



## Cash register - terminal protocol

Version 2.0.12  
2023.05.12

## © Copyright EVO Payments Inc., 2023

All information contained in this documentation, as well as the software described in it, is confidential and proprietary to EVO Payments Inc., or one of its subsidiaries. This documentation, is subject to a license agreement, and may be used or copied only in accordance with the terms of such license. Except as permitted by such license, no part of this documentation may be reproduced, stored in a retrieval system, or transmitted in any form or by electronic, mechanical, recording, or any other means, without the prior written permission of EVO Payments Inc., or one of its subsidiaries.

# Contents

<b>1</b>	<b>Introduction</b>	<b>8</b>
<b>2</b>	<b>Basic information</b>	<b>9</b>
2.1	Control marks . . . . .	9
2.2	Request message format . . . . .	9
2.3	Response message format . . . . .	10
2.4	Function return codes in response field . . . . .	11
2.5	Data formats . . . . .	11
2.6	List of available commands . . . . .	12
2.7	General remarks . . . . .	12
2.8	Communication logs (MANDATORY) . . . . .	13
<b>3</b>	<b>Terminal related commands</b>	<b>14</b>
3.1	Terminal status read . . . . .	14
3.2	Terminal general information read . . . . .	15
<b>4</b>	<b>Transaction processing related commands</b>	<b>17</b>
4.1	Start transaction . . . . .	17
4.1.1	Start purchase transaction . . . . .	19
4.1.2	Start preauthorization transaction . . . . .	19
4.1.3	Start completion transaction . . . . .	20
4.1.4	Start refund transaction . . . . .	20
4.1.5	Start reversal transaction . . . . .	21
4.1.6	Start giftcard's activation . . . . .	21
4.1.7	Start giftcard's balance check . . . . .	21
4.1.8	Start giftcard's topup . . . . .	22
4.1.9	Start reconciliation . . . . .	22
4.1.10	Start connection test . . . . .	22
4.1.11	Start connection to TMS . . . . .	23
4.1.12	Start function execution . . . . .	23
4.1.13	Start incremental authorization . . . . .	24
4.1.14	Start preauthorization cancel . . . . .	25
4.2	Transaction End / Transaction Data Read . . . . .	25
4.3	Transaction Data Input . . . . .	29
4.4	Last Transaction Data . . . . .	29
4.5	Transaction type . . . . .	30
4.6	Transaction status . . . . .	30
4.7	Transaction result code . . . . .	35
4.8	Transaction start tags . . . . .	36
4.9	Transaction end tags . . . . .	41
<b>5</b>	<b>Transaction processing</b>	<b>48</b>
5.1	General information . . . . .	48
5.2	Cashback transaction . . . . .	48
5.3	DCC transaction . . . . .	48

<b>6</b>	<b>Example message flows</b>	<b>50</b>
6.1	DCC transaction with PIN . . . . .	50
6.2	Cashback transaction with PIN . . . . .	52
6.3	Purchase transaction with PIN . . . . .	54
6.4	Purchase transaction with PIN, with transaction accepted status (12) enabled . . . .	55
6.5	Purchase transaction with PIN and signature, with multilanguage printout (UPI) . .	57
6.6	Purchase transaction referral . . . . .	58
6.7	Purchase transaction with invalid signature . . . . .	60
6.8	Purchase transaction with Erzsebet Benefit Program card . . . . .	61
6.9	Purchase transaction with ZenCard . . . . .	63
6.10	Purchase transaction with Fleet Card . . . . .	66
6.11	Purchase transaction with BLIK code given . . . . .	67
6.12	Purchase transaction with BLIK code given and PIN . . . . .	68
6.13	MultiSplit Payment Transaction . . . . .	69
6.14	Partial DCC refund . . . . .	71
6.15	DCC refund with waiting for amount status . . . . .	73
6.16	Reversal transaction . . . . .	75
6.17	Reversal transaction (extended transaction type) . . . . .	76
6.18	Giftcard's activation (accepted) . . . . .	77
6.19	Giftcard's activation (rejected) . . . . .	78
6.20	Giftcard's balance check . . . . .	78
6.21	Giftcard's topup . . . . .	79
6.22	Execute function - Zencard flow . . . . .	80
6.23	Incremental authorization . . . . .	81
6.24	Preauthorization cancel . . . . .	82
<b>7</b>	<b>Reconciliation handling related commands</b>	<b>84</b>
7.1	Single transaction read from batch . . . . .	84
7.2	Batch summary read . . . . .	85
7.2.1	Payment dayclose information . . . . .	88
7.2.2	Giftcard module dayclose information . . . . .	88
7.2.3	Additional data - file with batch transaction list . . . . .	90
<b>8</b>	<b>Transactions reports related commands</b>	<b>91</b>
8.1	Transaction Report . . . . .	91
8.1.1	Transaction Report - Generate . . . . .	91
8.1.2	Transaction Report - Get Summary of Cardsets Transactions . . . . .	93
8.1.3	Transaction Report - Get Total Summary . . . . .	96
8.2	Transaction Analysis . . . . .	98
8.2.1	Transaction Analysis - Generate . . . . .	99
8.2.2	Transaction Analysis - Get Transaction Details . . . . .	101
8.2.3	Transaction Analysis - Get Total Summary . . . . .	101
8.3	Transaction DCC Analysis . . . . .	103
8.3.1	Transaction DCC Analysis - Generate . . . . .	104
8.3.2	Transaction DCC Analysis - Get Currency Summary . . . . .	105
8.3.3	Transaction DCC Analysis - Get Total Summary . . . . .	106
8.4	Report summary entry id . . . . .	106
<b>9</b>	<b>Miscellaneous functions</b>	<b>108</b>
9.1	Set Parameters . . . . .	108
9.2	Set Standby . . . . .	108

## List of Tables

1	Function return codes. . . . .	11
2	Data formats. . . . .	11
3	List of available commands. . . . .	12
4	Terminal status read request description. . . . .	14
5	Terminal status read response description. . . . .	14
6	Expected data characteristics description. . . . .	15
7	Terminal general information read request description. . . . .	15
8	Terminal general information read response description. . . . .	16
9	Terminal general information description. . . . .	16
10	General request for start transaction command. . . . .	18
11	General response for start transaction command. . . . .	18
12	Start purchase transaction request description. . . . .	19
13	Start preauthorization transaction request description. . . . .	19
14	Start completion transaction request description. . . . .	20
15	Start refund transaction request description. . . . .	20
16	Start reversal transaction request description. . . . .	21
17	Start giftcard's activation request description. . . . .	21
18	Start giftcard's balance check request description. . . . .	22
19	Start giftcard's topup request description. . . . .	22
20	Start reconciliation request description. . . . .	22
21	Start connection test request description. . . . .	23
22	Start connection to the TMS request description. . . . .	23
23	Start function execution request description. . . . .	24
24	Start incremental authorization request description. . . . .	24
25	Start preauthorization cancel request description. . . . .	25
26	Transaction End / Transaction Data Read request description. . . . .	25
27	Transaction End / Transaction Data Read response description. . . . .	25
28	Transaction details description. . . . .	26
29	Transaction Data Input request description. . . . .	29
30	Transaction Data Input response description. . . . .	29
31	Last Transaction Data request description. . . . .	29
32	Transaction type. . . . .	30
33	Transaction status/Terminal state. . . . .	30
34	Transaction result code. . . . .	35
35	Transaction start tags. . . . .	36
36	Transaction end tags. . . . .	41
37	DCC transaction with PIN. . . . .	50
38	Cashback transaction with PIN. . . . .	52
39	Purchase transaction with PIN. . . . .	54
40	Purchase transaction with PIN, with transaction accepted status (12) enabled. . . . .	55
41	Purchase transaction with PIN and signature, with multilanguage printout (UPI). . . . .	57
42	Purchase transaction referral. . . . .	58
43	Purchase transaction with invalid signature. . . . .	60
44	Purchase transaction with Erzsebet Benefit Program card. . . . .	61
45	Purchase transaction with ZenCard. . . . .	63
46	Purchase transaction with Fleet Card. . . . .	66
47	Purchase transaction with BLIK code given. . . . .	67
48	Purchase transaction with BLIK code given and PIN. . . . .	68
49	MultiSplit Payment Transaction. . . . .	69

50	Partial DCC refund. . . . .	71
51	DCC Refund with waiting for amount status. . . . .	73
52	Reversal transaction. . . . .	75
53	Reversal transaction (extended transaction type). . . . .	76
54	Giftcard's activation (accepted). . . . .	77
55	Giftcard's activation (rejected). . . . .	78
56	Giftcard's balance check. . . . .	78
57	Giftcard's topup. . . . .	79
58	Execute function - ZenCard flow. . . . .	80
59	Incremental authorization. . . . .	81
60	Preauthorization cancel. . . . .	82
61	Single transaction read from batch request description. . . . .	85
62	Single transaction read from batch response description. . . . .	85
63	Batch summary read request description. . . . .	86
64	Batch summary read response description. . . . .	87
65	Content of payment multicurrency dayclose information. . . . .	88
66	Content of giftcard single currency dayclose information. . . . .	89
67	Content of giftcard multicurrency dayclose information. . . . .	89
68	Transaction Report - Generate request description. . . . .	92
69	Transaction Report - Generate response description. . . . .	93
70	Transaction Report - Generate response details. . . . .	93
71	Transaction Report - Get Summary of Cardsets Transactions request description. . . . .	93
72	Transaction Report - Get Summary of Cardsets Transactions response description. . . . .	94
73	Transaction Report - Content of the single currency Cardsets Transactions Summary description. . . . .	94
74	Transaction Report - Content of the multicurrency Total Summary description. . . . .	95
75	Transaction Report - Content of the Cardsets Transactions Summary description. . . . .	96
76	Transaction Report - Get Total Summary request description. . . . .	96
77	Transaction Report - Get Total Summary response description. . . . .	96
78	Transaction Report - Content of the single currency Total Summary description. . . . .	97
79	TransactionReport - Content of the multicurrency Total Summary description. . . . .	97
80	Content of Transaction Report Summary description. . . . .	98
81	Transaction Analysis - Generate request description. . . . .	100
82	Transaction Analysis - Generate response description. . . . .	100
83	Transaction Analysis - Get Transaction Details request description. . . . .	101
84	Transaction Analysis - Generate request description. . . . .	101
85	Transaction Analysis - Get Total Summary response description. . . . .	101
86	Transaction Analysis - Content of the single currency Total Summary description. . . . .	102
87	Transaction Analysis - Content of the multicurrency Total Summary description. . . . .	102
88	Transaction Analysis - Content of the Total Summary description. . . . .	103
89	Transaction DCC Analysis - Generate request description. . . . .	104
90	Transaction DCC Analysis - Generate response description. . . . .	105
91	Transaction DCC Analysis - Get Currency Summary request description. . . . .	105
92	Transaction DCC Analysis - Get Currency Summary response description. . . . .	105
93	Transaction DCC Analysis - Content of the currency summary description. . . . .	106
94	Transaction DCC Analysis - Get Total Summary request description. . . . .	106
95	Transaction DCC Analysis - Get Total Summary response description. . . . .	106
96	Report summary entry id. . . . .	107
97	Set Parameters request description. . . . .	108
98	Currently supported volatile parameters description. . . . .	108
99	Set Parameters response description. . . . .	108

100	Set Standby request description. . . . .	109
101	Set Standby response description. . . . .	109

## Revision History

Revision	Date	Author(s)	Description
2.0.0	2019.01.25	Albert Sitek	First version of the document.
2.0.1	2019.03.27	Jacek Kapuścik	Allow to send BLIK code in start transaction request. Additional status for TMS connection required after reconciliation.
2.0.2	2019.05.10	Zbigniew Tomica, Jacek Kapuścik	Support of fleet card added (tag 7F01). Detailed cardholder verification info added (in tag DF0B). Support of tip and additional printout data added (in tag DF11 and DF12)
2.0.3	2020.01.20	Zbigniew Tomica, Jacek Kapuścik	DCC Mark-up tags added (tags DF14, DF15, DF16, DF17)
2.0.4	2020.03.24	Zbigniew Tomica	Variable Symbol / Specific Symbol tags added (start tags DF06, DF07, end tags DF18, DF19). "Visa mobile transaction indicator" flag added (0x00000004)
2.0.5	2020.04.15	Zbigniew Tomica	Transaction start tags chapter added. Transaction end tags chapter added. Transaction type chapter updated. Multicurrency tags added (start tags DF08). Multilanguage tags added (start tags DF09, DF0A, DF0B, end tags: DF1A, DF1B)
2.0.6	2020.05.26	Zbigniew Tomica	"Abort transaction after card read flag" added (0x08)
2.0.7	2020.06.18	Artur Smoleń, Albert Sitek	Add MultiSplit Payment support. EMV Data to the Transaction End added (support for PSN added)
2.0.8	2020.08.10	Zbigniew Tomica	Multicurrency for batch read and reports added
2.0.9	2020.08.28	Artur Smoleń	"Card is blacklisted" result code added
2.0.10	2021.04.15	Jacek Kapuścik	Information about mandatory logging added.
2.0.11	2023.01.17	Tomasz Załuska	Rewriting documentation to latex. New transaction start tag 7F01 subtags added. New transaction end tag 7F01 subtags added.
2.0.12	2023.05.12	Zbigniew Tomica	Transaction end tag 7F01 fleet card subtags added (DF0F, DF10).



# 1 Introduction

This document is intended for integrators that implements integrations between eService payment terminal and their master devices (in this document denoted as ECR – Electronic Cash Register). The basic principles of the protocol are as follows:

- a. Terminal acts as a slave device, while ECR is a master device.
- b. Communication between terminal and ECR can be establish using TCP IP connection or via USB/RS-232 serial port. Default TCP IP port is set to 3000.
- c. Terminal, as a slave device, receives orders from ECR, processes it, and replies with proper answer, so terminal **never** sends messages to the ECR without being asked to do so (apart from some exceptions clearly described in this documentation).
- d. If there is no messages received from ECR within **60 seconds**, terminal will end transmission automatically – so to keep the connection established, ECR should send Terminal Status Read command periodically (keep alive).
- e. There must be some delay (around 200-300ms) between terminal response and next ECR request. Terminal may not be able to process next request immediately.

## 2 Basic information

Messages exchanged between terminal and ECR have following structure:

**<STX><Message body><ETX><LRC>[Additional data]**

Where:

- STX = 0x02 - Start of text
- ETX = 0x03 - End of text
- LRC - checksum calculated as modulo 2 sum (XOR) of whole message bytes (byte by byte) excluding STX and including ETX, for example:  
Whole frame without LRC:

**<STX>20<FS>1<ETX>**

The same frame in hex format:

**0x02 0x32 0x30 0x1C 0x31 0x03**

Calculated LRC is:

**0x02  $\oplus$  0x32  $\oplus$  0x30  $\oplus$  0x1C  $\oplus$  0x31  $\oplus$  0x03 = 0x2C**

- Additional data - Optional, used to transfer some binary data in certain messages

### 2.1 Control marks

Symbol	Hex value	Description	
STX	02	Start of text	Frame (message) begin mark.
ETX	03	End of text	Frame (message) end mark.
EOT	04	End of transmission	Request to disconnect. If sent by cash register terminal closes connection after receiving this char. Used in integration via RS232/USB interface and LAN.
ACK	06	Acknowledgement	
NAK	15	Acknowledge (negative)	Frame checksum negative. Sent in case of problems with LRC verification.
FS	1C	Field separator	

### 2.2 Request message format

General request message format is as format:

**<STX><FUN\_ID><FS><ARG\_IN1>... <FS><ARG\_INn><ETX><LRC>**

Field	Field length	Description
STX	1B	Start of message indicator – 0x02
FUN_ID	1B	Function identifier
FS	1B	Field separator – 0x1C
ARG_IN1	var	First input argument for function call – if needed.
...	...	...
FS	1B	Field separator – 0x1C
ARG_INn	var	n-th input argument
ETX	1B	End of message indicator – 0x03
LRC	1B	Checksum

## 2.3 Response message format

General response message format is as follows:

**<STX><FUN\_ID><FS><FUN\_RESULT>**  
**<FS><ARG\_OUT1>...<FS><ARG\_OUTn><ETX><LRC>[Additional data]**

Field	Field length	Description
STX	1B	Start of message indicator – 0x02
FUN_ID	1B	Function identifier – replied from request
FS	1B	Field separator – 0x1C
FUN_RESULT	1B	Result of function execution: 0 = SUCCESS !0 (other cases) = FAILURE
FS	1B	Field separator – 0x1C
ARG_OUT1	var	First output argument
...	...	...
FS	1B	Field separator – 0x1C
ARG_OUTn	var	n-th output argument
ETX	1B	End of message indicator – 0x03
LRC	1B	Checksum
Additional data	var	e.g. binary file data (optional)

**NOTE 1:** Output parameters are sent only when FUN\_RESULT is equal to 0 = SUCCESS.

**NOTE 2:** In case of FUN\_RESULT != 0, terminal may return one value with text description of an error.

**NOTE 3:** For the compatibility with future versions of the protocol, cash register should not assume certain number of values (or sub-values) in the response! The number of values for any function may increase in the future versions of protocol – in that case, existing cash register application, should ignore these additional fields.

**NOTE 4:** STX, ETX, FS and LRC will not be mentioned in further descriptions, but they must be present in every message.

## 2.4 Function return codes in response field

Table 1: Function return codes.

Value	Description
0	Function processed successfully
1	Generic error
2	Function code unknown
3	Wrong function arguments
7	Wrong order of calls (function not possible to call in current terminal state)
8	Requested data not available
9	Not enough memory to process request

## 2.5 Data formats

Data types used in this documentation are described below:

Table 2: Data formats.

Symbol	Name	Description
n	Numeric	ASCII characters from range <'0'-'9'>
x	Hexadecimal	ASCII characters from range <'0'-'9'; 'a'-'f'; 'A'-'F'>
a	Alphanumeric	ASCII characters from range <'0'-'9'; 'a'-'z'; 'A'-'Z'>
s	Special	<p>Special sequence:</p> <p>"\n" (hex. 0x5c, 0x6e) – OK (Green button)</p> <p>"\b" (hex. 0x5c, 0x62) – BACK (Yellow button)</p> <p>"\e" (hex. 0x5c, 0x65)– CANCEL (Red button)</p> <p><b>NOTE:</b> One special sequence is treated as single character</p>

## 2.6 List of available commands

Table 3: List of available commands.

Value	Name	Description
20	TrStatus	Terminal status read ( <a href="#">Section 3.1</a> )
21	TrBegin	Transaction beginning ( <a href="#">Section 4.1</a> )
22	TrEnd, GetData	Transaction data read ( <a href="#">Section 4.2</a> )
23	TrData	Transaction data input ( <a href="#">Section 4.3</a> )
24	TrReconData	Single transaction data read from batch ( <a href="#">Section 7.1</a> )
25	TrReconInfo	Batch summary read ( <a href="#">Section 7.2</a> )
26	TermInfo	Terminal general information read ( <a href="#">Section 3.2</a> )
27	TrLast	Last transaction data read ( <a href="#">Section 4.4</a> )
29	Set Standby Timeout	Sets timeout for standby (unattended only, <a href="#">Section 9.2</a> )
30	Transaction Report - Generate	Generates transaction report ( <a href="#">Section 8.1.1</a> )
31	Transaction Report - Get summary of cardsets transaction	Gets summary of given cardsets' transactions ( <a href="#">Section 8.1.2</a> )
32	Transaction Report - Get total summary	Gets total summary of all transactions ( <a href="#">Section 8.1.3</a> )
33	Transaction Analysis - Generate	Generates transaction analysis ( <a href="#">Section 8.2.1</a> )
34	Transaction Analysis - Get Transaction Details	Details of transaction analysis ( <a href="#">Section 8.2.2</a> )
35	Transaction Analysis - Get Total Summary	Generates summary for transaction analysis ( <a href="#">Section 8.2.3</a> )
36	Transaction DCC Analysis - Generate	Generates transaction DCC analysis ( <a href="#">Section 8.3.1</a> )
37	Transaction DCC Analysis - Get Currency Summary	Details of transaction DCC analysis ( <a href="#">Section 8.3.2</a> )
38	Transaction DCC Analysis - Get Total Summary	Generates summary for transaction DCC analysis ( <a href="#">Section 8.3.3</a> )
39	Set Parameters	Sets parameters ( <a href="#">Section 9.1</a> )

## 2.7 General remarks

In all messages sent to the terminal there is a place for "Cash register number" – there is no further processing on this field on the terminal side. When there is only one ECR communicating with the terminal, this field should be always set to '1'.

## 2.8 Communication logs (MANDATORY)

Collecting communication logs between the ECR and the payment terminal is obligatory. These logs are the basis for providing support in case of problems, both during development and production. The terminal cannot collect communication logs. Logs should include time stamps and complete protocol frames.

## 3 Terminal related commands

### 3.1 Terminal status read

This message should be sent by cash register periodically for detecting status of current transaction (during the transaction) and for checking terminal readiness in between of transactions. Via proper terminal status, terminal may ask ECR to input some data, answer questions etc.

#### Request:

Table 4: Terminal status read request description.

Field	Field length	Field value	Description
FUN_ID	2	"20"	Function identifier – Terminal Status Read
ARG_IN1	1	e.g. "1"	Cash register number

#### Response:

Table 5: Terminal status read response description.

Field	Field length	Field value	Description
FUN_ID	2	"20"	Function identifier – Terminal Status Read
FUN_RESULT	1-2	e.g. "0"	Function return code, ( <a href="#">Table 34</a> )
ARG_OUT1	1-2	e.g. "0"	Transaction status, ( <a href="#">Table 33</a> )
ARG_OUT2	0-4	e.g. "1234"	Sequence number of prompt, from 1 to 9999
ARG_OUT3	0-2048	e.g. "5,7,A,3,Enter code"	Expected data characteristics, ( <a href="#">Table 6</a> )

Table 6: Expected data characteristics description.

