Length: 3 The Cyclic Accumulator x Controls template may be obtained from the application using the GET DATA command with template tag 'BF3A', and may be updated using the PUT DATA command.
Format: b

Position	Data	Length	Value
byte 1-2	Accumulator Currency Code	2	Currency Code for the cyclic accumulator, coded according to ISO 4217
byte 3	Cyclic Accumulator parameters	1	See Table L-45.

Table L-44: Cyclic Accumulator x Control

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
x	x							Cycle Type
0	0							RFU
0	1							Daily Cycle
1	0							Weekly Cycle
1	1							Monthly Cycle
		1						Include online approved transactions
		x						RFU
				x	x	x		First Day of Cycle
				0	0	0		Friday
				0	0	1		Saturday
				0	1	0		Sunday
				0	1	1		Monday
				1	0	0		Tuesday
				1	0	1		Wednesday
				1	1	0		Thursday

Table L-45: Cyclic Accumulator x Control, Byte 3

The First Day of Cycle identifies the day of the week on which each cycle begins for the cyclic accumulator, when the cyclic accumulator has a weekly cycle.

Cyclic Accumulator x Data

(x = 1 to 14)

Template: 'BF42' A logical entity associating a cyclic accumulator value and
Tag: 'DF0x' the two possible formats for the cyclic accumulator reference
 'DF1x' date/day, as shown in Table L-46.
 'DF2x'
Length: 11
Format: varies

The Cycle Type for a cyclic accumulator may vary (between weekly – with a reference day, and daily/monthly – with a reference date); so for each Cyclic Accumulator x implemented, the application needs to support both possible formats for the start-of-cycle reference.

Position	Data	Length	Format
bytes 1-6	Cyclic Accumulator	6	n6
bytes 7-9	Cyclic Accumulator x Reference Date	3	n3
bytes 10-11	Cyclic Accumulator x Reference Day	2	binary

Table L-46: Cyclic Accumulator x Data**Cyclic Accumulator x
Reference Date**

(x = 1 to 14)

Template: 'BF42' For a daily or monthly cyclic accumulator, represents the
Tag: 'DF1x' Transaction Date of the last transaction that reset Cyclic
Length: 3 Accumulator x.
Format: n 6
 (YYMMDD)

**Cyclic Accumulator x
Reference Day**

(x = 1 to 14)

Template: 'BF42'
Tag: 'DF2x'
Length: 2
Format: b

The Reference Day is the Date in days for the day at the beginning of the week in which Cyclic Accumulator x was last reset (see Annex E).

Derivation Key Index (DKI)

Tag: —
Length: 1
Format: b

Part of the Issuer Application Data data element. The DKI identifies for an issuer the Issuer key used to derive the ICC key used to generate the Application Cryptogram.

DF Name

Template: '6F'
Tag: '84'
Length: 5-16
Format: b

Identifies the name of the DF as described in ISO/IEC 7816-4.

Dynamic Data Authentication Data Object List (DDOL)

Tag: '9F49'
Length: up to 252
Format: b

List of data objects (tag and length) to be passed to the ICC in the INTERNAL AUTHENTICATE command.

File Control Information (FCI) Proprietary Template

Template: '6F'
Tag: 'A5'
Length: var.
Format: var.

Issuer-discretionary part of the FCI.

File Control Information (FCI) Template

Tag: '6F' Identifies the FCI template according to ISO/IEC 7816-4.
Length: var. up to
 252
Format: var.

First GEN AC Log Data Table

Template: 'BF40'
 Tag: 'DF01'
 Length: var.
 Format: b

First GEN AC Log Data Table provides information necessary to support the inclusion of additional terminal-sourced data in the transaction log file.

The First GEN AC Log Data Table may be obtained from the application using the GET DATA command for template tag 'BF40'.

First GEN AC Log Data Table is the concatenation (without TLV coding) of the data elements identified in Table L-47.

Data Element	Length	Description
Count of Data Entries, n	1	The number of the following data entries that are logged after ICC-sourced data in the Log Entry.
Data Entry 1	2	Byte 1, binary: The position of the 1st block of additional terminal-sourced data in the First GENERATE AC Command Data that is to be appended to Log Entry. Note: If the position is 0, this block of data is not included in the log data. Byte 2, binary: The length of the 1st block of additional terminal-sourced data in the First GENERATE AC Command Data that is to be appended to Log Entry.
...	var.	
Data Entry n	2	The position and length of the nth block of additional terminal-sourced data in the First GENERATE AC Command Data that is to be appended to Log Entry.

Table L-47: First GEN AC Log Data Table

Note: Each Data Entry refers to a contiguous block of data in the First GENERATE AC Command Data that is to be appended to the Log Entry. It may contain multiple data elements that are listed in CDOL1 in the same order as they are listed in Log Format.

First GEN AC Log Data Table and First GEN AC Unchanging Log Data Table must be consistent with the contents personalised in Log Format and CDOL1.

First GEN AC Unchanging Log Data Table

Template: 'BF40'

Tag: 'DF03'

Length: var.

Format: b

First GEN AC Unchanging Log Data Table provides information necessary to support the inclusion of additional terminal-sourced data in the transaction log file. It contains the information needed by the application to locate transaction log data that is sent only in the first GENERATE AC command data.

The data identified by First GEN AC Unchanging Log Data Table is logged for all transactions that are to be logged, regardless of whether the transaction is logged during processing of the first or second GENERATE AC command. This enables the application not to request (in the second GENERATE AC command data) log data that would not change from the value sent in the first GENERATE AC command data.

