

GalaxyVend MDB MASTER - USB

**v1.002
2026-02-07**

Table of Contents

I. Introduction.....	7
II. Hardware.....	8
A. Core board overview.....	8
B. Core board connectors and buttons description.....	8
C. Optional relays board (uses GalaxyVend keypress board – sold separately).....	9
D. Relays board connectors.....	9
E. Communication interfaces.....	10
III. Communication protocol.....	11
A. MDB master related commands and answers.....	11
B. Bill validator/recycler related commands.....	11
1. Enable bill validator (no escrow support).....	11
2. Enable bill validator (with escrow support).....	11
3. Selective bills enable.....	12
4. Disable bill validator.....	12
5. Reset bill validator.....	13
6. Approve bill acceptance while a bill is in escrow position.....	13
7. Reject bill while a bill is in escrow position.....	13
8. Get last 10 bill status codes.....	14
9. Check if the bill validator was initialized by the interface.....	14
10. Check if the bill validator was enabled.....	14
11. Get the bill validator configured bills values.....	14
12. Get the bill validator information.....	15
13. Get the bill validator settings.....	15
14. Get the bill recycler bill type values.....	15
15. Get the bills set for recycling by the user application.....	16
16. Set the bills set for recycling by the user application.....	16
17. Get tot total value of bills available for recycling.....	16
18. Dispense bills as a change to customer.....	17
19. Dispense bills as a change to customer.....	17
20. Get current bill stacker status.....	17
C. Coin acceptor/changer related commands.....	18
1. Enable coin acceptor/changer.....	18
2. Selective coins enable.....	18
3. Disable coin acceptor.....	19
4. Reset coin acceptor.....	19
5. Get total value of coins in tubes (for coin changers only).....	19
6. Dispense some coins (change) to the customer – obsolete, try to use COINAP command whenever the coin acceptor/changer supports it.....	20
7. Dispense some coins using MDB alternative payout method.....	20
8. Check if the coin acceptor/changer was initialized by the interface.....	20
9. Check if the coin acceptor/changer was enabled.....	21
10. Get last 10 coin acceptor/changer codes.....	21
11. Get the coin acceptor/changer configured coins values.....	21
12. Get the coin acceptor/changer information.....	22
13. Get the coin acceptor/changer settings.....	22
14. Get the token values.....	23
15. Set the token values.....	23
16. Get coins quantity for each type.....	23
D. Cashless related commands.....	24
1. Reset cashless device.....	24

2. Enable cashless device.....	24
3. Disable cashless device.....	24
4. Cancel current cashless activity.....	25
5. Request cashless current revalue limit.....	25
6. Request approval for a vend (normal mode, basket functions not available or not enabled)	25
6a. Request approval for a vend (basket mode enabled).....	26
7. Confirm a success vend to the cashless device (normal mode, basket functions not available or not enabled).....	26
7a. Confirm a success vend to the cashless device (with basket mode enabled).....	27
8. Report a vend failure to the cashless device (normal mode, basket functions not available or not enabled).....	27
8a. Report a vend failure to the cashless device (basket mode enabled).....	28
9. Report a cash sale to the cashless device.....	28
10. Send a revalue request (the customer's account amount refill).....	29
11. Get last 10 cashless device codes.....	29
12. Check if the cashless device was initialized by the interface.....	29
13. Check if the cashless device was enabled.....	30
14. Get the cashless device information.....	30
15. Get the cashless device settings.....	30
16. Cashless force session complete.....	31
17. Activate/deactivate cashless basket function (if supported by your cashless device).....	31
19. Check if basket function is enabled/disabled (if supported by your cashless device).....	32
.....	32
20. Activate/deactivate cashless basket function (if supported by your cashless device).....	32
E. Interface (VMC) system related commands.....	33
1. Get VMC settings.....	33
2. Set VMC settings.....	33
3. Set VMC manufacturer code.....	33
4. Set VMC internal serial number.....	34
5. Set VMC internal model number.....	34
6. Reset interface internal counters.....	34
7. Interface reboot.....	34
8. Check if the interface is up and running.....	34
9. Read internal counters.....	35
10. Save settings.....	35
11. Factory reset.....	35
12. POLL command.....	36
13. Show current credit values.....	36
14. Delete current credit values.....	36
15. Reading device serial number.....	36
16. Reset all attached MDB devices.....	37
17. Enable all cash peripherals for a specific amount and time.....	37
18. Set RFID reader password (key-A) – only with the optional (sold separately) RFID reader	37
19. Get RFID reader password (key-A) – only with the optional (sold separately) RFID reader	38
20. Enable the RFID reader, using the password you set before with RDRPASS().....	38
21. Get last read card serial number.....	38
22. Erase last read card serial number.....	38
F. Relay related commands.....	39
1. Individual relay control.....	39
IV. Unsolicited messages.....	40

A. Bill validator related messages.....	40
1. Bill validator just reset time exceeded.....	40
2. Bill validator setup time exceeded.....	40
3. Bill validator expansion ID time exceeded.....	40
4. Bill validator expansion ID with options time exceeded.....	40
5. Bill validator optional feature enable time exceeded.....	41
6. Bill validator with recycling support has been identified.....	41
7. Interface will try to enable the recycling support.....	41
8. Bill recycler setup time exceeded.....	41
9. Bill device has no support to recycler any known bills.....	41
10. Bill recycler enabling time exceeded.....	41
11. Bill recycler enabling failed.....	42
12. Bill recycler function successfully enabled.....	42
13. Bill recycler answered with a NAK on enable function.....	42
14. Bill recycler reading dispense status time exceeded.....	42
16. Bill recycler reading dispense status returned a NAK.....	42
16. Bill recycler reading dispense status returned an ACK.....	42
17. Bill recycler reading dispense status returned an ACK.....	42
19. Bill recycler remaining stock value.....	43
19. Bill dispensing command time exceed.....	43
20. Bill dispensing command time exceed.....	43
21. Bill dispensing command time exceed.....	43
22. Bill stacker status command time exceeded.....	43
23. Bill stacker status – stacker full.....	43
24. Bill stacker status – stacker not full.....	43
25. Bill is not ready.....	44
26. Bill validator/recycler failed to answer on poll command.....	44
27. Bill validator – one bill stacked.....	44
28. Bill in escrow position.....	44
29. Bill returned to customer.....	44
30. Bill received in recycler.....	44
31. A disabled bill was rejected.....	45
32. A bill was manually loaded to recycler.....	45
33. A disabled bill was manually dispensed from the recycler.....	45
34. A disabled bill was transferred from the recycler to cashbox.....	45
35. Bill validator is in normal condition.....	45
36. Bill validator have a defective motor.....	45
37. Bill validator have a defective sensor.....	45
38. Bill validator is busy.....	45
39. Bill validator ROM error.....	46
40. Bill validator jam.....	46
41. Bill validator was reset.....	46
42. Bill removed from bill validator.....	46
43. Bill validaor cashbox has been removed.....	46
44. Bill validaor has been disabled by your application or by an internal error.....	46
45. Bill validator has been rejected a bill.....	46
46. Bill removed after it was credited.....	46
47. A bill was inserted while the bill validator is deactivated.....	46
48. Recycler has received a change request.....	46
B. Cashless related unsolicited messages.....	47
1. Cash sale reported to the cashless device.....	47
2. Cash sale successfully reported to the cashless device.....	47
3. Cash sale reporting to the cashless device failed.....	47

4. Cashless device setup time exceeded.....	47
5. Cashless device max/min prices reporting time exceed.....	47
6. Cashless device poll time exceed.....	47
7. Cashless device expansion request ID time exceed.....	48
8. Cashless device expansion enable options time exceed.....	48
9. Cashless device have Always Idle support and it will be enabled.....	48
10. Cashless device writing date/time exceed.....	48
11. Cashless device sent a display message.....	48
12. Cashless device sent a BEGIN SESSION message.....	49
13. Cashless device sent a VEND APPROVED message.....	49
14. Cashless device sent a VEND DENIED message.....	49
15. Cashless device sent an END SESSION message.....	49
16. Cashless device sent a CANCELED message.....	49
17. Cashless device is ready.....	49
18. Cashless device returned a malfunction error.....	49
19. Cashless device returned COMMAND OUT OF SEQUENCE message.....	50
20. Cashless device sent a REVALUE APPROVED message.....	50
21. Cashless device sent a REVALUE DENIED message.....	50
22. Cashless device sent a REVALUE LIMIT message.....	50
23. Cashless device sent a DATE/TIME request message.....	50
24. Interface successfully sent date/time command to the cashless device.....	50
24. Interface failed sending date/time command to the cashless device.....	50
25. Interface successfully enabled the cashless device.....	50
26. Interface successfully enabled the cashless device.....	51
C. Coin related unsolicited messages.....	52
1. Coin acceptor/changer just reset waiting time exceeded.....	52
2. Coin acceptor/changer setup time exceeded.....	52
3. Coin acceptor/changer expansion identification time exceeded.....	52
4. Coin acceptor/changer feature enable time exceeded.....	52
5. Coin acceptor/changer tube status time exceeded.....	52
6. Coin acceptor/changer is not ready for the issued command.....	53
7. Coin acceptor/changer tube status.....	53
8. Coin acceptor/changer poll time exceeded.....	53
9. Coin acceptor/changer is busy dispensing coins.....	53
10. Coin acceptor/changer temporarily unable to dispense coins.....	53
10. Coin acceptor/changer has failed to dispense all or some of the required coins.....	54
11. Coin acceptor/changer is reporting a manual coin dispense.....	54
12. Coin acceptor/changer received a token.....	54
13. Coin acceptor/changer received a coin.....	55
14. Coin acceptor/changer detected a slug.....	55
15. Coin acceptor/changer is in normal condition.....	55
16. Coin acceptor/changer received a change request.....	55
17. Coin acceptor/changer received a coin that was not credited.....	55
18. Coin acceptor/changer has a defective tube sensor.....	55
19. Coin acceptor/changer detected a double arrival.....	56
20. Coin changer detected an accep- a1 – first relay you want to energize (1-10).....	56
21. Coin acceptor/changer detected a tube jam.....	56
22. Coin acceptor/changer detected an internal ROM error.....	56
23. Coin acceptor/changer detected a routing error.....	56
24. Coin acceptor/changer detected reset condition.....	56
25. Coin acceptor/changer detected a coin jam.....	56
26. Coin acceptor/changer detected the removal of a credited coin.....	56
27. Cashless X basket functions are active or not (if supported by the cashless device).....	57

28. Cashless X has basket functions available.....	57
29. Cashless X basket is active when a normal mode function was called.....	58
30. Cashless X basket functions are not active when a normal function was called.....	58
D. RFID reader related unsolicited messages.....	59
1. RFID reader SN received.....	59
2. RFID card removed.....	59
3. RFID card reading failed.....	59
Appendix IV – Cashless basket transaction example.....	60
This section will describe a simple basket transaction for the cashless #1 device (if the device is supporting this function).....	60
A. Prerequisites.....	60
B. Cashless basket transaction.....	60
NOTES:.....	61

I. Introduction

This device was designed for an easy MDB payment systems management, development and integration. It is acting as a vending machine controller (VMC) for the MDB payment systems.

Depending on the loaded firmware, it offers the possibility to connect and develop an MDB master controller device by connecting it to:

- a computer over USB;
- a single board computer (like Raspberry Pi, Orange Pi, etc.) over USB serial interface.
- our SaaS backend over WiFi.