Subfield	Subfield length	Subfield value	Description
ARG_OUT3.1	0-2	e.g. "5"	Minimal number of <b>characters</b> to be entered. <b>NOTE:</b> If this subfield is empty or equal to 0, there is no need to enter any characters
SS	1	" , "	Subfield separator
ARG_OUT3.2	1-2	e.g. "7"	Maximal number of <b>characters</b> to be entered
SS	1	" , "	Subfield separator
ARG_OUT3.3	0-1	e.g. "A"	Allowed <b>characters</b> to be entered, possible values: n – numeric (0-9) A – alphanumeric S – any characters for ASCII range, including '.' etc.
SS	1	" , "	Subfield separator
ARG_OUT3.4	0-1	e.g. "3"	Allowed <b>special characters</b> to be entered, possible values: 1 – OK only 2 – CANCEL only 3 – OK and CANCEL 4 – BACK only 5 – BACK and OK 6 – BACK and CANCEL 7 – OK, BACK and CANCEL
SS	1	" , "	Subfield separator
ARG_OUT3.5	0-2048	e.g. "Enter code"	Prompt text. Special characters are encoded using Windows-1250 (Polish, default ones), or certain codepage (depending on terminal's configuration)

### 3.2 Terminal general information read

This message is being sent to get basic information about the terminal (address and identifier).

#### Request:

Table 7: Terminal general information read request description.

Field	Field length	Field value	Description
FUN_ID	2	"26"	Function identifier – Terminal General Information Read
ARG_IN1	1	e.g. "1"	Cash register number
ARG_IN2	1	"1"	Terminal index, should always be set to "1"



## Response:

Table 8: Terminal general information read response description.

Field	Field length	Field value	Description
FUN_ID	2	"26"	Function identifier – Terminal General Information Read
FUN_RESULT	1-2	e.g. "0"	Function return code, ( <a href="#">Table 1</a> )
ARG_OUT1	0-420	var	Terminal General Information, ( <a href="#">Table 9</a> )

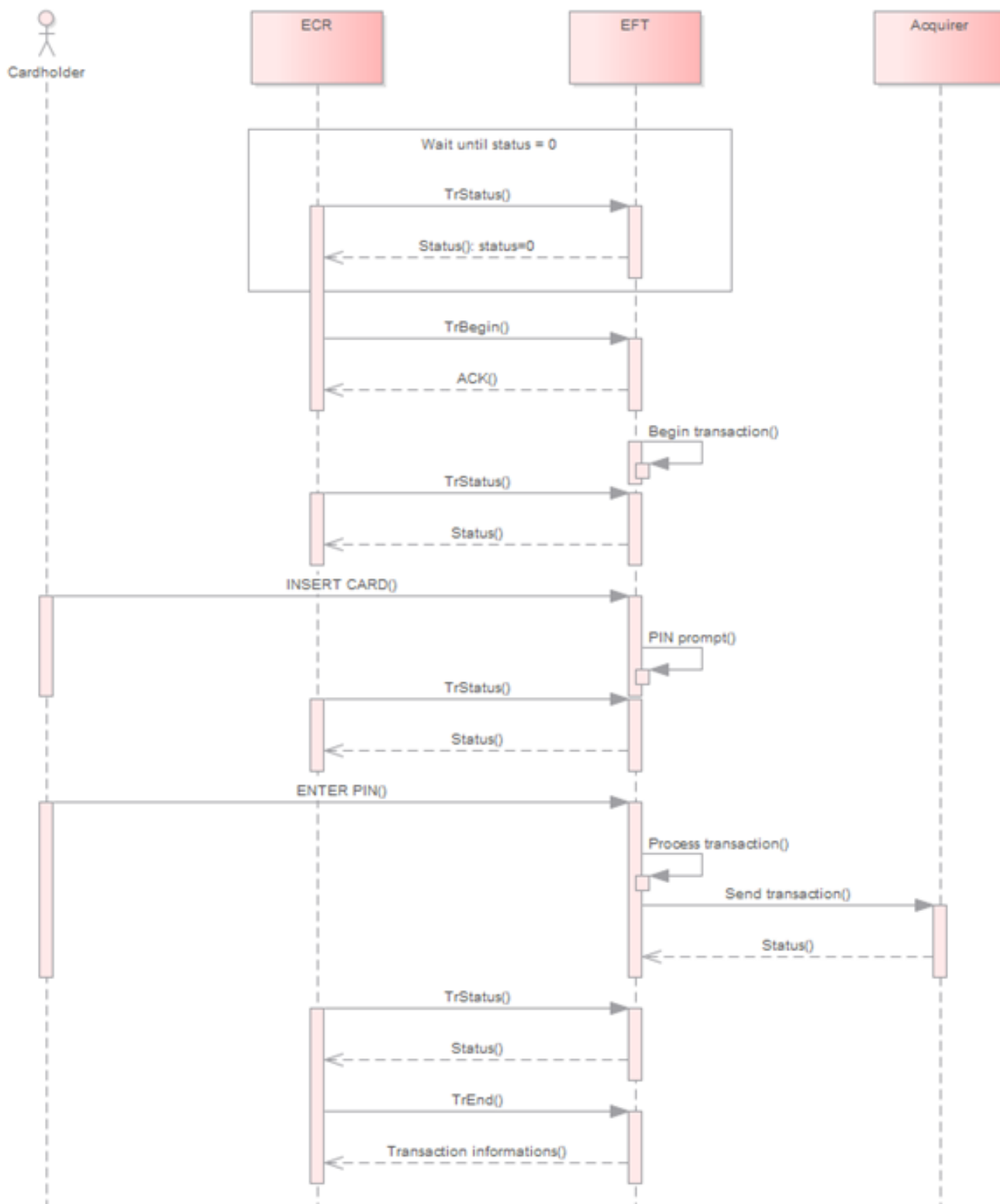
Table 9: Terminal general information description.

Subfield	Subfield length	Subfield value	Description
ARG_OUT1.1	0-128	e.g. "ABC Shop"	Address line 1
SS	1	", "	Subfield separator
ARG_OUT1.2	0-128	e.g. "Bliska 4"	Address line 2
SS	1	", "	Subfield separator
ARG_OUT1.3	0-128	e.g. "02-345 Warsaw"	Address 3
SS	1	", "	Subfield separator
ARG_OUT1.4	0-16	e.g. "0000002022000145"	Merchant identifier
SS	1	", "	Subfield separator
ARG_OUT1.5	0-16	e.g. "00300004"	Terminal identifier

## 4 Transaction processing related commands

### 4.1 Start transaction

This command can be sent to the terminal only if it's state is "Ready for transaction". Below one can find the general transaction flow diagram.



Start transaction command takes various parameters, depending on the type of transaction executed. All currently supported transaction and corresponding frames are describes in the rest part of this chapter.

**Request (general):**

Table 10: General request for start transaction command.

Field	Field length	Field value	Description
FUN_ID	2	"21"	Function identifier – Start Transaction
ARG_IN1	1	e.g. "1"	Cash register number
ARG_IN2	1-2	e.g. "1"	Transaction type, ( <a href="#">Section 4.5</a> )
ARG_IN3...N	var		Additional input arguments

As a response for start transaction command the terminal is sending following frame.

**Response:**

Table 11: General response for start transaction command.

Field	Field length	Field value	Description
FUN_ID	2	"21"	Function identifier – Start Transaction
FUN_RESULT	1-2	e.g. "0"	Function return code, ( <a href="#">Table 1</a> )

#### 4.1.1 Start purchase transaction

Table 12: Start purchase transaction request description.

Field	Field length	Field value	Description
FUN_ID	2	"21"	Function identifier – Start Transaction
ARG_IN1	1	e.g. "1"	Cash register number
ARG_IN2	1	"1"	Transaction type - Purchase
ARG_IN3	1-12	e.g. "1000"	Transaction amount in <b>minor units</b>
ARG_IN4	0-46	e.g. "Purchase 1234"	Transaction title <b>NOTE:</b> This field is optional and should be used when "transaction title" functionality is enabled in the terminal (please contact eService) <b>NOTE 2:</b> Do not use comma ",," in the transaction title.
ARG_IN5	0-256	e.g. "DF010101" denotes: 1. Enable Token Generation	Additional request data encoded as TLV structure, in ASCII format: <tag><len><value> See table ( <a href="#">Table 35</a> ) Tags supported: All

#### 4.1.2 Start preauthorization transaction

Table 13: Start preauthorization transaction request description.

Field	Field length	Field value	Description
FUN_ID	2	"21"	Function identifier – Start Transaction
ARG_IN1	1	e.g. "1"	Cash register number
ARG_IN2	1	"4"	Transaction type - Preauthorization
ARG_IN3	1-12	e.g. "1000"	Transaction amount in <b>minor units</b>
ARG_IN4	0-256		Additional request data encoded as TLV structure, in ASCII format: <tag><len><value> See table ( <a href="#">Table 35</a> ) Tags supported: All

### 4.1.3 Start completion transaction

Table 14: Start completion transaction request description.

Field	Field length	Field value	Description
FUN_ID	2	"21"	Function identifier – Start Transaction
ARG_IN1	1	e.g."1"	Cash register number
ARG_IN2	1	"5"	Transaction type - Completion
ARG_IN3	1-12	e.g."1000"	Transaction amount in <b>minor units</b>
ARG_IN4	0	blank	<b>Not used (for compatibility)</b>
ARG_IN5	1-8	e.g."1000"	Sequence number of preauthorization to be completed
ARG_IN6	0-256		Additional request data encoded as TLV structure, in ASCII format: <tag><len><value> See table ( <a href="#">Table 35</a> ) Tags supported: All

### 4.1.4 Start refund transaction

Table 15: Start refund transaction request description.

Field	Field length	Field value	Description
FUN_ID	2	"21"	Function identifier – Start Transaction
ARG_IN1	1	e.g."1"	Cash register number
ARG_IN2	1	"6"	Transaction type - Refund
ARG_IN3	1-12	e.g."1000"	Transaction amount in <b>minor units</b>
ARG_IN4	0	blank	<b>Not used (for compatibility)</b>
ARG_IN5	0-256		Additional request data encoded as TLV structure, in ASCII format: <tag><len><value> See table ( <a href="#">Table 35</a> ) Tags supported: All

#### 4.1.5 Start reversal transaction

Table 16: Start reversal transaction request description.

Field	Field length	Field value	Description
FUN_ID	2	"21"	Function identifier – Start Transaction
ARG_IN1	1	e.g."1"	Cash register number
ARG_IN2	2	"10"	Transaction type - Reversal
ARG_IN3	1-12	e.g."1000"	Transaction amount in <b>minor units</b>
ARG_IN4	1-8	e.g."123"	Sequence number of transaction to reverse
ARG_IN5	0-256		Additional request data encoded as TLV structure, in ASCII format: <tag><len><value> See table ( <a href="#">Table 35</a> ) Tags supported: DF09,DF0A,DF0B

There are two possible ways to execute reversal transaction:

1. Send transaction amount of last transaction in ARG\_IN3, without ARG\_IN4 - to reverse last transaction.
2. Send both ARG\_IN3 and ARG\_IN4 to reverse any transaction from current batch.

#### 4.1.6 Start giftcard's activation

This command allows to start giftcard's activation transaction on the terminal.

Table 17: Start giftcard's activation request description.

Field	Field length	Field value	Description
FUN_ID	2	"21"	Function identifier – Start Transaction
ARG_IN1	1	e.g."1"	Cash register number
ARG_IN2	2	"42"	Transaction type - Giftcard's activation
ARG_IN3	0-256		Additional request data encoded as TLV structure, in ASCII format: <tag><len><value> See table ( <a href="#">Table 35</a> ) Tags supported: DF09,DF0A,DF0B

#### 4.1.7 Start giftcard's balance check

This command allows to start giftcard's balance check transaction on the terminal.

Table 18: Start giftcard's balance check request description.

Field	Field length	Field value	Description
FUN_ID	2	"21"	Function identifier – Start Transaction
ARG_IN1	1	e.g."1"	Cash register number
ARG_IN2	2	"43"	Transaction type - Giftcard's Balance Check
ARG_IN3	0-256		Additional request data encoded as TLV structure, in ASCII format: <tag><len><value> See table ( <a href="#">Table 35</a> ) Tags supported: DF09,DF0A,DF0B

#### 4.1.8 Start giftcard's topup

Table 19: Start giftcard's topup request description.

Field	Field length	Field value	Description
FUN_ID	2	"21"	Function identifier – Start Transaction
ARG_IN1	1	e.g."1"	Cash register number
ARG_IN2	2	"44"	Transaction type - Giftcard's Topup
ARG_IN3	1-12	e.g."1000"	Transaction amount in <b>minor units</b>
ARG_IN4	0-256		Additional request data encoded as TLV structure, in ASCII format: <tag><len><value> See table ( <a href="#">Table 35</a> ) Tags supported: DF09,DF0A,DF0B

#### 4.1.9 Start reconciliation

Table 20: Start reconciliation request description.

Field	Field length	Field value	Description
FUN_ID	2	"21"	Function identifier – Start Transaction
ARG_IN1	1	e.g."1"	Cash register number
ARG_IN2	2	"31"	Transaction type - Reconciliation
ARG_IN3	0-256		Additional request data encoded as TLV structure, in ASCII format: <tag><len><value> See table ( <a href="#">Table 35</a> ) Tags supported: DF0A

#### 4.1.10 Start connection test

This command is used to check whether the terminal has proper connection with the authorization host.

Table 21: Start connection test request description.

Field	Field length	Field value	Description
FUN_ID	2	"21"	Function identifier – Start Transaction
ARG_IN1	1	e.g."1"	Cash register number
ARG_IN2	2	"33"	Transaction type - Connection test
ARG_IN3	0-256		Additional request data encoded as TLV structure, in ASCII format: <tag><len><value> See table ( <a href="#">Table 35</a> ) Tags supported: DF0A

#### 4.1.11 Start connection to TMS

This command is used to force the terminal to connects to the Terminal Management System for parameters update.

**NOTE:** It's highly recommended to implement this command in final integration.

Table 22: Start connection to the TMS request description.

Field	Field length	Field value	Description
FUN_ID	2	"21"	Function identifier – Start Transaction
ARG_IN1	1	e.g."1"	Cash register number
ARG_IN2	2	"40"	Transaction type - Connection to TMS
ARG_IN3	0-256		Additional request data encoded as TLV structure, in ASCII format: <tag><len><value> See table ( <a href="#">Table 35</a> ) Tags supported: DF0A

#### 4.1.12 Start function execution

This command is used to execute given function on the terminal. Currently it is used only for ZenCard loyalty program.



Table 23: Start function execution request description.

Field	Field length	Field value	Description
FUN_ID	2	"21"	Function identifier – Start Transaction
ARG_IN1	1	e.g. "1"	Cash register number
ARG_IN2	2	"85"	Transaction type - Execute function
ARG_IN3	1	"0"	Type of function to be set: 0 - Zencard's flow
ARG_IN4	0-128	e.g. "register register2"	AdditionalData. In case of ARG_IN3==0, it contains pipe ' ' separated list of flows to execute
ARG_IN5	0-256		Additional request data encoded as TLV structure, in ASCII format: <tag><len><value> See table ( <a href="#">Table 35</a> ) Tags supported: DF09,DF0A,DF0B

#### 4.1.13 Start incremental authorization

Table 24: Start incremental authorization request description.

Field	Field length	Field value	Description
FUN_ID	2	"21"	Function identifier – Start Transaction
ARG_IN1	1	e.g. "1"	Cash register number
ARG_IN2	2	"86"	Transaction type - Incremental authorization
ARG_IN3	2-10	e.g. "005802"	Authorization code of original preauthorization transaction
ARG_IN4	1-12	e.g. "1000"	Incremental authorization amount in <b>minor units</b>
ARG_IN5	0-256		Additional request data encoded as TLV structure, in ASCII format: <tag><len><value> See table ( <a href="#">Table 35</a> ) Tags supported: DF06,DF07,DF08,DF09,DF0A,DF0B

#### 4.1.14 Start preauthorization cancel

Table 25: Start preauthorization cancel request description.

Field	Field length	Field value	Description
FUN_ID	2	"21"	Function identifier – Start Transaction
ARG_IN1	1	e.g. "1"	Cash register number
ARG_IN2	2	"82"	Transaction type - Preauthorization cancel
ARG_IN3	2-10	e.g. "005802"	Authorization code of original preauthorization transaction
ARG_IN4	1-12	e.g. "1000"	Original preauthorization amount in <b>minor units</b>
ARG_IN5	0-256		Additional request data encoded as TLV structure, in ASCII format: <tag><len><value> See table ( <a href="#">Table 35</a> ) Tags supported: DF09,DF0A,DF0B

## 4.2 Transaction End / Transaction Data Read

This message is being sent when cash register needs to obtain result of current transaction. Message should be sent only when terminal reports status "Waiting Transaction End" or "Waiting for Signature Acceptance". Response to this message includes all data needed for printing ticket according to Acquirer's rules.

### Request:

Table 26: Transaction End / Transaction Data Read request description.

Field	Field length	Field value	Description
FUN_ID	2	"22"	Function identifier – Transaction End / Transaction Data Read
ARG_IN1	1	e.g. "1"	Cash register number

### Response:

Table 27: Transaction End / Transaction Data Read response description.

Field	Field length	Field value	Description
FUN_ID	2	"22"	Function identifier – Transaction End / Transaction Data Read
FUN_RESULT	1-2	e.g. "0"	Function return code, see table ( <a href="#">Table 1</a> )
ARG_OUT1	1-3	e.g. "1"	Transaction type, see table ( <a href="#">Section 4.5</a> )
ARG_OUT2	1-2	e.g. "0"	Transaction result code, see table ( <a href="#">Table 34</a> )
ARG_OUT3	0-8	e.g. "1234"	Transaction number
ARG_OUT4	0-2048	var	Transaction details, see table ( <a href="#">Table 28</a> )

Below one can find the detailed description of ARG\_OUT4. All subfields are comma (",") separated.

Table 28: Transaction details description.

Subfield	Subfield length	Subfield value	Description
ARG_OUT4.1	0-8	e.g. "A875F789"	Authorization Code
ARG_OUT4.2	0-64	var	Server message or error message
ARG_OUT4.3	0-16	e.g. "00000020220145"	Merchant ID
ARG_OUT4.4	0-20	e.g. "*****3465"	Masked card number
ARG_OUT4.5	0-10	e.g. "2015.10.23"	Date in format YYYY.MM.DD
ARG_OUT4.6	0-8	e.g. "10:23:22"	Time in format hh:mm:ss
ARG_OUT4.7	0-16	e.g. "Maestro"	Brand name of the card – cardset name
ARG_OUT4.8	0-1	e.g. "B"	Reader used to read card data. This character depends on the acquirer, example values: B – Contactless M- Manual C – Magstripe P – ICC ? – Unknown
ARG_OUT4.9	0-8	e.g. "00300004"	Terminal Identifier
ARG_OUT4.10	0-8	e.g. "1234"	Authorization message number – usually equal to transaction number
ARG_OUT4.11	0-1	e.g. "1"	Authorization type, possible values: 1 – online 3 – offline 4 – referral
ARG_OUT4.12	0-12	e.g. "1000"	Transaction amount (or balance). For DCC transaction this field contains amount in transaction currency. For payment with cashback this is authorized amount (purchase + cash)

Continued on next page

Table 28: Transaction details description. (Continued)

ARG_OUT4.13	0-1	e.g. "A"	Cardholder authorization method, possible values: A – PIN @ - Signature B – PIN and Signature ? – no authorization method has been used during the transaction <b>NOTE:</b> This subfield is obsolete and does not contain all CVMs. Use ARG_OUT4.34/DF0B instead.
ARG_OUT4.14	0-4	e.g. "123"	EMV card transaction counter (ATC)
ARG_OUT4.15	0-16	e.g. "7DBA673BF63298BD"	EMV Transaction Cryptogram (AC)
ARG_OUT4.16	0-32	e.g. "A0000000031010"	EMV Application Identifier (AID)
ARG_OUT4.17	0	blank	<b>Not used (for compatibility)</b>
ARG_OUT4.18	0	blank	<b>Not used (for compatibility)</b>
ARG_OUT4.19	0	blank	<b>Not used (for compatibility)</b>
ARG_OUT4.20	0	blank	<b>Not used (for compatibility)</b>
ARG_OUT4.21	0-10	e.g. "0034000080"	TVR (for EMV)
ARG_OUT4.22	0-4	e.g. "8000"	TSI (for EMV)
ARG_OUT4.23	0-46	e.g. "Invoice no 1234"	Transaction title
ARG_OUT4.24	1	e.g. "0"	Terminal printing indicator (value not equal 0 means that printout has been made by the terminal)
ARG_OUT4.25	0-3	e.g. "PLN"	Transaction currency. Should be always set. Example values: PLN, CHF, USD, EUR... In DCC transaction this currency is selected by user.

Continued on next page

Table 28: Transaction details description. (Continued)

ARG_OUT4.26	0-3	e.g. "PLN"	Terminal currency. Should be set only for DCC transaction. Example values: PLN, CHF, USD, EUR... For one terminal this is the same value for all DCC transactions, e.g. PLN for terminal in Poland.
ARG_OUT4.27	0-12	e.g. "1000"	Transaction amount in terminal currency. Should be set only for DCC transaction.
ARG_OUT4.28	0-12	e.g. "1.2344"	Currency exchange rate. Should be set only for DCC transaction. Uses dot '.' as a separator.
ARG_OUT4.29	0-512	var	DCC text 1. Should be set only for DCC transaction.
ARG_OUT4.30	0-512	var	DCC text 2. Should be set only for DCC transaction.
ARG_OUT4.31	0-12	e.g. "10000"	Cash amount in cashback transaction. Should be set only for cashback transaction.
ARG_OUT4.32	0-12	e.g. "12345"	For Visa Contactless cards: Visa available offline spending amount. For Erzsebet cards: remaining balance of the Voucher Type used. <b>NOTE:</b> should be printed on the customer receipt only, not on a merchant /control receipt
ARG_OUT4.33	0-2	e.g. "2"	DCC currency exponent
ARG_OUT4.34	0-1024	e.g. "DF01023132 DF0206313233343536" In this example: BLIK reference code = "12" BLIK entered code = "123456"	Additional information encoded as TLV structure, in ASCII format Hexadecimal: <tag><len><value> See table ( <a href="#">Table 36</a> )

## 4.3 Transaction Data Input

This message is being sent to the terminal to provide all data needed to perform transaction as it would be entered from terminal's keyboard. It may be used to remote control of terminal's functions and to provide details during transaction processing (accept/decline signature, enter authorization code etc.). When terminal receives this message, verifies it if current prompt number is equal to value given in "last prompt number" and if not the message will be ignored and function will return "bad order of calls" error.