The First GEN AC Unchanging Log Data Table may be obtained from the application using the GET DATA command for template tag 'BF40'.

First GEN AC Unchanging Log Data Table is the concatenation (without TLV coding) of the data elements identified in Table L-48.

Data Element	Length	Description
Count of Data Entries, n	1	The number of the following data entries that are logged after ICC-sourced data in the Log Entry.
Data Entry 1	2	<p>Byte 1, binary: The position of the 1st block of additional terminal-sourced data in the first GENERATE AC command data that is to be appended to Log Entry.</p> <p>Note: If the position is 0, this block of data is not included in the log data.</p> <p>Byte 2, binary: The length of the 1st block of additional terminal-sourced data in the first GENERATE AC command data that is to be appended to Log Entry.</p>
...	var.	
Data Entry n	2	The position and length of the nth block of additional terminal-sourced data in the first GENERATE AC command data that is to be appended to Log Entry.

Table L-48: First GEN AC Unchanging Log Data Table

Note: Each Data Entry refers to a contiguous block of data in the GENERATE AC command data that is to be appended to the Log Entry. It may contain multiple data elements that are listed in CDOL1 in the same order as they are listed in Log Format.

First GEN AC Unchanging Log Data Table must be consistent with the contents personalised in Log Format, CDOL1 Log Data Table and CDOL1. CDOL1 Unchanging Log Data Table must also be consistent with the contents personalised in Log Format, Second GEN AC Log Data Table and CDOL2.

For additional information, see Annex D and the Log Format data element in this annex.

First GENERATE AC Command Data Length

Tag: — Length of the First GENERATE AC Command Data for transactions processed using Profile x. The tags and lengths of these data elements were sent to the Terminal in CDOL1.

Length: 1

Format: b

This parameter is part of the Issuer Options Profile Control data element.

The value of First GENERATE AC Command Data Length must be consistent with the contents personalised in CDOL1.

GPO Command Data

Tag: — The value field of the template in PDOL Related Data. The
Length: var. tags and lengths of these data elements were sent to the
Format: b Terminal in the PDOL.

GPO Command Data Length

Tag: — The value expected by the application for the length of the
Length: 1 GPO Command Data field in the GET PROCESSING OPTIONS
Format: b template in PDOL Related Data. The tags and lengths of
 these data elements were sent to the Terminal in the PDOL.

This parameter is part of each GPO Parameters x data
element in the GPO Parameters template.

The value of GPO Command Data Length must be consistent with the contents personalised in the PDOL sent in the response to the SELECT command.

GPO Parameters x

Template: 'BF3E' Contains the following parameters used during processing of
Tag: 'DF0x' the GPO Command.
Length: 2 • GPO Command Data Length (Byte 1)
Format: b The value expected by the application for the length of
 the GPO Command Data (the value field of the template
 in PDOL Related Data). The tags and lengths of these
 data elements were sent to the Terminal in the PDOL.

 • Profile Selection Diversifier (Byte 2)
 An identifier of card data associated with the application
 at the time of Application Selection, used to support
 profile selection based on card data.

If the application does not support the Profile Selection Using Card Data implementer-option, then only GPO Parameters 1 (tag 'DF01') needs to be present in the application.

IAC - Default

Tag: '9F0D'	Specifies the issuer's conditions that cause a transaction to
Length: 5	be declined if it might have been approved online, but the
Format: b	terminal is unable to process the transaction online.

IAC - Denial

Tag: '9F0E'	Specifies the issuer's conditions that cause the decline of a
Length: 5	transaction without attempt to go online.
Format: b	

IAC - Online

Tag: '9F0F'	Specifies the issuer's conditions that cause a transaction to
Length: 5	be transmitted online.
Format: b	

ICC Dynamic Data

Tag: Dynamic application data included in the generation of the
Length: 9 or 38 dynamic signature for the card for DDA or CDA.
Format: b

Length	Data Element
1	ICC Dynamic Number Length
8	ICC Dynamic Number
1	CID
8	Application Cryptogram
20	Transaction Data Hash Code

Table L-49: ICC Dynamic Data for CDA

Length	Data Element
1	ICC Dynamic Number Length
8	ICC Dynamic Number

Table L-50: ICC Dynamic Data for DDA

ICC Dynamic Number

Tag: — Time-variant number generated by the ICC, to be captured
Length: 8 by the terminal.
Format: b

ICC PIN Encipherment Private Key

Tag: — If supported, the ICC private key part of the ICC PIN
Length: N_{PE} Encipherment public/private key pair. Used (in place of the
Format: b ICC Private key) exclusively for deciphering the enciphered
PIN.

ICC PIN Encipherment Public Key Certificate

Tag: '9F2D' ICC PIN Encipherment Public Key certified by the issuer.
Length: N_i
Format: b

ICC PIN Encipherment Public Key Exponent

Tag: '9F2E' ICC PIN Encipherment Public Key Exponent used for PIN
Length: 1 or 3 encipherment when the ICC PIN Encipherment Public Key
Format: b is used.

ICC PIN Encipherment Public Key Remainder

Tag: '9F2F' Remaining digits of the ICC PIN Encipherment Public Key
Length: $N_{PE} - N_i$ Modulus.
 + 42
Format: b

ICC Private Key

Tag: — The ICC private key part of the ICC public/private key pair
Length: N_{IC} used for offline DDA or CDA. It is also used for deciphering
Format: b the offline enciphered PIN if the ICC PIN Encipherment
 Private Key data is not present.