The device is using a simple ASCII protocol over USB interfaces. There is no need for you to have deep MDB knowledge, however, some knowledge about MDB will help you faster and better understand the interface functionality. For SaaS backend, the messages are in JSON format.

Standard package content:

- GalaxyVend MDB MASTER board, no cables provided.

NOTE!!! - this device can handle one MDB bill validator/recycler, one MDB coin acceptor/changer and two MDB cashless devices (Level 2 or Level 3 with always idle cashless device).

During its activity, the interface keeps some internal counters (total number of bills validated, total value of validated bills, coin, cashless transactions, etc.). At any moment, you can read those counters in order to obtain some statistics. Also, the counters may be used as a verification, if some messages are lost. For example, you may constantly poll for counters to check if you missed some bills or coins. USB serial interface is using same communication parameters:

- baudrate – 115200bps;
- data bits – 8;
- stop bits – 1;
- parity – none;
- flow control – no flow control (either hardware or software).

USB interface is using a driver for Windows OS. For Raspberry Pi (or compatible) and most of the Linux distributions with new kernels, the OS will automatically load the correct kernel modules.

You can [DONWLOAD WINDOWS DRIVER](#) on our website.

All answers on USB have <CR> and <LF> at the end. If you are using non-buffered serial interface reading, make sure your application reads until <LF>.

Sometimes, more than one message will be received (for example, a response to your command and an unsolicited message or an answer with the reason of the command fail). You need to receive the entire message and parse it by checking against all command answers related to your last command and also against all available unsolicited message (see unsolicited messages description below).

The MDB device with the master firmware accepts the GalaxyVend KEYPRESS board, but not to emulate key presses, but to control individual external circuits through small signal relays (max. 100mA/100V), active low. You can directly connect to PLSs digital inputs with active low level or you need to add external relays (preferably 24VDC coil, with a maximum 100mA coil draining current). It is also highly recommended to use flyback diodes in parallel with your relays coils.

II. Hardware

A. Core board overview

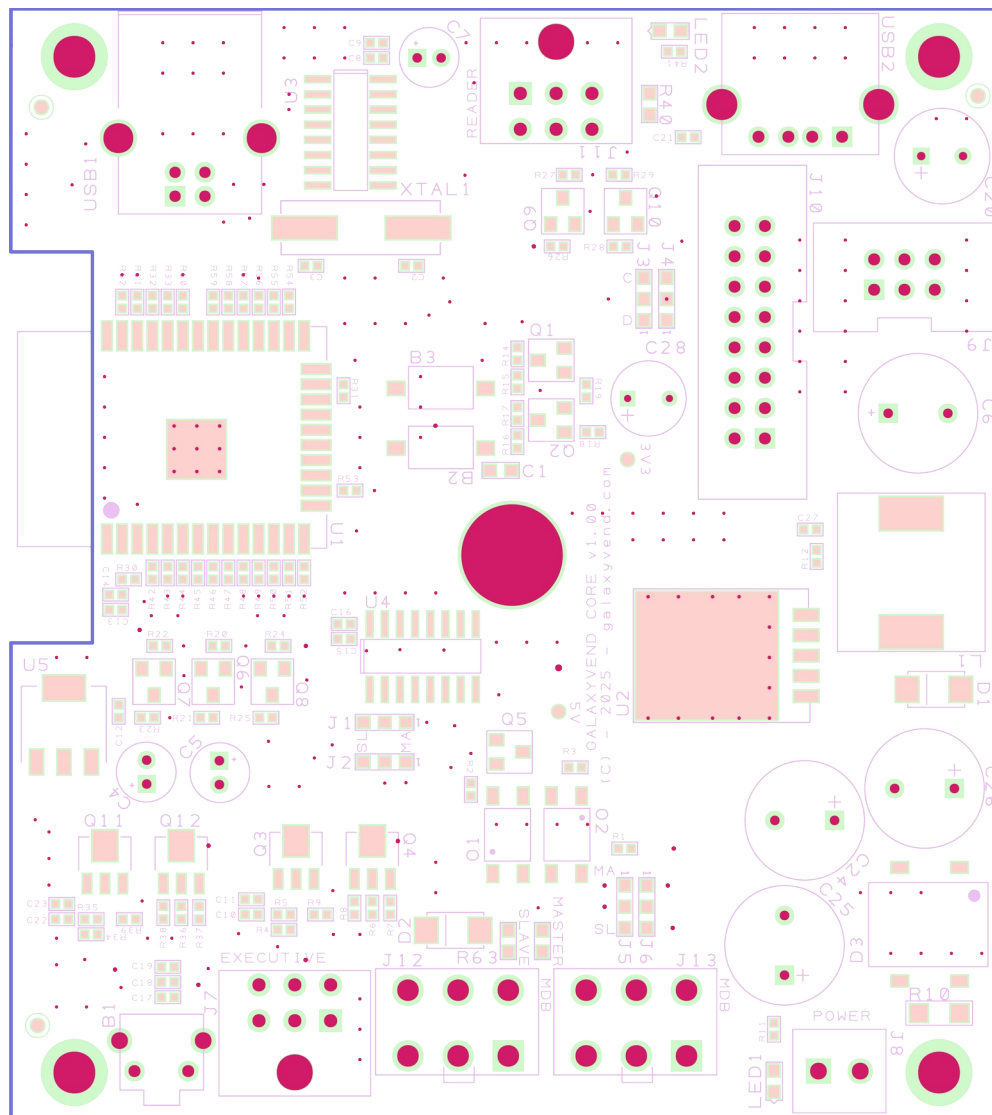


Figure 1: Core board

B. Core board connectors and buttons description

1. **B1** – multi purpose button – it sends the message “MINIBUTTONPRESSED” when pressed and “MINIBUTTONRELEASED” when released. Your application can use this button to activate some tasks (for example, enter application service mode when pressed for 3 seconds).
2. **J7** – **Executive connector** – not used for this device.
3. **J12 and J13** – **MDB connectors** – connectors to attach MDB peripherals (payment systems). You can use any of the connectors (or both, if some of your peripherals do not have an “Y” cable to cascade them).
4. **J8** – **POWER INPUT** – power connector. You need to apply power according to your MDB peripherals requirement. The accepted power input is:
 - 9 to 32VDC (polarity insensitive);
 - 7 to 24VAC.
5. **USB1** – USB-B connector for firmware update and for your app USB communication.
6. **J11** – **RFID reader** – connector to attach the optional (sold separately) RFID (MIFARE Classic) reader.
7. **USB2** – Connector to power external 5VDC devices (max. 1.2A, do not exceed this current)

C. Optional relays board (uses GalaxyVend keypress board – sold separately)

GalaxyVend keypress board is used with individual relays control, not double relays control (like it is used with GalaxyVend MDB Executive converter). MDB master device firmware can individually handle all 10 relays on GalaxyVend keypress board. You cannot simulate keypress functions with the MDB master. Just activating outputs individually is possible. The outputs are active low (tied to the GND when the relay is energized).

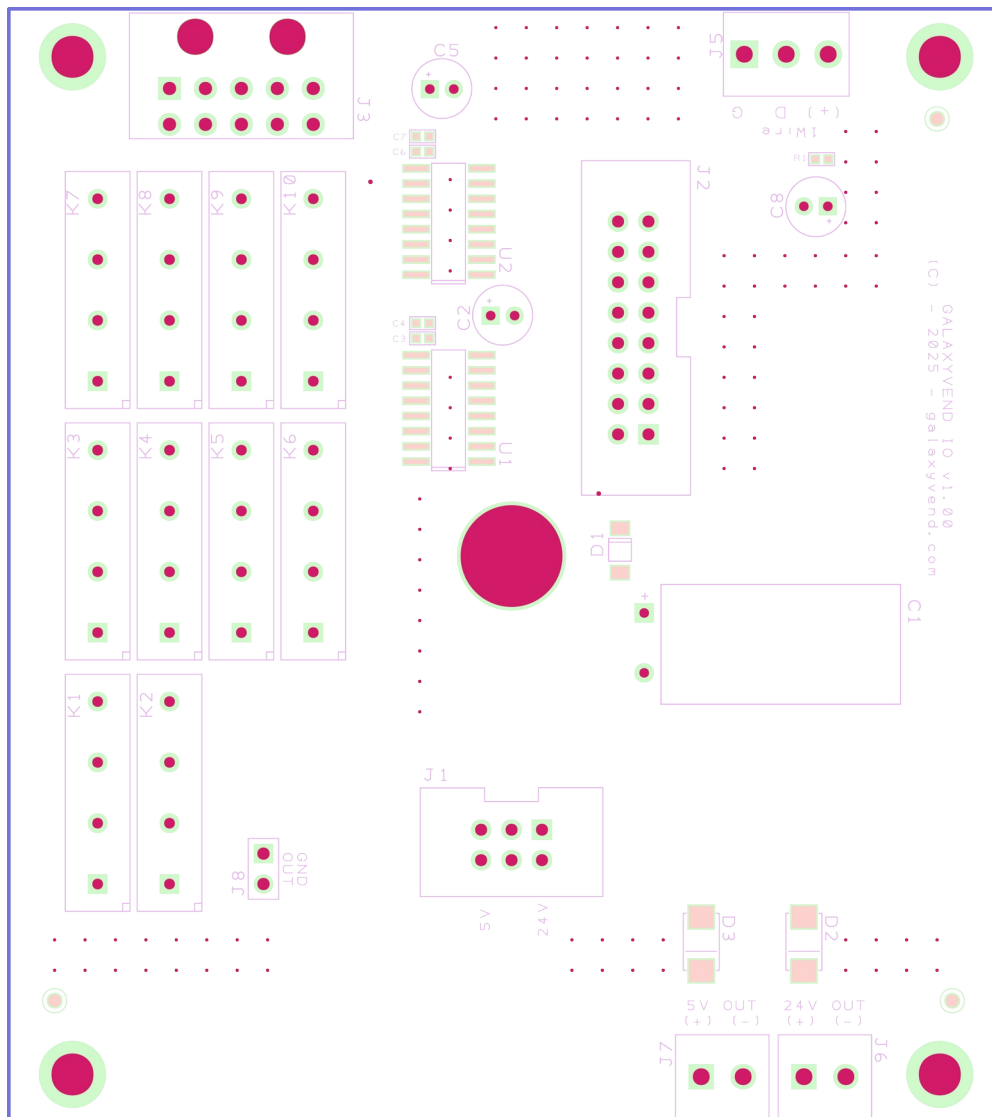


Figure 2: Relays board

D. Relays board connectors

1. J3 – relays output

- PIN #1 (squared pin) – relay #1
- PIN #2 (squared pin) – relay #2
- PIN #3 (squared pin) – relay #3
- PIN #4 (squared pin) – relay #4
- PIN #5 (squared pin) – relay #5
- PIN #6 (squared pin) – relay #6
- PIN #7 (squared pin) – relay #7
- PIN #8 (squared pin) – relay #8

- PIN #9 (squared pin) – relay #9
- PIN #10 (squared pin) – relay #10

2. J5 – 1wire - not used with this device (reserved for future options, for example, sensors). If you need any development that includes 1wire sensors, please contact us.