### Request:

Table 29: Transaction Data Input request description.

Field	Field length	Field value	Description
FUN_ID	2	"23"	Function identifier – Transaction Data Input
ARG_IN1	1	e.g. "1"	Cash register number
ARG_IN2	1-4	e.g. "1234"	Last prompt sequence number – the number returned in response to last Terminal Status Read command.
ARG_IN3	1-32	e.g. "\n"	Characters sequence to be sent to the terminal

### Response:

Table 30: Transaction Data Input response description.

Field	Field length	Field value	Description
FUN_ID	2	"23"	Function identifier – Transaction Data Input
FUN_RESULT	1-2	e.g. "0"	Function return code, see table ( <a href="#">Table 1</a> )

## 4.4 Last Transaction Data

This command is used to gather last transaction data. It is usually used in emergency situation: after terminal reboot during the transaction, or after break off of the communication.

### Request:

Table 31: Last Transaction Data request description.

Field	Field length	Field value	Description
FUN_ID	2	"27"	Function identifier – Last Transaction Data
ARG_IN1	1	e.g. "1"	Cash register number
ARG_IN2	1-8	e.g. "1"	Index of desired transaction, counting from 1

### Response:

Format of the Last Transaction Data response in similar to table ([Table 28](#)).

## 4.5 Transaction type

Table 32: Transaction type.

Value	Description
1	Purchase
2	Preauthorization
5	Completion
6	Refund
10	Reversal
31	Reconciliation
33	Connection test
40	Connection to TMS
42	Giftcard's activation
43	Giftcard's balance check
44	Giftcard's topup
82	Preauthorization cancel
85	Execute function
86	Incremental authorization

**NOTE for (Section 7.1) and (Section 8.2.2):** If transaction type is greater than 128, it means that current transaction is a reversal of transaction equal to transaction type – 128, e.g. “129” Means reversal of purchase; for cashback transaction type is equal to 3.

## 4.6 Transaction status

Table 33: Transaction status/Terminal state.

Value	Name	Description
00	Idle	Read for new transaction
01	WaitCard	Waiting for card
02	WaitPIN	Waiting for PIN
03	WaitEMVApp	Waiting for EMV application choice
04	WaitHost	Server communication in progress
05	WaitSign	Waiting for signature acceptance
06	WaitTrEnd	Waiting for transaction ending
07	WaitNoCard	Waiting for card removal
08	WaitBusy	Terminal busy (other request in progress)
09	InProgress	Transaction in progress (transaction is being processed)
0A	WaitCopy	Waiting for acceptance to print ticket copy

Continued on next page

Table 33: Transaction status/Terminal state. (Continued)

0B	WaitAuthCode	Waiting for entering authorization code
0C	WaitAction	Waiting for user action
0D	BatchCompleted	Batch completed, terminal is waiting until cash register will read batch data
0E	DCCCurrency	Selecting currency by cardholder
0F	CashBackAmount	Waiting for enter the cashback amount
10	<b>Used internally</b>	
11	<b>Used internally</b>	
12	Transaction accepted	This status is send depending on application configuration and when terminal knows that transaction is approved. Status sending depends on payment application parameters and is optional.
13	<b>Used internally</b>	
14	Waiting for amount	Status send when terminal waiting for amount entry. <b>IMPORTANT!</b> This status will be sent every time when terminal asking for amount like DCC, tip amount or cash back. To enable this status please set payment application parameter 2097 to value 1.
15	Waiting for selection	Terminal is waiting for item selection from the list. List is returned as comma separated string, first element is the title of the selection menu, e.g.: Select<20>Voucher<20>Type,Meal Culture Means that there is selection menu with title "Select Voucher Type", and two possible options:  1. Meal  2. Culture

Continued on next page



Table 33: Transaction status/Terminal state. (Continued)

16	Asking ECR to print data	<p>Terminal is asking ECR to print some text. This status can be returned by the terminal only if proper terminal parameter is set.</p> <p>Terminal will wait until OK button is sent to the terminal:</p> <p style="text-align: center;">&lt;STX&gt;20 0 16 4 1,1,,1,Text to be printed &lt;ETX&gt; &gt;&lt;LRC&gt;</p> <p>Text messages may contain the following HTML tags:</p> <ul style="list-style-type: none"> <li>• &lt;b&gt;bold &lt;/b&gt;</li> <li>• &lt;u&gt;underline &lt;/u&gt;</li> <li>• &lt;center&gt;center &lt;/center&gt;</li> <li>• &lt;h1&gt;font size &lt;/h1&gt;</li> <li>• &lt;br&gt;newline &lt;/br&gt;</li> </ul> <p>ECR needs to format the text according to the tags.</p>
----	--------------------------	---

Continued on next page

Table 33: Transaction status/Terminal state. (Continued)

17	Presenting ZenCard offer	<p>Complex status that contains ZenCard's offer to be presented on ECR. This status can be returned by the terminal only if the proper terminal parameter is set. Contains of following, pipe-separated (' ') subfields:</p> <ol style="list-style-type: none"> <li>1. discountMsg – message to be shown on ECR, format ans..256</li> <li>2. localDiscounts – a ';' separated list of pairs id, zId (';' separated), max size of list – 9, id format – ans..36, zId format – ans..128</li> <li>3. userId – User Id on ZenCard's side, format ans..36</li> <li>4. amountWithDiscount – proposition of new amount for the transaction, format n12</li> <li>5. zId – Identifier of proposed discount, format ans..128</li> </ol> <p>Each of those fields are optional, and can be empty. An example status is seen below:</p> <pre>&lt;STX&gt;20 0 17 7 1,64,,5,Text of promotion 122A, jdie8d8djjdj;kkxoox,jjisspe00008 998744hr 000000000100  fsie83n&lt;ETX&gt;&lt;LRC&gt;</pre> <p>As a response, terminal is expecting the ECR's decision about the proposal. There can be a possibility to accept, ignore (BACK key is supported), or to cancel it (CANCEL key is supported) – it depends on the offer. ECR should send to the terminal following ' ' separated data:</p> <ol style="list-style-type: none"> <li>1. Offer result (\n, \b, \e)</li> <li>2. ';' separated list of selected zIds (optional)</li> <li>3. New amount of transaction, format n12 (optional)</li> <li>4. ';' separated list of flows to be executed after transaction (optional)</li> </ol> <p>An example status without possibility to ignore offer is seen below (only CANCEL key is supported):</p> <pre>&lt;STX&gt;20 0 17 2 1,64,,5,Text of promotion 122A, jdie8d8djjdj;kkxoox,jjisspe00008 998744hr 000000000100  fsie83n&lt;ETX&gt;&lt;LRC&gt;</pre> <p>Fields are optional, and can be empty. An example response can be seen below:</p> <pre>&lt;STX&gt;23 1 2 \n jdie8d8djjdj; jjisspe00008 000000000100  register;register2&lt;ETX&gt;&lt;LRC&gt;</pre>
----	--------------------------	---

Continued on next page

Table 33: Transaction status/Terminal state. (Continued)

18	Waiting for fid	<p>This status can appear only just after the transaction (transaction data has been collected by the ECR via trEnd request). This status can be returned by the terminal only if parameter "ecr_fid.supported" has been previously set to 1.</p> <p>An example is shown below:</p> <pre>&lt;STX&gt;20 0 18 7 1,64,,2&lt;ETX&gt;&lt;LRC&gt;</pre> <p>In such a situation terminal is expecting fid (format ans..50) of recently finished transaction – example:</p> <pre>&lt;STX&gt;23 1 7 347583&lt;ETX&gt;&lt;LRC&gt;</pre>
19	Asking ECR to receive data	<p>Terminal is asking ECR to receive given data. . This status may appear while ZenCard's flow is running. Data is sent as ' ' separated key=value map. Terminal will wait until OK button is sent to the terminal:</p> <pre>&lt;STX&gt;20 0 19 4 1,1,,1,param1=some text number1=45 bool1=false&lt;ETX&gt;&lt;LRC&gt;</pre>
20	Asking ECR to provide data	<p>Terminal is asking to provide given data. This status may appear while ZenCard's flow is running. Required data is sent as ' ' separated list of variable identifiers. An example request can be found below:</p> <pre>&lt;STX&gt;20 0 20 4 1,128,,,param1 number1 bool1&lt;ETX&gt;&lt;LRC&gt;</pre> <p>As a response terminal is expecting ' ' separated key=value map of requested data. An example response can be as follows:</p> <pre>&lt;STX&gt;23 1 4 param1=test number1=23 bool1=true&lt;ETX&gt;&lt;LRC&gt;</pre> <p><b>NOTE:</b> Variables set to "_CANCEL" or "_ABORT" are specially treated by the terminal and will cause ZenCard's flow to be cancelled or aborted.</p>

Continued on next page

Table 33: Transaction status/Terminal state. (Continued)

21	Zencard continue	<p>This status is sent when Zencard transaction is finished and terminal continues Zencard processing ie. asks for fid, executes flow etc. After receiving this state, ECR can obtain transaction data using GetData function. Terminal will wait until OK button is sent to the terminal.</p> <p>An example is shown below:</p> <pre>&lt;STX&gt;20 0 21 6 1,1,1&lt;ETX&gt;&lt;LRC&gt;</pre> <p>As a response terminal can obtain transaction data, sending GetData command:</p> <pre>&lt;STX&gt;22 1&lt;ETX&gt;&lt;LRC &gt; &lt;STX&gt;22 0 1 0 0004 123456,,44444444,*****4123, 2017.06.28,16:24:24,Mastercard,C,12345678,0008,1,1000, A,,,,,011442,75C76392A871449840AE3942BD415F45,,,1, PLN,,,,,DF0E03313030 &lt;ETX &gt;&lt;LRC&gt;</pre> <p>Sending OK:</p> <pre>&lt;STX&gt;23 1 6 \n&lt;ETX &gt;&lt;LRC&gt; &lt;STX&gt;23 0\n&lt;ETX &gt;&lt;LRC&gt;</pre>
80	Application in error state	For instance when card readers are not operational
81	Reconciliation needed	Transaction executing is not possible until successful reconciliation

## 4.7 Transaction result code

Table 34: Transaction result code.

Value	Description
0	Transaction accepted
1	Transaction refused
2	No connection
7	Transaction interrupted by the user
9	Card is blacklisted

## 4.8 Transaction start tags

Table 35: Transaction start tags.

Value	Name	Description
DF01	Transaction flags	<p>1 byte container that contains various information about transaction. Possible flags:</p> <ul style="list-style-type: none"> <li>• 0x01 – Card token generation – if set, token for used card will be generated and returned in response for Transaction End command <b>NOTE:</b> This functionality is available on dedicated applications and terminals (please contact eService)</li> <li>• 0x02 – Reference number generation – if set, reference number will be generated and returned in response for Transaction End command</li> <li>• 0x04 – Transit transaction indicator – if set, transit transaction will be performed</li> <li>• 0x08 – Abort transaction after card read (for token generation only) – if set, transaction will be aborted after card read</li> </ul> <p>Example 1: "DF010101" denotes: 1. Enable Token Generation</p> <p>Example 2: "DF010106" denotes: 1. Enable Reference Number Generation 2. Enable Transit Transaction Indicator</p> <p>Example 3: "DF010109" denotes: 1. Enable Token Generation 2. Abort transaction after card read</p>
DF02	BLIK code	<p>If given, terminal will start transaction immediately using this code. Example: "DF0206313233343536" denotes: 1. Perform transaction with BLIK code "123456"</p>

Continued on next page

Table 35: Transaction start tags. (Continued)

DF03	Additional printout data	<p>Format ans..64</p> <p>Example:</p> <p><b>"DF0328544849532049532041444449544F4E414C205052494E544F555420444154412046524F4D20454352"</b></p> <p>denotes:</p> <p>Additional printout data: "THIS IS ADDITIONAL PRINTOUT DATA FROM ECR"</p>
DF04	"Hints"	<p>"Hints" – to be used to enable ZenCard's ECR extensions, format: ans..40</p> <p><b>NOTE:</b> This functionality is available on dedicated applications and terminals (please contact eService)</p> <p>Example: <b>"DF04085A656E48696E7473"</b></p> <p>denotes:</p> <p>Set "Hints" to value "ZenHints"</p>
DF05		RFU
DF06	Variable Symbol	<p>Format ans..20</p> <p>Example: <b>"DF06053132333435"</b></p> <p>denotes:</p> <p>Variable Symbol: "12345"</p>
DF07	Specific Symbol	<p>Format ans..20</p> <p>Example:</p> <p><b>"DF07 053637383930"</b></p> <p>denotes:</p> <p>Specific Symbol: "67890"</p>
DF08	Transaction currency code	<p>Transaction currency that will replace terminal default currency. ISO 4217 code. Format n3</p> <p>Example:</p> <p><b>"DF0803393835"</b></p> <p>denotes:</p> <p>Currency code "985"</p>
DF09	ECR supported languages	<p>List with languages supported by ECR. Terminal will allow only those languages when displaying selection menu. ISO 639-1 codes, comma separated. Format ans..64</p> <p>Example:</p> <p><b>"DF0908504C2C454E2C435A"</b></p> <p>denotes:</p> <p>Supported languages: "PL,EN,CZ"</p>
DF0A	Cashier language	<p>Language for cashier messages.ISO 639-1 code. Format an2</p> <p>Example:</p> <p><b>"DF0A02454E"</b></p> <p>denotes:</p> <p>Cashier language: "EN"</p>

Continued on next page

Table 35: Transaction start tags. (Continued)

DF0B	Cardholder language	Language for cardholder messages. ISO 639-1 code.Format an2 Example: "DF0B02435A" denotes: Cardholder language: "CZ"
------	---------------------	--

Continued on next page

Table 35: Transaction start tags. (Continued)

7F01	Sale Item List	<p>Constructed tag, that contains nested tags. Currently supported nested tags:</p> <ul style="list-style-type: none"> <li>• <b>7F01</b> - Sale Item.</li> </ul> <p>Constructed tag, that contains nested tags. Currently supported nested tags:</p> <ul style="list-style-type: none"> <li>• <b>DF01</b> - Item ID</li> <li>• <b>DF02</b> - Product Code</li> <li>• <b>DF03</b> - Amount (Total cost of item)</li> <li>• <b>DF04</b> - Unit Measure</li> <li>• <b>DF05</b> - Unit Price (Price of single unit of item)</li> <li>• <b>DF06</b> - Quantity</li> <li>• <b>DF07</b> - Tax Code</li> <li>• <b>DF08</b> - ECR Item ID</li> <li>• <b>DF09</b> - Group ID (0 — Unknown / 1 — Fuel product)</li> </ul> <p><b>NOTE:</b> This functionality is available on dedicated applications and terminals (please contact eService) Example: ”7F0161 7F012F DF010A31353933383335353230 DF02023737 DF03053130363330 DF04024541 DF05053130363330 DF060131 DF07014E 7F012C DF0109313239373734303832 DF02023833 DF030433333030 DF04024541 DF050433333030 DF060131 DF07014E DF080461303031 DF090130”</p> <p>denotes: Product list: Product 1: Item ID: 1593835520 Product Code: 77 Amount: 106.30 Unit Measure: EA Unit Price: 106.30 Quantity: 1 Tax Code: 4E</p>
------	----------------	--

Continued on next page



Table 35: Transaction start tags. (Continued)

		<p>Product 2:</p> <p>Item ID: 129774082</p> <p>Product Code: 83</p> <p>Amount: 33.00</p> <p>Unit Measure: EA</p> <p>Unit Price: 33.00</p> <p>Quantity: 1</p> <p>Tax Code: 4E</p> <p>ECR item ID: a001</p> <p>Group ID: 0</p>
7F02	MultiSplit Payment Item List	<p>Constructed tag, that contains nested tags.</p> <p>Currently supported nested tags:</p> <ul style="list-style-type: none"> <li>• <b>7F01</b> - Sale Item.</li> </ul> <p>Constructed tag, that contains nested tags.</p> <p>Currently supported nested tags:</p> <ul style="list-style-type: none"> <li>• <b>DF01</b> - Account type, range 70 - 90</li> <li>• <b>DF02</b> - Amount type 60 - for donation 00 - for other</li> <li>• <b>DF03</b> - Amount</li> <li>• <b>DF04</b> - Label (optional) MSP label on printout</li> <li>• <b>DF05</b> - Title list (optional) List of MSP titles separated by ", ", maximally 4 titles</li> </ul> <p><b>NOTE:</b> This functionality is available on dedicated applications and terminals (please contact eService)</p> <p>Example:</p> <p>"<b>7F024B</b></p> <p><b>7F0116</b></p> <p><b>DF01023730</b></p> <p><b>DF02023630</b></p> <p><b>DF0303353030</b></p> <p><b>DF05032C2C2C</b></p> <p><b>7F012F</b></p> <p><b>DF01023731</b></p> <p><b>DF02023030</b></p> <p><b>DF030431323030</b></p> <p><b>DF04074C6162656C2031</b></p> <p><b>DF05115469746C6520312C2C5469746C6520332C</b></p> <p>denotes:</p> <p>Accounts list:</p> <p>Account 1:</p> <p>Account type: 70</p> <p>Amount type: 60</p> <p>Amount: 500</p> <p>Title list: ,(empty)</p> <p>Account 2:</p> <p>Account type: 71</p> <p>Amount type: 00</p> <p>Amount: 1200</p> <p>Label: Label 1      Title list: Title 1,,Title 3,(empty)</p>

## 4.9 Transaction end tags

Table 36: Transaction end tags.

Value	Name	Description
DF01	BLIK reference code	BLIK reference code Example: "DF01023132" denotes: BLIK reference code: "12"
DF02	BLIK entered code	BLIK entered code Example: "DF0206313233343536" denotes: BLIK entered code: "123456"
DF03	Card token	Card token Example: "DF031263635f4a5f7744524b48366a6d4945625f38" denotes: Card token: "cc_J_wDRKH6jmIEb_8"
DF04	EMV Cryptogram type	Possible values: <ul style="list-style-type: none"> <li>• 0x00 - AAC</li> <li>• 0x40 - TC</li> <li>• 0x80 - ARQC</li> </ul> Example: "DF040180" denotes: EMV Cryptogram type: TC
DF05	Card expiry date	Returned if expiry date should be printed on the receipt. Format: YYMM Example: "DF05021610" denotes: Card expiry date: 1610
DF06 DF07 DF08 DF09 DF0A	Additional cardset value	Value to print on receipt (5 possible variables) Example: "DF060D49494E3A203330393030363136" denotes: Additional cardset value 1: "IIN: 30900616"

Continued on next page

Table 36: Transaction end tags. (Continued)

DF0B	Additional transaction flags	<p>4 bytes container that contains various information about transaction. Possible flags:</p> <ul style="list-style-type: none"> <li>• Byte 4: <ul style="list-style-type: none"> <li>– 0x00000001 – Multilanguage printout should be printed (for now only returned for UPI cards)</li> <li>– 0x00000002 – Flag set if reversal for transaction made manually</li> <li>– 0x00000004 – Visa mobile transaction indicator (from FFI)</li> </ul> </li> <li>• Byte 3: <p>Cardholder Verification Method flags.</p> <ul style="list-style-type: none"> <li>– 0x00000100 – No CVM</li> <li>– 0x00000200 – No CVM, signature on file</li> <li>– 0x00000400 – PIN Online</li> <li>– 0x00000800 – PIN Offline</li> <li>– 0x00001000 – Signature</li> <li>– 0x00002000 – CDCVM</li> <li>– 0x00004000 – WithID</li> </ul> </li> </ul> <p>Example: "DF0B0400000001" denotes: Additional transaction flags: Multilanguage printout should be printed</p>
DF0C	Client ticket PAN	<p>Properly formatted PAN that should be printed on customer copy (if needed). Used when masking is different on customer and merchant copy. Example: "DF0C1024242424242424242424242424243033353533" denotes: Masked client PAN: "\$\$\$\$\$\$\$\$\$\$\$03553"</p>
DF0D	Voucher type	<p>Name of Voucher Type to be printed for Benefit Program cards (Erzsebet). Example: "DF0D1245727A73656265743A2043756C747572616C" denotes: Voucher Type: "Erzsebet: Cultural"</p>

Continued on next page

Table 36: Transaction end tags. (Continued)

DF0E	Zencard discount amount	Present if ZenCard discount was used during the transaction. Example: " <b>DF0E</b> 03313030" denotes: Discount amount: 100
DF0F	Total preauthorization amount	Total preauthorization amount in terminal currency. Should be set only for DCC transaction. Example: " <b>DF0F</b> 0436363730" denotes: Total preauthorization amount in transaction currency: 6670
DF10	Total preauthorization amount in terminal currency	Total preauthorization amount in terminal currency. Should be set only for DCC transaction. Example: " <b>DF10</b> 53233303030" denotes: Total preauthorization amount in terminal currency: 23000
DF11	Additional printout data	Additional printout data from ECR. Example: " <b>DF11</b> 28544849532049532041444449544F4E414C205052494E544F555420444154412046524F4D20454352" denotes: Additional printout data: THIS IS ADDITONAL PRINTOUT DATA FROM ECR
DF12	Amount of tip	Amount of tip. Example: " <b>DF12</b> 03313030" denotes: Amount of tip: 100
DF13	Reference number	Reference number. Example: " <b>DF13</b> 19343534353031333930303030303030373034313033333039" denotes: Reference number: 454501390000000704103309
DF14	DCC Mark-up value	DCC Mark-up value. Example: " <b>DF14</b> 0B302E303031323334353637" denotes: DCC Mark-up: 0.001234567
DF15	DCC Mark-up text	DCC Mark-up text. Example: " <b>DF15</b> 1F4D41524B5550204F564552204543422045584348414E474520524154452A3A" denotes: DCC Mark-up text: "MARKUP OVER ECB EXCHANGE RATE*."