ICC Public Key (PK) Certificate

Tag: '9F46' ICC Public Key certified by the issuer.
Length: N_i
Format: b

ICC Public Key Exponent

Tag: '9F47' Exponent used with the ICC Public Key.
Length: 1 to 3
Format: b

ICC Public Key Remainder

Tag: '9F48' Remaining digits of the ICC Public Key Modulus.
Length: $N_{IC} - N_I$
 + 42
Format: b

ICC Unpredictable Number

Tag: — Time-variant number generated by the ICC, sent in the GET
Length: 8 CHALLENGE response to the terminal.
Format: b

Internal Flags

Tag: — Optional flags that may be implemented for use internal to
Length: var. the application.
Format: b **Note:** This is an implementer-optional data element that could be
 used to implement the requirements for the application.

The list of data elements included in Internal Flags is implementation-specific. For example, the following Internal Flags are used to describe CPA behaviour:

- 'Accumulator x Updated' flag, for all values of x for which an Accumulator x is supported in the application
- 'Counter y Updated' flag, for all values of y for which a Counter y is supported in the application
- 'Cyclic Accumulator z Updated' flag, for all values of z for which a Cyclic Accumulator z is supported in the application
- 'Script Received' flag
- 'Script Failed' flag

Issuer Action Codes (IACs)

See: **IAC - Denial**
IAC - Online
IAC - Default

Issuer Application Data

Tag: '9F10' The Issuer Application Data informs the issuer about the application during online transactions (in the authorisation request) and after transaction completion in the clearing record.
 Length: 32
 Format: b

Position	Content	Values
1	Length Indicator	'0F'
2	CCI	'A5' for CCD-compliant profiles
3	DKI	any
4-8	CVR	any
9-16	Counters	see below
17	Length Indicator	'0F'
18	Profile ID	any
19-32	Issuer-Discretionary	see below

Table L-51: Issuer Application Data

If the 'Send Accumulator Balance' bit in the Accumulator Profile Control is set to 1b for an Accumulator x that is sent in the Issuer Application Data, then the Accumulator x Balance is sent in place of the Accumulator x value.

If the 'Encipher Offline Velocity-Checking Counters' bit in Issuer Options Profile Control is set to the value 1b (for Encipher Offline Velocity-Checking Counters), bytes 9-16 of the Issuer Application Data contains the 8-byte cryptogram that results from enciphering the counts and amounts to be included in bytes 9-16.

Only Counters and Accumulators indicated to be sent in Issuer Application Data are included in bytes 9-16 (as described in sections 15.5.8.1 and 17.5.8.1).

The contents of bytes 19-32 are issuer-discretionary. It is optional to include Counters and Accumulators in these bytes. It is an issuer-option to use the CPA method to include counters and accumulators in these bytes. If the CPA method for filling bytes 19-32 is used, only Counters and Accumulators indicated to be sent in Issuer Application Data that are not sent in bytes 9-16 are included in bytes 19-32 (in the order described in sections 15 and 17).

Any portion of bytes 9-16 and 19-32 in the Issuer Application Data that are not filled by application functionality will be filled with the content of the corresponding bytes in the Default Issuer Application Data personalised for the application.

Using bytes 9-16 and the optional CPA content for bytes 19-32, there are a maximum of three data fields for amount or balance, and a maximum of four data fields for counts in Issuer Application Data.

Issuer Authentication Data

Tag: '91' Data sent from the issuer or its proxy to the ICC for online
Length: var. 8 - 16 Issuer Authentication.
Format: b

The format of Issuer Authentication Data expected by the card is shown in Table L-52.

Bytes	Content
1-4	Authentication Response Cryptogram (ARPC)
5-8	Card Status Update (CSU)
9-16	Proprietary Authentication Data (optional)

Table L-52: Issuer Authentication Data

Issuer Country Code

Tag: '5F28' Indicates the country of the issuer according to ISO 3166.
Length: 2
Format: n 3

Issuer Options Profile Control x (x = 1 to 14)

Template: 'BF3B'
 Tag: 'DF0x'
 Length: Min. 7
 Format: b

Indicates the issuer options that control card risk management and application behaviour within a Profile. Specified in a Profile Control using Issuer Options Profile Control ID. One or more Profile Controls may specify a single Issuer Options Profile Control.

The Issuer Options Profile Controls template may be obtained from the application using the GET DATA command with template tag 'BF3B', and may be updated using the PUT DATA command.

Byte	Contents
1	Issuer Options Profile Parameters (see Table L-54)
2	First GENERATE AC Command Data Length
3	Second GENERATE AC Command Data Length
4	Profile CCI
5	Profile DKI
6	RFU
7	Reserved for use by issuer

Table L-53: Issuer Options Profile Control x

Profile CCI indicates the value to be used for the Common Core Identifier (CCI) in the profile.

Profile DKI indicates the value to be used for the Derivation Key Index (DKI) in the profile.

First GENERATE AC Command Data Length (Byte 2) is the expected length of the data to be included in the First GENERATE AC Command Data. The tags and lengths of these data elements were sent to the Terminal in CDOL1.

The value of First GENERATE AC Command Data Length must be consistent with the contents personalised in CDOL1.

Second GENERATE AC Command Data Length (Byte 3) is the expected length of the data to be returned in the Second GENERATE AC Command Data. The tags and lengths of these data were sent to the terminal in CDOL2.

The value of Second GENERATE AC Command Data Length must be consistent with the contents personalised in CDOL2.

