Payment terminal
Integration with Cash Register

(v.3.00)

Contents

1 ABBREVIATIONS

<ACK>	(Acknowledgement) Byte with value of 0x06
<ETX>	(End of transmission text) Byte with value of 0x03
<NAK>	(Negative acknowledgement) Byte with value of 0x15
<ESC>	(Escape) Byte with value of 0x1B - used in meaning "BUSY"
<STX>	(Start of Text) Byte with value of 0x02
<FS>	(Field Separator) Byte with value of 0x1C
<EOT>	(End of Transmission) Byte with value of 0x04
<ENQ>	(Enquiry) Byte with value of 0x05
LRC	Longitudinal Redundancy Check - one binary Byte
ECR	Electronic Cash-register
RP	Cash-register
POS	Point Of Sale
AC	Authorization central (also called „Host")
UAT	**U**n**A**ttended **T**erminal
EFT	Electronic Funds Transfer
EFT-POST	EFT terminal

2 INTRODUCTION

This document describes communication protocol between Electronic Cash Register (ECR) and EFT-POST as parts of attended POS solution.

Communication is always initiated by ECR.

Packet exchange is synchronic (request -> response). EFT-POST is "server" and ECR is „client".

The basic state of EFT-POST application is IDLE state. EFT-POST reaches IDLE state automatically after power-up or after completing required task. In this state is EFT-POST waiting to receive packet from the ECR. ECR triggers action by sending request message to EFT-POST.

2.1 Supported EFT-POST services

ECR may request EFT-POST terminal to execute the following tasks:

- Card transaction:
  - Sale
  - Cancellation
  - Refund
  - Totals (Closure)
  - Subtotals
  - Link test to bank and meal HOST
- Loyalty:

  - Accrual
  - Redeem
  - Cancellation
  - Totals (Closure)
  - Line check
- TopUp:
  - Charge
  - Cancel
  - Totals (Closure)
  - Line check
- Kasa
  - Pay
  - Reversal
  - Totals (Closure)
  - Line check
- Supplementary services:
  - Totals – closure of all supplementary services
  - Subtotals – sub-totals of all supplementary services
  - Line check – connection test to supplementary services HOST
- Totals (Closure) – chained closure of all supported services including card transactions
- Subtotals – chained sub-totals of all supported services including card transactions
- Line check – chained connection test to all HOSTs (including card transactions HOSTs)
- Resend result

Some of these tasks are optional and do not have to be implemented and their usage depends on installation requirements.

2.2 Communication interfaces

Supported communication interfaces are RS232 and Ethernet.

2.2.1 SERIAL – RS232

Signals: Rx, Tx, GND

Available settings:

Transmission speed

Data bits

Parity
Stop bits
Flow control

2.2.2 ETHERNET - TCP/IP

This is preferred communication interface.

Both EFT-POST and ECR must have assigned static IP address in local network.

EFT-POST takes the role of server and „listens“ on specified TCP port (by default 20008). Port can be adjusted through EFT-POST settings.

ECR establishes TCPIP connection.

It is possible to create only one connection at the time. In other words EFT-POST accepts and handles only one connection with ECR.

2.2.3 PEERS CONFIGURATION

- ACK timeout
- Message timeout
- Paired/trusted device IDs:
    - EFT-POST ID
    - ECR ID (for example DKP)
- Printing:
    - characters per line (for receipt formatting purpose)
- Communication Interface:
    - TCPIP:
        - Destination IP (set only on ECR as EFT-POST is server)
        - Destination Port (set only on ECR as EFT-POST is server)
        - Trusted IP (optional parameter on EFT-POST)
    - RS232:
        - Transmission speed: 115200 bps
        - Data bits: 8
        - Parity: none
        - Stop bits: 1
        - Flow control: none

2.2.4 PEERS VERIFICATION

Serial communication

There is no verification. It is based on physical connection.

TCP/IP communication

Client uses the set of server's parameters. There is no other verification. Server ***optionally*** verifies client by comparing client‘s IP to trusted IP in his parameters. If IP verification fails, server rejects connection.

If trusted IP is not set server accepts connection from any IP.

Application level

On application level both sides should check if packet's source ID matches its own unique device ID.

POST uses either pre-configured value of or takes value of source ID from very first received packet.

ECR can bypass POST checking by using wildchar “*” at the beginning of particular ID. For example if DID check is not required DID value should start by “*” (the remaining characters are ignored).

3 PROTOCOL SPECIFICATION

3.1 Implementation rules

Each packet must be confirmed by acknowledge byte: <ACK>.

When EFT-POST receives the "Request" message, which contains all necessary information to be identified, it executes the required task.

When EFT-POST receives ENQ:

- sends ACK => ready to accept and process request
- Sends ESC => busy i.e. processing another request
- no response => see chapter Troubleshooting.

Notes:

- EFT-POST ignores messages received while processing previous ECR request.

Unknown fields - Data part content processing must be capable to handle data fields with unknown identifiers and empty data fields.

3.2 Packet structure

Message is transmitted enclosed by special characters STX and ETX. Each data packet is terminated by LRC.

<table>
<tr><td rowspan="2">&lt;STX&gt;</td><td colspan="2"><b>Message</b></td><td rowspan="2">&lt;ETX&gt;</td><td rowspan="2">{LRC}</td></tr>
<tr><td>Header</td><td>Data</td></tr>
</table>

Table 1. General packet structure

Header								
Protocol Name	Protocol Version	Command	Sub-command	Source ID	Destination ID	Session ID	Packet ID	Data Length
AN4	N2	AN1	AN2	ANS16	ANS16	N4	N4	N4

Table 2. Header structure

Source ID: Unique identification of sending device. Value must be right padded by space.
Destination ID: Unique identification of receiving device (the one that packet is addressed to). Value must be right padded by space.
Session ID: Identification number of session generated by ECR. It is expected to be unique within one day. It is recommended to be used as session sequence Nbr. (incremented by one)
Packet ID: Identification number that identifies each particular packet. It is expected to be used as packet sequence Nbr. (incremented by one) by both peers (ECR and POST).

Field list included in data is depends on used command. Packet templates for each particular command will be covered later on.

FieldID_1	Value_1	0x1C	FieldID_2	Value_2	0x1C	…	0x1C	FieldID_N	Value_N

Table 2. Message - data part structure

LRC is 1 binary byte value. It’s value is computed by applying XOR operation on complete message and ETX byte. If message length is N than LRC is computed:
LRC = Char1 XOR Char2 XOR Char3 ... XOR CharN XOR ETX

3.3 Command list

ID	Name	Source	Description
S	START_RQ	ECR	Starting communication session on application level
R	START_RSP	EFT-POST	Used to confirm that session has started or error occurs
F	FINISH	ECR	Triggers session completion by providing the list of services that are expected to be confirmed
C	CONFIRM	EFT-POST	Confirmation of service completion
E	END	ECR	Session termination
0	RQ_SRV	ECR	Request for service
1	RSP_SRV	EFT-POST	Response to service request packet
2	INFO	EFT-POST	Send information to ECR’s screen or printer
3	RQ_IN	EFT-POST	Request for input from keyboard or other device
4	RSP_IN	ECR	Response on RQ_IN command

3.4 Sub commands

ID	Task name	Host	Description
AT	Totals All	ALL	chained totals of all supported services
AS	Subtotals All	ALL	chained subtotals of all supported services
AL	Line test All	ALL	chained line check of all supported HOSTs
CP	C_Payment	Bank, Meal	Card payment

CC	C_Cancel	Bank, Meal	Cancellation of card payment
CR	C_Refund	Bank	Returning funds to bank card account
CT	C_Totals	Bank, Meal	Closure: get and reset totals
CS	C_Subtotals	Bank, Meal	Get totals
CL	C_Line_Check	Bank, Meal	Test connection with bank and meal card HOST
LA	L_Accrual	Loyalty	Add loyalty points
LR	L_Redeem	Loyalty	Redeem loyalty points
LC	L_A_Cancel	Loyalty	Cancel Add loyalty points
LB	L_R_Cancel	Loyalty	Cancel Redeem loyalty points
LL	L_Line_Check	Loyalty	Test connection with loyalty HOST
TC	T_Charge	TopUp	TopUp mobile credit
TR	T_Reverz	TopUp	Reverz TopUp mobile credit
TT	T_Totals	TopUp	TopUp totals/closure
TS	T_Subtotals	TopUp	TopUp subtotals
TL	T_Line_check	TopUp	Test connection with Topup HOST
KP	K_Pay	Kasa	Pay cheque (bill payment)
KR	K_Reverz	Kasa	Reverz cheque pay
KT	K_Totals	Kasa	Totals Kasa
KL	K_Test	Kasa	Test connection with Kasa HOST
RR	Resend result	N/A	Request resending last task result
00	None	None	Value used in commands where sub-command is not used
SS	S_Subtotals	Services	Subtotals of supplementary services (TopUp, Kasa, Loyalty)
ST	S_Totals	Services	Totals (closure) of supplementary services (TopUp, Kasa, Loyalty)
SL	S_LineCheck	Services	Line check of supplementary services HOST/HOSTs

3.5 Data structure

3.5.1 DATA FIELDS

Message consists of mandatory and optional fields. If any of mandatory fields is missing message should be considered as incomplete – appropriate error code is returned.

Fields are delimited by FS (Field Separator) [0x1C] and starts with character that defines field. Fields can be placed in message in any order. Except FS message includes only printable ASCII characters: letters, digits and symbols.