Continued on next page

Table 36: Transaction end tags. (Continued)

DF16	DCC Mark-up info	<p>DCC Mark-up info. Example: "DF16712A204D41524B55502049532053484F574E2041532041202520414741494E5354205448452045584348414E47452052415445205055424C495348454420425920544845204555524F5045414E2043454E5452414C2042414E4B20464F52205245464552454E434520555345204F4E4C592E"</p> <p>denotes: DCC Mark-up info: "* MARKUP IS SHOWN AS A % AGAINST THE EXCHANGE RATE PUBLISHED BY THE EUROPEAN CENTRAL BANK FOR REFERENCE USE ONLY."</p>
DF17	DCC wholesale rates text	<p>DCC wholesale rates text. Example: "DF174157686F6C6573616C6520726174657320617265207075626C697368656420756E64657220746865206C696E6B3A207777772E65736572766963652E636F6D2E706C"</p> <p>denotes: DCC wholesale rates text: "Wholesale rates are published under the link: <a href="http://www.eservice.com.pl">www.eservice.com.pl</a>"</p>
DF18	Variable Symbol	<p>Variable Symbol. Example: "DF18053132333435"</p> <p>denotes: Variable Symbol: "12345"</p>
DF19	Specific Symbol	<p>Specific Symbol. Example: "DF19053637383930"</p> <p>denotes: Specific Symbol: "67890"</p>
DF1A	Merchant copy ticket language	<p>ISO 639-1 code. Used when ticket should be printed with non-default language dictionary. Example: "DF1A02454E"</p> <p>denotes: Merchant ticket language: "EN"</p>
DF1B	Client copy ticket language	<p>ISO 639-1 code. Used when ticket should be printed with non-default language dictionary. Example: "DF1B02435A"</p> <p>denotes: Client ticket language: "CZ"</p>

Continued on next page

Table 36: Transaction end tags. (Continued)

DF1C	MultiSplit Payment data	Returns records of MultiSplit Payment labels correlated with amounts (present if MSP Data Item list was send and MSP was used). Label and amount shall be equality sign separated '=' and records shall be pipe ' ' separated. Encoding: CP1250. Example: "DF1C20446F6E6174696F6E3D313030307C4F74686572206163636F756E743D31303030" denotes: MultiSplit Payment data: "Donation=1000 Other account=1000"
DF1D	MultiMCC data	Returns the transaction type that has been made with MultiMCC functionality – this feature must be switched on by EVO. Label and type of MCC shall be equality sign separated '='. Encoding: CP1250. Example: "DF1D104D434320436F64653D5469636B657473" denotes: MultiMCC data: "MCC Code=Tickets"
DF1E	Cardset name	The name of the card used in the transaction. Encoding: CP1250. Example: "DF1E0456495341" denotes: Cardset name: "VISA"

Continued on next page

Table 36: Transaction end tags. (Continued)

7F01	Fleet card data	<p>Constructed tag, that contains nested tags. Currently supported tags:</p> <ul style="list-style-type: none"> <li>• <b>DF01</b> - Fleet card flags 1 byte container that contains information about transaction. Possible flags: Byte 1: <ul style="list-style-type: none"> <li>– 0x01 – Indicator - flag set if fleet card was used</li> <li>– 0x02 – Signature – flag set if extra signature lines are required on printout</li> <li>– 0x04 – Without amount and currency on customer copy - flag set if customer copy printout should not contain amount and currency</li> </ul> </li> <li>• <b>DF02</b> - Product names Dictionary records of product names correlated with particular Item IDs (present if Sale Item list was send and Fleet Card was used). Product name and Item ID shall be equality sign separated '=' and dictionary records shall be pipe ' ' separated.</li> <li>• <b>DF03</b> — Vehicle number</li> <li>• <b>DF04</b> — Driver number</li> <li>• <b>DF05</b> — Mileage</li> <li>• <b>DF06</b> — Car registration number</li> <li>• <b>DF07</b> — Unit number</li> <li>• <b>DF08</b> — Additional data</li> <li>• <b>DF09</b> — Car replacement indicator</li> <li>• <b>DF0A</b> — Driver code</li> <li>• <b>DF0B</b> — Third party TID</li> <li>• <b>DF0C</b> — Decline reason (Fleet card transaction detailed decline reason. If set, should be used to process specific errors.) Possible values: 1 — products not allowed (use DF0D, DF0E tags)</li> <li>• <b>DF0D</b> — Allowed products (Allowed product Item ID list, pipe ' ' separated. Set in case "Product not allowed" decline)</li> <li>• <b>DF0E</b> — Not allowed products (Not allowed product Item ID list, pipe ' ' separated. Set in case "Product not allowed" decline)</li> <li>• <b>DF0F</b> — MPK code</li> <li>• <b>DF10</b> — Transaction ID</li> </ul>
------	-----------------	---

Table 36: Transaction end tags. (Continued)

		<p>Example:  <b>”7F0144</b>  <b>DF010103</b>  <b>DF0234313539333833353532303D</b>  42656E7A796E612050623935  7C333836323939373235313D  532F484F5420444F47205041524F574B41  <b>DF0506313235303030</b>  <b>DF0B083132333431323334”</b></p> <p>denotes:  Fleet card flags: 0x03  Indicator flag: set  Signature flag: set  Product names: 1593835520=Benzyna Pb95 3862997251=S/HOT DOG PAROWKA  Mileage: 125000  Third party TID: 12341234</p>
7F03	EMV Data	<p>Contains EMV data from the transaction. Currently used tags are:  <b>5F34</b> - Application PAN Sequence Number  Example:  <b>”7F03045F340101”</b></p>



## 5 Transaction processing

### 5.1 General information

**NOTE:** Each transaction can be initiated only when Terminal Status indicates 00 - Idle.

During typical transaction processing cash register only initiates changes from WaitTrEnd and Idle states. Cash register have also possibility to send transaction cancel request, but only if response to terminal status request includes prompt number and there is a possibility to send \e sequence. In case of transaction cancelation or when an error occurs during transaction, terminal changes its state to WaitTrEnd. All other state changes are triggered by a user action or automatically during transaction processing. There is a possibility to check transaction status at any time. However, transaction status may change in the meantime between status request and transaction begin or end request, so terminal may respond with an error (**wrong order of calls**). Another status check should be performed to assure terminal is not busy. After terminal or cash register power on first operation should be status checking. If terminal is in Idle state it is possible to begin new transaction, check the last transaction status if its result is unknown or read terminal localization information. If last transaction is approved and unknown to cash register it could be reversed (TrBegin(10)). If BatchCompleted state is set cash register can only read batch data with TrReconTrans and TrReconInfo commands. All other states mean that transaction has been started but was not finished yet. Than cash register should continue transaction form current state to the end.

### 5.2 Cashback transaction

Transaction with cash back is started in the same way as ordinary purchase transaction. After inserting the card, payment application checks if cash back functionality is available. If yes, payment application is switched to the state 15 with prompt to the cash register in order to input cash back amount. Cash register should send back message "Transaction data input". If cash back functionality is not available for this particular transaction, the transaction will be processed as a ordinary purchase transaction.

At this stage a merchant should enter in cash register interface the cash back amount. There are 2 possible answers from cash register:

- Cash back amount. If cash back amount is too low or too high the whole transaction will be stopped.
- \e. It means that the whole transaction should be stopped.

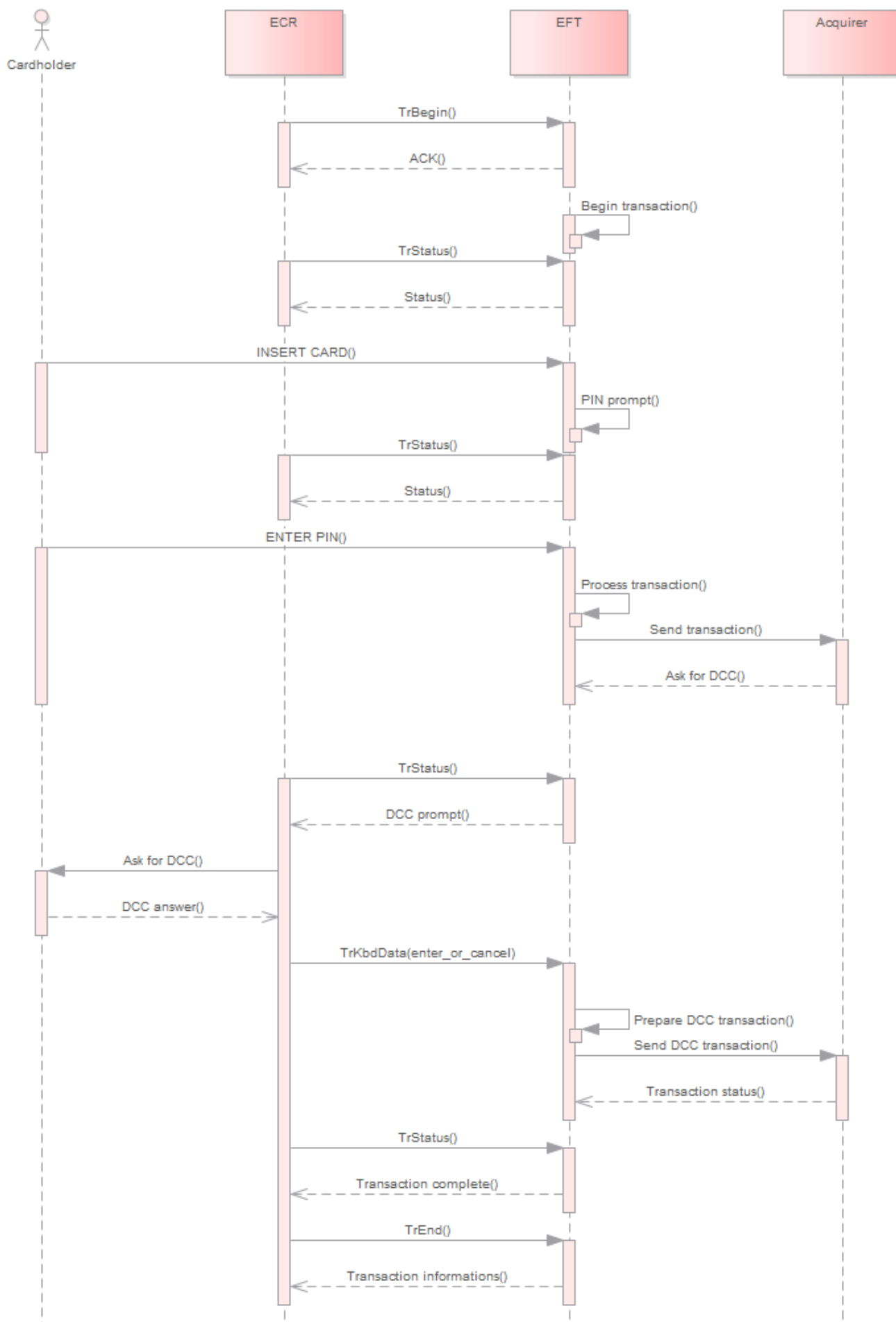
There is one additional field in transaction data sent at the end of processing in the message "Transaction data read". The field val4.30 is set only for transaction with cash back. The field val4.12 contains total authorized amount – purchase + cash back.

**Important:** every cash back transaction is non-DCC transaction.

### 5.3 DCC transaction

The whole processing of DCC transaction from cash register protocol is the same as for typical purchase transaction with one difference. There is an additional question on the terminal side to decide whether a DCC transaction should be performed or not. The selection of currency is possible to perform only on terminal / pinpad side. There are additional fields in transaction data sent at the end of processing in the message "Transaction data read". Data stored in fields "ARG\_OUT4\_25 – ARG\_OUT4\_29" are set only for DCC transaction. For non-DCC transaction these fields will be empty.

**Important:** Field "ARG\_OUT4\_12" contains amount in transaction currency in DCC transaction.



## 6 Example message flows

In this chapter one can find example message exchanges between ECR and Terminal.

**NOTE:** Field separator (FS, 0x1C) is presented on traces below as a pipe ('|').

**NOTE 2:** Please be aware that in case of some messages, a pipe ('|') can be a part of the message body (e.g. Execute function command).

### 6.1 DCC transaction with PIN

Table 37: DCC transaction with PIN.

Sender	Message	Description
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 8<ETX><LRC >	Terminal is busy
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0<ETX><LRC >	Terminal ready for new transaction
ECR	<STX>21 1 1 20000<ETX><LRC>	Request for new transaction for amount of 200.00
T	<STX>21 0<ETX><LRC >	Transaction began
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 01 1 0,1,,2,Enter card<ETX ><LRC>	Waiting for card (possible to cancel using \e sequence)
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 09<ETX><LRC >	Transaction in progress
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 2 2 0,1,,2,Enter PIN<ETX ><LRC>	Waiting for PIN (possible to cancel)
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 04<ETX><LRC >	Waiting for transaction authorization
ECR	<STX>20 1<ETX><LRC >	Terminal status check

Continued on next page

Table 37: DCC transaction with PIN. (Continued)

T	<STX>20 0 0C 25 1,1,,3,Change currency?<ETX><LRC>	Does customer want to change currency? Exchange rate are displayed on the terminal screen by default and can be send in message being sent to ECR.
ECR	<STX>23 1 25 \n<ETX><LRC>	DCC confirmation
T	<STX>23 0<ETX><LRC >	Answer accepted
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 09<ETX><LRC >	Transaction in progress
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 2 2 0,1,,2, <ETX><LRC>	Waiting for second try PIN
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 04<ETX><LRC >	Waiting for transaction authorization
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 06<ETX><LRC >	Waiting for transaction finalization
ECR	<STX>22 1<ETX><LRC >	Transaction ending

Continued on next page

Table 37: DCC transaction with PIN. (Continued)

T	<STX>22 0 1 0 0133 013492,,000000000002009, *****0166,2016.07.25,17:32:09, MAESTRO,P,00300004,000133,1,5150,A,014C, 899865F188781478,A0000000043060,,,,,0200008000 ,E800,,0,EUR,PLN,20000,0.2575000,I understand that MasterCard has a currency conversion process and that I have chosen not to use the MasterCard currency conversion process and I will no recourse against MasterCard with respect to any matter related to the currency conversion. ,,0,2,DF040140 DF140B302E303031323334353637DF151E4D 41524B5550204F5645522045434220455843 48414E474520524154452ADF16712A204D41 524B55502049532053484F574E2041532041 202520414741494E53542054484520455843 48414E47452052415445205055424C495348 454420425920544845204555524F5045414E 2043454E5452414C2042414E4B20464F5220 5245464552454E434520555345204F4E4C592E <ETX><LRC>	Transaction data
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0<ETX><LRC >	Terminal is ready for new transaction

## 6.2 Cashback transaction with PIN

Table 38: Cashback transaction with PIN.

Sender	Message	Description
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 8<ETX><LRC >	Terminal is busy
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0<ETX><LRC >	Terminal ready for new transaction
ECR	<STX>21 1 1 20000<ETX><LRC>	Request for new transaction for amount of 200.00
T	<STX>21 0<ETX><LRC >	Transaction began
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 01 1 0,1,,2,Enter card<ETX ><LRC>	Waiting for card (possible to cancel using \e sequence)

Continued on next page

Table 38: Cashback transaction with PIN. (Continued)

ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0C 2 1,1,,3,Payment with cashback? <ETX><LRC>	Waiting for cashback confirmation
ECR	<STX>23 1 2 \n<ETX><LRC>	Confirmation with \n sequence
T	<STX>23 0<ETX ><LRC >	Confirmation accepted
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0F 3 1,8,n,3,Enter cashback amount <ETX><LRC>	Waiting for amount of the cashback
ECR	<STX>23 1 3 5000<ETX><LRC>	Cash back amount (50.00)
T	<STX>23 0 <ETX><LRC >	Sequence accepted
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 2 4 0,1,,2, <ETX><LRC>	Waiting for PIN (possible to cancel)
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 2 5 0,1,,2, <ETX><LRC>	Waiting for second try PIN
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 04<ETX><LRC >	Waiting for transaction authorization
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 06<ETX><LRC >	Waiting for transaction finalization
ECR	<STX>22 1<ETX><LRC >	Transaction ending
T	<STX>22 0 1 0 0004 123456,, 44444444,*****4123,2009.04.28,16:24:24,Mae- stro,C,12345678,0008, 1,20000,A,,,,,011442,75C76392A871449840AE3942BD415F45 ,,1,PLN,,,,,5000 <ETX><LRC>	Transaction data
ECR	<STX>20 1<ETX><LRC >	Terminal status check

Continued on next page

Table 38: Cashback transaction with PIN. (Continued)

T	<STX>20 0 0<ETX><LRC >	Terminal is ready for new transaction
---	------------------------	---------------------------------------

## 6.3 Purchase transaction with PIN

Table 39: Purchase transaction with PIN.

Sender	Message	Description
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 8<ETX><LRC >	Terminal is busy
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0<ETX><LRC >	Terminal ready for new transaction
ECR	<STX>21 1 1 20000<ETX><LRC>	Request for new transaction for amount of 200.00
T	<STX>21 0<ETX><LRC >	Transaction began
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 01 1 0,1,,2,Enter card<ETX ><LRC>	Waiting for card (possible to cancel using \e sequence)
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 7<ETX><LRC >	Waiting for card removal
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 2 2 0,1,,2, <ETX><LRC>	Waiting for PIN (possible to cancel)
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 04<ETX><LRC >	Waiting for transaction authorization
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 2 5 0,1,,2, <ETX><LRC>	Waiting for second try PIN

Continued on next page

Table 39: Purchase transaction with PIN. (Continued)

ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 04<ETX><LRC >	Waiting for transaction authorization
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 06<ETX><LRC >	Waiting for transaction finalization
ECR	<STX>22 1<ETX><LRC >	Transaction ending
T	<STX>22 0 1 0 0004 123456,,44444444, *****4123,2009.04.28,16:24:24,Maestro,C,12345678, 0008,1,20000,A,,,,,011442, 75C76392A871449840AE3942BD415F45 ,,1,PLN,,,,, <ETX><LRC>	Transaction data
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0<ETX><LRC >	Terminal is ready for new transaction

## 6.4 Purchase transaction with PIN, with transaction accepted status (12) enabled

Table 40: Purchase transaction with PIN, with transaction accepted status (12) enabled.

Sender	Message	Description
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0<ETX><LRC >	Terminal ready for new transaction
ECR	<STX>21 1 1 20000<ETX><LRC>	Request for new transaction for amount of 200.00
T	<STX>21 0<ETX><LRC >	Transaction began
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 09<ETX><LRC >	Transaction in progress
ECR	<STX>20 1<ETX><LRC >	Terminal status check

Continued on next page



Table 40: Purchase transaction with PIN, with transaction accepted status (12) enabled. (Continued)

T	<STX>20 0 01 1 0,1,,2,Enter card<ETX><LRC>	Waiting for card (possible to cancel using \e sequence)
ECR	<STX>20 1<ETX><LRC>	Terminal status check
T	<STX>20 0 09<ETX><LRC>	Transaction in progress
ECR	<STX>20 1<ETX><LRC>	Terminal status check
T	<STX>20 0 2 2 0,1,,2,Enter PIN <ETX><LRC>	Waiting for PIN (possible to cancel)
ECR	<STX>20 1<ETX><LRC>	Terminal status check
T	<STX>20 0 04<ETX><LRC>	Waiting for transaction authorization
ECR	<STX>20 1<ETX><LRC>	Terminal status check
T	<STX>20 0 12 6 0,1,,1,Transaction Accepted<ETX><LRC>	Transaction accepted – with new status 12
ECR	<STX>20 1<ETX><LRC>	Terminal status check
T	<STX>20 0 09<ETX><LRC>	Transaction in progress
ECR	<STX>20 1<ETX><LRC>	Terminal status check
T	<STX>20 0 12 8 0,1,,1,Transaction Accepted<ETX><LRC>	Final transaction result - Transaction accepted – with new status 12
ECR	<STX>20 1<ETX><LRC>	Terminal status check
T	<STX>20 0 06<ETX><LRC>	Waiting for transaction finalization
ECR	<STX>22 1<ETX><LRC>	Transaction ending
T	<STX>22 0 1 0 0108 011284,,0000000000002009, *****1109,2016.06.29,17:22:42,MASTER- CARD,P,00300004, 000108,1,20000,A,0091,E5F45C14BB7BF55D,A0000000041010 ,,,,,0000048000,E800,,0,PLN,,,,,,0,,DF040140 <ETX><LRC>	Transaction data

Continued on next page

Table 40: Purchase transaction with PIN, with transaction accepted status (12) enabled. (Continued)

ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0<ETX><LRC >	Terminal is ready for new transaction

## 6.5 Purchase transaction with PIN and signature, with multilanguage printout (UPI)

Table 41: Purchase transaction with PIN and signature, with multilanguage printout (UPI).