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
x								Log Transactions
0								Do Not Log Transactions
1								Log Transactions
	1							Activate Additional Check Table 1 Check
		1						Activate Additional Check Table 2 Check
			1					Activate Maximum Number of Days Offline Check
				1				Reset Maximum Number of Days offline with online response
					1			Allow Override of CIAC-Default for Transactions at Terminal Type 26 (unattended offline-only POS device)
						1		Encipher Counters portion of IAD
							x	RFU

Table L-54: Issuer Options Profile Parameters

The Issuer Options Profile Control may be extended for support of additional application resources, but the format for additional bytes is beyond the scope of this specification.

Issuer Public Key (PK) Certificate

Tag: '90'

Issuer public key certified by a certification authority.

Length: N_{CA}

Format: b

Issuer Public Key Exponent

Tag: '9F32'

Issuer public key exponent used for the verification of the Signed Static Application Data and the ICC Public Key Certificate.

Length: 1 or 3

Format: b

Issuer Public Key Remainder

Tag: '92' Remaining digits of the Issuer Public Key Modulus.
 Length: $N_I - N_{CA} + 36$
 Format: b

Issuer Script Command Counter

Tag: — Indicates the number of script commands processed previously. The right nibble is included in the Card Verification Results sent to the Issuer in Issuer Application Data.
 Length: 1
 Format: b

Table L-55 describes the coding for the Script Counter.

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
x	x	x	x					RFU
				x	x	x	x	Script Counter

Table L-55: Issuer Script Counter Coding

Only the right nibble of the Script Counter is used. The number of script commands is not limited to 15. The Script Counter is cyclic: '0F' + 1 = '00'.

The Script Counter is updated for each script command successfully processed, for example:

- PUT DATA
- UPDATE RECORD
- PIN CHANGE/UNBLOCK
- APPLICATION UNBLOCK

Issuer Script Results

Tag: — Indicates the result of the terminal script processing.
 Length: var. For details, see *EMV Book 4*, Annex A, Table 33.
 Format: b

Last Online Transaction Date in days

Tag: —	The Transaction Date in days of the last transaction that successfully went online.
Length: 2	
Format: b	The difference between the current Transaction Date and the Last Online Transaction Date is compared to the Number of Days Offline Limit, triggering issuer-specified actions if the limit is exceeded.

Limits Entry x (x = 1 to 14)

Template: 'BF3C'	An issuer's choice of limits that may be used with the Maximum Transaction Amount and Cyclic Accumulators.
Tag: 'DF0x'	
Length: 6 – 84 (in multiples of 6)	Each limit is 6 bytes long.
Format: n 12	The Limits Entries template may be obtained from the application using the GET DATA command with template tag 'BF3C', and may be updated using the PUT DATA command.

The Limit Entry used for the Maximum Transaction Amount is identified in the MTA Profile Control for the transaction. The Limit Entry used for a Cyclic Accumulator is identified in the Cyclic Accumulator Profile Control for the Cyclic Accumulator. If the Limit Entry ID in the MTA Profile Control or Cyclic Accumulator Profile Control = y, then Limit Entry y will be used.

Log Entry

Tag: '9F4D'

Length: 2.

Format: b

Devices that read the transaction log use the Log Entry data element to determine the location (SFI) and the maximum number of transaction log records.

A terminal will obtain this value from the SELECT response in the FCI. This data element is also personalized outside the SELECT response, so that the application knows where to log transactions and the maximum number of records supported in the transaction log file.

Byte	Format	Length	Value
1	b	1	bits b8-b4: SFI containing the cyclic transaction log file bit b3-b0: RFU
2	b	1	Maximum number of records in the transaction log file

Table L-56: Log Entry

Log Format

Tag: '9F4F' List in tag and length format of data elements that are
 Length: var. logged by the transaction.
 Format: b This value may be obtained from the application using the
 GET DATA command.

Log Format contains the tags and lengths of the data elements listed in Table L-57.

Data Element	Tag	Length	Source	Condition
Amount, Authorised	'9F02'	6	Terminal (First or Second GENERATE AC Command Data)	Mandatory
Transaction Currency Code	'5F2A'	2	Terminal (First GENERATE AC Command Data)	Mandatory
Transaction Date	'9A'	3	Terminal (First GENERATE AC Command Data)	Mandatory
CVR	'9F52'	5	Application	If the 'Log the CVR' bit in Application Control is set to 1
ATC	'9F36'	2	Application	If the 'Log the ATC' bit in Application Control is set to 1
CID	'9F27'	1	Application	If the 'Log the CID' bit in Application Control is set to 1

Table L-57: Log Format
(continues)

Data Element	Tag	Length	Source	Condition
Issuer-defined list of data elements (first list)	Var.	Var.	Terminal (First GENERATE AC Command Data)	Optional Data Extracted from First GENERATE AC Command Data using “First GEN AC Unchanging Log Data Table.” Note: When the first GENERATE AC command returns an ARQC cryptogram, the card saves these values to allow their availability during the second GENERATE AC command processing.
Issuer-defined list of data elements (second list)	Var.	Var.	Terminal (First GENERATE AC Command Data and Second GENERATE AC Command Data)	Optional. When the record is logged during the first GENERATE AC command, Data Extracted from First GENERATE AC Command Data using “First GEN AC Log Data Table”. When the record is logged during the second GENERATE AC command, Data Extracted from Second GENERATE AC Command Data using “Second GEN AC Log Data Table.”