3.5.2 FIELD FORMAT AND ATTRIBUTES

The following table lists field attributes and their meaning:

Attribute	Meaning
A	Mandatory if transaction was approved
C	Conditional – depends on presence or value of another field
D	Mandatory if transaction was declined
E	Mandatory if transaction was terminated with error
M	Mandatory
O	Optional
R	Mandatory in response if present in request
F	Mandatory in financial transactions
X	Mandatory for resend task response
P	Mandatory when data for printer are present

Attributes can be combined if necessary. When symbol „!“ precedes attribute, it negates attribute’s meaning (similar to programming language C syntax).

Format defines field length and which characters are permitted to appear in field (combinations are allowed). C language annotation is used. Permitted characters are followed by length information. If there is single number length is accurate but it can be variable too. We use common mathematic notation to specify intervals (a, b) and <c, d>.

Format	Meaning
A	alphanumeric characters ‘A’-‘Z’, ‘a’-‘z’
N	digits ‘0’-‘9’
X	hexadecimal characters ‘0’-‘9’, ‘A’-‘F’
S	special characters (printable), ‘*’ etc.
B	binary characters (any value ASCII 0-255)

Example:

A1 is field that contains single letter.

AN<2, 8> means that field is created by alphanumeric characters and may have length from 2 up to 8 characters.

3.5.3 FIELD LIST

Field ID	Description	Description
C	Amount	Amount authorized (total amount: goods, services, cashback, etc.)
M	Meal Amount	Amount for goods classified as food - used for payment by meal card.
c	CashBack Amount	Amount for cashback part of transaction

A	Authorization code	Field value depends on card brand and issuers AC. The only permitted symbol is „ “ (space ASCII 0x20).
a	Account	Loyalty account identification
N	Customer ID	Customer Identification number – phone number, card number, etc.
T	Timeout	Time required for completing action
t	Time-stamp	Transaction Date and time in format: YYYYMMDDhhmmss
S	Variable symbol	Variable symbol (Transaction unique Id)
E	Electronic Code	TopUp – EAN product code; Kasa – Cheque Bar Code
J	Balance	Loyalty account balance
P	Print	ASCII Text to print on receipt (for text format see …)
D	Display	Text for ECR to display on the screen (for text format see …)
b	Card Brand	“VISA”, “Maestro”, …
s	Signature	If signature is required value is Y otherwise N
p	PIN	PIN entered: Y/N
x	Print type	Customer / Merchant
f	Force print	Force printing right after receiving message
R	Resp. Code	For example authorization response code received from HOST or ECR operation result
d	Details	More specific information about task execution process and/or result
F	Transaction ID	Identification number used to pair cancellation transaction with original transaction on HOST server side
m	Response Message	Textual representation of overall result
I	Task ID	Value used to identify particular task in session.
r	Overall result	Operation result category: 0 – approved 1 – declined 2-8 – reserved 9 – local declined by card or terminal or other error (communication error, etc.).
O	Card type	P-pay; M-meal; L-Loyalty
k	Card interface	1-Magstripe; 2-chip; 3-CTLS
n	EFT-POS Totals	Terminal side totals
h	Host Totals	Server side totals
Z	Input device	Device required to get input from: B – bar code reader K – keyboard
X	Composed field	Multi-record field
V	Values	Multi-functional field. See data templates for details.
K	Cheque issuer	Identity which issues cheque (Orange, ZSE, SPP, etc.)
i	Original ID	Sequence ID of original service request
B	BIN	Card number prefix (usually first 6 digits of card number)

g	Terminal ID	Terminal ID used for current service transaction
G	Control	Value controls terminal behaviour during transaction. It overwrites default terminal behaviour. See dedicated chapter *Task processing control*

3.5.4 DISPLAY AND PRINT DATA FORMATTING

Text fields that are addressed to be displayed on ECR screen or printed on ECR printer are formatted by using following control character sequences:

“\n” - new line

“\c” – center (applied to complete line)

“\b” – bold (applied to complete line)

“\h” – double high (applied to complete line)

“\e” – end of single receipt

Print data maximum line width is by default related to POST HW capabilities. This may be configurable in future versions of EFT-POST applications.

Any limitations on ECR printer side (such as number of characters per line), must be known in advance.

3.6 Task processing control

Default behaviour of POST during task processing:

- Signature verification (if required by card processing) is handled by ECR
- Receipts are not printed on POST
- Receipt data is sent to ECR

This behaviour can be changed by value of field “G” (“Control”) included in RQ_SRV command. When POST receives field “G” it applies changes and uses it until another “G” field is received or until next POST application restart.

Valid values are:

1 - Handle signature check on POST

2 - Print receipt on POST (receipt data are still send to ECR)

4 - turn off sending status INFO messages -> i.e. POST screen mirroring (INFO messages with receipt data are still send to ECR)

8 - turn off sending receipt data INFO messages to ECR

and any other value get by combine above options i.e. simple adding above values.

Example:

Value “3” means - Both signature check and receipt printing handled on POST but receipt data are still send to ECR.

3.7 Screen Mirroring

Most of messages displayed on POST screen are mirrored to ECR display.

This is realized by sending INFO command each time when on the POST screen is displayed a message important for transaction progress.

3.8 Receipts printing

By default terminal:

- do not print receipts
- receipt data (data for printing on ECR) are provided through dedicated field within INFO command (command ID = "2"). When EFT-POST issues an INFO command is covered in chapter *4.3 Implementation schemas*. Receipt data formatting is covered in chapter *3.5.4. Display and print data formatting*.

Such behavior and can be overwritten (See dedicated chapter *3.7 Task processing control*).

3.9 Message templates

3.9.1 COMMON RULES

It is recommended to use unique identification for packet source and destination device IDs:

- ECR ID (for example DKP may be used or any other unique value) space right padded

- EFT-POST ID (may not match with authorization "Terminal ID") right padded by space

For example if packet source is ECR then packet header should use:

- ECR ID as Source ID

- EFT-POST ID right padded by space as Destination ID

Session ID should be the same for all packets sent within one session (starting by "START_RQ" and ending by "END" command).
Packet ID is incremented to be unique per each particular packet.

3.9.2 START_RQ

This message initiates communication session between ECR and EFT-POST. It is always issued by ECR. Message does not include any data part. It is expected that EFT-POS responds by START_RSP message. If EFT-POST receive this command while previous session (with same sessionID) is still active, it will cancel all reservation services and open a new session.

Command ID: S

Sub-command: 00
Data: No data.

Header								
Protocol Name	Protocol Version	Command	Sub-command	Source ID	Destination ID	Session ID	Packet ID	Data Length
AN4	N2	N1	AN2	ANS16	ANS16	N4	N4	N4
POST	03	S	00	ECR_DKP	TerminalID	ECR gen.	ECR gen.	0000

Table 2. Header structure

Example:
<02>POST03S00DKP1234567890123TERMID12   123412340000<03>{LRC}

3.9.3 START_RSP

This is a response message to START_RQ. EFT-POS informs ECR about START_RQ result.
Command ID: R
Sub-command: 00
Data: Field "R" – response code. Field value on success is "0000", otherwise particular error code is generated (see chapter *Error Codes*).

Field	Name	Format	Attributes	Description
R	Response code	AN<1,4>	M	Alphanumeric result code
m	Response Message	ANS<1,40>	O	Textual representation of overall result

Example:
<02>POST03R00TERMID12   DKP1234567890123123412340005R0000<03>{LRC}

3.9.4 RQ_SRV

This packet starts execution of single service on EFT-POST. We will call it a "TASK".
Command ID: 0
Supported sub-commands: all except "00".
Data: see templates for each sub-command below.
Response: During service processing EFT-POST may use command packets: INFO, RQ_IN.
Task is considered as completed once command packet RSP_SRV is received.

3.9.4.1 Closure/Subtotals

Triggers "chained" closure on each supported HOST.

Sub-command: AT - Closure

Sub-command: AS - Subtotals

ID	Field name	Format	Attributes	Description
I	Task ID	AN<3,16>	M	Alfa-numeric Identification used to keep task request tracked.

3.9.4.2 Line check

Triggers "chained" connection check to each supported HOST in this order:

- Bank host
- Meal-Card host
- Supplementary services host (TopUp, Kasa, Loyalty HOST)

Terminal stops execution if any of partial Line checks fail and generates appropriate error code (see chapter *Error Codes*).

Sub-command: AL

Data: see table below

Field	Name	Format	Attributes	Description
I	Task ID	AN<3,16>	M	Alfa-numeric Identification used to keep task request tracked.

3.9.4.3 Card – Payment

Sub-command: CP

Field	Name	Format	Attributes	Description
C	Amount	N<1,12>	M	Amount authorized (in eurocents)
M	Meal Amount	N<1,12>	O	Meal Amount authorized (in eurocents)
c	Cashback amount	N<1,12>	O	CashBack Amount (in eurocents)
I	Task ID	AN<3,16>	M	Alfa-numeric Identification used to keep task request tracked.
S	Variable symbol	AN<0,20>	O	Unique transaction identification generated by ECR for matching transaction in accounting
G	Control	N<1,3>	O	Value controls terminal behaviour during transaction. See chapter *3.7 Task processing control*

3.9.4.4 Card – Cancel

Sub-command: CC

Payment cancellation request must include amount and Transaction ID of original transaction from field “F” in RSP_SRV message. EFT-POST checks these data. It is possible to cancel only last authorized transaction. Partial cancellation is not permitted (final amount must be 0.00).