3. J7 – 5VDC output (max. 1.2A, if you are not using core board 5VDC output, otherwise using this is prohibited in order to avoid draining too much current from the internal PSU).

5. J6 – POWER OUTPUT - this connector outputs the voltage you apply to the core board's power connector (minus 0.6V-0.8V drop on the protection diode) if you apply DC voltage on core board.

E. Communication interfaces

USB serial interface is practically an USB to serial converter embedded on the board and is using the following communication parameters:

- baudrate – 115200bps;
- data bits – 8;
- stop bits – 1;
- parity – none;
- flow control – no flow control (either hardware or software).

USB interface is using a driver for Windows OS. For Raspberry Pi (or compatible), Android and most of the Linux distributions with new kernels, the OS will automatically load the correct kernel modules.

All answers have <CR> and <LF> at the end. If you are using non-buffered serial interface reading, make sure your application reads until <LF>.

Sometimes, more than one message will be received (for example, a response to your command and an unsolicited message or an answer with the reason of the command fail). Your application needs to receive the entire message and parse it by checking against all command answers related to your last command and also against all available unsolicited message (see unsolicited messages description below), breaking the entire payload to sub-strings/messages.

III. Communication protocol

A. MDB master related commands and answers

Commands are case-sensitive and you must use all upper case or all lower case for a command. No mixed characters accepted.

For simplicity, we will only use upper case in our documentation.

Answers are always upper case.

B. Bill validator/recycler related commands

1. Enable bill validator (no escrow support)

Command	
BILLENABLE	This command will enable all supported bill acceptance on the bill validator. It enables the bill validator with bill escrow function disabled (even if the bill validator is configured with the escrow function active).
Possible answers	
- BILLENABLEFAIL	- If the bill validator could not be enabled when the BILLENABLE is received, you will receive this answer. Possible reasons (but not limited to those) could be: bill validator was not initialized, bill validator already enabled, MDB communication error, etc. - If the command is correctly received and interpreted by the interface.
- BILLENABLEOK	

2. Enable bill validator (with escrow support)

Command	
BILLENABLEESCROW	This command will enable all supported bill acceptance on the bill validator. It enables the bill validator with bill escrow function enabled (if then validator is configured with the escrow function active).
Possible answers	
- BILLENABLEESCROWFAIL	- If the bill validator could not be enabled when the BILLENABLE is received, you will receive this answer. Possible reasons (but not limited to those) could be: bill validator was not initialized, bill validator already enabled, MDB communication error, etc. - If the command is correctly received and interpreted by the interface.
- BILLENABLEESCROWOK	

3. Selective bills enable

Command	
BILLSELECTENABLE(X)	This command will enable some of supported bill acceptance on the bill validator. If the bill validator is supporting escrow function, it will be automatically activated. - X is a 16bit number corresponding with MDB BILL TYPE COMMAND (0x34) BILL ENABLE parameter. Bit 0 correspond with bill type 0 and bit 15 correspond with the bill type 15. This command allows your application to enable only specific bill/bills. You can obtain bill type values after the interface is automatically initialized the bill validator/recycler using BILLVALUES? command, explained later below.
Possible answers	
- BILLSELECTENABLEFAIL - BILLSELECTENABLEOK	- If the bill validator could not be enabled when the BILLSELECTENABLE is received, you will receive this answer. Possible reasons (but not limited to those) could be: bill validator was not initialized, bill validator already enabled, MDB communication error, etc. - If the command is correctly received and interpreted by the interface.

4. Disable bill validator

Command	
BILLDISABLE	This command will disable all bills acceptance. Your application may disable the bill validator when the maximum allowed credit value has been reached, when a malfunction occurs or during product dispensing/preparation
Possible answers	
- BILLDISABLEFAIL - BILLDISABLEOK	- If the bill validator could not be disabled when the BILLDISABLE is received, you will receive this answer. Possible reasons (but not limited to those) could be: bill validator was not initialized, bill validator already disabled, MDB communication error, etc. - If the command is correctly received and interpreted by the interface.

5. Reset bill validator

Command	
BILLRESET	This command will reset the bill validator. The interface will automatically initialize the bill validator again and your application needs to enable or selective enable it do make it available for receiving bills. After issuing this command, you will receive some unsolicited messages while the interface is initializing the bill validator. You may receive the following messages: - BILLSTACKNOTFULL(X) - BILLREADY - BILLOK Please check the unsolicited messages information below in the "Interface unsolicited messages" section.
Possible answers	
- BILLRESETFAIL	- If the bill validator could not be reset when the BILLRESET is received, you will receive this answer. Possible reasons (but not limited to those) could be: bill validator was not initialized, bill validator already disabled, MDB communication error, etc. - If the command is correctly received and interpreted by the interface.
- BILLRESETOK	

6. Approve bill acceptance while a bill is in escrow position

Command	
BILLACCEPT	This command will send the ACCEPT command to the bill validator after the BILLESCROW(X) unsolicited message was received.
Possible answers	
- BILLACCEPTFAIL	- If the bill validator could not be reset when the BILLACCEPT is received, you will receive this answer. Possible reasons (but not limited to those) could be: bill validator was not initialized, bill validator already disabled, MDB communication error, etc. - If the command is correctly received and interpreted by the interface.
- BILLACCEPTOK	

7. Reject bill while a bill is in escrow position

Command	
BILLREJECT	This command will send the REJECT command to the bill validator after the BILLESCROW(X) unsolicited message was received.
Possible answers	
- BILLREJECTFAIL	- If the bill validator could not be reset when the BILLACCEPT is received, you will receive this answer. Possible reasons (but not limited to those) could be: bill validator was not initialized, bill validator already disabled, MDB communication error, etc. - If the command is correctly received and interpreted by the interface.
- BILLREJECTOK	

8. Get last 10 bill status codes

Command	
BILLSTATUS?	This command will ask for the last 10 bill validator status codes. Your application can use this to periodically ask the bill validator status, if it missed some unsolicited messages.
Possible answers	
- BILLSTATUS(A,B,C,D,E,F,G,H,I,J)	- A to J are some byte values, corresponding with the bill validator status bytes received on bill poll. You need to check with MDB documentation for the bytes interpretation. For example, 8 means "cashbox removed". This vector is a FIFO loop and you may need to read it periodically.

9. Check if the bill validator was initialized by the interface

Command	
BILLINITED?	This command will check if the bill validator was initialized by the interface after power-up or after issuing BILLRESET command
Possible answers	
- BILLINITEDOK - BILLNOTINITED	- The bill validator was successfully initialized - The bill validator was not initialized (missing or not initialized, yet).

10. Check if the bill validator was enabled

Command	
BILLACTIVE?	This command will check if the bill validator was previously activated by a BILLEENABLE or a BILLSELECTENABLE command.
Possible answers	
- BILLACTIVEOK - BILLNOTACTIVE	- The bill validator is currently enabled - The bill validator is not currently enabled

11. Get the bill validator configured bills values

Command	
BILLVALUES?	This command will read the bill validator configured bills values. This vector is read during automatic bill validator initialization phase, after a power-up or after issuing BILLRESET command.
Possible answers	
- BILLVALUES(A,B,C,D,E,F,G,H,I,J,K,L,M,N,O,P) - BILLNOTINITED	- A to P are the scaled values of the bills recognized and accepted by the bill validator. You can use this to obtain necessary information for selective bill activation in order to avoid accepting bills values higher than the maximum accepted credit. - The bill validator was not previously initialized and the bills values information is not available.

12. Get the bill validator information

Command	
BILLINFOREQ?	This command will read the bill validator information for statistics and payment systems inventory tracking. This information is read during automatic bill validator initialization phase, after a power-up or after issuing BILLRESET command.
Possible answers	
- BILLINFOREQ(A,B,C) - BILLNOTINITED	- A is the bill validator manufacturer code, fixed length – 3 characters (ASCII) - B is the bill validator internal serial number, fixed length, 12 characters (ASCII) - C is the bill validator internal model number, fixed length, 12 characters (ASCII) - The bill validator was not previously initialized and the information is not available

13. Get the bill validator settings

Command	
BILLSETTINGS?	This command will read the bill validator settings. This information is read during automatic bill validator initialization phase, after a power-up or after issuing BILLRESET command.
Possible answers	
- BILLSETTINGS(A,B,C,D,E,F) - BILLNOTINITED	- A is the bill validator feature level (decimal) - B is the bill validator country code (HEX) - C is the bill validator scaling factor (decimal) - D is the bill validator decimal places (decimal) - E is the bill validator stacker capacity (decimal) - F is the bill validator escrow support (1 if the bill validator supports escrow function or 0 if the bill validator does not support escrow function) - The bill validator was not previously initialized and the information is not available

14. Get the bill recycler bill type values

Command	
RECYCLERBILLS?	This command will read the bill recycler accepted bills values. This information is read during automatic bill recycler initialization phase, after a power-up or after issuing BILLRESET command.
Possible answers	
- RECYCLERBILLS(A,B,C,D,E,F,G,H,I,J,K,L,M,N,O,P) - BILLNOTINITED	- A to P is the flag for the bills that the bill recycler can recycle (can give back to the customers for payout/change). If the value is 0, the corresponding bill value is not available for recycling. If the value is 1, the recycler can use the corresponding bill For recycling. Use BILLVALUES to obtain the real bills value. - The bill validator was not previously initialized and the information is not available

15. Get the bills set for recycling by the user application

Command	
RECYCLERSETBILLS?	This command will read the bills enabled for recycling, by the user application
Possible answers	
- RECYCLERSETBILLS(A,B,C,D,E,F,G,H,I,J,K,L,M,N,O,P)	- A to P is the value for recycling mode: - 0 – this bill is not enabled for recycling; - 1 – only high quality bills are enabled for recycling; - 2 – only high and medium bills are enabled for recycling; - 3 – use all possible bills for recycling (this is the recommended option)

16. Set the bills set for recycling by the user application

Command	
RECYCLERSETBILLS(A,B,C,D,E,F,G,H,I,J,K,L,M,N,O,P)	This command will set the bills enabled for recycling, by the user application - A to P is the value for recycling mode: - 0 – this bill is not enabled for recycling; - 1 – only high quality bills are enabled for recycling; - 2 – only high and medium bills are enabled for recycling; - 3 – use all possible bills for recycling (this is the recommended option)
Possible answers	
- RECYCLERSETBILLSOK	- The bill enabled for recycling were successfully set.

17. Get tot total value of bills available for recycling

Command	
RECYCLERSTOCKVALUE?	This command will read the total value of the bill available for recycling in the bill recycler
Possible answers	
- RECYCLERSTOCKVALUE(X)	- X is total value of the bills available for recycling in the bill recycler

18. Dispense bills as a change to customer

Command	
BILLDISPENSEVALUE(X)	This command will start the bill dispensing operation. - X is the value that bill recycler should dispense to the customer.
Possible answers	
- BILLDISPENSEVALUEOK - BILLDISPENSEVALUEFAIL	- If the command is successfully sent to the bill recycler - If the command fails while sending to recycler

19. Dispense bills as a change to customer

Command	
BILLDISPENSEVALUE(X)	This command will start the bill dispensing operation. - X is the value that bill recycler should dispense to the customer.
Possible answers	
- BILLDISPENSEVALUEOK - BILLDISPENSEVALUEFAIL - BILLDISPENSERNOTENABLED	- If the command is successfully sent to the bill recycler - If the command fails while sending to recycler - if the dispensing function was not enabled

20. Get current bill stacker status

Command	
BILLSTACKER?	This command will read the current bill stacker status
Possible answers	
- BILLSTACKER(X) - BILLSTACKEROK - BILLSTACKERFAIL	- X is the number of bills currently in the bill validator stacker. - If the command is successfully sent to the bill validator. - If the command fails while sending to the bill validator.