Table L-57: Log Format, continued

Master Key for AC

Tag: — Master Key used for Application Cryptogram Generation.
Length: 16
Format: b

Master Key for SMC

Tag: — Master Key used for Secure Messaging for Confidentiality.
Length: 16
Format: b

Master Key for SMI

Tag: — Master Key used for Secure Messaging for Integrity.
Length: 16
Format: b

Maximum Transaction Amount (MTA) Profile Control x

(x = 1 to 14)

Template: 'BF3D' Indicates the issuer's choice of data to configure the
 Tag: 'DF0x' Maximum Transaction Amount Check within a Profile.
 Length: 4 The MTA Profile Controls template may be obtained from
 Format: b the application using the GET DATA command with template
 tag 'BF3D', and may be updated using the PUT DATA
 command.

Position	Data	Length	Value
byte 1-2	MTA Currency Code	2	Currency Code for MTA, coded according to ISO 4217
byte 3-4	MTA parameters	2	See Table L-59 and Table L-60.

Table L-58: Maximum Transaction Amount Profile Control x, Bytes

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
x	x	x	x					Limits Entry ID
				x	x	x	x	Currency Conversion Table ID
				1	1	1	1	Currency Conversion Not Allowed

Table L-59: Maximum Transaction Amount Profile Control x, Byte 3

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

Table L-60: Maximum Transaction Amount Profile Control x, Byte 4

Number of Days Offline Limit

Tag: 'C3'
Length: 2
Format: n 4

The limit associated with the number of days since the application has successfully sent a transaction online. The Number of Days Offline is measured from the date of the previous transaction that went online and the terminal did not indicate it was unable to go online.

The difference between the current Transaction Date and the Last Online Transaction Date is compared to the Number of Days Offline Limit, triggering issuer-specified actions if the limit is exceeded.

The Number of Days Offline Limit may be obtained from the application using the GET DATA command, and may be updated in the application using the PUT DATA command.

Offline Balance Currency Code

Tag: 'C9'
Length: 2
Format: n 3

The Currency Code in which Accumulator 1 is managed (used with Offline Balance).

Offline Balance Currency Code may be obtained from the application using the GET DATA command.

Offline Balance

Tag: '9F50'	Represents the amount of offline spending available for the application.
Length: 6	
Format: 12 n	The value of Offline Balance may be obtained from the application using the GET DATA command, if allowed by the Application Control.

Offline Balance is retrievable by the GET DATA command, if allowed by the 'Allow Retrieval of Values and Limits of Accumulators and Counters' bit in the Application Control.

Offline Balance is computed as follows:

Offline Balance = Accumulator 1 Upper Limit 0 minus Accumulator 1 value.

If Accumulator 1 Upper Limit 0 < Accumulator 1 value, then the value for Offline Balance is zero ('00 00 00 00 00 00').

Note: The Offline Balance is calculated using the Accumulator 1 Upper Limit 0, regardless of which Upper Limit is used for Accumulator 1 in the profile.

PDOL

Tag: '9F38'	Contains a list of terminal resident data objects (tags and lengths) needed by the ICC in processing the GET PROCESSING OPTIONS command.
Length: var.	
Format: see EMV Book 3, section 5.4	

PDOL Related Data

Tag: —	The template sent in the GET PROCESSING OPTIONS command. PDOL Related Data begins with template tag '83' and the template length, so the minimum length for PDOL Related Data is 2 bytes. The value field of the template, if present, is the GPO Command Data.
Length: var.	
Format: b	

PIN Decipherments Error Counter

Tag: —	The PIN Decipherments Error Counter counts the number of unsuccessful offline PIN decipherments in the application lifetime.
Length: 2	
Format: b	Note: This is an optional counter that could be used to implement security requirements in the application instead of in the platform. See Annex F. Requires 'Dynamic RSA' implementer option.

PIN Decipherments Error Counter Limit

Tag: —	The PIN Decipherments Error Counter Limit limits the number of unsuccessful offline PIN decipherments in the application lifetime.
Length: 2	
Format: b	Note: This is an optional limit that could be used to implement security requirements in the application instead of in the platform. See Annex F. Requires 'Dynamic RSA' implementer option.

PIN Try Counter

Tag: '9F17'	Number of PIN tries remaining.
Length: 1	The value of the PIN Try Counter may be obtained from the application using the GET DATA command.
Format: b	

PIN Try Limit

Tag ¹⁴ : 'C6'	The maximum number of consecutive incorrect PIN entries allowed.
Length: 1	
Format: b	

¹⁴ This tag is required for this data element only when the EMV CPS implementer-option is supported. If EMV CPS is not supported, this specification does not require that the data element be tagged, and does not require this value for the tag associated with this data element.

Previous Transaction History (PTH)

Tag¹⁵: 'C7'

Length: 2

Format: b

Used to store in non-volatile memory information about the previous transactions. Used in Card Risk Management.

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
1								Offline Data Authentication Failed on Previous Transaction
	1							Last Online Transaction Not Completed
		x						RFU
			1					Application Blocked
				1				Go Online On Next Transaction ¹⁶
					1			Issuer Authentication Failed (that is, was performed and did not pass)
						1		Script Received
							1	Script Failed

Table L-61: Previous Transaction History, Byte 1

¹⁵ This tag is required for this data element only when the EMV CPS implementer-option is supported. If EMV CPS is not supported, this specification does not require that the data element be tagged, and does not require this value for the tag associated with this data element.

¹⁶ The value 0b in this bit means do not force transaction online. It does not prevent a transaction from going online if card risk management or the terminal request results in a decision to attempt to go online.

If an issuer wants new cards to attempt to go online, the issuer may personalise this bit in the PTH to the value 1b. Personalisation of the PTH to set this bit is an issuer-option.