Field	Name	Format	Attributes	Description
C	Amount	N<1,12>	M	Amount authorized (in eurocents)
I	Task ID	AN<3,16>	M	Alfa-numeric Identification used to keep task request tracked.
F	TransactionID	AN<1,32>	M	Transaction ID of original transaction used to match it at HOST side
S	Variable symbol	AN<0,20>	O	Unique transaction identification generated by ECR for matching transaction in accounting
G	Control	N<1,3>	O	Value controls terminal behaviour during transaction. See chapter *3.7 Task processing control*

3.9.4.5 Card – Refund

Payment task used to return funds to bank card account.

Sub-command: CR

Data:

Field	Name	Format	Attributes	Description
C	Amount	N<1,12>	M	Amount authorized (in eurocents)
I	Task ID	AN<3,16>	M	Alfa-numeric Identification used to keep task request tracked.
S	Variable symbol	AN<0,20>	O	Unique transaction identification generated by ECR for matching transaction in accounting
G	Control	N<1,3>	O	Value controls terminal behaviour during transaction. See chapter *3.7 Task processing control*

3.9.4.6 Card – Totals (Closure)

Sub-command: CT

Data:

Field	Name	Format	Attributes	Description
I	Task ID	AN<3,16>	O	Alfa-numeric Identification used to keep task request tracked.

G	Control	N<1,3>	O	Value controls terminal behaviour during transaction. See chapter *3.7 Task processing control*

3.9.4.7 Card – Subtotals

Sub-command: CS

Data: Same as card closure.

3.9.4.8 Card – Line check

Sub-command: CL

Data:

Field	Name	Format	Attributes	Description
I	Task ID	AN<3,16>	O	Alfa-numeric Identification used to keep task request tracked.
G	Control	N<1,3>	O	Value controls terminal behaviour during transaction. See chapter *3.7 Task processing control*

3.9.4.9 TopUp – Charge

On execution of this sub-command POST asks for phone number and validates it.

Command FINISH is mandatory to be used within same session. Otherwise service request will voided i.e. not completed.

Sub-command: TC

Field	Name	Format	Attributes	Description
I	Task ID	AN<3,16>	M	Alfa-numeric Identification used to keep task request tracked.
E	Electronic Code	N13	M	EAN product code

3.9.4.10 TopUp – Cancel (Reversal)

This sub-command may be used only within the session in which corresponding "TopUp Charge" task was performed. Otherwise it will finish with error as mobile operators do not allow cancelling completed transactions.

Sub-command: TR

Field	Name	Format	Attributes	Description
I	Task ID	AN<3,16>	M	Alfa-numeric Identification used to keep task request tracked.
E	Electronic Code	N13	M	EAN product code

A	Authorization code	ANS<2, 8>	M	Approval code of original transaction
i	Original Task ID	AN<3,16>	M	ID of original request

3.9.4.11 TopUp – Subtotals

TopUp sub-totals retrieves local counts and sums. There is no support for HOST counters retrieval at the moment.

Sub-command: TS

Data:

Field	Name	Format	Attributes	Description
I	Task ID	AN<3,16>	O	Alfa-numeric Identification used to keep task request tracked.

Obsolete: Use *Services Sub-Totals* instead

3.9.4.12 TopUp – Totals (Closure)

In addition to TopUp sub-totals this sub-command clears counters.

Sub-command: TT

Data: Same as TopUp subtotals.

Obsolete: Use *Services Totals* instead

3.9.4.13 TopUp – Line check

Sub-command: TL

Data:

Field	Name	Format	Attributes	Description
I	Task ID	AN<3,16>	O	Alfa-numeric Identification used to keep task request tracked.

Obsolete: Use *Services Line check* instead

3.9.4.14 Loyalty – Add

Sub-command: LA

ID	Field name	Format	Attributes	Description
C	Amount	N<1,12>	M	Total amount
N	Customer ID	N<1, 16>	M	Card number
I	Task ID	AN<3,16>	M	Alfa-numeric Identification used to keep task request tracked.

3.9.4.15 Loyalty – Redeem

Sub-command: LR

ID	Field name	Format	Attributes	Description
C	Amount	N<1,12>	M	Total amount
N	Customer ID	N<1, 16>	M	Card number
I	Task ID	AN<3,16>	M	Alfa-numeric Identification used to keep task request tracked.

3.9.4.16 Loyalty – Cancel Add

Sub-command: LC

ID	Field name	Format	Attributes	Description
C	Amount	N<1,12>	M	Total amount
N	Customer ID	N<1, 16>	M	Card number
I	Task ID	AN<3,16>	M	Alfa-numeric Identification used to keep task request tracked.
A	Authorization code	ANS<2, 8>	M	Approval code of original transaction
i	Original Task ID	AN<3,16>	M	ID of original request

3.9.4.17 Loyalty – Cancel Redeem

Sub-command: LB

ID	Field name	Format	Attributes	Description
C	Amount	N<1,12>	M	Total amount
N	Customer ID	N<1,16>	M	Card number
I	Task ID	AN<3,16>	M	Alfa-numeric Identification used to keep task request tracked.
A	Authorization code	ANS<2,8>	M	Approval code of original transaction
i	Original Task ID	AN<3,16>	M	ID of original request

3.9.4.18 Loyalty – Subtotals

This sub-command retrieves POST loyalty counters.

Sub-command: LS

Data:

Field	Name	Format	Attributes	Description

I	Task ID	AN<3,16>	O	Alfa-numeric Identification used to keep task request tracked.

Obsolete: Use *Services Sub-Totals* instead

3.9.4.19 Loyalty – Closure (Totals)

This sub-command returns and clears counters.

Sub-command: LT

Data:

Field	Name	Format	Attributes	Description
I	Task ID	AN<3,16>	O	Alfa-numeric Identification used to keep task request tracked.

Obsolete: Use *Services Totals* instead

3.9.4.20 Loyalty – Line check

Sub-command: LL

Data:

Field	Name	Format	Attributes	Description
I	Task ID	AN<3,16>	O	Alfa-numeric Identification used to keep task request tracked.

Obsolete: Use *Services Line check* instead

3.9.4.21 Kasa – Pay cheque

On execution of this sub-command POST validates received cheque data.

Command FINISH is mandatory to be used within same session. Otherwise service request will voided i.e. not completed.

Sub-command: KP

Data:

ID	Field name	Format	Attributes	Description
E	Electronic Code	AN<1,550>	M	Cheque Bar Code
I	Task ID	AN<3,16>	M	Alfa-numeric Identification used to keep task request tracked.

3.9.4.22 Kasa – Reverz cheque

Sub-command: KR

Data:

ID	Field name	Format	Attributes	Description
i	Original	AN<3,16>	M	ID of Original request

	Task ID			
I	Task ID	AN<3,16>	M	Alfa-numeric Identification used to keep task request tracked.
A	Authorization code	ANS<1,8>	M	Approval code of original transaction

3.9.4.23 Kasa – Subtotals

This sub-command retrieves POST Kasa counters.

Sub-command: KS

Data:

Field	Name	Format	Attributes	Description
I	Task ID	AN<3,16>	O	Alfa-numeric Identification used to keep task request tracked.

Obsolete: Use *Services Sub-Totals* instead

3.9.4.24 Kasa – Totals (Closure)

In addition to subtotals this sub-command not only retrieves counters but also clears them.

Sub-command: KT

Data:

Field	Name	Format	Attributes	Description
I	Task ID	AN<3,16>	O	Alfa-numeric Identification used to keep task request tracked.

Obsolete: Use *Services Totals* instead

3.9.4.25 Kasa – Line check

Sub-command: KL

Data:

Field	Field name	Format	Attributes	Description
I	Task ID	AN<3,16>	M	Alfa-numeric Identification used to keep task request tracked.

Obsolete: Use *Services Line check* instead

3.9.4.26 Services – Subtotals

Supplementary services subtotals retrieves local POST counts and sums. TopUp, Kasa and Loyalty counters are included. There is no support for HOST counters retrieval at the moment.

Sub-command: SS

Data:

Field	Name	Format	Attributes	Description
I	Task ID	AN<3,16>	O	Alfa-numeric Identification used to keep task request tracked.

3.9.4.27 Services – Totals (Closure)

In addition to Services subtotals this sub-command clears all related counters.

Sub-command: ST

Data: Same as Services subtotals.

3.9.4.28 Services – Line check

This sub-command triggers execution of line check to supplementary services HOST.

Sub-command: SL

Data:

Field	Name	Format	Attributes	Description
I	Task ID	AN<3,16>	O	Alfa-numeric Identification used to keep task request tracked.

3.9.4.29 Resend result

On receiving this command POST resends identified by field “Original Task ID” and all related INFO messages. In other words ECR should expect more then one message. POST sends INFO messages first and RSP_SRV message comes at the end of message queue.

If screen mirroring was used for particular task (it is used by default) a whole task process can be restored.

Command purpose is to handle short term communication problems. It is intended to be used within troubleshooting process. (i.e. It is the way how to retrieve lost response/result for particular RQ_SRV)

Sub-command: RR

Field	Name	Format	Attributes	Description
I	Task ID	AN<3,16>	M	Alfa-numeric Identification used to keep task request tracked.
i	Original Task ID	AN<3,16>	O	ID of Original request. If not present, last result is returned.