Sender	Message	Description
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0<ETX><LRC >	Terminal ready for new transaction
ECR	<STX>21 1 1 20000<ETX><LRC>	Request for new transaction for amount of 200.00
T	<STX>21 0<ETX><LRC >	Transaction began
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 09<ETX><LRC >	Transaction in progress
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 01 1 0,1,,2,Enter card<ETX ><LRC>	Waiting for card (possible to cancel using \e sequence)
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 09<ETX><LRC >	Transaction in progress
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 2 2 0,1,,2,Enter PIN <ETX ><LRC>	Waiting for PIN (possible to cancel)
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 04<ETX><LRC >	Waiting for transaction authorization

Continued on next page

Table 41: Purchase transaction with PIN and signature, with multilanguage printout (UPI). (Continued)

ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 05 6 1,1,,3,Is sign correct?<ETX ><LRC>	Signature verification required
ECR	<STX>22 1<ETX><LRC >	Transaction data read
T	<STX>22 0 1 0 0108 abc123,,0000000000002009, *****0029,2016.07.25,16:41:37,UnionPay,P, 00300004,000121,1,20000,B,0012,E87FC4AF6FFE7B30, A000000333010101,,,,,0800048040,F800,,0,PLN,,,,,,0,, DF040140DF060D49494E3A203330393030363136 DF0B0400000001 <ETX><LRC>	Transaction data
ECR	<STX>23 1 6 \n<ETX><LRC>	Sign accepted
T	<STX>23 0<ETX><LRC >	Answer accepted
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0C 7 0,1,,1,Transaction Accepted<ETX><LRC>	Final transaction result - Transaction accepted
T	<STX>20 0 06<ETX><LRC >	Waiting for transaction finalization
ECR	<STX>22 1<ETX><LRC >	Transaction ending
T	<STX>22 0 1 0 0121 abc123,,0000000000002009, *****0029,2016.07.25,16:41:37,UnionPay,P, 00300004,000121,1,20000,B,0012,E87FC4AF6FFE7B30, A000000333010101,,,,,0800048040,F800,,0,PLN,,,,,,0,, DF040140DF060D49494E3A203330393030363136 DF0B0400000001 <ETX><LRC>	Transaction data
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0<ETX><LRC >	Terminal is ready for new transaction

## 6.6 Purchase transaction referral

Table 42: Purchase transaction referral.

Sender	Message	Description
ECR	<STX>20 1<ETX><LRC >	Terminal status check

Continued on next page

Table 42: Purchase transaction referral. (Continued)

T	<STX>20 0 0<ETX><LRC >	Terminal ready for new transaction
ECR	<STX>21 1 1 20000<ETX><LRC>	Request for new transaction for amount of 200.00
T	<STX>21 0<ETX><LRC >	Transaction began
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 1 0,1,,2,<ETX ><LRC>	Waiting for card
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 04<ETX><LRC >	Waiting for transaction authorization
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0C 123 1,1,,1,Call eService: 022 533 33 33<ETX><LRC>	Waiting for user action (\n sequence)
ECR	<STX>23 1 123 \n<ETX ><LRC>	Sending sequence \n (2 chars!)
T	<STX>23 0 <ETX><LRC >	Sequence accepted
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0B 124 4,6,A,3,Enter authorization code<ETX><LRC>	Waiting for authorization code (4 to 6 alphanumerical chars)
ECR	<STX>23 1 124 123ABC<ETX><LRC>	Authorization code (ABC123)
T	<STX>20 0 0C 126 1,2,n,2,Enter operator's code<ETX><LRC>	Waiting for operator's code (1 to 2 numerical chars)
ECR	<STX>23 1 126 47<ETX><LRC>	Operator's code (47)
T	<STX>23 0 <ETX><LRC >	Sequence accepted
ECR	<STX>20 1<ETX><LRC >	Terminal status check
ECR	<STX>22 1<ETX><LRC >	Transaction ending
T	<STX>20 0 5 125 1,1,,3,Is sign correct?<ETX><LRC>	Waiting for signature acceptance
ECR	<STX>22 1<ETX><LRC >	Transaction data read

Continued on next page

Table 42: Purchase transaction referral. (Continued)

T	<STX>22 0 0 0008 ABC123,,44444444,*****4523, 2009.04.28,16:29:24,Maestro,C,12345678, 0009,1,20000,@,,,,,011442, 75C76392A871449840AE3942BD415F45,,,1  <ETX><LRC>	Transaction data
ECR	<STX>23 1 6 \n<ETX><LRC>	Sign accepted
T	<STX>23 0<ETX><LRC >	Answer accepted
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 06<ETX><LRC >	Waiting for transaction ending
ECR	<STX>22 1<ETX><LRC >	Transaction ending
T	<STX>22 0 1 0 0008 ABC123,,44444444, *****4523,2009.04.28,16:29:24,Maestro,C, 12345678,0009,1,20000,@,,,,,011442, 75C76392A871449840AE3942BD415F45,,,1,PLN,,,,,  <ETX><LRC>	Transaction data
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0<ETX><LRC >	Terminal is ready for new transaction

## 6.7 Purchase transaction with invalid signature

Table 43: Purchase transaction with invalid signature.

Sender	Message	Description
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0<ETX><LRC >	Terminal ready for new transaction
ECR	<STX>21 1 1 20000<ETX><LRC>	Request for new transaction for amount of 200.00
T	<STX>21 0<ETX><LRC >	Transaction began
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 1 0,1,,2 <ETX><LRC>	Waiting for card
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 04<ETX><LRC >	Waiting for transaction authorization

Continued on next page

Table 43: Purchase transaction with invalid signature. (Continued)

ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 5 125 1,1,,3,Is sign correct?<ETX><LRC>	Waiting for signature acceptance
ECR	<STX>22 1<ETX><LRC >	Transaction data read
T	<STX>22 0 1 0 0122 008451,,0000000000002009, *****0078,2014.07.05,20:22:23,MASTERCARD,P, 00300004,000122,1,20000,@,0004,7BF990FF8AA6C833, A0000000041010,,,,,0000008000,E800,,0,PLN,,,,,,0,  <ETX><LRC>	Transaction data
ECR	<STX>23 1 125 \e<ETX ><LRC>	Signature invalid
T	<STX>23 0 <ETX><LRC >	Sequence accepted
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 04 128 0,0,,1,WAITING;20;FOR;20; CONNECTION<ETX><LRC>	Void in progress
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0C 130 1,2,,1,Transaction;20;canceled <ETX><LRC>	Transaction voided
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 06<ETX><LRC >	Waiting for transaction finalization
ECR	<STX>22 1<ETX><LRC >	Transaction ending
T	<STX>22 0 1 1 0122 008451,,0000000000002009, *****0078,2014.07.05,20:22:23,MASTERCARD,P, 00300004,000122,1,20000,@,0004,7BF990FF8AA6C833, A0000000041010,,,,,0000008000,E800,,0,PLN,,,,,,0,  <ETX><LRC>	Transaction data <b>NOTE:</b> Transaction result code is set to 1 - Transaction refused
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0<ETX><LRC >	Terminal is ready for new transaction

## 6.8 Purchase transaction with Erzsebet Benefit Program card

Table 44: Purchase transaction with Erzsebet Benefit Program card.

Sender	Message	Description
--------	---------	-------------

Continued on next page

Table 44: Purchase transaction with Erzsebet Benefit Program card. (Continued)

ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 8<ETX><LRC >	Terminal is busy
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0<ETX><LRC >	Terminal ready for new transaction
ECR	<STX>21 1 1 20000<ETX><LRC>	Request for new transaction for amount of 200.00
T	<STX>21 0<ETX><LRC >	Transaction began
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 01 1 0,1,,2,Enter card<ETX ><LRC>	Waiting for card (possible to cancel using \e sequence)
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 15 3 1,2,n,2,Select<20> Voucher<20> Type,Meal, Gift,School,Child<20>Protection,Cultural,Sport<ETX><LRC>	Terminal is waiting for user selection with prompt: "Select Voucher Type". <b>NOTE:</b> Transaction status is set to 15. First element is the title of the selection menu. Items are indexed from 1. Length of list may vary and depends on terminal's configuration
ECR	<STX>23 1 2 5<ETX><LRC >	Selected Voucher Type - Cultural
T	<STX>23 0 <ETX><LRC >	Sequence accepted
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 2 2 0,1,,2, <ETX><LRC>	Waiting for PIN (possible to cancel)
ECR	<STX>20 1<ETX><LRC >	Terminal status check

Continued on next page

Table 44: Purchase transaction with Erzsebet Benefit Program card. (Continued)

T	<STX>20 0 2 2 0,1,,2, <ETX><LRC>	Waiting for transaction authorization
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 2 2 0,1,,2, <ETX><LRC>	Waiting for second try PIN
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 2 2 0,1,,2, <ETX><LRC>	Waiting for transaction authorization
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 06<ETX><LRC >	Waiting for transaction finalization
ECR	<STX>22 1<ETX><LRC >	Transaction ending
T	<STX>22 0 1 0 0004 123456,,44444444, *****4123,2017.04.17,16:24:24,Erzse- bet,C,12345678,0008,1,20000,A, ,,,,011442,75C76392A871449840AE3942BD415F45,,,1,PLN,,,, 10000,DF0D1245727A73656265743A2043756C747572616C  <ETX><LRC>	Transaction data – with balance left = 100,00 and "Erzsebet: Cultural" Voucher Type (See table (), description of ARG_OUT4.34)
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0<ETX><LRC >	Terminal is ready for new transaction

## 6.9 Purchase transaction with ZenCard

Table 45: Purchase transaction with ZenCard.

Sender	Message	Description
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 8<ETX><LRC >	Terminal is busy
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0<ETX><LRC >	Terminal ready for new transaction

Continued on next page



Table 45: Purchase transaction with ZenCard. (Continued)

ECR	<STX>39 1 0 ecr_fid_supported=1<ETX><LRC>	Set Parameters – enable printing support and fid handling. Should be executed only once, before first transaction
T	<STX>39 0<ETX><LRC>	Operation Successful
ECR	<STX>20 1<ETX><LRC>	Terminal status check
T	<STX>20 0 0<ETX><LRC>	Terminal ready for new transaction
ECR	<STX>21 1 1 20000<ETX><LRC>	Request for new transaction for amount of 200.00
T	<STX>21 0<ETX><LRC>	Transaction began
ECR	<STX>20 1<ETX><LRC>	Terminal status check
T	<STX>20 0 01 1 0,1,,2,Enter card<ETX><LRC>	Waiting for card (possible to cancel using \e sequence)
ECR	<STX>20 1<ETX><LRC>	Terminal status check
T	<STX>20 0 2 2 0,1,,2,<ETX><LRC>	Waiting for PIN (possible to cancel)
ECR	<STX>20 1<ETX><LRC>	Terminal status check
T	<STX>20 0 2 2 0,1,,2,<ETX><LRC>	Waiting for transaction authorization
ECR	<STX>20 1<ETX><LRC>	Terminal status check
T	<STX>20 0 17 5 1,64,,5,Text of promotion 122A,jdie8d8djjdj;kkxoox,jjisspe00008 998744hr 000000000100 fsie83n<ETX><LRC>	Waiting for selection ZenCard discount
ECR	<STX>23 1 5 jdie8d8djjdj;jjisspe00008 000000000100 register<ETX><LRC>	ECR chose discount, and asks terminal to execute "register" flow after the transaction
T	<STX>23 0<ETX><LRC>	Sequence accepted
ECR	<STX>20 1<ETX><LRC>	Terminal status check

Continued on next page

Table 45: Purchase transaction with ZenCard. (Continued)

T	<STX>20 0 21 6 1,1,,1<ETX><LRC>	Zencard continue
ECR	<STX>22 1<ETX><LRC >	Get transaction data
T	<STX>22 0 1 0 0004 123456,,44444444, *****4123,2017.06.28,16:24:24,Master- card,C,12345678,0008,1,20000,A, ,,,011442,75C76392A871449840AE3942BD415F45 ,,,1,PLN,,,,,DF0E03313030  <ETX><LRC>	Transaction data – with discount information – 1,00
ECR	<STX>23 1 6 \n<ETX><LRC>	Sending OK
T	<STX>23 0 <ETX><LRC >	Sequence accepted
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 18 7 1,64,,2<ETX><LRC>	Waiting for fid
ECR	<STX>23 1 7 347583<ETX><LRC>	Fid entered
T	<STX>23 0 <ETX><LRC >	Sequence accepted
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 9<ETX><LRC >	In progress
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 16 9 1,1,,1,Text to be printed<ETX><LRC>	Terminal asks ECR to print some data
ECR	<STX>23 1 9 \n<ETX><LRC>	Printing finished
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 19 11 1,1,,1,param1=some text number1=45  bool1=false<ETX><LRC>	Terminal asks ECR to receive some data
ECR	<STX>23 1 11 \n<ETX><LRC>	ECR confirms receiving data
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 20 13 1,128,,1,param1 number1 bool1 <ETX><LRC>	Terminal is asking ECR to provide some data
ECR	<STX>23 1 13 param1=test number1=23 bool1=true <ETX><LRC>	ECR provides data that the terminal asked for
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 06<ETX><LRC >	Waiting for transaction finalization

Continued on next page

Table 45: Purchase transaction with ZenCard. (Continued)

ECR	<STX>22 1<ETX><LRC >	Transaction ending
T	<STX>22 0 1 0 0004 123456,,44444444, *****4123,2017.06.28,16:24:24,Mastercard,C, 12345678,0008,1,20000,A,,,,,011442, 75C76392A871449840AE3942BD415F45 ,,1,PLN,,,,,DF0E03313030 <ETX><LRC>	Transaction data – with discount information – 1,00
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0<ETX><LRC >	Terminal is ready for new transaction

## 6.10 Purchase transaction with Fleet Card

Table 46: Purchase transaction with Fleet Card.

Sender	Message	Description
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 8<ETX><LRC >	Terminal is busy
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0<ETX><LRC >	Terminal ready for new transaction
ECR	<STX>21 1 1 13900   <b>7F01</b> 1617F012FDF02023737DF030531 30363030DF04024541DF0505313036 3030DF060131DF07014EDF010A3135 39333833353532307F012CDF020238 33DF030433333030DF04024541DF05 0433333030DF060131DF07014EDF01 09313239373734303832 <ETX><LRC>	Request for new transaction for amount of 139,00 <b>7F01</b> – Sale Item List
T	<STX>21 0<ETX><LRC >	Transaction began
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 1 1 0,1,,2, <ETX ><LRC>	Waiting for card (possible to cancel using \e sequence)
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 7<ETX><LRC >	Waiting for card removal
ECR	<STX>20 1<ETX><LRC >	Terminal status check

Continued on next page

Table 46: Purchase transaction with Fleet Card. (Continued)

T	<STX>20 0 2 0,1,,2, <ETX><LRC>	Waiting for PIN (possible to cancel)
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 01 3 0,12,S,3,Podaj przebieg samochodu <ETX><LRC>	Waiting for fleet card additional info (possible to cancel)
ECR	<STX>23 1 3 125000<ETX><LRC>	Fleet card additional info (125.000)
T	<STX>23 0 <ETX><LRC >	Sequence accepted
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 4<ETX><LRC >	Waiting for transaction authorization
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 6<ETX><LRC >	Waiting for transaction finalization
ECR	<STX>22 1<ETX><LRC >	Transaction ending
T	<STX>22 0 1 0 75 123456,,0000000000002009, 636200*****1785,2019.05.10,13:15:07,UTA,C, 45450139,000075,1,13900,A,,,,,,,,,1,PLN,,,,,,,,, DF0B0400000000 <b>7F0144DF010103DF02</b> 34313539333833353532303D42656E7A 796E6120506239357C33383632393937 3235313D532F484F5420444F47205041 524F574B41 <b>DF05</b> 06313235303030<ETX><LRC>	Transaction data 7F01 – Fleet Card data: DF01 – Fleet card flags: Indicator: set Signature: set DF02 - Product names DF05 – Mileage
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0<ETX><LRC >	Terminal is ready for new transaction

## 6.11 Purchase transaction with BLIK code given

Table 47: Purchase transaction with BLIK code given.

Sender	Message	Description
ECR	<STX>20 1<ETX><LRC >	Terminal status check

Continued on next page

Table 47: Purchase transaction with BLIK code given. (Continued)

T	<STX>20 0 8<ETX><LRC >	Terminal is busy
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0<ETX><LRC >	Terminal ready for new transaction
ECR	<STX>21 1 1 1000  DF0206313233343536 <ETX><LRC>	Request for new transaction for amount of 10,00 with given BLIK code "123456"
T	<STX>21 0<ETX><LRC >	Transaction began
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 4<ETX><LRC >	Waiting for transaction authorization
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 6<ETX><LRC >	Waiting for transaction finalization
ECR	<STX>22 1<ETX><LRC >	Transaction ending
T	<STX>22 0 1 0 8 341431,,44444444, *****0001,2019.02.05,14:31:33,BLIK,C,12345678, 000008,1,1000,?,,,,,,,,,,0,PLN,,,,,,,,, DF0111313233343536373839383736 3534333231DF0206313233343536DF 0C1035393924242424242424242424303031 <ETX><LRC>	Transaction data
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0<ETX><LRC >	Terminal is ready for new transaction

## 6.12 Purchase transaction with BLIK code given and PIN

Table 48: Purchase transaction with BLIK code given and PIN.

Sender	Message	Description
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 8<ETX><LRC >	Terminal is busy

Continued on next page

Table 48: Purchase transaction with BLIK code given and PIN. (Continued)

ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0<ETX><LRC >	Terminal ready for new transaction
ECR	<STX>21 1 1 1000   <b>DF02</b> 09313233343536373839<ETX><LRC >	Request for new transaction for amount of 10,00 with given BLIK code "123456789"
T	<STX>21 0<ETX><LRC >	Transaction began
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 2 2 0,1,,2, <ETX><LRC>	Waiting for PIN (possible to cancel)
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 4<ETX><LRC >	Waiting for transaction authorization
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 6<ETX><LRC >	Waiting for transaction finalization
ECR	<STX>22 1<ETX><LRC >	Transaction ending
T	<STX>22 0 1 0 0004 123456,,44444444, *****0001,2019.02.05,13:46:38,BLIK,C,12345678, 000006,1,1000,A,,,,,,,,,0,PLN,,,,,,,,, DF0111313233343536373839383736 3534333231DF02092A2A2A34353637 3839DF0C10353939242424242424242424303031 <ETX><LRC>	Transaction data
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0<ETX><LRC >	Terminal is ready for new transaction

## 6.13 MultiSplit Payment Transaction

Table 49: MultiSplit Payment Transaction.

Sender	Message	Description
--------	---------	-------------

Continued on next page

Table 49: MultiSplit Payment Transaction. (Continued)

ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 8<ETX><LRC >	Terminal is busy
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0<ETX><LRC >	Terminal ready for new transaction
ECR	<STX>21 1 1 2000   7F02567F0122DF01023730DF02023630DF030431303030DF0408446F6E6174696F6EDF05032C2C2C7F012EDF01023731DF02023030DF030431303030DF040D4F74686572206163636F756E74DF050A2C5469746C6520322C2C<ETX><LRC>	Request for new transaction for amount of 20.00 with two MSP account: Donation – 10.00 Other account – 10.00
T	<STX>21 0<ETX><LRC >	Transaction began
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 1 1 0,1,,2, <ETX><LRC>	Waiting for card (possible to cancel using \e sequence)
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 7<ETX><LRC >	Waiting for card removal
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 2 2 0,1,,2, <ETX><LRC>	Waiting for PIN (possible to cancel)
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 4<ETX><LRC >	Waiting for transaction authorization
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 6<ETX><LRC >	Waiting for transaction finalization
ECR	<STX>22 1<ETX><LRC >	Transaction ending

Continued on next page

Table 49: MultiSplit Payment Transaction. (Continued)

T	<STX>22 0 1 0 135 00828Z,,0000000000002009, *****3553,2020.06.18,20:01:57,MASTERCARD,C, 00163367,000135,1,2000,A,,,,,,,,,1,PLN,,,,,,,,, DF0B0400000400DF1C20446F6E6174 696F6E3D313030307C4F7468657220 6163636F756E743D31303030<ETX><LRC>	Transaction data DF1C – MSP data: Donation=1000  Other account=1000
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0<ETX><LRC >	Terminal is ready for new transaction

## 6.14 Partial DCC refund

Table 50: Partial DCC refund.

Sender	Message	Description
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0<ETX><LRC >	Terminal ready for new transaction
ECR	<STX>21 1 6 1000 <ETX><LRC>	Request for new transaction for amount 10.00
T	<STX>21 0<ETX><LRC >	Transaction began
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0C 1 1,1,,3,Change<20>currency?<ETX><LRC>	Ask for DCC
ECR	<STX>23 1 1 \n <ETX><LRC>	DCC Confirmation
T	<STX>23 0<ETX><LRC >	Sequence accepted
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0E 3 1,2,n,2,EUR,USD,GBP,NOK,SEK,LTL,CZK, DKK,RUB,LVL,CAD,CHF,UAH,BYR,HUF,ISK,ILS,JPY, KRW,RON,TRY<ETX><LRC>	Currency list – terminal is waiting for currency index, starting from 1 <b>NOTE:</b> Transaction status is set to 0E
ECR	<STX>23 1 3 2<ETX><LRC >	Chosen currency at index 2 - USD
T	<STX>23 0<ETX><LRC >	Sequence accepted
ECR	<STX>20 1<ETX><LRC >	Terminal status check

Continued on next page



Table 50: Partial DCC refund. (Continued)

T	<STX>20 0 0C 5 1,10,S,,Enter<20>exchange<20>rate<20><ETX><LRC>	Waiting for exchange rate <b>NOTE:</b> allowed characters are set to S. Expected exchange rate is dot separated
ECR	<STX>23 1 5 1.2<ETX><LRC>	Rate – 1.2
T	<STX>23 0<ETX><LRC>	Sequence accepted
ECR	<STX>20 1<ETX><LRC>	Terminal status check
T	<STX>20 0 0C 7 1,13,n,,Enter<20>amount<20>in<20>card<20>currency<ETX><LRC>	Request for amount in dcc currency
ECR	<STX>23 1 7 1200<ETX><LRC>	Amount in dcc currency – 12.00
T	<STX>23 0<ETX><LRC>	Sequence accepted
ECR	<STX>20 1<ETX><LRC>	Terminal status check
T	<STX>20 0 0C 8 1,1,,3,Refund<20>of<20>the<20>full<20>amount?<ETX><LRC>	Ask for refund for whole amount
ECR	<STX>23 1 8 \e<ETX><LRC>	Partial refund chosen
T	<STX>23 0<ETX><LRC>	Sequence accepted
ECR	<STX>20 1<ETX><LRC>	Terminal status check
T	<STX>20 0 0C 10 1,13,n,,Enter<20>amount<20>in<20>terminal<20>currency<ETX><LRC>	Request for amount in terminal currency
ECR	<STX>23 1 10 800<ETX><LRC>	Amount in terminal currency – 8.00
T	<STX>23 0<ETX><LRC>	Sequence accepted
ECR	<STX>20 1<ETX><LRC>	Terminal status check
T	<STX>20 0 01 15 0,2,,2,Insert<20>card<ETX><LRC>	Card entry request
ECR	<STX>20 1<ETX><LRC>	Terminal status check
T	<STX>20 0 05 17 1,1,,3,Is<20>sign<20>correct?<ETX><LRC>	Is signature correct?
ECR	<STX>23 1 17 \n<ETX><LRC>	Signature correct
T	<STX>23 0<ETX><LRC>	Sequence accepted
ECR	<STX>20 1<ETX><LRC>	Terminal status check
T	<STX>20 0 0C 50 0,0,,1,Refund<20>transaction<20>ACCEPTED<ETX><LRC>	Authorization status

Continued on next page

Table 50: Partial DCC refund. (Continued)

ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 6<ETX><LRC >	Waiting for transaction finalization
ECR	<STX>22 1<ETX><LRC >	Transaction ending
T	<STX>22 0 1 0 0038 ,,0000000000002009, *****3561,2014.08.12,15:09:46,MASTER- CARD,C,00300004, 000038,3,960,@,,,,,,,,,0,USD,PLN,800,1.200000,*<20>THIS <20>CURRENCY<20>CONVERSION<20> >IS<20>PROVIDED<20>BY <20>eService,,0,2<ETX><LRC >	Transaction data <b>NOTE:</b> Transaction amount is set to 8PLN converted to 9.60USD
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0<ETX><LRC >	Terminal is ready for new transaction

## 6.15 DCC refund with waiting for amount status

Table 51: DCC Refund with waiting for amount status.