C. Coin acceptor/changer related commands

1. Enable coin acceptor/changer

Command	
COINENABLE	This command will enable all supported coins acceptance on the coin acceptor.
Possible answers	
- COINENABLEFAIL	- If the coin acceptor/changer could not be enabled when the COINENABLE is received, you will receive this answer. Possible reasons (but not limited to those) could be: coin acceptor was not initialized, coin acceptor already enabled, MDB communication error, etc.
- COINENABLEOK	- If the command is correctly received and interpreted by the interface.

2. Selective coins enable

Command	
COINSELECTENABLE(X)	This command will enable some of supported coins acceptance on the coin acceptor. If the coin acceptor is supporting change function, the manual coin dispense will be automatically enabled - X is a 16bit number corresponding with MDB COIN TYPE COMMAND (0x0C) COIN ENABLE parameter. Bit 0 correspond with bill type 0 and bit 15 correspond with the bill type 15. This command allows your application to enable only specific coin/coins. You can obtain coin type values after the interface is automatically initialized the coin acceptor/changer using COINVALUES? command, explained later below.
Possible answers	
- COINSELECTENABLEFAIL	- If the coin acceptor could not be enabled when the COINSELECTENABLE is received, you will receive this answer. Possible reasons (but not limited to those) could be: coin acceptor was not initialized, coin acceptor already enabled, MDB communication error, etc.
- COINSELECTENABLEOK	- If the command is correctly received and interpreted by the interface.

3. Disable coin acceptor

Command	
COINDISABLE	This command will disable all coins acceptance. Your application may disable the coin acceptor when the maximum allowed credit value has been reached, when a malfunction occurs or during product dispensing/preparation
Possible answers	
- COINDISABLEFAIL	- If the bill validator could not be disabled when the BILLDISABLE is received, you will receive this answer. Possible reasons (but not limited to those) could be: bill validator was not initialized, bill validator already disabled, MDB communication error, etc. - If the command is correctly received and interpreted by the interface.
- COINDISABLEOK	

4. Reset coin acceptor

Command	
COINRESET	This command will reset the coin acceptor/changer. The interface will automatically initialize the coin acceptor/changer again and your application needs to enable or selective enable it do make it available for receiving coins. After issuing this command, you will receive some unsolicited messages while the interface is initializing the bill validator. You may receive the following messages: - COINREADY - COINOK Please check the unsolicited messages information below in the “Interface unsolicited messages” section.
Possible answers	
- COINRESETFAIL	- If the coin acceptor/changer could not be reset when the COINRESET is received, you will receive this answer. Possible reasons (but not limited to those) could be: coin acceptor was not initialized, coin acceptor already disabled, MDB communication error, etc. - If the command is correctly received and interpreted by the interface.
- COINRESETOK	

5. Get total value of coins in tubes (for coin changers only)

Command	
COINTBSTATUS?	This command will get the total coins value in changer's tubes. For changers with more than 255 same type coins on a tube or multiple tubes, the changer always returns 255 for a tube. Do not use this command for inventory management.
Possible answers	
- COINTBSTATUS(X)	- X is the total scaled value of the coins in the coin changer tubes. - If the command is correctly received and interpreted by the interface. - If the command was not correctly received and interpreted by the interface.
- COINTBSTATUSOK	
- COINTBSTATUSFAIL	

6. Dispense some coins (change) to the customer – obsolete, try to use COINAP command whenever the coin acceptor/changer supports it.

Command	
COINDISPENSE(X)	This command will start coin dispensing for the X value (for example, COINDISPENSE(120) will dispense 1.20EUR. Use this command instead of COINDISPENSE whenever the coin changer is supporting it.
Possible answers	
- COINDISPENSEOK - COINPAYBUSY - COINDISPENSEFAIL - REMAINING(X)	- If the command is correctly received and interpreted by the interface and, also, the changer managed to successfully or not dispensed the coins - You will receive this message until the changer manages to return the entire amount or fails for some reason (not enough change, - If the command was not correctly received and interpreted by the interface or if the changer is returning an error. - X is the total value that could not be dispensed by the changer (due to an internal error, missing coins stock, etc.)

7. Dispense some coins using MDB alternative payout method

Command	
COINAP(X)	This command will start coin dispensing for the X value (for example, COINDISPENSE(120) will dispense 1.20EUR. This command is obsolete and you must use COINAP instead if the coin changer supports it. Using this command is much slower than the COINAP command since it will dispense one coin at a time.
Possible answers	
- COINAPOK - COINPAYBUSY - COINAPFAIL - REMAINING(X)	- If the command is correctly received and interpreted by the interface and, also, the changer managed to successfully or not dispensed the coins - You will receive this message until the changer manages to return the entire amount or fails for some reason (not enough change, - If the command was not correctly received and interpreted by the interface or if the changer is returning an error. - X is the total value that could not be dispensed by the changer (due to an internal error, missing coins stock, etc.)

8. Check if the coin acceptor/changer was initialized by the interface

Command	
COININITED?	This command will check if the coin acceptor was initialized by the interface after power-up or after issuing COINRESET command
Possible answers	
- COININITEDOK - COINOTINITED	- The bill validator was successfully initialized - The bill validator was not initialized (missing or not initialized, yet).

9. Check if the coin acceptor/changer was enabled

Command	
COINACTIVE?	This command will check if the coin acceptor/changer was previously activated by a COINENABLE or a COINSELECTENABLE command.
Possible answers	
- COINACTIVEOK - COINNOTACTIVE	- The coin acceptor/changer is currently enabled - The coin acceptor/changer is not currently enabled

10. Get last 10 coin acceptor/changer codes

Command	
COINSTATUS?	This command will ask for the last 10 coin acceptor/changer status codes. Your application can use this to periodically ask the coin acceptor/changer status, if it missed some unsolicited messages.
Possible answers	
- COINSTATUS(A,B,C,D,E,F,G,H,I,J)	- A to J are some byte values, corresponding with the coin acceptor/changer status bytes received on coin poll. You need to check with MDB documentation for the bytes interpretation. For example, 7 means "tube jam". This vector is a FIFO loop and you may need to read it periodically.

11. Get the coin acceptor/changer configured coins values

Command	
COINVALUES?	This command will read the coin acceptor/changer configured coins values. This vector is read during automatic coin validator initialization phase, after a power-up or after issuing COINRESET command.
Possible answers	
- COINVALUES(A,B,C,D,E,F,G,H,I,J,K,L,M,N,O,P) - COINNOTINITED	- A to P are the scaled values of the coins recognized and accepted by the coin acceptor/changer. You can use this to obtain necessary information for selective coin activation in order to avoid accepting coins values higher than the maximum accepted credit. - The coin acceptor/changer was not previously initialized and the coins values information is not available.

12. Get the coin acceptor/changer information

Command	
COININFOREQ?	This command will read the coin acceptor/changer information for statistics and payment systems inventory tracking. This information is read during automatic coin acceptor/changer initialization phase, after a power-up or after issuing COINRESET command.
Possible answers	
- COININFOREQ(A,B,C) - COINNOTINITED	- A is the coin acceptor/changer manufacturer code, fixed length – 3 characters (ASCII) - B is the coin acceptor/changer internal serial number, fixed length, 12 characters (ASCII) - C is the coin acceptor/changer internal model number, fixed length, 12 characters (ASCII) - The coin acceptor was not previously initialized and the information is not available

13. Get the coin acceptor/changer settings

Command	
COINSETTINGS?	This command will read the bill validator settings. This information is read during automatic bill validator initialization phase, after a power-up or after issuing COINRESET command.
Possible answers	
- COINSETTINGS(A,B,C,D,E,F,G,H,I,J,K,L,M,N,O,P,Q,R,S) - COINNOTINITED	- A is the coin acceptor feature level (decimal) - B is the coin acceptor country code (HEX) - C is the coin acceptor scaling factor (decimal) - D is the coin acceptor decimal places (decimal) - E to S are tube flags. Each coin type where the corresponding flag is set to 1, can be stored in changer's tubes and used for change. Each coin type where the corresponding flag is 0, cannot be stored in changer's tubes. - The coin acceptor/changer was not previously initialized and the information is not available

14. Get the token values

Command	
TOKENVALUES?	This command will read coin tokens set into the interface memory.
Possible answers	
- TOKENVALUES(A,B,C,D,E,F,G,H,I,J,K,L,M,N,O,P) - COINNOTINITED	- A to P are the value set for each token. These values are used if you have connected a coin acceptor/changer that is sending 0XFF for coin values if a token is accepted. You don't need to use ththat if your coin acceptor/changer is directly reporting the token value. - The coin acceptor/changer was not previously initialized and the information is not available

15. Set the token values

Command	
- TOKENVALUES(A,B,C,D,E,F,G,H,I,J,K,L,M,N,O,P)	This command will set the token values in the interface. - A to P are the value set for each token. These values are used if you have connected a coin acceptor/changer that is sending 0xFF for coin values if a token is accepted. You don't need to use ththat if your coin acceptor/changer is directly reporting the token value.
Possible answers	
- TOKENVALUESOK - COINNOTINITED	- The coin acceptor/changer have been set. - The coin acceptor/changer was not previously initialized and you cannot set this value.

16. Get coins quantity for each type

Command	
- COINTBVALUES?	This command is returning the number of each coin type available in the changer's tubes
Possible answers	
- COINTBVALUES(A,B,C,D,E,F,G,H,I,J,K,L,M,N,O,P) - COINNOTINITED	- A to P is the number of each coin in tubes, reported by the coin changer. The accuracy of those values are directly dependable by the coin changer sensors and settings (for example, the security stock settings, etc.). You can calculate the available value for each coin type in the tubes by multiplying each A to P value on this command answer with the corresponding A to P value in COINVALUES? command. CONTBVALUES? is returning the number of coins in each tube, not the value of coins in tubes.

D. Cashless related commands

All commands and messages have the following format CSLS<X>CMD where <X> can be “1” or “2”, depending on the cashless number you want to address (1 is for the cashless with MDB address 0x10 and 2 is for the cashless with MDB address 0x60), for example CSLS1RESET or CSLS2RESET.

1. Reset cashless device

Command	
CSLS<X>RESET	This command will reset the cashless device After issuing this command, you will receive some unsolicited messages while the interface is initializing the cashless device. You may receive the following messages: - CSLS<X>READY - CSLS<X>OK Please check the unsolicited messages information below in the “Interface unsolicited messages” section.
Possible answers	
- CSLS<X>RESETFAIL - CSLS<X>RESETOK	- If the cashless device could not be reset when the CSLS<X>RESET is received, you will receive this answer. Possible reasons (but not limited to those) could be: cashless device was not initialized, MDB communication error, etc. - If the command is correctly received and interpreted by the interface.