Note:

To ensure that “requested result” can be identified by value of “i” field, ECR must generate values for field “I” that are unique within observed time period (e.g. day).

Currently only last 10 “results” are kept.

3.9.5 RSP_SRV

Data part templates depends on used sub-commands. All supported sub-commands are listed below.

3.9.5.1 Closure/Subtotals

Response on "chained" closure on each supported HOST.

Sub-command: AT - Closure

Sub-command: AS - Subtotals

Field	**Name**	**Format**	**Attributes**	**Description**
r	Transaction result	N1	M	Operation result category: 0 – approved/success 1 – declined 2-8 – reserved 9 – declined by card or terminal or other local error (communication error, etc.).
I	Task ID	AN<3,16>	M	Alfa-numeric Identification used to keep task request tracked.
~~P~~	~~Print~~	~~ANS<0,500>~~	~~O~~	~~Present if ECR prints receipt.~~
D	Display	ANS<0,100>	O	Present if ECR prints receipt.
t	Time-stamp	N14	A	Date and time of approved transaction (YYYYMMDDhhmmss)
n	Counters	ANS<0,500>	M	Totals from EFT-POST
h	Host Counters	ANS<0,200>	M	Totals from Host
d	Details	ANS<0,100>	O	More specific information about task execution process and/or result
m	Response Message	ANS<1,80>	M	Textual representation of overall result
R	Response code	AN<1,4>	M	Alphanumeric result code

Field value						
Record1	";"	Record2	";"	…	";"	RecordN

Table: Field "n" and "h" structure.

Data	Data type	Example
RecordID	AN<1,8>	"BankCard"

Delimiter	S1	“:”
Debit Count	N<1,8>	23
Delimiter	S1	“,”
Debit Amount	N<1,12>	134000
Delimiter	S1	“,”
Credit Count	N<1,8>	2
Delimiter	S1	“,”
Credit Amount	N<1,12>	1400

Table: Totals record structure

Maximum record size is 52 characters.

Field value includes only non-zero records of allowed transactions (for example if TopUp).

Supported Record IDs:

- “BankCard” - Payment transactions authorized on bank HOST
- “BankCB” - Bank CashBack counters
- “Topup”
- “Loyalty”
- “Cheque” - Kasa cheque payments
- “Callio” - Callio Gastro
- “DOXX” - Doxx meal card
- “UPSK” - LE Cheque DEJENEUR Slovakia meal card
- “UPCZ” - LE Cheque DEJENEUR Czech meal card
- “AXA” - AXA meal card
- “EDENRED” - Edenred meal card
- “BENEFIT” - Benefit Plus meal card
- “SODEXO” - Sodexo meal card
- “GUSTO” - Gusto meal card
- “ESL” - counters from meal card HOST
- „VISA“
- „MC“ – Mastercard
- „MAESTRO“
- „AMEX“
- „DINERS“
- “CUPI” - China Union Pay International
- “JCB”

This list may be extended if needed (e.g. for more card brands, bank or meal card, etc.)

3.9.5.2 Line check

Response on “chained” test link on each supported HOST.

If partial line check fails, POST does not continue line check to next HOST/server (see HOST order specified in request), response will contain information about that particular task.

Sub-command: AL

Data: see table below

Field	Name	Format	Attributes	Description
r	Transaction result	N1	M	Operation result category: 0 – approved/success 1 – declined 2-8 – reserved 9 – declined by card or terminal or other local error (communication error, etc.).
I	Task ID	AN<3,16>	M	Alfa-numeric Identification used to keep task request tracked.
D	Display	ANS<0,100>	O	Present if ECR prints receipt.
m	Response Message	ANS<1,80>	M	Textual representation of overall result
R	Response code	AN<1,4>	M	Alphanumeric result code
d	Details	ANS<0,100>	O	More specific information about task execution process and/or result

If Field “r” value is not “0” – success see field “R”.

For list of error codes see chapter Error codes.

3.9.5.3 Card – Payment

Sub-command: CP

Field	Name	Format	Attributes	Description
r	Transaction result	N1	M	Operation result category: 0 – approved 1 – declined 2-8 – reserved 9 – declined by card or terminal or other local error (communication error, etc.).
I	Task ID	AN<3,16>	M	Alfa-numeric Identification used to keep task request tracked.
A	Authorization code	ANS<2,8>	A	Field value depends on card brand and issuers AC. If field is empty, value wasn’t received from AC (this means that transaction was declined). Only „ “ (space) symbol is permitted.

p	PIN indicator	A1	A	Allowed values are Y a N.
s	Signature	A1	A	Allowed values are Y a N.
b	Card brand	AN<1,30>	A	VISA, MCI, AMEX etc.
R	Host Response code	AN<1,4>	D	Alphanumeric coding of AC decision during authorizing transaction.
D	Display	ANS<0,100>	O	Text for Display. Present if ECR prints receipt.
~~P~~	~~Print~~	~~ANS<0,500>~~	~~O~~	~~Text for printer.~~ ~~Present if ECR prints receipt.~~
t	Time-stamp	N14	A	Date and time of approved transaction (YYYYMMDDhhmmss)
S	Variable symbol	AN<0,20>	R	Unique transaction identification generated by ECR for matching transaction in accounting.
F	Transaction ID	AN<1,32>	M	Transaction ID dedicated to be used for cancellation/reversal to match transaction at HOST side
m	Response Message	ANS<0,40>	M	Textual representation of overall result
O	Card type	A1	M	P-pay; M-meal; L-Loyalty
k	Card interface	N1	M	0-Unknown; 1-Magstripe; 2-chip; 3-CTLS
d	details	ANS<0,100>	OP	More specific information about task execution process and/or result
C	Amount	N<1,12>	M	Amount authorized (in eurocents)
B	BIN	N<1,20>	M	Card number prefix (usually first 6 digits of card number)

3.9.5.4 Card – Cancel

Sub-command: CC

Field	Name	Format	Attributes	Description
r	Transaction result	N1	M	Operation result category: 0 – approved 1 – declined 2-8 – reserved 9 – declined by card or terminal or other local error (communication error, etc.).
I	Task ID	AN<3,16>	R	Alfa-numeric Identification used to keep task request tracked.

D	Display	ANS<0,100>	O	Text for Display. Present if ECR prints receipt.
P	Print	ANS<0,500>	O	Text for printer. Present if ECR prints receipt.
F	Transaction ID	AN<1,32>	O	Transaction ID dedicated to be used for cancellation/reversal to match transaction at HOST side
R	Response code	AN<1,4>	M	Alphanumeric coding of AC decision during authorizing transaction.
t	Time-stamp	N14	A	Date and time of approved transaction (YYYYMMDDhhmmss)
S	Variable symbol	AN<0,20>	R	Unique transaction identification generated by ECR for matching transaction in accounting.
d	details	ANS<0,100>	OP	More specific information about task execution process and/or result
m	Response Message	ANS<1,40>	M	Textual representation of overall result

3.9.5.5 Card – Refund

Sub-command: CR

Field	Name	Format	Attributes	Description
r	Transaction result	N1	M	Operation result category: 0 – approved 1 – declined 2-8 – reserved 9 – declined by card or terminal or other local error (communication error, etc.).
I	Task ID	AN<3,16>	M	Alfa-numeric Identification used to keep task request tracked.
A	Authorization code	ANS<2,8>	A	Field value depends on card brand and issuers AC. If field is empty, value wasn't received from AC (this means that transaction was declined). Only „ " (space) symbol is permitted.
p	PIN indicator	A1	A	Allowed values are Y a N.
s	Signature	A1	A	Allowed values are Y a N.
b	Card brand	AN<1,30>	A	VISA, MCI, AMEX etc.
R	Response code	AN<1,4>	D	Alphanumeric coding of AC decision during authorizing transaction.

D	Display	ANS<0,100>	O	Text for Display. Present if ECR prints receipt.
~~P~~	~~Print~~	~~ANS<0,500>~~	~~O~~	~~Text for printer.~~ ~~Present if ECR prints receipt.~~
t	Time-stamp	N14	A	Date and time of approved transaction (YYYYMMDDhhmmss)
S	Variable symbol	AN<0,20>	R	Unique transaction identification generated by ECR for matching transaction in accounting.
F	Transaction ID	AN<1,32>	M	Transaction ID dedicated to be used for cancellation/reversal to match transaction at HOST side
m	Response Message	ANS<0,40>	M	Textual representation of overall result
O	Card type	A1	M	P-pay; M-meal; L-Loyalty
k	Card interface	N1	M	0-Unknown; 1-Magstripe; 2-chip; 3-CTLS
d	details	ANS<0,100>	OP	More specific information about task execution process and/or result
B	BIN	N<1,20>	M	Card number prefix (usually first 6 digits of card number)

3.9.5.6 Card – Totals (Closure)/Subtotals

Response on Card transaction closure.
Sub-command: CT - Closure
Sub-command: CS - Subtotals
Data: see chapter Closure/Subtotals.

Note: Fields "n" and "h" will include only records related to Card transactions (payment and meal).

3.9.5.7 Card – Line check

Response on test link to bank and meal HOST.
Sub-command: CL
Data structure: see chapter "Line Check".
If Field "r" value is not "0" – success see field "R".
For list of error codes see chapter Error codes.

3.9.5.8 TopUp - Charge

On execution of this sub-command POST asks for phone number and validates it. Success means that entered phone number can be TopUp.