Sender	Message	Description
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0<ETX><LRC >	Terminal ready for new transaction
ECR	<STX>21 1 6 1000 <ETX><LRC>	Request for new transaction for amount 10.00
T	<STX>21 0<ETX><LRC >	Transaction began
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0C 1 1,1,,3,Change<20>currency?<ETX><LRC>	Ask for DCC
ECR	<STX>23 1 1 \n <ETX><LRC>	DCC Confirmation
T	<STX>23 0<ETX><LRC >	Sequence accepted
ECR	<STX>20 1<ETX><LRC >	Terminal status check

Continued on next page

Table 51: DCC Refund with waiting for amount status. (Continued)

T	<STX>20 0 0E 3 1,2,n,2,EUR,USD,GBP,NOK,SEK,LTŁ,CZK,DKK,RUB,LVL,CAD,CHF,UAH,BYR,HUF,ISK,ILS,JPY,KRW,RON,TRY<ETX><LRC>	Currency list – terminal is waiting for currency index, starting from 1 <b>NOTE:</b> Transaction status is set to 0E
ECR	<STX>23 1 25 1<ETX><LRC >	Chosen currency at index 1 - EUR
T	<STX>23 0<ETX><LRC >	Sequence accepted
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0C 27 1,10,S,,Enter<20>exchange<20>rate<20><ETX><LRC>	Waiting for exchange rate <b>NOTE:</b> allowed characters are set to S. Expected exchange rate is dot separated
ECR	<STX>23 1 27 1<ETX><LRC >	Rate – 1
T	<STX>23 0<ETX><LRC >	Sequence accepted
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 14 29 1,13,n,,Enter<20>amount<20>in<20>card<20>currency<ETX><LRC >	Request for amount in dcc currency
ECR	<STX>23 1 29 1000<ETX><LRC >	Amount in dcc currency – 10.00
T	<STX>23 0<ETX><LRC >	Sequence accepted
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0C 30 1,1,,3,Refund<20>of<20>the<20>full<20>>amount?<ETX><LRC>	Ask for refund for whole amount
ECR	<STX>23 1 30 \n<ETX><LRC>	Full refund chosen
T	<STX>23 0<ETX><LRC >	Sequence accepted
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 09  <ETX><LRC >	Transaction in progress
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 01 15 0,2,,2,Insert<20 >card<ETX><LRC>	Card entry request
ECR	<STX>20 1<ETX><LRC >	Terminal status check

Continued on next page

Table 51: DCC Refund with waiting for amount status. (Continued)

T	<STX>20 0 05 31 1,1,,3,Is<20>sign<20>correct? <ETX><LRC >	Is signature correct?
ECR	<STX>23 1 31 \n<ETX ><LRC>	Signature correct
T	<STX>23 0<ETX><LRC >	Sequence accepted
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0C 50 0,0,,1,Refund <20>transaction<20>ACCEPTED<ETX><LRC>	Authorization status
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 6<ETX><LRC >	Waiting for transaction finalization
ECR	<STX>22 1<ETX><LRC >	Transaction ending
T	<STX>22 0 6 0 25 ,,454500000002017, 541\$\$\$\$\$\$\$\$\$640,2017.04.14,09:00:35,MASTER- CARD,P,20170003, 000025,3,1000,@,0185,C8532EEA5645A360,A0000000041010,,,, 0000000000,A000,,0,EUR,PLN,1000,1.000000,*<20>THIS <20>CURRENCY<20>CONVERSION<20 >IS<20>PROVIDED<20>BY <20 >eService,,,2, DF040100DF0C10353431242424242424242424242424363430 <ETX><LRC>	Transaction data
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0<ETX><LRC >	Terminal is ready for new transaction

## 6.16 Reversal transaction

Table 52: Reversal transaction.

Sender	Message	Description
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0<ETX><LRC >	Terminal ready for new transaction
ECR	<STX>21 1 10 2000 82<ETX><LRC>	Request for reversal transaction for amount 20.00 and sequence number 82
T	<STX>21 0<ETX><LRC >	Transaction began

Continued on next page

Table 52: Reversal transaction. (Continued)

ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0C 11 0,1,,1,Transaction<20>reversed<ETX><LRC>	Reversal status
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 6<ETX><LRC >	Waiting for transaction end
ECR	<STX>22 1<ETX><LRC >	Transaction ending
T	<STX>22 0 10 0 0083 010467,,0000000000002009,*****0018,2017.02.14,12:13:44,MAESTRO,P,45450009,000083,1,2000,?,0578,0FFE0EF602024377,A0000000043060,,,,,0000008000,E800,,0,PLN,,,,,,0,,DF040140<ETX><LRC>	Transaction data
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0<ETX><LRC >	Terminal is ready for new transaction

## 6.17 Reversal transaction (extended transaction type)

Table 53: Reversal transaction (extended transaction type).

Sender	Message	Description
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0<ETX><LRC >	Terminal ready for new transaction
ECR	<STX>21 1 10 2000 82<ETX><LRC>	Request for reversal transaction for amount 20.00 and sequence number 82
T	<STX>21 0<ETX><LRC >	Transaction began
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0C 11 0,1,,1,Transaction<20>reversed<ETX><LRC>	Reversal status
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 6<ETX><LRC >	Waiting for transaction end
ECR	<STX>22 1<ETX><LRC >	Transaction ending

Continued on next page

Table 53: Reversal transaction (extended transaction type). (Continued)

T	<STX>22 0 129 0 0083 010467,,0000000000002009, *****0018,2017.02.14,12:13:44,MAESTRO,P, 45450009,000083,1,2000,?,0578,0FFE0EF602024377, A0000000043060,,,,,0000008000,E800,,0,PLN,,,,,, 0,,DF040140DF0B0400000002<ETX><LRC>	Transaction data. <b>Note:</b> new status for transaction is 129 which means: 128 – reversal 01 – original transaction type (purchase) Additional flags : DF0B0400000002 – manual void
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0<ETX><LRC >	Terminal is ready for new transaction

## 6.18 Giftcard's activation (accepted)

Table 54: Giftcard's activation (accepted).

Sender	Message	Description
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0<ETX><LRC >	Terminal ready for new transaction
ECR	<STX>21 1 42<ETX><LRC>	Request Giftcard's Activation
T	<STX>21 0<ETX><LRC >	Transaction began
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0A 9 1,1,,3,Print copy? <ETX ><LRC>	Question about printing a copy
ECR	<STX>23 1 9 \e<ETX><LRC>	Answer "no"
T	<STX>23 0<ETX><LRC >	Transaction data
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 06  <ETX ><LRC>	Waiting for transaction finalization
ECR	<STX>22 1<ETX><LRC >	Transaction ending

Continued on next page

Table 54: Giftcard's activation (accepted). (Continued)

T	<STX>22 0 42 0 0191 123456,00,0000000000002009, *****0917,2015.11.27,15:18:22,GIFTCARD,C, 00300001,000191,1,0,?,,,,,,,,,,0,PLN,,,,,,,,0,, DF05022006 <ETX><LRC>	Transaction end, card activated
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0<ETX><LRC >	Terminal is ready for new transaction

## 6.19 Giftcard's activation (rejected)

Table 55: Giftcard's activation (rejected).

Sender	Message	Description
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0<ETX><LRC >	Terminal ready for new transaction
ECR	<STX>21 1 42<ETX><LRC>	Request Giftcard's Activation
T	<STX>21 0<ETX><LRC >	Transaction began
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0A 9 1,1,,3,Print copy? <ETX ><LRC>	Question about printing a copy
ECR	<STX>23 1 9 \e<ETX><LRC>	Answer "no"
T	<STX>23 0<ETX><LRC >	Transaction data
ECR	<STX>22 1<ETX><LRC >	Transaction ending
T	<STX>22 0 42 1 0078 ,12,0000000000002009, *****0891,2015.11.24,17:50:47,GIFTCARD,C, 00300001,000078,1,0,?,,,,,,,,,,0,DF05022006 <ETX><LRC>	Transaction end, response from terminal
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0<ETX><LRC >	Terminal is ready for new transaction

## 6.20 Giftcard's balance check

Table 56: Giftcard's balance check.

Sender	Message	Description
--------	---------	-------------

Continued on next page

Table 56: Giftcard's balance check. (Continued)

ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0<ETX><LRC >	Terminal ready for new transaction
ECR	<STX>21 1 43<ETX><LRC>	Request Giftcard's Balance Check
T	<STX>21 0<ETX><LRC >	Transaction began
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0A 18 1,1,,3,Print copy? <ETX ><LRC>	Question about printing a copy
ECR	<STX>23 1 18 \n<ETX><LRC>	Answer "yes"
T	<STX>23 0<ETX><LRC >	Answer accepted
ECR	<STX>22 1<ETX><LRC >	Transaction ending
T	<STX>22 0 43 0 0163 161000,00,0000000000002009,*****0891,2015.11.26,10:31:55,GIFTCARD,C,00300001,000163,1,102125,?,,,,,,,,,,0,PLN,,,,,,,, 0,,DF05021610<ETX><LRC>	Transaction end, response from terminal (balance is 1021,25 PLN)
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0<ETX><LRC >	Terminal is ready for new transaction

## 6.21 Giftcard's topup

Table 57: Giftcard's topup.

Sender	Message	Description
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0<ETX><LRC >	Terminal ready for new transaction
ECR	<STX>21 1 44<ETX><LRC>	Request giftcard topup
T	<STX>21 0<ETX><LRC >	Transaction began
ECR	<STX>20 1<ETX><LRC >	Terminal status check

Continued on next page



Table 57: Giftcard's topup. (Continued)

T	<STX>20 0 0C 28 1,1,,3,Balance: 1 000,81 PLN. Should transaction be continued?<ETX><LRC>	Card balance returned from terminal, question about the will of continuation
ECR	<STX>23 1 28 \n<ETX><LRC>	Affirmative answer
T	<STX>23 0<ETX><LRC >	Answer accepted
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0A 36 1,1,,3,Print copy?i<ETX ><LRC>	Question about printing a copy
ECR	<STX>23 1 28 \n<ETX><LRC>	Affirmative answer
T	<STX>23 0<ETX><LRC >	Answer accepted
ECR	<STX>22 1<ETX><LRC >	Transaction ending
T	<STX>22 0 44 0 0081 161000,00,0000000000002009,*****0891,2015.11.24,18:03:28,GIFTCARD,C,00300001,000081,1,1022,?,,,,,,,,,,0,DF05021610 <ETX><LRC>	Transaction end, response from terminal
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0<ETX><LRC >	Terminal is ready for new transaction

## 6.22 Execute function - Zencard flow

Table 58: Execute function - ZenCard flow.

Sender	Message	Description
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 8<ETX><LRC >	Terminal is busy
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0<ETX><LRC >	Terminal ready for new transaction
ECR	<STX>21 1 85 0 flowECR17<ETX><LRC>	Request for executing flowECRF17
T	<STX>21 0<ETX><LRC >	Flow began
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 9<ETX ><LRC>	In progress

Continued on next page

Table 58: Execute function - ZenCard flow. (Continued)

ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 16 1 1,1,,1,<h1><b>Letnia promocja!</b></h1> <u>Uruchomiono flow. </u> <center>Flow ECRF17.</center> <ETX><LRC>	Terminal asks ECR to print some data
ECR	<STX>23 1 1 \n<ETX><LRC>	Printing finished
T	<STX>23 0<ETX><LRC >	Sequence accepted
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 06<ETX><LRC >	Waiting for transaction finalization
ECR	<STX>22 1<ETX><LRC >	Transaction ending
T	<STX>22 0 85 0   ,,,,,,,000000,?,0,?, ,,,,,,,0, ,,,,,,, <ETX><LRC>	Executing result
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0<ETX><LRC >	Terminal is ready for new transaction

## 6.23 Incremental authorization

Table 59: Incremental authorization.

Sender	Message	Description
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 8<ETX><LRC >	Terminal is busy
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0<ETX><LRC >	Terminal ready for new transaction
ECR	<STX>21 1 86 005802 1000<ETX><LRC>	Incremental authorization request, authorization code of original preauthorization – 005802, incremental amount - 1000
T	<STX>21 0<ETX><LRC >	Transaction began

Continued on next page

Table 59: Incremental authorization. (Continued)

ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 9<ETX ><LRC>	In progress
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0C 38 0,1,,1,TRANSAKCJA ZAAKCEPTOWANA<ETX><LRC>	Transaction accepted
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 06<ETX><LRC >	Waiting for transaction finalization
ECR	<STX>22 1<ETX><LRC >	Transaction ending
T	<STX>22 0 86 0 305 005900,,0000000000002009, *****0069,2017.08.22,15:33:21,MASTERCARD,M, 00199991,000302,1,290,@,,,,,,,,,1,USD,PLN,1000, 0.2900000,* THIS CURRENCY CONVERSION IS PROVIDED BY eService,MAKE SURE YOU UNDERSTAND\n THE COSTS OF CURRENCY CONVERSION\n AS THEY MAY BE DIFFERENT DEPENDING\n ON WHETHER YOU SELECT YOUR HOME \n CURRENCY OR THE TRANSACTION CURRENCY,,,2, DF0610534E204E72204D4153544552 43415244DF0C103534312424242424 2424242424303639DF0F0436363730 DF10053233303030 <ETX><LRC>	Transaction data
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0<ETX><LRC >	Terminal is ready for new transaction

## 6.24 Preauthorization cancel

Table 60: Preauthorization cancel.

Sender	Message	Description
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 8<ETX><LRC >	Terminal is busy
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0<ETX><LRC >	Terminal ready for new transaction

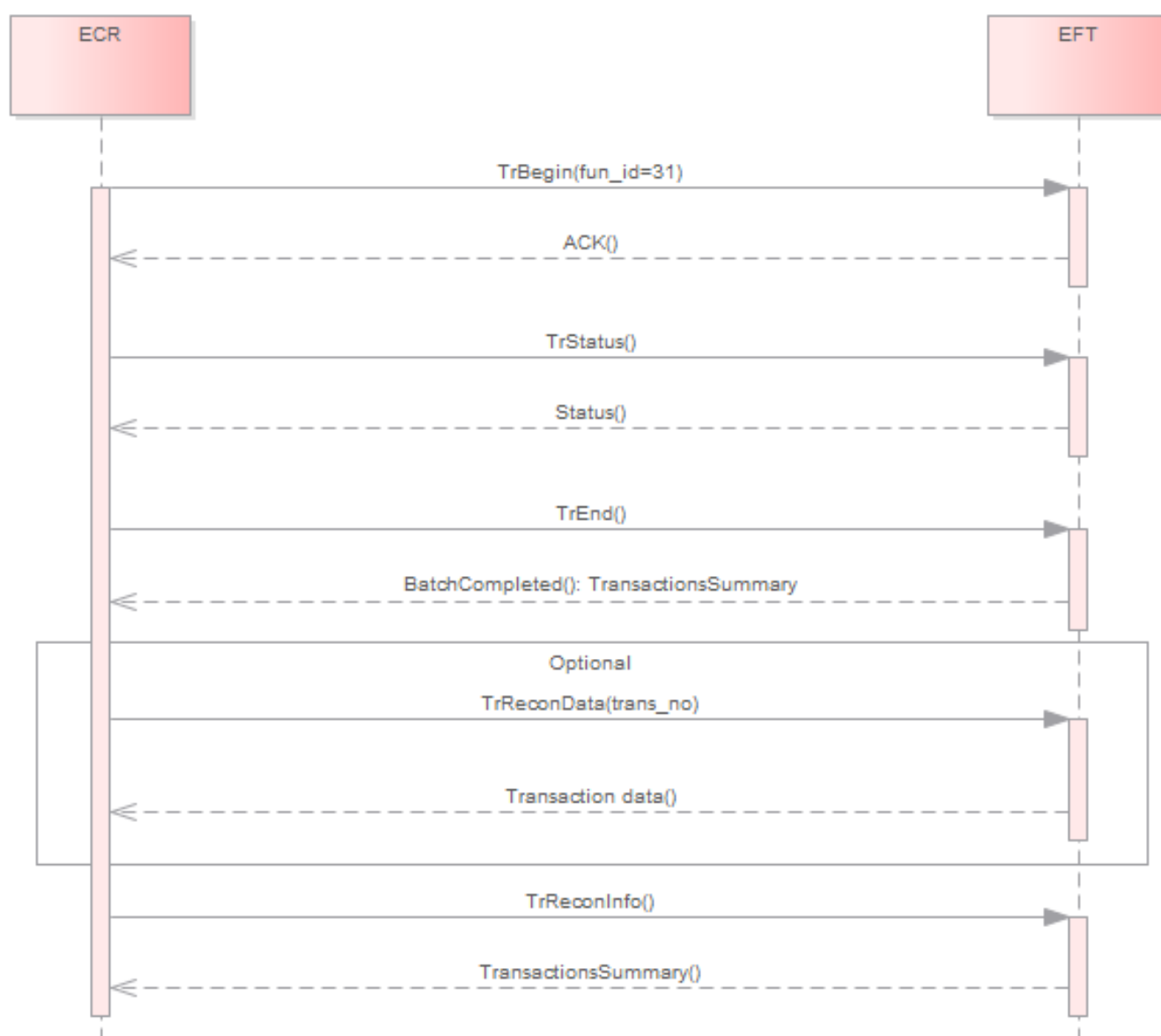
Continued on next page

Table 60: Preauthorization cancel. (Continued)

ECR	<STX>21 1 82 00947Z 1000<ETX><LRC>	Preauthorization cancel request, authorization code of original preauthorization – 00947Z, original preauthorization amount - 1000
T	<STX>21 0<ETX><LRC >	Transaction began
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 9<ETX ><LRC>	In progress
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0C 38 0,1,,1,TRANSAKCJA ZAAKCEPTOWANA<ETX><LRC>	Transaction accepted
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 06<ETX><LRC >	Waiting for transaction finalization
ECR	<STX>22 1<ETX><LRC >	Transaction ending
T	<STX>22 0 82 0 45 00947Z,,0000000000002009,*****3553,2019.08.29,09:20:52,MASTERCARD,C,45450139,000045,1,1000,?,,,,,,,,,,1,PLN,,,,,,,,,DF0B0400000100<ETX><LRC>	Transaction data
ECR	<STX>20 1<ETX><LRC >	Terminal status check
T	<STX>20 0 0<ETX><LRC >	Terminal is ready for new transaction

## 7 Reconciliation handling related commands

When the terminal will perform a reconciliation (batch), its data will be available after a few seconds and terminal will report BatchCompleted state. It is possible that BatchCompleted state will appear only at specified time even if batch were sent earlier. Batch summary is available even if no payment transactions were done. It is possible to have more than one summary on terminal. Each summary could contain transactions. If two transactions read from single batch have the same sequence number, it means that transaction has been reversed and shall not be included in totals. Diagram below explains how to read batch data.



### 7.1 Single transaction read from batch

This message is being sent to terminal just after batch was sent to acquirer (BatchCompleted status is set). Using this message cash register is able to read transaction data which were sent to acquirer. Main input parameter is an index of transaction stored on the terminal. When a transaction with specified index does not exist, the terminal returns empty record with return code equal to 'Requested data not available'. First transaction index is always 1. Each transaction is available until batch summary will be read.

## Request:

Table 61: Single transaction read from batch request description.

Field	Field length	Field value	Description
FUN_ID	2	"24"	Function identifier – TrReconData
ARG_IN1	1	e.g. "1"	Cash register number
ARG_IN2	1-8	e.g."1"	Index of desired transaction, counting from 1

## Response:

A response has the same format as regular response for Transaction End command (Table).

Table 62: Single transaction read from batch response description.