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
1								Issuer Authentication Data Not Received in Online Response
	1							Unable to Go Online
		x						RFU
			x					RFU
				x				RFU
					x			RFU
						x		RFU
							x	RFU

Table L-62: Previous Transaction History, Byte 2

Processing Options Data Object List (PDOL)

See PDOL.

Profile Control x

(x = '01' to '7E')

Template: 'BF3F'

Tag: 'DFxx'

Length: min. 8

Format: b

A list of resource IDs and resource control IDs that identify the Profile-specific data and behaviour used by the application when processing a transaction using Profile ID x.

The Profile Controls template may be obtained from the application using the GET DATA command with template tag 'BF3F', and may be updated using the PUT DATA command.

Byte	b8-b5	b4-b1
1	Issuer Options Profile Control ID	AIP/AFL ID
2	CIACs ID	Accumulator Profile Control ID for Accumulator 1
3	Accumulator Profile Control ID for Accumulator 2	Counter Profile Control ID for Counter 1
4	Counter Profile Control ID for Counter 2	Counter Profile Control ID for Counter 3
5	Cyclic Accumulator Profile Control ID for Cyclic Accumulator 1	Cyclic Accumulator Profile Control ID for Cyclic Accumulator 2
6	MTA (Maximum Transaction Amount) Profile Control ID	VLP Profile Control ID ¹⁷
7	RFU	RFU
8	RFU	RFU

Table L-63: Profile Control

The Profile Control may be extended beyond 8 bytes for support of additional application resources, but the format for the additional bytes is beyond the scope of this specification.

¹⁷ If VLP is not supported in the application, this field is set to 'F' to indicate it is not used.

Profile ID

Tag: —
Length: 1
Format: b

Identifies the Profile to be used in processing the transaction. The Profile ID is the output of the Profile Selection algorithm, and has a value in the range '01' to '7F'. The value '7F' is reserved to indicate that the GET PROCESSING OPTIONS command is to be rejected by the application.

The values '70' to '7C' are RFU for EMVCo.

The following Profile IDs are pre-assigned in this specification:

- default Profile ('01')
- VLP Profile ('7D')
- Authentication Token Profile ('7E')
- reject GPO Command ('7F')

Profile Selection Diversifier

Tag: — An identifier of card data that was associated with the application at the time of Application Selection, used to support profile selection based on card data.

Length: 1

Format: b

This parameter is part of each GPO Parameters x data element in the GPO Parameters template.

Profile Selection Entries

Tag: — The Profile Selection File is a file which contains records called Profile Selection Entries. Each Profile Selection Entry contains values that are compared to the contents of the GET PROCESSING OPTIONS command data. Use of the Profile Selection Entries is described in Annex A.

Length: var.

Format: b

The Profile Selection File Processing is performed only if the 'Activate Profile Selection File' bit in Application Control is set to 1b.

At a minimum, the application shall be able to support up to 15 Profile Selection Entries.

Each Profile Selection Entry x is the concatenation of the data elements identified in Table L-64.

Data Element	Length	Format	Description	
Length	1	b	Profile Selection Entry x length	
Position in GPO Command Data	1	b	<p>The starting position (in bytes) of the portion of the data extracted from the GET PROCESSING OPTIONS (GPO) command data that is compared to the Comparison Value(s) in this Profile Selection Entry.</p> <p>If the first byte in the GPO command data is extracted, the value of Position in GPO Command Data is '01'.</p>	
Comparison Block Length	1	b	The length in bytes of the data extracted from the GET PROCESSING OPTIONS command data that is compared to the Comparison Value(s).	
Number Of Comparison Blocks (n)	1	b	The number of Comparison Blocks in the Profile Selection Entry. The first Comparison Block is a Bit Mask. The second and any subsequent Comparison Blocks are Comparison Values that are used for the comparison.	
Comparison Blocks	Var.	b	Contains the concatenation of the Bit Mask and one or more Comparison Values. The first entry is used as a Bit Mask.	
	Comparison Block Length	b	Bit Mask	Used to mask the comparison data to allow the comparison with the Comparison Value to use only selected portions of the extracted data.
	Comparison Block Length	b	Comparison Value	A value compared to the masked data extracted from the GPO command data.

Table L-64: Profile Selection Entry x
(continues)

Data Element	Length	Format	Description
Check Type	1	b	The type of comparison to make with the personalised value: '00' = match '01' = less than '02' = greater than
Positive Action	1	b	Action to take when the check is positive (TRUE): <ul style="list-style-type: none"> selected Profile ID move down a specified number of Profile Selection Entries in the file discontinue processing the GPO command and respond with SW1 SW2 = '6985'
Negative Action	1	b	Action to take when the check is negative (FALSE): <ul style="list-style-type: none"> selected Profile ID move down a specified number of Profile Selection Entries in the file discontinue processing the GPO command and respond with SW1 SW2 = '6985'

Table L-64: Profile Selection Entry x, continued

Profile Selection File Entry

Tag: 'C2'
Length: 2.
Format: b

Devices that read the Profile Selection File from the card use the Profile Selection File Entry to determine the location (SFI) and the number of records (that is, the number of Profile Selection Entries) in the table.

This value may be obtained from the application using the GET DATA, and may be updated using the PUT DATA command.

Byte	Format	Length	Value
1	b	1	SFI containing Profile Selection File
2	b	1	Number of Profile Selection Entries in the Profile Selection File

Table L-65: Profile Selection File Entry

Reference PIN

Tag: —
Length: var. 2 to 6
Format: n 4 to 12 digits

The value of the offline PIN that is compared to a PIN entered by the cardholder.