2. Enable cashless device

Command	
CSLS<X>ENABLE	This command will enable the cashless device.
Possible answers	
- CSLS<X>ENABLEFAIL - CSLS<X>ENABLEOK	- If the cashless device could not be enabled when the CSLS<X>ENABLE is received, you will receive this answer. Possible reasons (but not limited to those) could be: cashless device was not initialized, cashless device already enabled, MDB communication error, etc. - If the command is correctly received and interpreted by the interface.

3. Disable cashless device

Command	
CSLS<X>DISABLE	This command will disable the cashless device.
Possible answers	
- CSLS<X>DISABLEFAIL - CSLS<X>DISABLEOK	- If the cashless device could not be disabled when the CSLS<X>DISABLE is received, you will receive this answer. Possible reasons (but not limited to those) could be: cashless device was not initialized, cashless device already disabled, MDB communication error, etc. - If the command is correctly received and interpreted by the interface.

4. Cancel current cashless activity

Command	
CSLS<X>CANCEL	This command will cancel all current cashless device activities.
Possible answers	
- CSLS<X>CANCELFAIL	- If the cashless device could not be disabled when the CSLS<X>CANCEL is received, you will receive this answer. Possible reasons (but not limited to those) could be: cashless device was not initialized, cashless device already idle, MDB communication error, etc. - If the command is correctly received and interpreted by the interface.
- CSLS<X>CANCELOK	

5. Request cashless current revalue limit

Command	
CSLS<X>REVALLIMITREQ?	This command will read the current revalue limit.
Possible answers	
- CSLS<X>REVALLIMIT(X)	- X is the maximum revalue amount accepted by the cashless device for further CSLS<X>REVALREQ (cashless revalue request) command. - Cashless device is not in session so, the revalue is not available. - If the cashless device or media does not support revalue command - Revalue limit request command was not successfully executed.
- CSLS<X>NOSESSION	
- CSLS<X>NOREVALSUPPORT	
- CSLS<X>REVALLIMITFAIL	

6. Request approval for a vend (normal mode, basket functions not available or not enabled)

Command	
CSLS<X>VNDREQ(A,B)	This command will request a vend approval from the cashless device in normal mode (Level 2 cashless devices and Level 3 without active basket functions) - A is the scaled price (16bit value maximum) - B is the item ID/selection number (16bit value maximum)
Possible answers	
- CSLS<X>NOSESSION	- You will receive this answer if you are requesting for a vend approval and the cashless device is Level 2 or Level 3 without Always Idle support and a cashless session is not opened. - If the interface successfully received and parsed the command. - If the interface was not able to successfully receive and parse the command.
- CSLS<X>VNDREQOK	
- CSLS<X>VNDREQFAIL	

6a. Request approval for a vend (basket mode enabled)

Command	
CSLS<X>VNDREQ(A,B,C)	This command will request a vend approval from the cashless device - A is the total scaled price (16bit value maximum) of all items in the basket that will be sold (must include the options price (C value – products + options); - B is the number of items in the basket - C is the total scaled price of the options if applicable (ex. Sugar, pallets, etc.) Usually, you will include in A the value of C and set C to 0 all the time, if not optional items will be dispensed (like a box, a bag, a cup cover, etc.)
Possible answers	
- CSLS<X>NOSESSION - CSLS<X>VNDREQOK - CSLS<X>VNDREQFAIL	- You will receive this answer if you are requesting for a vend approval and the cashless device is Level 2 or Level 3 without Always Idle support and a cashless session is not opened. - If the interface successfully received and parsed the command. - If the interface was not able to successfully receive and parse the command.

7. Confirm a success vend to the cashless device (normal mode, basket functions not available or not enabled)

Command	
CSLS<X>VNDSUCC(A)	This command will confirm the product dispensing was successful - A is the item ID/selection number (16bit value maximum) that was successfully dispensed
Possible answers	
- CSLS<X>NOSESSION - CSLS<X>VNDSUCCOK - CSLS<X>VNDSUCCFAIL	- You will receive this answer if you are trying to send a vend success in a stage that is not expecting this command - If the interface successfully received and parsed the command. - If the interface was not able to successfully receive and parse the command.

7a. Confirm a success vend to the cashless device (with basket mode enabled)

Command	
CSLS<X>VNDSUCC(A,B,C,D)	This command will confirm the product dispensing was successful - A is the item ID/selection number (16bit value maximum) that was successfully dispensed; - B is the item's price; - C is the number of items remaining to dispense in the basket after this (current) product was dispensed. For the last product dispensed in a basket, C must be 0. If you need to interrupt the basket products at any time, you just need to set C to 0 and the basket transaction will be closed even if you did not finished all products dispensing; - D is the optional items price (must be included in A), usually 0 unless you are dispensing optional items like cups covers, etc.
Possible answers	
- CSLS<X>NOSESSION - CSLS<X>VNDSUCCOK - CSLS<X>VNDSUCCFAIL	- You will receive this answer if you are trying to send a vend success in a stage that is not expecting this command - If the interface successfully received and parsed the command. - If the interface was not able to successfully receive and parse the command.

8. Report a vend failure to the cashless device (normal mode, basket functions not available or not enabled)

Command	
CSLS<X>VNDFAIL	This command will report a vend failure to the cashless device. Usually, the cashless device must restore funds to the customer's account.
Possible answers	
- CSLS<X>NOSESSION - CSLS<X>VNDFAILOK - CSLS<X>VNDFAILFAIL	- You will receive this answer if you are trying to send a vend failure in a stage that is not expecting this command - If the interface successfully received and parsed the command. - If the interface was not able to successfully receive and parse the command.

8a. Report a vend failure to the cashless device (basket mode enabled)

Command	
CSLS<X>VNDFAIL(A,B,C,D)	This command will communicate that the product dispensing failed - A is the item ID/selection number (16bit value maximum) that was successfully dispensed; - B is the item's price; - C is the number of items remaining to dispense in the basket after this (current) product was dispensed. For the last product dispensed in a basket, C must be 0. If you need to interrupt the basket products at any time, you just need to set C to 0 and the basket transaction will be closed even if you did not finished all products dispensing; - D is the optional items price (must be included in B), usually 0 unless you are dispensing optional items like cups covers, etc.
Possible answers	
- CSLS<X>NOSESSION - CSLS<X>VNDFAILOK - CSLS<X>VNDFAILFAIL	- You will receive this answer if you are trying to send a vend failure in a stage that is not expecting this command - If the interface successfully received and parsed the command. - If the interface was not able to successfully receive and parse the command.

9. Report a cash sale to the cashless device

Command	
CSLS<X>CASHSALE(A,B)	This command will report a cash sale to the cashless device. This is used for statistic purposes and not all cashless devices may recognize this command. You should test with the cashless device prior to use that. - A is the scaled price (16bit value maximum) - B is the item ID/selection number (16bit value maximum)
Possible answers	
- CSLS<X>NOCASHSALESUPPORT - CSLS<X>CASHSALEOK - CSLS<X>CASHSALEFAIL	- You will receive this answer if you are trying to send a cashless cash sale command, but the cashless device is not supporting this sale subcommand. - If the interface successfully received and parsed the command. - If the interface was not able to successfully receive and parse the command.

10. Send a revalue request (the customer's account amount refill)

Command	
CSLS<X>REVALREQ(A)	This command will add some amount to customer's account. - A is the scaled amount your application needs to add to customer's account (16bit value maximum)
Possible answers	
- CSLS<X>NOREVALSUPPORT	- You will receive this answer if you are trying to send a cashless revalue command, but the cashless device is not supporting revalue (is not able to load the amount to the customer's account) - If there is no cashless session opened, the cashless device will not be able to load any amount to customer's account. - The specified amount exceeds the cashless maximum revalue capacity for the current session. - If the interface successfully received and parsed the command. - If the interface was not able to successfully receive and parse the command.
- CSLS<X>NOSESSION	
- CSLS<X>REVALOVER	
- CSLS<X>REVALREQOK	
- CSLS<X>REVALREQFAIL	

11. Get last 10 cashless device codes

Command	
CSLS<X>STATUS?	This command will ask for the last 10 cashless device status codes. You application can use this to periodically ask the cashless device status, if it missed some unsolicited messages.
Possible answers	
- CSLS<X>STATUS(A,B,C,D,E,F,G,H,I,J)	- A to J are some byte values, corresponding with the cashless device status bytes received on cashless poll. You need to check with MDB documentation for the bytes interpretation. For example, 8 means "cashbox removed". This vector is a FIFO loop and you may need to read it periodically.

12. Check if the cashless device was initialized by the interface

Command	
CSLS<X>INITED?	This command will check if the cashless device was initialized by the interface after power-up or after issuing CSLSxRESET command
Possible answers	
- CSLS<X>INITEDOK - CSLS<X>NOTINITED	- The cashless device was successfully initialized - The cashless device was not initialized (missing or not initialized, yet).

13. Check if the cashless device was enabled

Command	
CSLS<X>ACTIVE?	This command will check if the cashless was previously activated by a CSLSxENABLE.
Possible answers	
- CSLS<X>ACTIVEOK - CSLS<X>NOTACTIVE	- The cashless device is currently enabled - The cashless device is not currently enabled

14. Get the cashless device information

Command	
CSLS<X>INFOREQ?	This command will read the cashless device information for statistics and payment systems inventory tracking. This information is read during automatic coin acceptor/changer initialization phase, after a power-up or after issuing CSLSxRESET command.
Possible answers	
- CSLS<X>INFOREQ(A,B,C) - CSLS<X>NOTINITED	- A is the cashless device manufacturer code, fixed length – 3 characters (ASCII) - B is the cashless device internal serial number, fixed length, 12 characters (ASCII) - C is the cashless device internal model number, fixed length, 12 characters (ASCII) - The cashless device was not previously initialized and the information is not available

15. Get the cashless device settings

Command	
CSLS<X>SETTINGS?	This command will read the cashless device settings. This information is read during automatic cashless device initialization phase, after a power-up or after issuing CSLSxRESET command.
Possible answers	
- CSLS<X>SETTINGS(A,B,C,D,E,F) - CSLS<X>NOTINITED	- A is the cashless device feature level (decimal) - B is the cashless device country code (HEX) - C is the cashless device scaling factor (decimal) - D is the cashless device decimal places (decimal) - E is the cashless device maximum application time (decimal) - F is the cashless device option bits as described in the MDB specifications: - b0 – if set, the payment media is able to accept revalue command; - b1 – if set, the cashless device is multivend capable; - b2 – if set, the cashless device has it's own display; - b3 – if set, the cashless device is supporting cash sale reporting - The bill cashless device was not previously initialized and the information is not available

16. Cashless force session complete

Command	
CSLS<X>SESSCOMPLETE?	This command will force closing current cashless session. For multi vend cashless devices there is no effect, since they will immediately begin a new session if the media support still inserted. Usually if the cashless device is configured with single vend option, it will automatically close the current session right after settlement. Still there are some poorly implemented cashless devices on the market that may require this command. CSLS<x>RESET can be also used as a workaround for those cashless device that are not automatically request session closing.
Possible answers	
- CSLS<X>SESSCOMPLETEOK - CSLS<X>SESSCOMPLETEFAIL	- Command successfully sent - Command could not be sent