Sub-command: TC

Field	Name	Format	Attributes	Description
r	Transaction result	N1	M	Operation result category: 0 – approved 1 – declined 2-8 – reserved 9 – declined by card or terminal or other local error (communication error, etc.).
I	Task ID	AN<3,16>	R	Alfa-numeric Identification used to keep task request tracked.
A	Authorization code	ANS<2,8>	A	If field is empty, value wasn't received from AC (this means that transaction was declined).
D	Display	ANS<1,100>	O	Present if ECR prints receipt.
~~P~~	~~Print~~	~~ANS<1,500>~~	~~O~~	~~Present if ECR prints receipt.~~
m	Response Message	ANS<1,40>	M	Textual representation of overall result
R	Response code	AN<1,4>	M	Alphanumeric result code
d	Details	ANS<0,100>	O	More specific information about task execution process and/or result
C	Amount	N<1,12>	M	Amount authorized

3.9.5.9 TopUp - Cancel

Sub-command: TR

Field	Name	Format	Attributes	Description
r	Transaction result	N1	M	Operation result category: 0 – approved 1 – declined 2-8 – reserved 9 – declined by card or terminal or other local error (communication error, etc.).
I	Task ID	AN<3,16>	R	Alfa-numeric Identification used to keep task request tracked.
A	Authorization	ANS<2,8>	A	If field is empty, value wasn't

	code			received from AC (this means that transaction was declined).
D	Display	ANS<1,100>	O	Present if ECR prints receipt.
~~P~~	~~Print~~	~~ANS<1,500>~~	~~O~~	~~Present if ECR prints receipt.~~
m	Response Message	ANS<1,40>	M	Textual representation of overall result
R	Response code	AN<1,4>	M	Alphanumeric result code
d	Details	ANS<0,100>	O	More specific information about task execution process and/or result

3.9.5.10 TopUp – Subtotals

Response on TopUp Subtotals.

Sub-command: TS

Data:

Field	Name	Format	Attributes	Description
r	Transaction result	N1	M	Operation result category: 0 – approved/success 1 – declined 2-8 – reserved 9 – declined by card or terminal or other local error (communication error, etc.).
I	Task ID	AN<3,16>	M	Alfa-numeric Identification used to keep task request tracked.
~~P~~	~~Print~~	~~ANS<0,500>~~	~~O~~	~~Present if ECR prints receipt.~~
D	Display	ANS<1,100>	O	Present if ECR prints receipt.
t	Time-stamp	N14	A	Date and time of approved transaction (YYYYMMDDhhmmss)
n	Counters	ANS<0,500>	M	Totals from EFT-POST
d	details	ANS<0,100>	O	More specific information about task execution process and/or result
m	Response Message	ANS<1,80>	M	Textual representation of overall result
R	Response code	AN<1,4>	M	Alphanumeric result code

For field “n” structure see chapter Closure/Subtotals. Field will include only records related to TopUp.

3.9.5.11 TopUp – Closure

Response on TopUp Closure. In addition to TopUp Subtotals this task will clear counters.
Sub-command: TT
Data structure: see chapter TopUp Subtotals.

3.9.5.12 TopUp – Line check

Response on test link to TopUp HOST.
Sub-command: TL
Data structure: see chapter "Line Check".
If Field "r" value is not "0" – success see field "R".
For list of error codes see chapter Error codes.

3.9.5.13 Loyalty – Add

Sub-command: LA

Field	Name	Format	Attributes	Description
r	Transaction result	N1	M	Operation result category: 0 – approved 1 – declined 2-8 – reserved 9 – declined by card or terminal or other local error (communication error, etc.).
I	Task ID	AN<3,16>	M	Alfa-numeric Identification used to keep task request tracked.
A	Authorization code	ANS<2,8>	A	If field is empty, value wasn't received from AC (this means that transaction was declined).
D	Display	ANS<1,100>	O	Present if ECR prints receipt.
~~P~~	~~Print~~	~~ANS<1,500>~~	~~O~~	~~Present if ECR prints receipt.~~
m	Response Message	ANS<1,40>	M	Textual representation of overall result
C	Amount	N<1,12>	M	Amount authorized
J	Balance	N<1,18>	M	Loyalty account balance
R	Response code	AN<1,4>	M	Alphanumeric result code
d	Details	ANS<0,100>	O	More specific information about task execution process and/or result

3.9.5.14 Loyalty – Redeem

Sub-command: LR

Field	Name	Format	Attributes	Description
r	Transaction result	N1	M	Operation result category: 0 – approved 1 – declined 2-8 – reserved 9 – declined by card or terminal or other local error (communication error, etc.).
I	Task ID	AN<3,16>	M	Alfa-numeric Identification used to keep task request tracked.
A	Authorization code	ANS<2, 8>	A	If field is empty, value wasn't received from AC (this means that transaction was declined).
D	Display	ANS<1,100>	O	Present if ECR prints receipt.
~~P~~	~~Print~~	~~ANS<1,500>~~	~~O~~	~~Present if ECR prints receipt.~~
m	Response Message	ANS<1,40>	M	Textual representation of overall result
C	Amount	N<1,12>	M	Amount authorized
J	Balance	N<1,18>	M	Loyalty account balance
R	Response code	AN<1,4>	M	Alphanumeric result code
d	Details	ANS<0,100>	O	More specific information about task execution process and/or result

3.9.5.15 Loyalty – Cancel Redeem

Sub-command: LB

Field	Name	Format	Attributes	Description
r	Transaction result	N1	M	Operation result category: 0 – approved 1 – declined 2-8 – reserved 9 – declined by card or terminal or other local error (communication error, etc.).
I	Task ID	AN<3,16>	R	Alfa-numeric Identification used to keep task request tracked.

A	Authorization code	ANS<2,8>	A	If field is empty, value wasn't received from AC (this means that transaction was declined).
D	Display	ANS<1,100>	O	Present if ECR prints receipt.
~~P~~	~~Print~~	~~ANS<1,500>~~	~~O~~	~~Present if ECR prints receipt.~~
m	Response Message	ANS<1,40>	M	Textual representation of overall result
R	Response code	AN<1,4>	M	Alphanumeric result code
d	Details	ANS<0,100>	O	More specific information about task execution process and/or result

3.9.5.16 Loyalty – Line check

Response on test link to Loyalty HOST.

Sub-command: LL

Data structure: see chapter "Line Check".

If Field "r" value is not "0" – success see field "R".

For list of error codes see chapter Error codes.

3.9.5.17 Loyalty - Subtotals

Response on loyalty subtotals.

Sub-command: LS

Data structure: see chapter Closure/SubTotals. Field "n" will include only records related to loyalty transactions.

3.9.5.18 Loyalty - Closure (Totals)

Response on Loyalty closure.

Sub-command: LT

Data structure: see chapter Closure/SubTotals. Field "n" will include only records related to loyalty transactions.

3.9.5.19 Kasa – pay cheque

Sub-command: KP

Field	Name	Format	Attributes	Description
r	Transaction result	N1	M	Operation result category: 0 – approved 1 – declined

				2-8 – reserved 9 – declined by card or terminal or other local error (communication error, etc.).
I	Task ID	AN<3,16>	M	Alfa-numeric Identification used to keep task request tracked.
C	Amount	N<1,12>	M	Amount authorized
D	Display	ANS<1,100>	O	Present if ECR prints receipt.
~~P~~	~~Print~~	~~ANS<1,500>~~	~~O~~	~~Present if ECR prints receipt.~~
A	Authorization code	ANS<2, 8>	A	If field is empty, value wasn't received from AC (this means that transaction was declined).
m	Response Message	ANS<1,80>	M	Textual representation of overall result
K	Cheque issuer	A20	M	
R	Response code	AN<1,4>	M	Alphanumeric result code
d	Details	ANS<0,100>	O	More specific information about task execution process and/or result

3.9.5.20 Kasa – reverz pay cheque

Sub-command: KR

Field	Name	Format	Attributes	Description
r	Transaction result	N1	M	Operation result category: 0 – approved 1 – declined 2-8 – reserved 9 – declined by card or terminal or other local error (communication error, etc.).
I	Task ID	AN<3,16>	M	Alfa-numeric Identification used to keep task request tracked.
C	Amount	N<1,12>	M	Amount authorized
D	Display	ANS<1,100>	O	Present if ECR prints receipt.
~~P~~	~~Print~~	~~ANS<1,500>~~	~~O~~	~~Present if ECR prints receipt.~~
A	Authorization code	ANS<2, 8>	A	If field is empty, value wasn't received from AC (this means that transaction was declined).
m	Response Message	ANS<1,80>	M	Textual representation of overall result
K	Cheque issuer	A20	M	Issuer name

R	Response code	AN<1,4>	M	Alphanumeric result code
d	Details	ANS<0,100>	O	More specific information about task execution process and/or result

3.9.5.21 Kasa - Subtotals

Response on Kasa subtotals.

Sub-command: KS

Data structure: see chapter Closure/SubTotals. Field “n” will include only records related to Kasa transactions.

3.9.5.22 Kasa - Closure

Response on Kasa closure.

Sub-command: KC

Data structure: see chapter Closure/SubTotals. Field “n” will include only records related to Kasa transactions.

3.9.5.23 Kasa – Line check

Response on test link to Kasa HOST.

Sub-command: TL

Data structure: see chapter “Line Check”.