Registered Application Identifier (RID)

Tag: —
Length: 5
Format: b

The first 5 bytes of the AID, unique to an application provider and assigned according to ISO/IEC 7816-5.

SDA Tag List

Tag: '9F4A'

Length: var.

Format: —

List of tags of primitive data objects defined in the EMV specification whose value fields are to be included in the Signed Static Application Data or ICC Public Key Certificate or ICC PIN Encipherment Public Key Certificate.

If supported, the SDA Tag List contains only the tag of the Application Interchange Profile.

Second GEN AC Log Data Table

Template: 'BF40'
 Tag: 'DF02'
 Length: var.
 Format: b

Second GEN AC Log Data Table provides information necessary to support the inclusion of additional terminal-sourced data in the transaction log file.

The Second GEN AC Log Data Table may be obtained from the application using the GET DATA command for template tag 'BF40'.

Second GEN AC Log Data Table is the concatenation (without TLV coding) of the data elements identified in Table L-66.

Data Element	Length	Description
Count of Data Entries, n	1	The number of the following data entries that are logged after ICC-sourced data in the Log Entry.
Data Entry 1	2	Byte 1, binary: The position of the 1st block of additional terminal-sourced data in the Second GENERATE AC Command Data that is to be appended to Log Entry. Note: If the position is 0, this block of data is not included in the log data. Byte 2, binary: The length of the 1st block of additional terminal-sourced data in the Second GENERATE AC Command Data that is to be appended to Log Entry.
...	var.	
Data Entry n	2	The position and length of the nth block of additional terminal-sourced data in the Second GENERATE AC Command Data that is to be appended to Log Entry.

Table L-66: Second GEN AC Log Data Table

Note: Each Data Entry refers to a contiguous block of data in the GENERATE AC command data that is to be appended to the Log Entry. It may contain multiple data elements that are listed in CDOL2 in the same order as they are listed in Log Format.

Second GEN AC Log Data Table and CDOL1 Unchanging Log Data Table must be consistent with the contents personalised in Log Format, CDOL1 and CDOL2.

Second GENERATE AC Command Data Length

Tag: — Length of the Second GENERATE AC Command Data for transactions processed using Profile x. The tags and lengths of these data elements were sent to the Terminal in CDOL2.
 Length: 1
 Format: b

This parameter is part of the Issuer Options Profile Control data element.

The value of Second GENERATE AC Command Data Length must be consistent with the contents personalised in CDOL2.

Security Limits

Tag: 'C5'
 Length: 6
 Format: b

Concatenation of limits used for internal security-related counters if the associated security-related counters are implemented in CPA (see section 20 and annex F). The associated security counters can be used to prevent further use of the keys.

If supported in the application, the Security Limits are not retrievable from the card and may be updated in the application using the PUT DATA command.

Note: This is an implementer-optional data element that could be used to implement security requirements in the application instead of in the platform. See Annex F.

Security Limits is the concatenation of the values in Table L-67.

For DDA or CDA applications	For SDA-only applications	Length
Element	Element	
AC Session Key Counter Limit	AC Session Key Counter Limit	2
SMI Session Key Counter Limit	SMI Session Key Counter Limit	2
PIN Decipherments Error Counter Limit	'0000'	2

Table L-67: Security Limits

Security Limits Status

Tag: 'C4'

Length: 1

Format: b

Indicates whether the limit for a security counter that limits the number of times a key is used has been reached, if the associated security-related counters are implemented in CPA (see section 20).

If supported in the application, this data element may be retrieved using the GET DATA command

Security Limits Status is the concatenation of the values in Table L-68.

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
1								AC Session Key Counter Limit Exceeded
	1							SMI Session Key Counter Limit Exceeded
		1						PIN Decipherments Error Counter Limit Exceeded
			x					RFU
				x				RFU
					x			RFU
						x		Issuer Proprietary
							x	Issuer Proprietary

Table L-68: Security Limits Status

Service Code

Tag: '5F30'

Length: 2

Format: n 3

Service code as defined in ISO/IEC 7813 for track 1 and track 2.

Session Key for AC

Tag: — Session Key used for Application Cryptogram Generation.
Length: 16
Format: b

Session Key for SMC

Tag: — Session Key used for Secure Messaging for Confidentiality.
Length: 16
Format: b

Session Key for SMI

Tag: — Session Key used for Secure Messaging for Integrity.
Length: 16
Format: b

Short File Identifier (SFI)

Tag: '88' Identifies the file to be used in the commands related to a
Length: 1 given AEF or DDF. The SFI data object is a binary field
Format: b with the three high order bits set to zero.

Signed Dynamic Application Data

Tag: '9F4B' Digital signature on critical application parameters for DDA
Length: N_{IC} or CDA.
Format: b

Signed Static Application Data (SSAD)

Tag: '93' Digital signature on critical application parameters for SDA.
Length: N_1
Format: b

SMI Session Key Counter

Tag: — The SMI Session Key Counter counts the number of Secure Messaging for Integrity session key derivations that are not followed by successful validation of a Secure Messaging MAC, over the lifetime of the card, and is used to limit this number to the SMI Session Key Counter Limit.

Note: This is an implementer-optional counter that could be used to implement security requirements in the application instead of in the platform. See Annex F.

SMI Session Key Counter Limit

Tag: — The SMI Session Key Counter Limit limits the number of Secure Messaging for Integrity session key derivations that are not followed by successful validation of a Secure Messaging MAC, over the lifetime of the card.

Note: This is an implementer-optional limit that could be used to implement security requirements in the application instead of in the platform. See Annex F.