17. Activate/deactivate cashless basket function (if supported by your cashless device)

Command	
CSLS<X>BASKET(A)	This command will activate or deactivate cashless X basket function (if supported by the cashless itself, check with the manufacturer, first). "A" can take the following values: - 0 – deactivate basket functions; - 1 – activate basket functions. In order to make this setting persistent (keep the parameter value after reboot), you need to issue the SAVESETTINGS command. Also, after changing this parameter value, you will need to issue a CSLS<X>RESET command in order to reinitialize the cashless device with the new context. When the basket functions are enabled, you need to use the alternative commands for VEND REQUEST, VEND SUCCESS and VEND FAILURE (see the VNDREQ, VNDSUCC and VNDFAIL commands descriptions for details).
Possible answers	
- CSLS<X>BASKETOK(X) - CSLS<X>BASKETFAIL	- You will receive this answer if the parameter is correctly configured - You will receive this answer if the parameter configuration failed

19. Check if basket function is enabled/disabled (if supported by your cashless device)

Command	
CSLS<X>BASKET?	This command checks if the basket functions are enabled or not for the connected cashless device.
Possible answers	
- CSLS<X>BASKET(A)	- "A" is the current interface basket configuration (0 – the interface will not activate cashless X basket functions, 1 – the interface will activate cashless X basket functions). The interface will not activate basket functions if the

20. Activate/deactivate cashless basket function (if supported by your cashless device)

Command	
CSLS<X>BASKET(A)	This command will activate or deactivate cashless X basket function (if supported by the cashless itself, check with the manufacturer, first). "A" can take the following values: - 0 – deactivate basket functions; - 1 – activate basket functions. In order to make this setting persistent (keep the parameter value after reboot), you need to issue the SAVESETTINGS command. Also, after changing this parameter value, you will need to issue a CSLS<X>RESET command in order to reinitialize the cashless device with the new context. When the basket functions are enabled, you need to use the alternative commands for VEND REQUEST, VEND SUCCESS and VEND FAILURE (see the VNDREQ, VND SUCC and VND FAIL commands descriptions for details).
Possible answers	
- CSLS<X>BASKETOK(X)	- You will receive this answer if the parameter is correctly configured
- CSLS<X>BASKETFAIL	- You will receive this answer if the parameter configuration failed

E. Interface (VMC) system related commands

1. Get VMC settings

Command	
VMCCONFIG?	This command will read interface internal settings.
Possible answers	
- VMCCONFIG(A,B,C,D)	- A is the VMC configured feature level (this interface can only work as a Level 2 and level 3 VMC) - B is number of characters on display/columns (maximum 16). If this value is set to 0, the VMC will inform cashless devices that it is not supporting display messages. - C is the number of rows on display - D is the display type, according to MDB specifications, cashless display message section.

2. Set VMC settings

Command	
- VMCCONFIG(A,B,C,D)	This command will set interface internal settings - A is the VMC configured feature level (this interface can only work as a Level 2 and level 3 VMC) - B is number of characters on display/columns (maximum 16). If this value is set to 0, the VMC will inform cashless devices that it is not supporting display messages. - C is the number of rows on display - D is the display type, according to MDB specifications, cashless display message section.
Possible answers	
- FTLVLERROR - VMCCONFIGOK	- The VMC feature level you mentioned in parameters is invalid. - Command correctly received and parsed.

3. Set VMC manufacturer code

Command	
- VMCSETMFCODE(AAA)	This command will set interface internal manufacturer code that it is reporting to cashless device during automated initialization phase. - AAA is a fixed length, 3 characters (ASCII) value
Possible answers	
- VMCSETMFCODEERR1 - VMCSETMFCODEOK	- The VMC manufacturer code length you mentioned as a parameter is invalid. - Command correctly received and parsed.

4. Set VMC internal serial number

Command	
- VMCSETSN(AAAAAAAAAAAAAA)	This command will set interface internal serial number that it is reporting to cashless device during automated initialization phase. - AAAAAAAAAAAAAA is a fixed length, 12 characters (ASCII) value
Possible answers	
- VMCSETSNERR1	- The VMC interface serial number length you mentioned as a parameter is invalid.
- VMCSETSNOK	- Command correctly received and parsed.

5. Set VMC internal model number

Command	
- VMCSETMN(AAAAAAAAAAAAAA)	This command will set interface internal model number that it is reporting to cashless device during automated initialization phase. - AAAAAAAAAAAAAA is a fixed length, 12 characters (ASCII) value
Possible answers	
- VMCSETMNERR1	- The VMC interface model number length you mentioned as a parameter is invalid.
- VMCSETMNOK	- Command correctly received and parsed.

6. Reset interface internal counters

Command	
- COUNTERSRESET	This command will clear all internal counters. The interface will automatically reboot 3 seconds after issuing this command. You need to reactivate your payment systems if required.
Possible answers	
- COUNTERSRESETOK	- The interface correctly received and executed the command.
- COUNTERSRESETFAIL	- The interface failed executing this command.

7. Interface reboot

Command	
- SYSRESET	This command will force interface reboot after 3 seconds.
Possible answers	
- SYSRESETOK	- The interface correctly received and parsed the command.

8. Check if the interface is up and running

Command	
- ALIVE?	This command will request a simple ACK response from the interface, in order to check it is normally working.
Possible answers	
- ALIVEACK	- The interface correctly received the message and is running.

9. Read internal counters

Command	
- CNTR?	This command will read interface's internal counters. Counters are automatically incremented on some events (bill validated, coin accepted, bill rejected, coin rejected, etc.)
Possible answers	
- CNTR(A,B,C,D,E,F,G,H,I,J,K,L,M)	- A is the total number of received bills. - B is the total value of received bills. - C is not used in this version. - D is the total number of rejected bills (you can monitor this counter in order to decide when you need to clean/recalibrate the bill validator). - E is the total number of received coins. - F is the total value of received coins. - G is the total number of cashless transactions (both #1 and #2) - H is the total value of the cashless transactions (both #1 and #2) - I is not used in this version. - J is the total number of received tokens. - K is the total number of rejected coins. - L is the total value of received tokens. - M - N/A

10. Save settings

Command	
- SAVESETTINGS	This command will save modified settings to the non-volatile memory. You must use this command after you modify at least one of the interface settings.
Possible answers	
- SAVESETTINGSOK	- If the interface successfully saved the settings to the non-volatile memory.
- SAVESETTINGSFAIL	- If the interface failed to save settings to non-volatile memory.

11. Factory reset

Command	
- FACTORYRESET	This command will force a complete erasure of all settings and parameters. The interface will reboot after 3 seconds and will load the default (factory) settings.
Possible answers	
- FACTORYRESETOK	- If the command was successfully received and executed.
- FACTORYRESETFAIL	- If the command was not successfully received and executed.

12. POLL command

Command	
- POLL	This command can be used to check if the interface is connected and alive. Please note that, once you will send this command first time after power-up, the interface requires POLL command to be sent at least once at every 10 seconds. If you are not sending this command, the interface will automatically reset, initialize all attached MDB peripherals and let them disabled (not receiving money). It is intended as a security mechanism (if your application is locked or communication broken, this mechanism will automatically inhibit money acceptance, in order to not accept money that cannot be processed). Do not send POLL command if you do not to activate this mechanism. If you want to activate this mechanism, send POLL after every power-up, then constantly POLL the interface at least once at 10 seconds.
Possible answers	
- POLLOK	- If the command was successfully received.

13. Show current credit values

Command	
- SHOWCREDIT	This command will display current accumulated credit values
Possible answers	
- CREDITS(A,B,C,D,E)	- A – current general credit (cash + cashless) - B – current cashless #1 available credit - C – current cashless #1 authorized credit - D – current cashless #2 available credit - E – current cashless #2 authorized credit

14. Delete current credit values

Command	
- DELETECREDIT	This command will clear all accumulated credits (use before or after any transaction, depending on your app business flow)
Possible answers	
- DELETECREDITOK	- credit values successfully erased

15. Reading device serial number

Command	
- SNREAD?	Responds with the device SN
Possible answers	
- SNREAD(a1)	- Command correctly executed - a1 – device serial number

16. Reset all attached MDB devices

Command	
- MDBRESET	This command will send reset command to all attached MDB peripherals
Possible answers	
- MDBRESETOK	- command received

17. Enable all cash peripherals for a specific amount and time

Command	
- GETCASH(a1, a2)	This command enables all attached cash peripherals, waiting for a specific amount and time. - a1 – the value for cash payment that needs to be received (in cents – 200 = 2.00EUR) - a2 – the value for cash receiving timeout (seconds). If the requested (a1) amount is not received during the a2 time, a cash receive failure message will be received. The time is updated (extended) after each inserted bill or coin received or rejected.
Possible answers	
- GETCASHOK - GETCASHFAIL There are two possible results when the execution finished or timed out: - CASHTIMEOUT(r1) - CASHFULLYRECEIVED(r1)	- command successfully executed - command execution failed - Operation timed out: - r1 is the total accumulated cash since the GETCASH command was issued, until the timeout occurs. - Operation complete: - r1 is the total accumulated cash since the GETCASH command was issued, until the (a1) is accumulated or. Can be higher than (a1) so you need to handle it (use for another purchase or return change), otherwise it will expire.

18. Set RFID reader password (key-A) – only with the optional (sold separately) RFID reader

Command	
- RDRPASS(a1)	This command sets the password that the RFID reader will use in order to read the cards. - a1 – the password, ASCII, fixed length – 6 characters. Special characters permitted, as well. If you want to set to the factory default (0xFFFFFFFF), then send the word "default" as a1 value.
Possible answers	
- RDRPASSOK - RDRPASSFAIL - RDRDEFAULTPASS	- command successfully executed - command execution failed - if you set the default password

19. Get RFID reader password (key-A) – only with the optional (sold separately) RFID reader

Command	
- RDRPASS?	This command sets the password that the RFID reader will use in order to read the cards.
Possible answers	
- RDRPASS(a1)	- a1 – the password, ASCII, fixed length – 6 characters or “default” if the current password is set to factory default (0xFFFFFFFF).

20. Enable the RFID reader, using the password you set before with RDRPASS()

Command	
- RDREENABLE(a1)	This command enables RFID readings. - a1 – the sector you want to read, default 0. If you store other information in the first 4 bytes of other sector, you can read that as SN by simply providing RDREENABLE(3), if your data is in sector 3.
Possible answers	
- RDREENABLEOK - RDREENABLEFAIL	- command successfully executed - command execution failed

21. Get last read card serial number

Command	
- RDRLASTCARD?	This command gets the serial number of the last read card.
Possible answers	
- RDRLASTCARD(a1, a2)	- a1 – latest read card SN - a2 – if == 0, the card is no longer in front of the reader, if == 1, the card is still in front of the reader.

22. Erase last read card serial number

Command	
- RDRCLEAN	This command deletes the last read card SN (sets the last read SN to 0xFFFFFFFF)
Possible answers	
- RDRCLEANFAIL - RDRCLEANOK	- If the card is still in front of the reader, you cannot delete last read SN - command successfully executed.

F. Relay related commands

This set of commands is working only when an optional GalaxyVend KEYPRESS board. Each relay can be addressed individually. You can control each of the 10 relays of the expansion board. A maximum of 4 relays can be energized simultaneously (in order to control multiple outputs with a single command)

1. Individual relay control

Command	
- RELAY(a1, ..., a5)	This command will energize one or multiple relays for a specified amount of time. - a1 – first relay you want to energize (1-10, 0 for not enabled) - a2 – second relay you want to energize (1-10, 0 for not enabled) - a3 – third relay you want to energize (1-10, 0 for not enabled) - a4 – fourth relay you want to energize (1-10, 0 for not enabled) - a5 – the time (seconds) you want to keep energized For example if you want the relays 2 and 5 energized for 4 seconds, you will send the command as follows: RELAY(2,5,0,0,4)
Possible answers	
- RELAYOK - RELAYFAIL	- If the command was successfully received and executed. - If the command was not successfully received and executed.