Field	Field length	Field value	Description
FUN_ID	2	"24"	Function identifier – TrReconData
FUN_RESULT	1-2	e.g. "0"	Function return code, see table ( <a href="#">Table 1</a> )
ARG_OUT1	1-3	e.g."1"	Transaction type, see section ( <a href="#">Section 4.5</a> )
ARG_OUT2	1-2	e.g."0"	Transaction result code, see table ( <a href="#">Table 34</a> )
ARG_OUT3	0-8	e.g."1234"	Transaction number
ARG_OUT4	0-2048	var	Transaction details, see table ( <a href="#">Table 28</a> )

## 7.2 Batch summary read

This message is being sent to terminal just after batch was sent, batch complete status is set and cash register had read transmitted transactions data. After sending this message, cash register is no longer able to read batch summary data or single transaction data from given batch – they will be removed from the terminal. A response contains basic information about transmitted batch.

## Request:

Table 63: Batch summary read request description.

Field	Field length	Field value	Description
FUN_ID	2	"25"	Function identifier – TrReconInfo
ARG_IN1	1	e.g. "1"	Cash register number
ARG_IN2	0-2	e.g. "1"	<p>Batch summary read flags</p> <p>Possible flags:</p> <p>Byte 1:</p> <p>0x01 – repeat last TrReconInfo response. Used when ECR didn't receive previous frame correctly, ie file CRC was incorrect.</p> <p>0x02 – send file with batch transaction.</p> <p>It can be used instead of sending every transaction with TrReconData.</p> <p>0x04 – multicurrency report</p> <p>In this case terminal will generate report data grouped by currencies Used when terminal supports multicurrency and ECR requires currency report.</p> <p>Example:</p> <p>"3"</p> <p>denotes:</p> <ol style="list-style-type: none"> <li>1. Repeat last TrReconInfo</li> <li>2. Send file with batch transaction.</li> </ol>

**Response:**

Table 64: Batch summary read response description.

Field	Field length	Field Value	Description
FUN_ID	2	"25"	Function identifier – TrReconInfo
FUN_RESULT	1-2	e.g. "0"	Function return code, see table ( <a href="#">Table 1</a> )
ARG_OUT1	1-12	e.g. "10000"	Total amount counted as sum of debit transactions minus sum of credit transactions, in minor units.
ARG_OUT2	1-12	e.g. "10"	Batch transaction count
ARG_OUT3	1-2	e.g. "0"	Batch status Possible values: <ul style="list-style-type: none"> <li>• 0 – success</li> <li>• 1 – error</li> <li>• 2 – success, but not completed preauthorizations were present in batch – unattended terminals only</li> <li>• 4 – success, but TMS connection is required</li> </ul>
ARG_OUT4	1-16	e.g. "13244534"	Acquirer identifier
ARG_OUT5	1-16	e.g. "1234"	Batch number
ARG_OUT6	0-10	e.g. "2016.04.25"	date (format: YYYY.MM.DD)
ARG_OUT7	0-8	e.g. "22:43:25"	time (format: hh:mm:ss)
ARG_OUT8	1-2	e.g. "0"	Terminal printing indicator (value not equal 0 means that printout was made by terminal)
ARG_OUT9	1-8	e.g. "00300004"	Terminal identifier (TID)
ARG_OUT10	1-12	e.g. "123"	Activation sequence number
ARG_OUT11	1-2	e.g. "1"	Cashier identifier
ARG_OUT12	0-1024		eVoucher module dayclose information, see Appendix A
ARG_OUT13	0-1024		Giftcard module dayclose information, see ( <a href="#">Section 7.2.2</a> )
ARG_OUT14	var	e.g. "1421"	File size
ARG_OUT15	4	e.g. "A45C"	File CRC
ARG_OUT16	0-1024		Payment dayclose information, see ( <a href="#">Section 7.2.1</a> )
...	...	...	...
Additional data	var		File data – batch transaction list. See ( <a href="#">Section 7.2.3</a> ) for details.



## 7.2.1 Payment dayclose information

This information depends on ARG\_IN2 flags:

- If flag 0x04 (multicurrency report) is disabled – single currency content will be generated (see (7.2.1.1) for details)
- If flag 0x04 (multicurrency report) is enabled – multicurrency content will be generated (see (7.2.1.2) for details)

### 7.2.1.1 Payment single currency dayclose information

Field is empty.

**Example:**

<STX>25|0|15000|2|0||96|2020.03.10|13:08:44| 1|45450139|||||<ETX><LRC>

### 7.2.1.2 Payment multicurrency dayclose information

Contains batch summary (ARG\_OUT1 and ARG\_OUT2) grouped by the transaction currencies.

Table 65: Content of payment multicurrency dayclose information.

Subfield	Subfield length	Subfield value	Description
ARG_OUT16.1	0-40	e.g. "PLN 10 25943"	Single entry containing following items (separated with ' '):  1. Currency to print  2. Total amount counted as sum of debit transactions minus sum of credit transactions, in minor units (similar as ARG_OUT1 field)  3. Batch transaction count (similar as ARG_OUT2 field)
SS	1	","	Subfield separator
ARG_OUT16.2	0-40	e.g. "EUR 3 2000"	Next entry
...	...	...	...

**Example:**

<STX>25|0|15000|2|0||41|2020.08.05|14:13:39| 1|45450139|||||PLN|1|10000;EUR|1|5000<ETX>

## 7.2.2 Giftcard module dayclose information

In case of any giftcard transaction there will be additional information about its dayclose. This information depends on ARG\_IN2 flags:

- If flag 0x04 (multicurrency report) is disabled – single currency content will be generated (see (7.2.2.1) for details)
- If flag 0x04 (multicurrency report) is enabled – multicurrency content will be generated (see (7.2.2.2) for details)

### 7.2.2.1 Giftcard single currency dayclose information

Table 66: Content of giftcard single currency dayclose information.

Subfield	Subfield length	Subfield value	Description
ARG_OUT13.1	0-1024	e.g. "14044 4 10000 1 2400 2 0"	Giftcard dayclose information entry, see (7.2.2.3)

**Example:**

<STX>25|0|2400|14|0||135|2015.11.23|11:08:50| 0|00300001||||14044|4|10000|1|2400|2|0<ETX>

### 7.2.2.2 Giftcard multicurrency dayclose information

Contains giftcard module dayclose information grouped by the transaction currencies.

Table 67: Content of giftcard multicurrency dayclose information.

Subfield	Subfield length	Subfield value	Description
ARG_OUT13.1	0-1024	e.g. "PLN 14044 4 10000 1 2400 2 0"	Single giftcard entry containing following items (separated with ' '):  1. Currency to print  2. Giftcard dayclose common information, see (7.2.2.3)
SS	1	","	Subfield separator
ARG_OUT13.2	0-1024	e.g. "EUR 14044 4 10000 1 2400 2 0"	Next entry
...	...	...	...

**Example:**

<STX>25|0|2400|14|0||135|2015.11.23|11:08:50| 0|00300001||||PLN|14044|4|10000|1|2400|2|0;  
CZK|14044|4|10000|1|2400|2|0||PLN|1|2000;CZK|1|400 <ETX >

### 7.2.2.3 Giftcard dayclose common information

Giftcard dayclose common information structure: A|B|C|D|E|F|G

A – total amount of giftcard purchase transactions

B – number of giftcard purchase transactions

C – total amount of reversed giftcard purchase transactions

D – number of such reversals

E – total amount of giftcard topup transactions

F – number of such topups

G – number of giftcard activations

**Example:**

14044|4|10000|1|2400|2|0

## 7.2.3 Additional data - file with batch transaction list

This file contains list of transaction from according batch. Each line contains information about a single transaction. It has the same format as regular response for Transaction End command ([Table 27](#)). **Example of file data:**

```
25|0|1|0|10|00770Z,,000000000002009,*****3553,2020.04.03,09:05:25 ,MASTERCARD,C,45450139,000010,1,1000,A,,,,,,,,,1,PLN
,,,,,,,,,DF0B0400000400
25|0|1|0|11|00771Z,,000000000002009,*****3553,2020.04.03,09:05:44 ,MASTERCARD,C,45450139,000011,1,1000,A,,,,,,,,,1,PLN
,,,,,,,,,DF0B0400000400
25|0|1|0|12|00771Z,,000000000002009,*****3553,2020.04.03,09:06:09 ,MASTERCARD,C,45450139,000012,1,1000,A,,,,,,,,,1,PLN
,,,,,,,,,DF0B0400000400
```

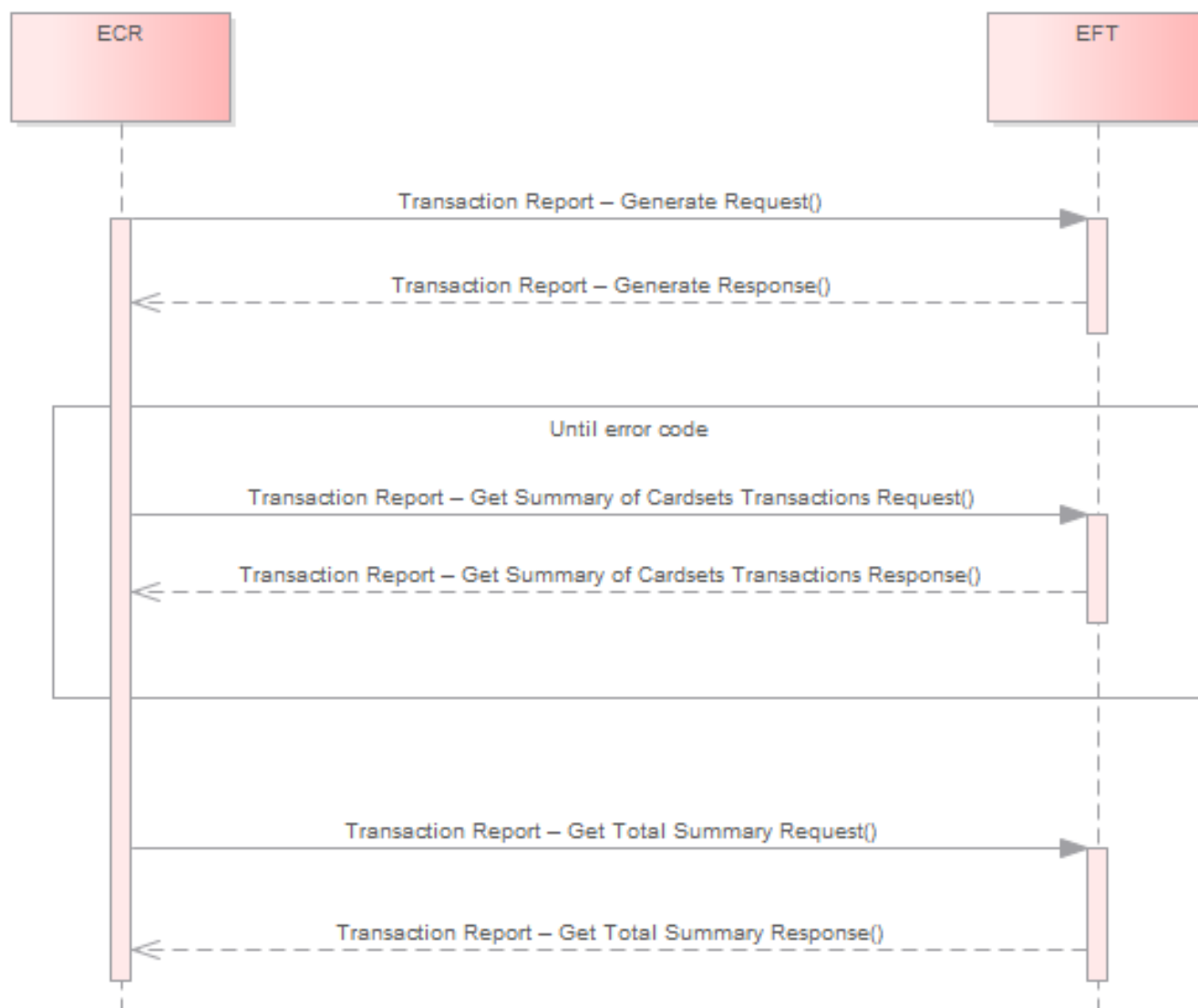
**Example of the whole response:**

```
<STX>25|0|3000|3|0||13|2020.04.03|09:07:00|1 |45450139||||450|7808<ETX><LRC>
25|0|1|0|10|00770Z,,000000000002009,*****3553,2020.04.03,09:05:25 ,MASTERCARD,C,45450139,000010,1,1000,A,,,,,,,,,1,PLN
,,,,,,,,,DF0B0400000400
25|0|1|0|11|00771Z,,000000000002009,*****3553,2020.04.03,09:05:44 ,MASTERCARD,C,45450139,000011,1,1000,A,,,,,,,,,1,PLN
,,,,,,,,,DF0B0400000400
25|0|1|0|12|00771Z,,000000000002009,*****3553,2020.04.03,09:06:09 ,MASTERCARD,C,45450139,000012,1,1000,A,,,,,,,,,1,PLN
,,,,,,,,,DF0B0400000400
```

## 8 Transactions reports related commands

### 8.1 Transaction Report

Following commands allow ECR to gather all information from the terminal necessary to create transaction report – short and long version. Report contains transaction details since last report reset till new report generation.



#### 8.1.1 Transaction Report - Generate

Following commands allow ECR to collect data to prepare data information for Report generation. There is possibility to reset Report by sending AGR\_2 set to 1. Terminal will return transaction numbers range for generated report.

**NOTE:** Report is generated with transactions since last report reset.

**Request:**

Table 68: Transaction Report - Generate request description.

Field	Field length	Field value	Description
FUN_ID	2	"30"	Function identifier – Transaction Report - Generate
ARG_IN1	1	e.g. "1"	Cash register number
ARG_IN2	0-2	e.g. "1"	<p>Report generate flags: Possible flags:</p> <ol style="list-style-type: none"> <li>0x01 - Reset report counters Indicates if report should be reset after generation, possible values: disabled – do not reset enabled – reset</li> <li>0x02 - RFU</li> <li>0x04 - multicurrency report In this case terminal will generate report data grouped by currencies. Used when terminal supports multicurrency and ECR requires currency report.</li> <li>0x08 - use summary entry id Used when ECR needs entry id instead of transaction names, ie. report is used for some additional calculations.</li> </ol> <p>Example 1: "1" denotes: 1. Reset report counters</p> <p>Example 2: "0C" denotes: 1. Generate multicurrency report 2. Use summary entry id</p>
ARG_IN3	0-256		<p>Additional request data encoded as TLV structure, in ASCII format: &lt;tag&gt;&lt;len&gt;&lt;value&gt; see (Table 35) Tags supported: DF0A</p>

**Response:**

Table 69: Transaction Report - Generate response description.

Field	Field length	Field value	Description
FUN_ID	2	"30"	Function identifier – Transaction Report - Generate
FUN_RESULT	1-2	e.g. "0"	Function return code, see ( <a href="#">Table 1</a> )
ARG_OUT1	13	var	See below

Table 70: Transaction Report - Generate response details.

Subfield	Subfield length	Subfield value	Description
ARG_OUT1.1	6	e.g. "000333"	Sequence number of first transaction included in the report
SS	1	" , "	Subfield separator
ARG_OUT1.2	6	e.g. "000444"	Sequence number of last transaction included in the report

**Example:**

```
<STX>30|1|4<ETX><LRC >
<STX>30|0|000000,000041<ETX><LRC>
```

### 8.1.2 Transaction Report - Get Summary of Cardsets Transactions

Allows ECR to get details of each transaction performed by terminal in report requested range.

**Request:**

Table 71: Transaction Report - Get Summary of Cardsets Transactions request description.

Field	Field length	Field value	Description
FUN_ID	2	"31"	Function identifier – Transaction Report - Get Summary of Cardsets Transactions
ARG_IN1	1	e.g. "1"	Cash register number
ARG_IN2	1-2	e.g. "1"	Index of cardset counting from 1

**Response:**

Table 72: Transaction Report - Get Summary of Cardsets Transactions response description.

Field	Field length	Field value	Description
FUN_ID	2	"31"	Function identifier – Transaction Report - Get Summary of Cardsets Transactions
FUN_RESULT	1-2	e.g. "0"	Function return code, see (Table 1)
ARG_OUT1	0-1024	var	Content of summary, see (8.1.2.1) below
ARG_OUT2	0-40	e.g. "MASTERCARD"	Cardset name (only when report multicurrency flag 0x04 was set during report generate)

### 8.1.2.1 Content of the summary

This information depends on ARG\_IN2 flags set during Transaction Report – Generate.

- If flag 0x04 (multicurrency report) is disabled – single currency content will be generated (see (8.1.2.2) for details)
- If flag 0x04 (multicurrency report) is enabled – multicurrency content will be generated (see (8.1.2.3) for details)
- If flag 0x08 (use summary entry id) is disabled – report with text to print will be generated
- If flag 0x08 (use summary entry id) is enabled – report with summary entry id will be generated (see (8.1.2.4) for details)

### 8.1.2.2 Single currency content of the summary

Contains report summary.

Table 73: Transaction Report - Content of the single currency Cardsets Transactions Summary description.

Subfield	Subfield length	Subfield value	Description
ARG_OUT1.1	0-1024	See below	Transaction report summary content, see table (Table 75). <b>NOTE:</b> Summary title contains text to print.

#### Example:

```
<STX>31|0|MASTERCARD,Sales|10|82000,Refunds|0|0, Completions|0|0,TOTAL|10|82000,
CASHBACK|0|0<ETX><LRC>
```

### 8.1.2.3 Multicurrency content of the summary

Contains report summary grouped by the transaction currencies.

Table 74: Transaction Report - Content of the multicurrency Total Summary description.

Subfield	Subfield length	Subfield value	Description
ARG_OUT1.1	0-1024	See below	Transaction report summary content for the first currency, see table (Table 75). <b>NOTE:</b> Summary title contains currency.
SS	1	”.”	Subfield separator
ARG_OUT1.2	0-1024	See below	Next entry
...	...	...	...

**Example:**

```
<STX>31|0|PLN,Sales|10|82000,Refunds|0|0, Completions|0|0,TOTAL|10|82000,CASHBACK|0|0;
CZK,Sales|9|42000,Refunds|0|0,Completions|0|0,TOTAL|9|42000, CASHBACK|0|0;
EUR,Sales|1|1000,Refunds|0|0,Completions|0|0,TOTAL|1|1000, CASHBACK|0|0;
USD,Sales|1|1000,Refunds|0|0,Completions|0|0,TOTAL|1|1000, CASHBACK|0|0|MASTERCARD
<ETX><LRC>
```

#### 8.1.2.4 Entry id in the summary

In this case terminal uses entry id instead of the text to print (see (section) for the possible values).

**Example:**

```
<STX>31|0|PLN,0|10|82000,1|0|0,8|0|0,21|10 |82000,4|0|0;
CZK,0|9|42000,1|0|0,8|0|0,21|9|42000,4|0|0;
EUR,0|1|1000,1|0|0,8|0|0,21|1|1000,4|0|0;
USD,0|1|1000,1|0|0,8|0|0,21|1|1000,4|0|0|MASTERCARD
<ETX><LRC>
```



### 8.1.2.5 Transaction report summary content

Table 75: Transaction Report - Content of the Cardsets Transactions Summary description.

Subfield	Subfield length	Subfield value	Description
ARG_OUT2.1	0-40	e.g. "VISA"	Summary title. Text to print or Currency depending on report multicurrency flag.
SS	1	","	Subfield separator
ARG_OUT2.2	0-40	e.g. "SALE 10 25943"	Single report entry containing following items (separated with ' '): <ol style="list-style-type: none"> <li>1. Transaction type (text to print or id depending on report flag 0x08)</li> <li>2. Transaction count</li> <li>3. Summary amount in minor units</li> </ol>
SS	1	","	Subfield separator
ARG_OUT2.3	0-40	e.g. "REFUND 3 2000"	Next report entry
...	...	...	...

### 8.1.3 Transaction Report - Get Total Summary

Allows ECR to get summary of all cardsets transactions performed by the terminal.

#### Request:

Table 76: Transaction Report – Get Total Summary request description.

Field	Field length	Field value	Description
FUN_ID	2	"32"	Function identifier – Transaction Report – Get Total Summary

#### Response:

Table 77: Transaction Report – Get Total Summary response description.

Field	Field length	Field value	Description
FUN_ID	2	"32"	Function identifier – Transaction Report – Get Total Summary
FUN_RESULT	1-2	e.g. "0"	Function return code, see table ( <a href="#">Table 1</a> )
ARG_OUT1	0-1024	var	Content of summary, see ( <a href="#">8.1.3.1</a> )

### 8.1.3.1 Content of the summary

This information depends on ARG\_IN2 flags set during Transaction Report – Generate.

- If flag 0x04 (multicurrency report) is disabled – single currency content will be generated (see (8.1.3.2) for details)
- If flag 0x04 (multicurrency report) is enabled – multicurrency content will be generated (see (8.1.3.3) for details)
- If flag 0x08 (use summary entry id) is disabled – report with text to print will be generated
- If flag 0x08 (use summary entry id) is enabled – report with summary entry id will be generated (see (8.1.3.4) for details)

### 8.1.3.2 Single currency content of the summary

Contains report summary.

Table 78: Transaction Report - Content of the single currency Total Summary description.

Subfield	Subfield length	Subfield value	Description
ARG_OUT1.1	0-1024	See below	Transaction report summary content, see table (Table 80). <b>NOTE:</b> Summary title contains text to print.

#### Example:

<STX>32|0|Overall summary:;Sales|10|82000,Refunds| 0|0,Completions|0|0,TOTAL|10|82000,CASHBACK|0|0 <ETX><LRC>

### 8.1.3.3 Multicurrency content of the summary

Contains report summary grouped by the transaction currencies.

Table 79: TransactionReport - Content of the multicurrency Total Summary description.

Subfield	Subfield length	Subfield value	Description
ARG_OUT1.1	0-1024	See below	Transaction report summary content for the first currency, see table (Table 80). <b>NOTE:</b> Summary title contains currency.
SS	1	”;	Subfield separator
ARG_OUT1.2	0-1024	See below	Next entry
...	...	...	...