If Field “r” value is not “0” (success), see field “R”.

For list of error codes see chapter Error codes.

3.9.5.24 Services – Subtotals

Response on supplementary services subtotals request. Local TopUp, Kasa and Loyalty counters are included in data. There is no support for HOST counters retrieval at the moment.

Sub-command: SS

Data structure: see chapter Closure/SubTotals. Field “n” will include records related to all supplementary services.

3.9.5.25 Services – Totals (Closure)

Response on supplementary services totals request.

Sub-command: ST

Data structure: see chapter Closure/SubTotals. Field “n” will include records related to all supplementary services.

3.9.5.26 Services – Line check

Response on “line check” to supplementary services HOST.
Sub-command: SL
Data structure: see chapter “Line Check”.
If Field “r” value is not “0” (success), see field “R”.
For list of error codes see chapter Error codes.

3.9.5.27 Resend result

Sub-command: RR

Field	Name	Format	Attribute s	Description
i	Original Task ID	AN<3,16>	M	Identification of original Task request
All fields from original packet.				

3.9.6 INFO

Purpose of this command message is to enable EFT-POST sending formatted data to ECR screen or printer. In other words by using this message all transaction progress messages can be sent to ECR and displayed on ECR screen.

Another usage is to print EFT-POST transaction receipts on ECR.

Command ID: “2”
Sub-command: 00
Data: see Table
Response: No response.

Field	Name	Format	Attributes	Description
D	Display	ANS<0,100>	M	Text for Display.
P	Print	ANS<0,500>	O	Text for printer.
I	Task ID	AN<3,16>	M	Alfa-numeric Identification used to keep task request tracked.
x	Print type	A1	P	C=Customer, M=Merchant
f	Force print	N1	P	Y= Force printing right after receiving message N (and any other value) = No forced printing
T	Timeout	N3	O	Indicates time period within which next POST command should be issued

Table: Info command data structure

3.9.7 RQ_IN

This packet command enables EFT-POST to request input from ECR keyboard or other ECR external device (reserved for future use). It may be used only after RQ_SRV

Command: 3

Sub-command: 00

Data: see Table below

Field	Name	Format	Attributes	Description
I	Task ID	AN<3,16>	M	Alfa-numeric Identification used to keep task request tracked.
D	Display	ANS<0,100>	M	Text for Display.
~~P~~	~~Print~~	~~ANS<0,500>~~	~~O~~	~~Text for printer.~~
T	Timeout	N3	O	Timeout in seconds for requested input
Z	Input device	A1	M	Device required to get input from: K – keyboard B – bar code reader (for future use)
V	Values	ANS<1,100>	C	Used if field “Z” has value “K”. List of values that may ECR return in RSP_IN as response to particular request

Table: Command RQ_IN - data structure

Keyboard input request example:

<02>POST03300TERMID12 DKP12345678901231234123400491I1234<1C>DDoes signature match?\n1-Yes\n2-No<1C>ZK<1C>T125<1C>V1,2<03>{LRC}

3.9.8 RSP_IN

This is a response packet to command enables EFT-POST to request input from ECR keyboard or other ECR external device.

Command: 4

Sub-command: 00

Field	Name	Format	Attributes	Description
I	Task ID	AN<3,16>	M	Alfa-numeric Identification used to keep task request tracked.
V	Values	ANS<1,100>	C	Value that identifies ECR input

R	Response code	AN<1,4>	M	Alphanumeric result code
m	Response Message	ANS<1,80>	O	Textual representation of overall result

Table: RSP_IN command data structure

Keyboard input response example (option "1" was chosen = signature match):
<02>POST03400DKP1234567890123 TERMID12 123412340014R0000<1C>I1234
<1C>V1<03>{LRC}

3.9.9 FINISH

By using this command ECR announces EFT-POS that services and goods were paid and no more requests for services will be sent within current session. Its usage is mandatory within sessions where any service processed as reservation was requested (e.g. TopUp-Charge).

Data part includes information about all services requested within this session that ECR considers as successfully authorized. It is a trigger for EFT-POST to complete all pre-authorized transactions (service reservations) included in provided service list. POST may issue several INFO commands with receipt data for particular service during processing this command. Reservations (TopUp and cheque transactions) which are not included in message will be considered as voided. Card payments and loyalty transactions must be explicitly cancelled by using appropriate command before issuing this command.

Command: "F".

Sub-command: 00

Response: COMPLETE

Data:

Field	**Name**	**Format**	**Attributes**	**Description**
X	Composed field	ANS<0,300>	M	Multi-record field

Field value						
Record1	";"	Record2	";"	…	";"	RecordN

Table: Field "X" structure.

Sub-Field ID	Data type	Example	Explanation
Sub-Command	AN2	TC	Sub-command ID from header of command RQ_SRV
Value Delimiter	S1	","	
Task ID	AN<3,16>	001	Field "I" value from command RQ_SRV

Value Delimiter	S1	“,”	
Authorization code	ANS<2,8>	12345678	Field “A” value from command RSP_SRV

Table Composed field “X” - record structure

Field value may include only records of successfully preauthorized TopUp and Kasa transactions requested within particular session.

3.9.10 COMPLETE

“COMPLETE” message is sent by EFT-POST as response to FINISH command. By this packet EFT-POS confirms completion of requested services.

Command: “C”.

Sub-command: 00

Data:

Field	**Name**	**Format**	**Attributes**	**Description**
X	Composed field	ANS<0,300>	M	Multi-record field
R	Response code	AN<1,4>	M	Alphanumeric result code

Composed field “X” (same structure as in command FINISH) includes records from command FINISH for tasks that EFT-POST failed to complete. Each record carries information about particular task. Task is identified by Task ID. It is up to ECR how to handle failed tasks completions.

Field “R” – response code. Field value on success (all tasks included in X field completed) is “0000”, otherwise there is a specific error code (see chapter *Error Codes*).

Next package should be “END” command.

3.9.11 END

This command terminates communication session between ECR and EFT-POST.

It should follow command COMPLETE and is always issued by ECR.

After receiving this message EFT-POST:

- voids all services reservations
- closes connection with ECR (configurable)
- performs POST administration tasks (if there is any pending)
- returns to IDLE state.

Message does not include any data part.

Command ID: E

Sub-command: 00
Data: No data.
There is no response expected on this message.

As test if POST is in IDLE state ECR may use simple <ENQ>.

4 IMPLEMENTATION

4.1 Simple packet exchange

This scenario covers single task/service execution.

	ECR		EFT-POST
optional connection test	ENQ	——>	
		<——	<ACK> -> EFT-POST ready <ESC> = BUSY
Packet transmission	Request	——>	
		<——	<ACK> = LRC match - READY <ESC> = LRC match - BUSY <NAK> = otherwise
	In case of <NAK> or timeout resend request max. 2x		
INFO		<——	Request
	<ACK>	——>	
...			
INFO		<——	Request
	<ACK>	——>	
Packet transmission		<——	Response
	<ACK> = LRC match - READY <ESC> = LRC match - BUSY <NAK> = otherwise	——>	
			In case of <NAK> or timeout resend response max. 2x

Timeout for <ACK>, <NAK> or <ESC> is 1 second.

Timeout for EFT-POST response should be adjustable by parameter and its value should be gained by experiment. It depends on request type.

If <ESC> was returned by POST on packet, packet is dropped and sending peer must resend packet until it gets <ACK> or <NAK>.

Handling of temporary communication problems is covered by chapter „*Troubleshooting*“.

Note:

Timeout may be different for each particular task.