IV. Unsolicited messages

Unsolicited messages are messages that are coming as a result of the payment systems activity and not as a result of a command from your application. They may occur at any moment so your application is responsible to constantly listen on the serial or USB interface, parse unsolicited messages and react accordingly.

A. Bill validator related messages

1. Bill validator just reset time exceeded

Message	Description
BILLJRESETEXCEED	- The interface failed waiting for reset message from the bill validator. It will automatically reset all validator related variables and start sending bill reset message.

2. Bill validator setup time exceeded

Message	Description
BILLSETUPEXCEED	- The interface repeatedly failed to receive a valid answer on the MDB BILL SETUP command. It will automatically restore all validator related variables and start sending bill reset message to retry the bill initialization operation.

3. Bill validator expansion ID time exceeded

Message	Description
BILLEXPIDEXCEED	- The interface repeatedly failed to receive a valid answer on the MDB BILL EXPANSION ID command. It will automatically restore all validator related variables and start sending bill reset message to retry the bill initialization operation.

4. Bill validator expansion ID with options time exceeded

Message	Description
BILLEXPIDOPTEXCEED	- The interface repeatedly failed to receive a valid answer on the MDB BILL EXPANSION ID WITH OPTIONS command. It will automatically restore all validator related variables and start sending bill reset message to retry the bill initialization operation.

5. Bill validator optional feature enable time exceeded

Message	Description
BILLENOPTFEATEXCEED	- The interface repeatedly failed to receive a valid answer on the MDB BILL OPTIONAL FEATURES ENABLE command. It will automatically restore all validator related variables and start sending bill reset message to retry the bill initialization operation.

6. Bill validator with recycling support has been identified

Message	Description
BILLHASRECYCLER	- The interface identified a bill validator with recycling support during bill initialization phase.

7. Interface will try to enable the recycling support

Message	Description
BILLTRYENRECYCLER	- The interface will perform needed operations in order to enable bill recycling support for the bill validator/recycler

8. Bill recycler setup time exceeded

Message	Description
BILLRECYCLERSETUPEXCEED	- The interface repeatedly failed to receive a valid answer on the MDB BILL RECYCLER SETUP command. It will automatically restore all validator related variables and start sending bill reset message to retry the bill initialization operation.

9. Bill device has no support to recycler any known bills

Message	Description
BILLNOAVAILRECYCLINGBILLS	- The bill validator does not support recycling for any of the known (configured) bills.

10. Bill recycler enabling time exceeded

Message	Description
BILLRECYCLERENEXCEED	- The interface repeatedly failed to receive a valid answer on the MDB BILL RECYCLER ENABLED command. It will automatically restore all validator related variables and start sending bill reset message to retry the bill initialization operation.

11. Bill recycler enabling failed

Message	Description
BILLRECYCLERENFAIL	- The interface failed to enable the bill recycler functions.

12. Bill recycler function successfully enabled

Message	Description
BILLRECYCLERONOK	- The interface failed to enable the bill recycler functions.

13. Bill recycler answered with a NAK on enable function

Message	Description
BILLRECYCLERENACK	- The interface received a NAK while trying to enable bill recycler functions. It will retry until the bill recycler will correctly answer or until the retry time exceed.

14. Bill recycler reading dispense status time exceeded

Message	Description
BILLDISPENSESTATEXCEED	- The interface repeatedly failed to receive a valid answer on the MDB BILL RECYCLER DISPENSE STATUS command. It will automatically restore all validator related variables and start sending bill reset message to retry the bill initialization operation.

16. Bill recycler reading dispense status returned a NAK

Message	Description
BILLDISPENSESTATNAK	- The interface received a NAK while trying to obtain a dispense status. It will retry until the bill recycler will correctly answer or until the retry time exceed.

16. Bill recycler reading dispense status returned an ACK

Message	Description
BILLDISPENSESTATACK	- The interface received a simple ACK while trying to obtain a dispense status. It will retry until the bill recycler will correctly answer or until the retry time exceed.

17. Bill recycler reading dispense status returned an ACK

Message	Description
BILLDISPENSESTATACK	- The interface received a simple ACK while trying to obtain a dispense status. It will retry until the bill recycler will correctly answer or until the retry time exceed.

19. Bill recycler remaining stock value

Message	Description
RECYCLERSTOCKVALUE(X)	- X is the scaled total bills value remaining for recycling after the bill recycler finished dispensing bills.

19. Bill dispensing command time exceed

Message	Description
BILLDISPENSEVALUEXCEED	- The interface repeatedly failed to receive a valid answer on the MDB BILL RECYCLER DISPENSE command. It will automatically restore all validator related variables and start sending bill reset message to retry the bill initialization operation.

20. Bill dispensing command time exceed

Message	Description
BILLDISPENSESETOTAL(X)	- X is the scaled total value of the bills to dispense

21. Bill dispensing command time exceed

Message	Description
BILLDISPENSED(X)	- X is the scaled total value of the dispensed bills.

22. Bill stacker status command time exceeded

Message	Description
BILLSTACKEXCEED	- The interface repeatedly failed to receive a valid answer on the MDB BILL STACKER command. It will automatically restore all validator related variables and start sending bill reset message to retry the bill initialization operation.

23. Bill stacker status – stacker full

Message	Description
BILLSTACKFULL(X)	- X is the number of the bills in the bill validator stacker and the stacker is reported full.

24. Bill stacker status – stacker not full

Message	Description
BILLSTACKNOTFULL(X)	- X is the number of the bills in the bill validator stacker and the stacker is not full, yet.

25. Bill is not ready

Message	Description
BILLNOTREADY	- Bill validator/recycler is not ready to execute the last received command, probably because it was not initialized, enabled or it's current status does not allow this command.

26. Bill validator/recycler failed to answer on poll command

Message	Description
BILLPOLLEXCEED	- The interface repeatedly failed to receive a valid answer on the MDB BILL POLL command. It will automatically restore all validator related variables and start sending bill reset message to retry the bill initialization operation.

27. Bill validator – one bill stacked

Message	Description
BILLSTACKED(A,B,C)	- One bill was successfully stacked - A is the scaled value of the last stacked bill - B is the number of total stacked bill (internal non-volatile counter) - C is the total value of the stacked bills (internal non-volatile counter)

28. Bill in escrow position

Message	Description
BILLESCROW(X)	- One bill is in the escrow position - X is the scaled value of the bill in escrow position. Your application should send a BILLACCEPT or a BILLREJECT command on this stage, depending on it's flow, maximum credit, etc

29. Bill returned to customer

Message	Description
BILLRETURNED(X)	- The bill in escrow position returned to customer - X is the scaled value of the returned bill.

30. Bill received in recycler

Message	Description
BILLTORECYCLER(X)	- A bill was received and stored in the recycling box - X is the scaled value of the stored bill.

31. A disabled bill was rejected

Message	Description
BILLDISREJ(X)	- A bill was rejected because it was previously disabled by the user application. - X is the scaled value of the rejected bill.

32. A bill was manually loaded to recycler

Message	Description
BILLRECMANFILL(X)	- A bill manually loaded to recycler stock - X is the scaled value of the loaded bill.

33. A disabled bill was manually dispensed from the recycler

Message	Description
BILLMANDISP(X)	- A bill was manually dispensed from the recycler - X is the scaled value of the dispensed bill.

34. A disabled bill was transferred from the recycler to cashbox

Message	Description
BILLTRANSFER(X)	- A bill was transferred from the recycler box to the cashbox - X is the scaled value of the transferred bill.

35. Bill validator is in normal condition

Message	Description
BILLOK	- Bill was correctly initialized after reset or has been recovered after an error.

36. Bill validator have a defective motor

Message	Description
BILLDEFMOTOR	- Bill validator encountered one of it's motors failure

37. Bill validator have a defective sensor

Message	Description
BILLSENSORFAIL	- Bill validator encountered one of it's sensors failure

38. Bill validator is busy

Message	Description
BILLBUSY	- Bill validator is in a busy state doing something

39. Bill validator ROM error

Message	Description
BILLROMERROR	- Bill validator encountered an internal ROM error

40. Bill validator jam

Message	Description
BILLJAM	- Bill validator encountered a bill jam error

41. Bill validator was reset

Message	Description
BILLRESET	- Bill validator has just been reset.

42. Bill removed from bill validator

Message	Description
BILLREMOVED	- A bill was removed from the bill validator

43. Bill validaor cashbox has been removed

Message	Description
BILLCSBOXREMOVED	- Bill validator's cashbox has been removed

44. Bill validaor has been disabled by your application or by an internal error

Message	Description
BILLDISABLED	- Bill validator has been disabled by your application or due an internal error

45. Bill validator has been rejected a bill

Message	Description
BILLREJECTED(X)	- Bill validator has been rejected a bill - X is the total number of rejected bills.

46. Bill removed after it was credited

Message	Description
BILLCREDITEDREMOVED	- A bill was removed from the bill validator after it was credited.

47. A bill was inserted while the bill validator is deactivated

Message	Description
BILLINSERTWHILEDISABLED	- A bill was inserted while the bill validator is deactivated

48. Recycler has received a change request

Message	Description
RECYCLERCHGREQUEST	- Recycler has received a change request

B. Cashless related unsolicited messages

1. Cash sale reported to the cashless device

Message	Description
CSLS<X>CASHSALE(A,B)	- Cash sale was reported to the cashless device - A is the item price - B is the item ID/selection number

2. Cash sale successfully reported to the cashless device

Message	Description
CSLS<X>CASHSALEOK	- A cash sale was successfully reported to the cashless device.

3. Cash sale reporting to the cashless device failed

Message	Description
CSHS<X>CASHSALEFAIL	- A cash sale reporting to the cashless device failed.

4. Cashless device setup time exceeded

Message	Description
CSLS<X>SETUPEXCEED	- The interface repeatedly failed to receive a valid answer on the MDB CASHLESS SETUP command. It will automatically restore all cashless device related variables and start sending cashless reset message to retry the cashless device initialization operation.

5. Cashless device max/min prices reporting time exceed

Message	Description
CSLS<X>MAXMINEXCEED	- The interface repeatedly failed to receive a valid answer on the MDB MAX/MIN PRICES command. It will automatically restore all cashless device related variables and start sending cashless reset message to retry the cashless device initialization operation.

6. Cashless device poll time exceed

Message	Description
CSLS<X>POLLEXCEED	- The interface repeatedly failed to receive a valid answer on the MDB CASHLESS POLL command. It will automatically restore all cashless device related variables and start sending cashless reset message to retry the cashless device initialization operation.

7. Cashless device expansion request ID time exceed

Message	Description
CSLS<X>EXPREQIDEXCEED	- The interface repeatedly failed to receive a valid answer on the MDB CASHLESS REQUEST ID command. It will automatically restore all cashless device related variables and start sending cashless reset message to retry the cashless device initialization operation.