#### Example:

<STX>32|0|PLN,Sales|10|82000,Refunds|0|0, Completions|0|0,TOTAL|10|82000,CASHBACK|0|0;  
CZK,Sales|9|42000,Refunds|0|0,Completions|0|0,TOTAL|9|42000, CASHBACK|0|0;  
EUR,Sales|1|1000,Refunds|0|0,Completions|0|0,TOTAL|1|1000,CASHBACK |0|0;  
USD,Sales|1|1000,Refunds|0|0,Completions|0|0,TOTAL|1|1000,CASHBACK |0|0 <ETX><LRC>

### 8.1.3.4 Entry id in the summary

In this case terminal uses entry id instead of the text to print (see (section) for the possible values).

**Example:**

```
<STX>32|0|PLN,0|10|82000,1|0|0,8|0|0,21|10 |82000,4|0|0;
CZK,0|9|42000,1|0|0,8|0|0,21|9|42000,4|0|0;
EUR,0|1|1000,1|0|0,8|0|0,21|1|1000,4|0|0;
USD,0|1|1000,1|0|0,8|0|0,21|1|1000,4|0|0 <ETX ><LRC>
```

### 8.1.3.5 Transaction report summary content

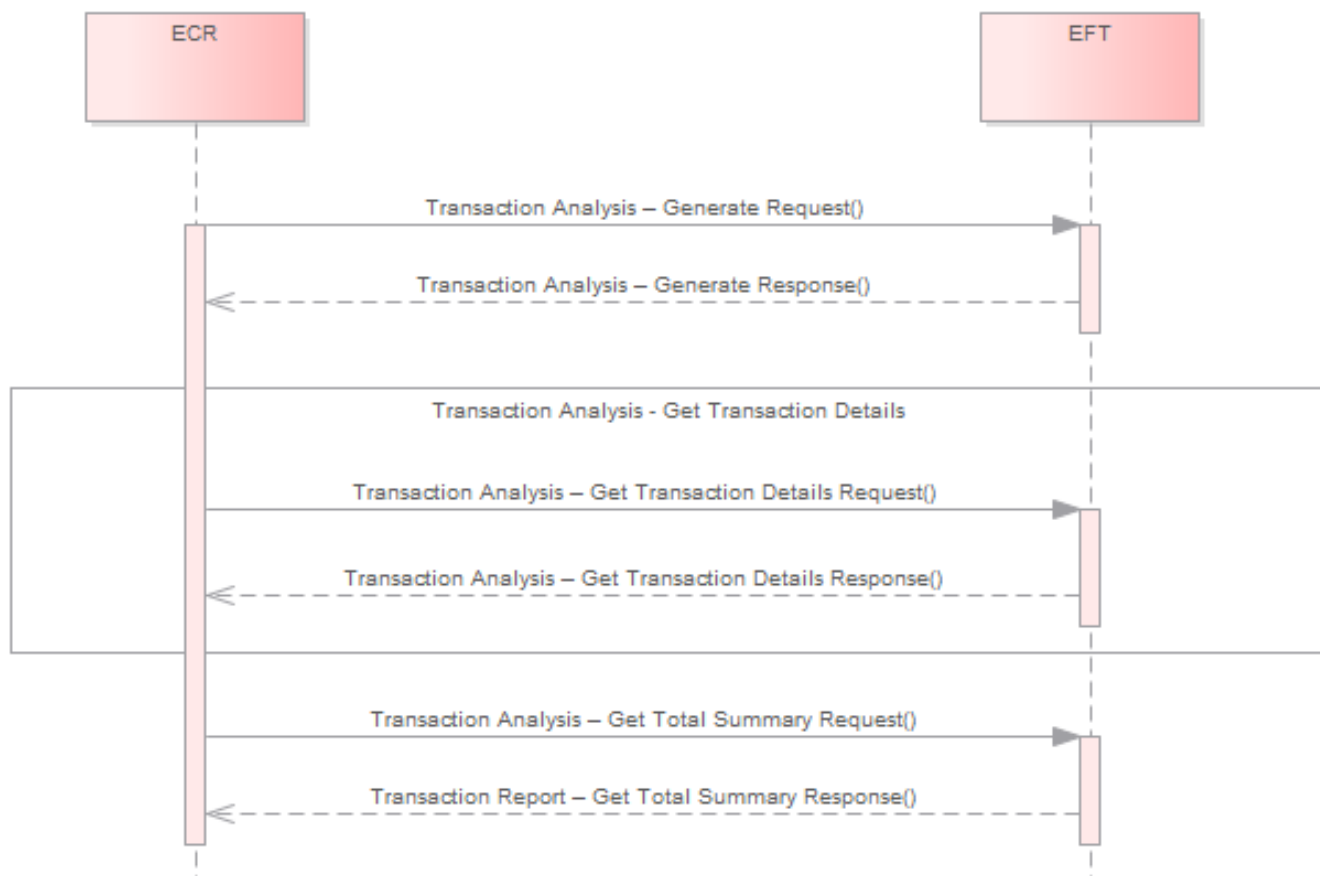
Table 80: Content of Transaction Report Summary description.

Subfield	Subfield length	Subfield value	Description
ARG_OUT2.1	0-40	e.g. "Total"	Summary title. Text to print or Currency depending on report multicurrency flag.
SS	1	" , "	Subfield separator
ARG_OUT2.2	0-40	e.g. "SALE: 10 25943"	Single report entry containing following items (separated with ' '):  1. Transaction type (text to print or id depending on report flag 0x08)  2. Transaction count  3. Summary amount in minor units
SS	1	" , "	Subfield separator
ARG_OUT2.3	0-40	e.g. "REFUND: 3 2000"	Next report entry
...	...	...	...

## 8.2 Transaction Analysis

Allows ECR to get report with details for all transactions performed since last successful End of Day till Report execution. Report can be executed in short or long mode. For long mode see chapter (Section 8.2.2).

**NOTE:** Each Reconciliation process will erase this report data from terminal. After EOD terminal will start summary from scratch.



### 8.2.1 Transaction Analysis - Generate

Allows ECR to collect data for Analysis generation.

**Request:**

Table 81: Transaction Analysis – Generate request description.

Field	Field length	Field value	Description
FUN_ID	2	"33"	Function identifier – Transaction Analysis - Generate
ARG_IN1	1	e.g. "1"	Cash register number
ARG_IN2	0-256		Additional request data encoded as TLV structure, in ASCII format: <tag><len><value> See table (Table 35) Tags supported: DF0A
ARG_IN3	0-2	"4"	Report generate flags Possible flags: Byte 1: <ul style="list-style-type: none"> <li>• 0x01 - RFU</li> <li>• 0x02 - RFU</li> <li>• 0x04 - multicurrency report In this case terminal will generate report data grouped by currencies. Used when terminal supports multicurrency and ECR requires currency report.</li> <li>• 0x08 - use summary entry id Used when ECR needs entry id instead of transaction types to print, ie. report is used for some additional calculations.</li> </ul> <p>Example: "0C" denotes: 1. Generate multicurrency report 2. Use transaction id</p>

### Response:

Table 82: Transaction Analysis – Generate response description.

Field	Field length	Field value	Description
FUN_ID	2	"33"	Function identifier – Transaction Analysis - Generate
FUN_RESULT	1-2	e.g. "0"	Function return code, see table (Table 1)
ARG_OUT1	13	var	The same like in Transaction Generate. See table (Table 70)

### Example:

<STX>33|1||4 <ETX><LRC >  
<STX>33|0|000038,000041 <ETX><LRC>

### 8.2.2 Transaction Analysis - Get Transaction Details

Allows ECR to get data for Transaction Analysis. Collected data can be used to generate Summarized and Detailed Analysis. Detailed information's for transaction performed by terminal and included in Analysis.

**Request:**

Table 83: Transaction Analysis – Get Transaction Details request description.

Field	Field length	Field value	Description
FUN_ID	2	"34"	Function identifier – Transaction Analysis – Get Transaction Details
ARG_IN1	1	e.g. "1"	Cash register number
ARG_IN2	1-3	e.g. "1"	Index of transaction, counting from 1

The response for that command has the same format as regular transaction details, see table (Table 27).

### 8.2.3 Transaction Analysis - Get Total Summary

While Detailed transaction information can be collected by option, this function is mandatory and contains summarized information's of all cardset transactions performed by the terminal.

**Request:**

Table 84: Transaction Analysis – Generate request description.

Field	Field length	Field value	Description
FUN_ID	2	"35"	Function identifier – Transaction Analysis – Get Total Summary
ARG_IN1	1	e.g. "1"	Cash register number

**Response:**

Table 85: Transaction Analysis – Get Total Summary response description.

Field	Field length	Field value	Description
FUN_ID	2	"35"	Function identifier – Transaction Report - Get Total Summary
FUN_RESULT	1-2	e.g. "0"	Function return code, see table (Table 1)
ARG_OUT1	0-1024	var	Content of the summary, see (8.2.3.1)

#### 8.2.3.1 Content of the summary

This information depends on ARG\_IN3 flags set during Transaction Analysis – Generate.

- If flag 0x04 (multicurrency report) is disabled – single currency content will be generated (see (8.2.3.2) for details)
- If flag 0x04 (multicurrency report) is enabled – multicurrency content will be generated (see (8.2.3.3) for details)

- If flag 0x08 (use summary entry id) is disabled – report with text to print will be generated
- If flag 0x08 (use summary entry id) is enabled – report with summary entry id will be generated (see (8.2.3.4) for details)

### 8.2.3.2 Single currency content of the summary

Contains report summary.

Table 86: Transaction Analysis - Content of the single currency Total Summary description.

Subfield	Subfield length	Subfield value	Description
ARG_OUT1.1	0-1024	See below	Transaction analysis summary content, see table (Table 88). <b>NOTE:</b> Summary title contains text to print

#### Example:

```
<STX>35|0|SUMMARY:;Sales|2|15000,Sales voids.|0|0 ,Refunds|0|0,Refunds voids|0|0,
Completions|0|0,Completion voids|0|0,TOTAL|2|15000,CASHBACK|0|0, Pre-Auth|0|0,
ZWIEKSZ.PRE-AUT.|0|0,Pre-Auth voids|0|0 <ETX><LRC>
```

### 8.2.3.3 Multicurrency content of the summary

Contains report summary grouped by the transaction currencies.

Table 87: Transaction Analysis - Content of the multicurrency Total Summary description.

Subfield	Subfield length	Subfield value	Description
ARG_OUT1.1	0-1024	See below	Transaction analysis summary content for the first currency, see table (Table 88). <b>NOTE:</b> Summary title contains currency
SS	1	”.”	Subfield separator
ARG_OUT1.2	0-1024	See below	Next entry
...	...	...	...

#### Example:

```
<STX>35|0|PLN,Sales|1|10000,Sales voids.|0|0, Refunds|0|0,Refunds voids|0|0,Completions|0|0,
Completion voids|0|0,TOTAL|1|10000,CASHBACK|0|0,Pre-Auth|0|0, Increm. auth.|0|0,
Pre-Auth voids|0|0;CZK,Sales|1|5000,Sales voids.|0|0,Refunds|0|0, Refunds voids|0|0,
Completions|0|0,Completion voids|0|0,TOTAL|1|5000,CASHBACK|0|0, Pre-Auth|0|0,Increm. auth.
|0|0,Pre-Auth voids|0|0 <ETX><LRC>
```

### 8.2.3.4 Entry id in the summary

In this case terminal uses entry id instead of the text to print (see (section) for the possible values).

#### Example:

```
<STX>35|0|PLN,0|1|10000,2|0|0,1|0|0,3|0|0 ,8|0|0,9|0|0,21|1|10000,4|0|0,6|0|0,24|0|0,7|0|0;
CZK,0|1|5000,2|0|0,1|0|0,3|0|0,8|0|0,9|0|0,21|1|5000, 4|0|0,6|0|0,24|0|0,7|0|0 <ETX><LRC >
```

### 8.2.3.5 Transaction analysis summary content

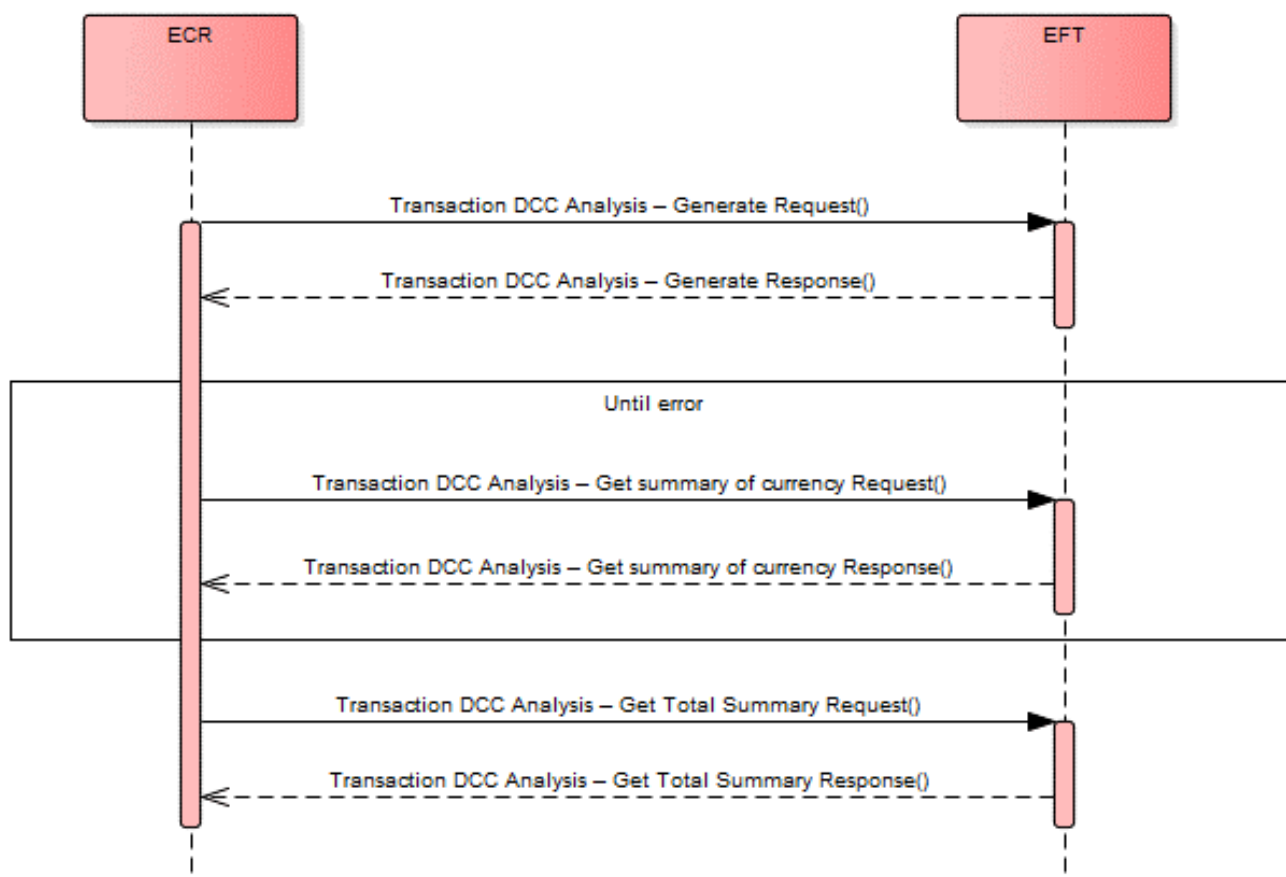
Table 88: Transaction Analysis - Content of the Total Summary description.

Subfield	Subfield length	Subfield value	Description
ARG_OUT2.1	0-40	e.g. "SUMMARY:"	Summary title. Text to print or Currency depending on report multicurrency flag
SS	1	", "	Subfield separator
ARG_OUT2.2	0-40	e.g. "SALE: 10 25943"	Single report entry containing following items (separated with ' '):  1. Transaction type (text to print or id depending on report flag 0x08)  2. Transaction count  3. Summary amount in minor units
SS	1	", "	Subfield separator
ARG_OUT2.3	0-40	e.g. "REFUND: 3 2000"	Next report entry
...	...	...	...

## 8.3 Transaction DCC Analysis

Allows ECR to get report with details for all dcc transactions performed since last successful End of Day till Report execution.





### 8.3.1 Transaction DCC Analysis - Generate

Allows ECR to collect data for DCC analysis generation.

#### Request:

Table 89: Transaction DCC Analysis – Generate request description.

Field	Field length	Field value	Description
FUN_ID	2	"36"	Function identifier – Transaction DCC Analysis - Generate
ARG_IN1	1	e.g. "1"	Cash register number
ARG_IN2	0-256		Additional request data encoded as TLV structure, in ASCII format: <tag><len><value> See table ( <a href="#">Table 35</a> ) Tags supported: DF0A

#### Response:

Table 90: Transaction DCC Analysis – Generate response description.

Field	Field length	Field value	Description
FUN_ID	2	"36"	Function identifier – Transaction DCC Analysis - Generate
FUN_RESULT	1-2	e.g. "0"	Function return code, see table (Table 1)
ARG_OUT1	13	var	See table (Table 70)

**Example:**

<STX>36|1||4 <ETX><LRC >  
 <STX>36|0|000038,000041 <ETX><LRC>

### 8.3.2 Transaction DCC Analysis – Get Currency Summary

Allows ECR to get DCC transactions summary for currency.

**Request:**

Table 91: Transaction DCC Analysis – Get Currency Summary request description.

Field	Field length	Field value	Description
FUN_ID	2	"37"	Function identifier – Transaction DCC Analysis – Get Currency Summary
ARG_IN1	1	e.g. "1"	Cash register number
ARG_IN2	1-3	e.g. "1"	Index of currency, counting from 1

**Response:**

Table 92: Transaction DCC Analysis – Get Currency Summary response description.

Field	Field length	Field value	Description
FUN_ID	2	"37"	Function identifier – Transaction DCC Analysis - Get Currency Summary
FUN_RESULT	1-2	e.g. "0"	Function return code, see table (Table 1)
ARG_OUT1	0-1024	var	Content of summary, see table (Table 93)

Table 93: Transaction DCC Analysis - Content of the currency summary description.

Subfield	Subfield length	Subfield value	Description
ARG_OUT2.1	0-40	e.g. "EUR"	Summary title.
SS	1	", "	Subfield separator
ARG_OUT2.2	0-40	e.g. "SALE: 10 25943"	Single report entry containing following items (separated with ' '): 1. Transaction type to print 2. Transaction count 3. Summary amount in minor units
SS	1	", "	Subfield separator
ARG_OUT2.3	0-40	e.g. "REFUND: 3 2000"	Next report entry
...	...	...	...

### 8.3.3 Transaction DCC Analysis – Get Total Summary

Allows ECR to get total summary of DCC transactions.

#### Request:

Table 94: Transaction DCC Analysis – Get Total Summary request description.

Field	Field length	Field value	Description
FUN_ID	2	"38"	Function identifier – Transaction DCC Analysis – Get Total Summary
ARG_IN1	1	e.g. "1"	Cash register number

#### Response:

Table 95: Transaction DCC Analysis – Get Total Summary response description.

Field	Field length	Field value	Description
FUN_ID	2	"38"	Function identifier – Transaction DCC Analysis - Get Total Summary
FUN_RESULT	1-2	e.g. "0"	Function return code, see table ( <a href="#">Table 1</a> )
ARG_OUT1	0-1024	var	Content of summary, see table ( <a href="#">Table 80</a> )

## 8.4 Report summary entry id

Below table describes summary entry id's used in reports.

Table 96: Report summary entry id.

Value	Description
0	Purchase
1	Refund
2	Reversed Purchase
3	Reversed Refund
4	Cashback
5	Reversed Cashback
6	Preauthorization
7	Reversed Preauthorization
8	Completion
9	Reversed Completion
11	TIP
13	Giftcard Topup
16	Giftcard Purchase
17	Giftcard Activation
18	Reversed Giftcard Purchase
19	Cash
20	Reversed Cash
21	Summary
22	Zencard Discount
24	Incremental Authorization
25	Reversed Incremental Authorization

## 9 Miscellaneous functions

### 9.1 Set Parameters

This command is used to set terminal's parameters. There are two types of parameters:

- Volatile - those parameters are reset to their default values after terminal reboot, so they should be set at the beginning of each communication.
- Permanent – stored permanently in terminal's memory (RFU)

Parameters are sent in the message body, as '|' separated list of '=' map entries: key=value;

#### Request:

Table 97: Set Parameters request description.

Field	Field length	Field value	Description
FUN_ID	2	"39"	Function identifier – Set Parameters
ARG_IN1	1	e.g. "1"	Cash register number
ARG_IN2	1	"0"	Type of parameters to be set: <ul style="list-style-type: none"> <li>0 - volatile</li> <li>1 - permanent (RFU)</li> </ul>
ARG_IN3	0-128	e.g. "ecr_fid_supported=1"	List of parameters to be set and it's values.

Table 98: Currently supported volatile parameters description.

Parameter	Value	Default value	Description
ecr_fid_supported	0 - not supported 1- supported	0	Indicates, if ECR is capable to send fid (fiscal id) to the terminal after the transaction.

#### Response:

Table 99: Set Parameters response description.

Field	Field length	Field value	Description
FUN_ID	2	"39"	Function identifier – Set Parameters
FUN_RESULT	1-2	e.g. "0"	Function return code, see table ( <a href="#">Table 1</a> )

### 9.2 Set Standby

This command allows to put the terminal in Standby mode (power saving). It works only on certain unattended devices.

#### Request:

Table 100: Set Standby request description.

Field	Field length	Field value	Description
FUN_ID	2	"29"	Function identifier – Set Standby
ARG_IN1	1	e.g. "1"	Cash register number
ARG_IN2	1-2	"10"	Timeout (in second) after which the terminal will turn into Standby mode.

**Response:**

Table 101: Set Standby response description.

Field	Field length	Field value	Description
FUN_ID	2	"29"	Function identifier – Set Standby
FUN_RESULT	1-2	e.g. "0"	Function return code, see table ( <a href="#">Table 1</a> )