Source Currency Code

Tag: —
Length: 2
Format: n 3

Indicates the currency from which a Currency Conversion parameter converts amounts into the Target Currency for the Currency Conversion Table. This parameter is part of each Currency Conversion Parameter in a Currency Conversion Table x data element.

A Currency Conversion Parameter can be used only to convert a transaction amount if the Transaction Currency Code matches the Source Currency Code of the Currency Conversion Parameter. Coded according to ISO 4217.

Static Data Authentication Tag List

See **SDA Tag List**,

Target Currency Code

Tag: —
Length: 2
Format: n 3

Indicates the currency to which a Currency Conversion Table converts amounts in other currencies. This parameter is part of the Currency Conversion Table x data element.

A Currency Conversion Table can be used only to convert currency if the Accumulator Currency Code matches the Target Currency Code of the Currency Conversion Table. Coded according to ISO 4217.

Terminal Action Code – Default

Tag: — Specifies the acquirer's conditions that cause a transaction to be declined if it might have been approved online, but the terminal is unable to process the transaction online.
Length: 5
Format: b

Terminal Action Code – Denial

Tag: — Specifies the acquirer's conditions that cause the decline of a transaction without attempt to go online.
Length: 5
Format: b

Terminal Action Code – Online

Tag: — Specifies the acquirer's conditions that cause a transaction to be transmitted online.
Length: 5
Format: b

Terminal Action Codes (TACs)

See: **Terminal Action Code – Denial**
Terminal Action Code – Online
Terminal Action Code – Default

Terminal Capabilities

Tag: '9F33' Indicates the card data input, CVM, and security capabilities of the terminal.
Length: 3
Format: b For details, see *EMV Book 4*, Annex A, Table 24 through Table 26.

Terminal Country Code

Tag: '9F1A'	Indicates the country of the terminal, represented according to ISO 3166.
Length: 2	
Format: n 3	

Terminal Type

Tag: '9F35'	Indicates the environment of the terminal, its communications capability, and its operational control.
Length: 1	
Format: n 2	For details, see <i>EMV Book 4</i> , Annex A, Table 23.

Terminal Verification Results (TVR)

Tag: '95'	Status of the different functions as seen from the terminal.
Length: 5	For details, see <i>EMV Book 3</i> , Annex C, Table 42.
Format: b	

Track 1 Discretionary Data

Tag: '9F1F'	Discretionary part of track 1 according to ISO/IEC 7813.
Length: var.	
Format: ans	

Track 2 Equivalent Data

Tag: '57'	Contains the data elements of track 2 according to ISO/IEC 7813.
Length: var. up to 19	
Format: b	For details, see <i>EMV Book 3</i> , Annex A, Table 33 (Data Elements Dictionary).

Transaction Currency Code

Tag: '5F2A'	Indicates the currency code of the transaction according to
Length: 2	ISO 4217.
Format: n 3	

Transaction Date

Tag: '9A'	Local date that the transaction was authorised.
Length: 3	
Format: n 6	
	(YYMMDD)

Transaction Date in Days

Tag: —	Local date that the transaction was authorised, represented
Length: 2	as a count of days since a base date (see Annex E).
Format: b	

Transaction PIN

Tag: —	Data entered by the cardholder for the purpose of the PIN
Length: var. 2 to 6	verification. For offline PIN Verification, the data is
Format: b	compared with the Reference PIN.

Transaction Type

Tag: '9C'	Indicates the type of financial transaction, represented by
Length: 1	the first two digits of ISO 8583:1987 Processing Code. The
Format: n 2	actual values to be used are defined by the relevant
	payment systems.

Unpredictable Number

Tag: '9F37'	Value to provide variability and uniqueness to the
Length: 4	generation of a cryptogram.
Format: b	

VLP Available Funds

Tag: '9F79'	An accumulator that is decremented by the transaction
Length: 6	amount when a VLP transaction is approved.
Format: n 12	

VLP Funds Limit

Tag: '9F77'	The issuer limit for VLP Available Funds that is used by the
Length: 6	card to reset VLP Available Funds after an online approved
Format: n 12	transaction.
	If supported by the application, the VLP Funds Limit may
	be obtained from the application using the GET DATA
	command, and may be updated using the PUT DATA
	command.

VLP Issuer Authorisation Code

Tag: '9F74'	A code on the card that indicates that the transaction is
Length: 6	approved; may be sent in the clearing message if the
Format: a	transaction is approved offline. The format is defined by the
	payment system.

VLP Profile Control

Tag: — If VLP is supported in the application, indicates the issuer's choice of data and behaviour to configure the VLP accumulator within a Profile.
 Length: 2
 Format: b

Note: The application uses an Accumulator Profile Control x resource for the VLP Profile Control, but only bits b7 and b6 in byte 1 are active. All other functionality is specified for the VLP Available Funds.

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
x								RFU
	1							Reset Accumulator with online response
		1						Send Accumulator in IAD
			x	x	x	x	x	RFU

Table L-69: VLP Profile Control, Byte 1

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

Table L-70: VLP Profile Control, Byte 2

VLP Single Transaction Limit

Tag: '9F78' The maximum amount allowed for a single VLP transaction.
 Length: 6 If supported by the application, the VLP Single Transaction Limit may be obtained from the application using the GET DATA command, and may be updated using the PUT DATA command.
 Format: n 12

VLP Terminal Support Indicator

Tag: '9F7A'	Indicates that the terminal supports VLP processing
Length: 1	0 = VLP not supported
Format: n 1	1 = VLP supported