4.2 Complex packet exchange

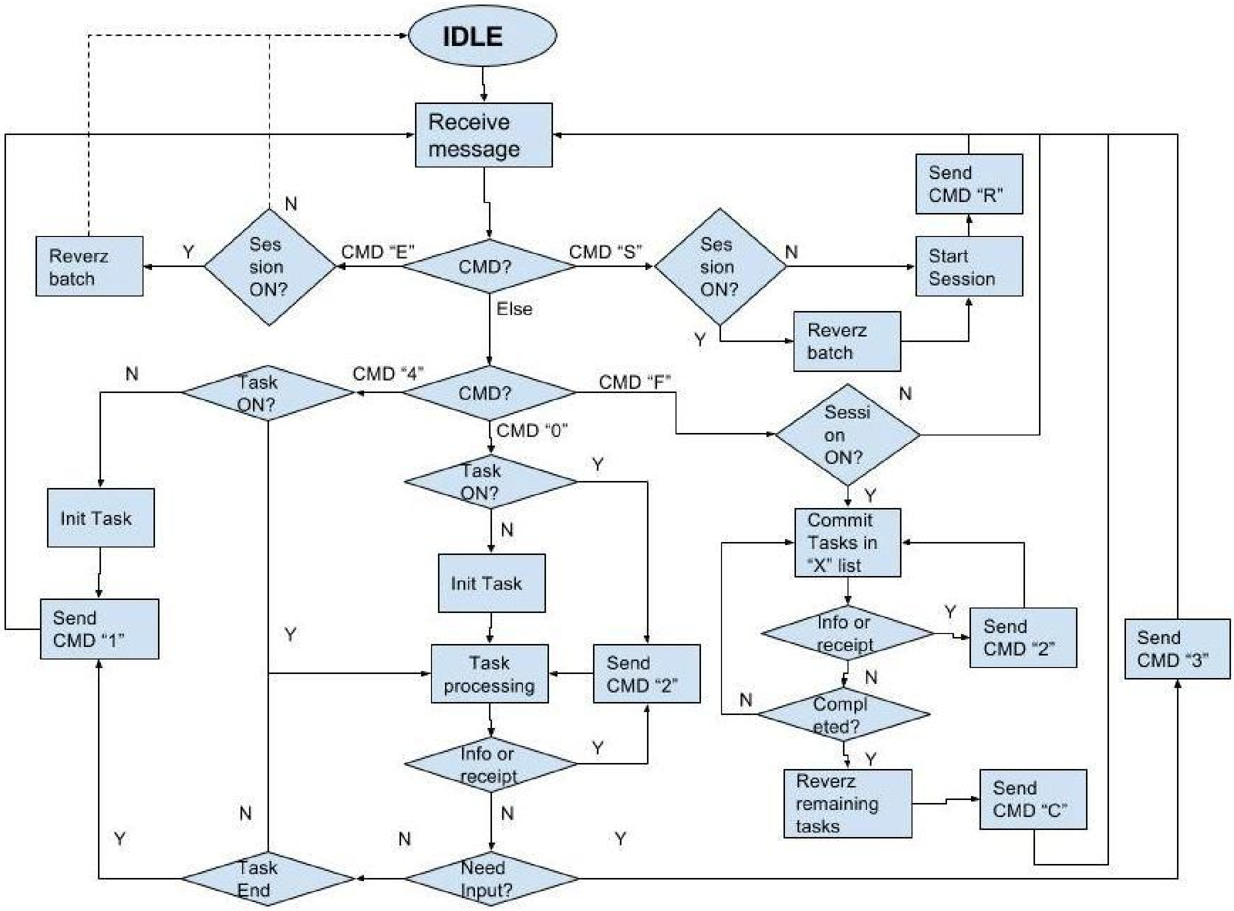
This scenario implements full functionality of protocol. It allows ECR to execute EFT-POS services in bunch within so called session.

Session encapsulates multiple requests for EFT-POS service.

	ECR		**EFT-POST**
optional connection test	ENQ	——>	
		<——	<ACK> -> EFT-POST ready
START_RQ	Request	——>	
		<——	<ACK>
START_RSP		<——	Response
	<ACK>	——>	
Request EFT-POS service	Request	——>	
		<——	<ACK>
EFT-POST RQ_IN		<——	Request for input
	<ACK>	——>	
RSP_IN	Input Response	——>	
		<——	<ACK>
INFO		<——	Request
	<ACK>	——>	
INFO		<——	Request
	<ACK>	——>	
EFT-POST RSP		<——	Response
	<ACK>	——>	
FINISH	Request	——>	
		<——	<ACK>
COMPLETE		<——	Request
	<ACK>	——>	
END	Request	——>	
		<——	<ACK>

4.3 Implementation schemas

4.3.1 MAIN SCHEMA



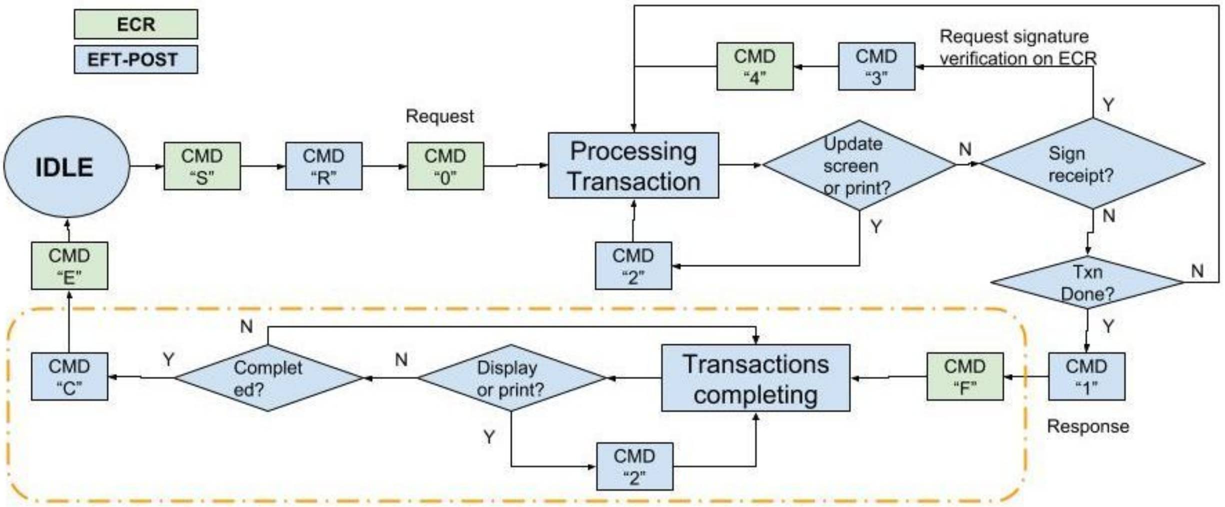
Note:

Please, mind that POST may issue INFO command (cmd “3”) and send receipt for particular service during processing RQ_SRV or during processing FINISH command (cmd “F”).

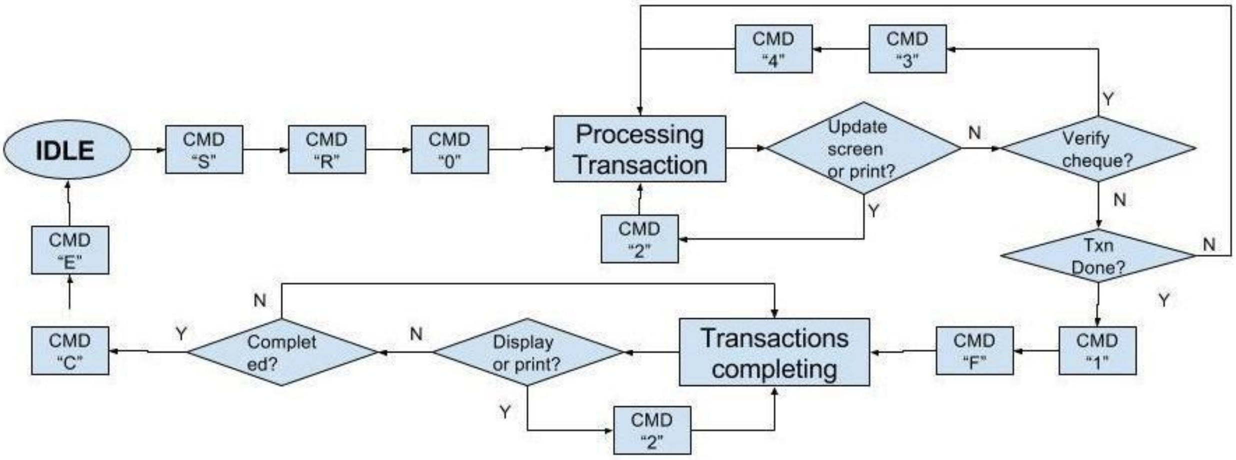
4.3.2 CARD PAYMENT

This schema shows “Card-Payment” service implementation if used as isolated service (not combined with other more complex services) within single session. Mind that

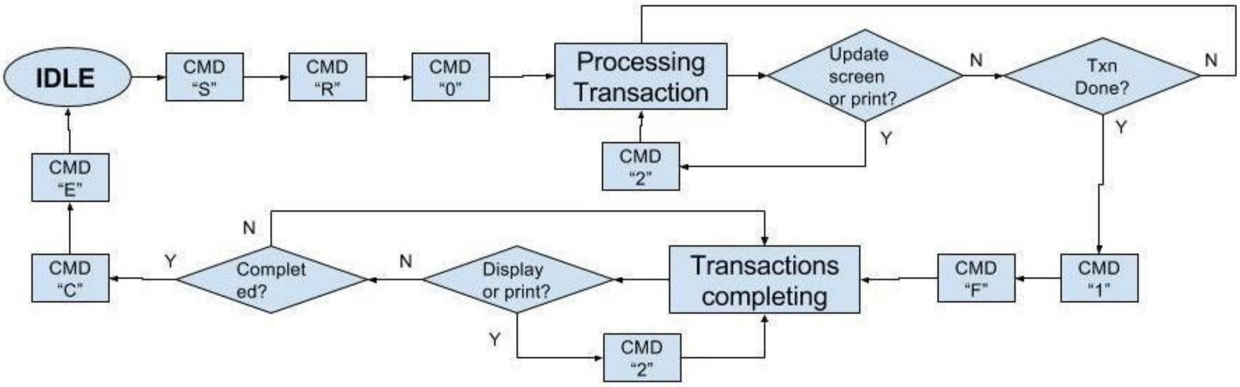
FINISH/COMPLETE commands are optional (marked by orange dashed line) for this service.



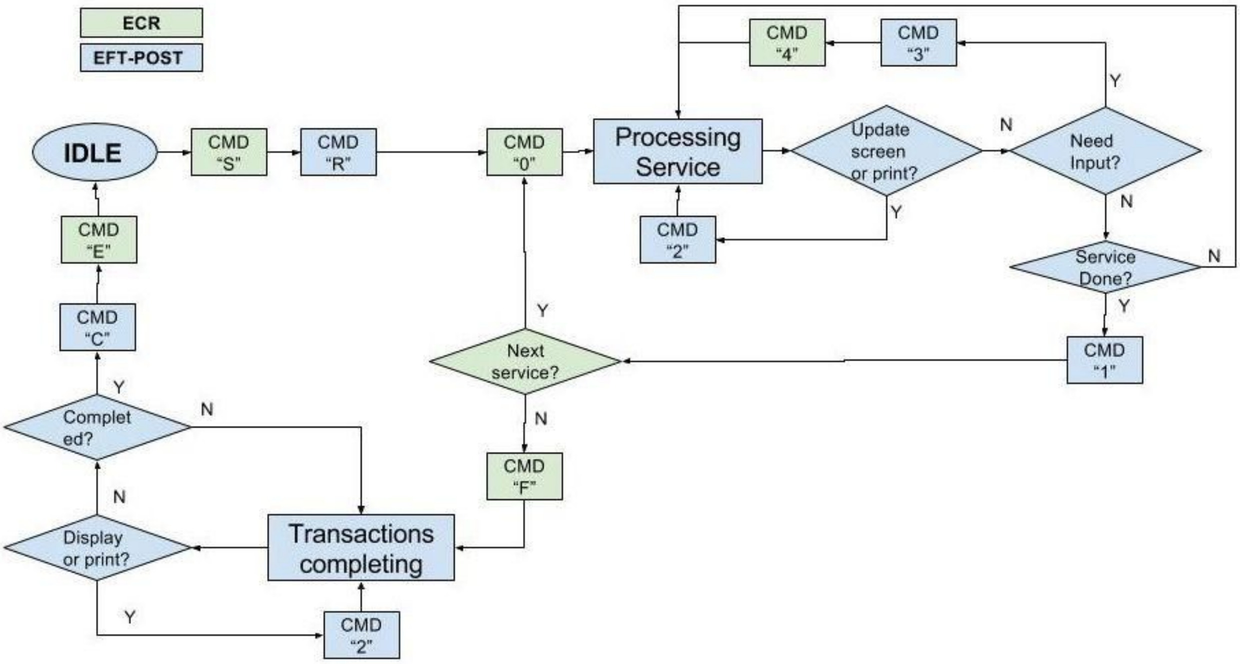
4.3.3 KASA – PAY CHEQUE



4.3.4 TOPUP - CHARGE



4.3.5 MULTI SERVICE SESSION



Note:

Please, mind that POST may issue INFO command (cmd "3") and send receipt for particular service during processing RQ_SRV or during processing FINISH command (cmd "F").

5 TROUBLESHOOTING

5.1 Connection not accepted

This issue is related only to TCPIP interface.

Reason:

1. POST handles another connection (only one connection possible)

Solution:

Check destination IP – possible conflict if more ECR are configured to work with the same POST.

Wait for socket inactivity timeout to elapse.

5.2 ECR – no response from EFT-POST on ENQ

RS232 interface:

Reason:

1. Cable connection issue
2. EFT-POST not configured properly ready (administration or service state).

Solution:

Check cable, check communication settings.

ECR resend ENQ after specified timeout (several seconds).

5.3 ECR – no acknowledge on packet

ECR did not receive acknowledge byte <ACK> on packet.

There are two possible scenarios:

a) EFT-POST received and processed request. Afterwards EFT-POST tries to send response packet. ECR may not expect response at the moment so next attempts will most probably fail.

b) EFT-POST did not receive request

=> EFT-POST stays in IDLE state and keeps waiting for requests from ECR

Solution:

Step1. ECR resends packet (same packetID) when timeout for receiving ACK expires. Maximum 3 attempts are allowed (original request and two resends).

Step2. ECR should use message “Resend result”. If ECR gets response, it performs analyses and tries to match response “Task ID” with original request. If response “Task ID” does not match it means that EFT-POST did not receive previous request.

5.4 ECR – no response packet

After sending request packet, ECR received <ACK> byte, but did not receive anything else within expected time period (usually specified in field “T”).

ECR should issue START_RQ message with the same session ID as original request. POST will respond by START_RSP message with “R” field value:

- “1400”=continue session, so ECR may send message “Resend result”.
- “0000”= new session, session data was lost.

Attempts are repeated until:

- connection with EFT-POST is established and EFT-POST send last response packet,
- ECR obtain precise information about task result and EFT-POST state.

5.5 EFT-POST – no ACK on response packet

EFT-POST resends packet, but only 2 times. Result packet is stored in EFT-POST memory and will be available through “*Resend result*” request issued by ECR.

If message was not delivered it is up to ECR to handle this case as described above.

5.6 Wrong packet LRC

If any peer receives packet with non-matching LRC value, it should act as specified in chapter “*Simple packet exchange*”. If other side keeps sending bad packet (more than 2x), handling is case dependent:

- RSP_SRV: ECR should cancel operation by using appropriate cancellation RQ_SRV (if exists) and issue END session. If problem persists, then POST support incident should be raised.
- RQ_IN: ECR should issue RSP_IN with error code “1003” and optionally error message.
- COMPLETE, START_RSP: Issue END session. If problem persists, POST support incident should be raised.

5.7 Session handling issues

5.7.1 NO FINISH COMMAND

If session is terminated with no “FINISH” command (command “END” or “START” received), EFT-POST ends session and all preauthorized services are cancelled/reversed.

5.7.2 NO COMPLETE COMMAND

If ECR does not receive command COMPLETE as response on FINISH command, it should resend FINISH command.

5.7.3 TIMEOUT ON RQ_IN

EFT-POST did not get response command RSP_IN with in timeout value provided in RQ_IN message. It terminates task with error and sends RSP_SRV.

In special cases both sides may simultaneously send messages and cause temporary dead-lock.

So ECR may not get ACK on RSP_IN and it may miss RSP_SRV too. In this case it is recommended to ECR to issue a RQ_SRV command with “Resend result” sub-command.

5.8 Error codes

Here after the list of codes is going to be described. This codes you may find in field “R” within response packages sent from EFT-POST.

AC decision codes

The range of this codes is “000” to “989”. The meaning of values depends on the AC used to authorize particular transaction/operation.

This codes are returned in RSP_SRV packets if transaction was sent to be online authorized by AC (Authorization Center/Host).

Packet errors

1001 = Packet source ID mismatch

1002 = Packet destination ID mismatch

1003 = Packet format error

1004 = Packet: session ID mismatch

1005 = Packet data: missing mandatory field

1006 = Packet data: syntax error

1007 = Packet data: unsupported command

1008 = Packet data: unsupported sub-command

1009 = Packet data: wrong field value

Communication errors

1100: communication error general

1101: communication error with Bank HOST

1102: communication error with TopUp HOST

1103: communication error with Kasa HOST

1104: communication error with Loyalty HOST

1105: communication error with Meal-card HOST

1106: communication error with Supplementary services HOST

Command COMPLETE error codes

1201: missing task ID (one of TaskIDs not found in list).

1202: some tasks failed to complete (at least one of Tasks was not completed for other reason).

Command START_RSP error codes

1400: Session already started - continue session (All records are kept)

1401: Session start OK - previous session successfully reversed (may be used if previous session was not properly end by using END command)

1402: Session start NOK - previous session reversal failed

RSP_SRV error codes

1500: TaskID not found

1501: Task not found: parameters do not match (used if LoyaltyRedeemCancel was issued and some input parameters do not match)

1502: Card not allowed. Use different card type.

1503: Wrong input parameter value: amount

1504: RSP_IN message not received within expected timeout.

Other

2000: Busy - processing another task

2001: Timeout - operation not completed within specified time

2002: UserCancel - operation interrupted by user: cancel button pressed

2999: Other unspecified error

ECR Server

3000:reserved

...

3999:reserved

Change log:

2017.05.16 - Added field “m” to message RSP_IN
- Added error handling for missing response on FINISH command
- Added error handling for wrong packet LRC value
- Clarifications regarding START_RQ and FINISH commands
- Changed field “TransactionID” flag to “M” for sub-command “Card-Cancel”

2017.09.06 - Added clarification for command *COMPLETE*.

2017.12.01 - Changed filed “E” size from 120 to max. 550 (required to support QR code)

2018.05.29 - Added totals Record IDs for additional meal cards: Benefit, sodexo and UPCZ
- Changed port default value to 20008

2018.06.13 - Clarification of EFT-POST ID value for source and destination IDs (chapter *COMMON RULES*)

2018.06.13 - Fix data type for source and destination IDs (chapter *START_RQ*)

2018.06.18 - Document revision 04

2018.06.21 - Added clarification for *Receipts printing* and *Screen Mirroring*.

2018.07.06 - Updated schema (*CARD PAYMENT*) to be easier to read.

2018.07.06 - Document revision 05

2018.08.20 - Fixed duplicate field identifier “S” in chapture *Field list*. Identifier for field “Signature” set to “s”.

2018.09.05 - Updated schema (*CARD PAYMENT*) to emphasize optional implementation part (i.e. that FINISH is not mandatory).

2018.09.05 - Added *FINISH* command clarification - when it is mandatory and when optional.

2018.09.05 - Updated *4.1 Simple packet exchange* to include INFO messages.

2018.09.05 - Document revision 06

2018.10.24 - *“Resend result”* request - usage extended (removed limitation to one session) and described in more details.

2018.11.25 - Field “G” defined and added to Card related tasks.
Defined bypass POST IDs check by using wildchar “*” (*Peers verification*).

2018.11.08 - Extended list of totals record IDs by: GUSTO, CUPI and JCB.

2018.11.12 - Document revision 07