8. Cashless device expansion enable options time exceed

Message	Description
CSLS<X>EXPENOPTEXCEED	- The interface repeatedly failed to receive a valid answer on the MDB CASHLESS EXPANSION ENABLE OPTIONS command. It will automatically restore all cashless device related variables and start sending cashless reset message to retry the cashless device initialization operation.

9. Cashless device have Always Idle support and it will be enabled

Message	Description
CSLS<X>ALWAYSIDLE	- The cashless device have Always Idle support and the interface will try to enable it

10. Cashless device writing date/time exceed

Message	Description
CSLS<X>WRDTEXCEED	- The interface repeatedly failed to receive a valid answer on the MDB DATE/TIME command. It will automatically restore all cashless device related variables and start sending cashless reset message to retry the cashless device initialization operation.

11. Cashless device sent a display message

Message	Description
CSLS<X>DISPMSG(A,B)	- The cashless device sent a display message to the interface - A is the time to keep message on display (A x 0.1sec) - B is the message to display

12. Cashless device sent a BEGIN SESSION message

Message	Description
CSLS<X>BEGIN(A,B,C)	- The cashless device sent a BEGIN SESSION message to the interface - A is the scaled available credit value - B is the media ID (for example, the card serial number) - C is the media type

13. Cashless device sent a VEND APPROVED message

Message	Description
CSLS<X>VNDAPP(A,B,C)	- The cashless device sent a VEND APPROVED message to the interface - A is the scaled approved value - B is the total number of cashless transactions (internal counter) - C the total scaled value of cashless transactions (internal counter)

14. Cashless device sent a VEND DENIED message

Message	Description
CSLS<X>VNDDEN	- The cashless device sent a VEND DENIED message to the interface

15. Cashless device sent an END SESSION message

Message	Description
CSLS<X>ENDSESSION	- The cashless device sent an END SESSION message to the interface

16. Cashless device sent a CANCELED message

Message	Description
CSLS<X>CANCELED	- The cashless device sent a CANCELED message to the interface

17. Cashless device is ready

Message	Description
CSLS<X>READY	- The cashless device was correctly initialized and is ready to be enabled.

18. Cashless device returned a malfunction error

Message	Description
CSLS<X>MALFUNCTION(X)	- The cashless device returned a malfunction message - X is the internal malfunction code, it's value depends on the cashless device and you can find more information in it's manual

19. Cashless device returned COMMAND OUT OF SEQUENCE message

Message	Description
CSLS<X>CMDOUTOFSEQ	- The cashless device returned a COMMAND OUT OF SEQUENCE message

20. Cashless device sent a REVALUE APPROVED message

Message	Description
CSLS<X>REVALAPP	- The cashless device returned a REVALUE APPROVED message

21. Cashless device sent a REVALUE DENIED message

Message	Description
CSLS<X>REVALDEN	- The cashless device returned a REVALUE DENIED message

22. Cashless device sent a REVALUE LIMIT message

Message	Description
CSLS<X>REVALLIMIT(X)	- The cashless device returned a REVALUE LIMIT message - X is the maximum amount it will accept for the next REVALUE REQUEST command

23. Cashless device sent a DATE/TIME request message

Message	Description
CSLS<X>DTREQ	- The cashless device is requesting a date/time command to synchronize its internal RTC

24. Interface successfully sent date/time command to the cashless device

Message	Description
CSLS<X>DTSENDOK	- The interface successfully sent date/time command to the cashless device.

24. Interface failed sending date/time command to the cashless device

Message	Description
CSLS<X>DTSENDFAIL	- The interface failed sending date/time command to the cashless device.

25. Interface successfully enabled the cashless device

Message	Description
CSLS<X>ENABLED	- The interface successfully enabled the cashless device.

26. Interface successfully enabled the cashless device

Message	Description
CSLS<X>ENABLED	- The interface successfully enabled the cashless device.

C. Coin related unsolicited messages

1. Coin acceptor/changer just reset waiting time exceeded

Message	Description
COINJUSTRESETEXCEED	- The interface repeatedly failed to receive a valid answer on waiting for JUST RESET message. It will automatically restore all coin acceptor/changer related variables and start sending coin reset message to retry the coin acceptor/changer initialization operation.

2. Coin acceptor/changer setup time exceeded

Message	Description
COINSETUPEXCEED	- The interface repeatedly failed to receive a valid answer on COIN SETUP command. It will automatically restore all coin acceptor/changer related variables and start sending coin reset message to retry the coin acceptor/changer initialization operation.

3. Coin acceptor/changer expansion identification time exceeded

Message	Description
COINEXPIDEXCEED	- The interface repeatedly failed to receive a valid answer on COIN EXPANSION IDENTIFICATION command. It will automatically restore all coin acceptor/changer related variables and start sending coin reset message to retry the coin acceptor/changer initialization operation.

4. Coin acceptor/changer feature enable time exceeded

Message	Description
COINFTENABLEEXCEED	- The interface repeatedly failed to receive a valid answer on COIN FEATURE ENABLE command. It will automatically restore all coin acceptor/changer related variables and start sending coin reset message to retry the coin acceptor/changer initialization operation.

5. Coin acceptor/changer tube status time exceeded

Message	Description
COINTBSTATEXCEED	- The interface repeatedly failed to receive a valid answer on COIN TUBE STATUS command. It will automatically restore all coin acceptor/changer related variables and start sending coin reset message to retry the coin acceptor/changer initialization operation.

6. Coin acceptor/changer is not ready for the issued command

Message	Description
COINNOTREADY	- The coin acceptor/changer is not ready to execute the last issued command

7. Coin acceptor/changer tube status

Message	Description
COINTBSTATUS(X)	- The coin acceptor/changer returned the TUBE STATUS answer. - X is the total scaled value of the coins available for change. If the number of coins in a tube is bigger than 255, the coin changer will return 255 as a value for that tube. So, this command is not appropriate for coins stock management since it will return the same value for a tube until the number of coins in that tube falls below 255. But you can use it to set an alarm on lower coins stock, for example.

8. Coin acceptor/changer poll time exceeded

Message	Description
COINPOLLEXCEED	- The interface repeatedly failed to receive a valid answer on COIN POLL command. It will automatically restore all coin acceptor/changer related variables and start sending coin reset message to retry the coin acceptor/changer initialization operation.

9. Coin acceptor/changer is busy dispensing coins

Message	Description
COINPAYBUSY	- The coin acceptor/changer is busy dispensing coins following a COINDISPENSE or a COINAP command. This message will occur repeatedly until the coin changer finish the dispense operation. The number of those messages depends on the number of the coins it should dispense and the dispensing method (COINAP method is faster than COINDISPENSE).

10. Coin acceptor/changer temporarily unable to dispense coins

Message	Description
COINCHGNOTNOW	- The coin acceptor/changer is temporarily unable to dispense coins due to it's working stage. Your application should retry later.

10. Coin acceptor/changer has failed to dispense all or some of the required coins

Message	Description
CHANGEREMAINING(X)	- The coin acceptor/changer has failed to dispense all or some of the required coins. - X is the scaled value of the coins changer was unable to dispense for some reasons. You will use this value to display the remaining credit to the customer.

11. Coin acceptor/changer is reporting a manual coin dispense

Message	Description
COINMANDISP(A,B,C)	- The coin changer has manually dispensed one or more coins (usually by pressing one or more buttons on it's front panel). - A is the scaled coin type value - B is the total number of manually dispensed coins - C is the total number of coins remaining in tubes for the A type value

12. Coin acceptor/changer received a token

Message	Description
TOKENIN(A,B,C,D)	- One token has been received by the coin acceptor/changer - A is the token value (you need to set the token values correctly on the interface settings section) - B is the token routing (0 – to cashbox, 1 – to tubes, 3 - rejected) - C is the total number of received tokens (lifetime internal counter) - D is the total value of received tokens (lifetime internal counter)

13. Coin acceptor/changer received a coin

Message	Description
COININ(A,B,C,D,E)	- One coin has been received by the coin acceptor/changer - A is the scaled coin value - B is the token routing (0 – to cashbox, 1 – to tubes, 3 – rejected) - C is the total number of coins with the same value available in tubes - D is the total number of received coins (lifetime internal counter) - E is the total value of received coins (lifetime internal counter)

14. Coin acceptor/changer detected a slug

Message	Description
COINSLUG(A,B)	- One slug detected by the coin acceptor/changer - A is the slug counter, reported by the coin acceptor/changer - B is the total number of coins/tokens rejected by the coin acceptor/changer (lifetime internal counter)

15. Coin acceptor/changer is in normal condition

Message	Description
COINOK	- Coin was correctly initialized after reset or has been recovered after an error.

16. Coin acceptor/changer received a change request

Message	Description
COINCHGREQ	- Coin acceptor/changer has received a change request (usually by pressing the coin changer mechanical lever). Customer pressed the change lever in order to cancel the transaction or request the change after transaction. Your application should act accordingly.

17. Coin acceptor/changer received a coin that was not credited

Message	Description
COINNOTCRDT	- Coin acceptor/changer received a coin that was routed, but not credited.

18. Coin acceptor/changer has a defective tube sensor

Message	Description
COINDEFTBSSENSOR	- Coin acceptor/changer detected a defective tube sensor.

19. Coin acceptor/changer detected a double arrival

Message	Description
COINDBLARRIVAL	- Coin acceptor/changer detected a double arrival (two or more coins/tokens were inserted too fast in order to allow the coin acceptor to validate them).

20. Coin changer detected an accep- a1 – first relay you want to energize (1-10)

Message	Description
COINACCUNPL	- Coin changer detected an acceptor disconnection.

21. Coin acceptor/changer detected a tube jam

Message	Description
COINTBJAM	- Coin acceptor/changer detected a tube jam

22. Coin acceptor/changer detected an internal ROM error

Message	Description
COINROMERR	- Coin acceptor/changer detected an internal ROM error

23. Coin acceptor/changer detected a routing error

Message	Description
COINROUTERR	- Coin acceptor/changer detected a routing error for the last accepted coin/token

24. Coin acceptor/changer detected reset condition

Message	Description
COINRST	- Coin acceptor/changer detected a reset condition

25. Coin acceptor/changer detected a coin jam

Message	Description
COINJAM	- Coin acceptor/changer detected a coin jam, most probably in the flight deck area. Your application can indicate the customer to press the escrow lever in order to release the blocked coins

26. Coin acceptor/changer detected the removal of a credited coin

Message	Description
COINCRREM	- Coin acceptor/changer detected the removal of a credited coin.

27. Cashless X basket functions are active or not (if supported by the cashless device)

Message	Description
CSLS<X>ACTIVATEBASKET(A)	- This message is received on boot or after issuing a SYSRESET command - If "A" == 1 then, the basket functions for cashless device "X" will be activated if that cashless device is supporting those functions. - If "A" == 0 then, the basket functions for cashless device "X" will be deactivated if that cashless device is supporting those functions.

28. Cashless X has basket functions available

Message	Description
CSLS<X>BASKET	- This message is received after issuing a CSLS<X> RESET function if, during the initialization procedure, the interface is detecting that the "X" attached cashless device can handle basket functions.

29. Cashless X basket is active when a normal mode function was called

Message	Description
CSLS<X>BASKETMODE	- This message is received if you are trying to use the normal VNDREQ, VNDSUCC or VNDFAIL functions when the basket functions are enabled, because basket VEND REQUEST, VEND SUCCESS and VEND FAILURE require additional parameters. The message will be immediately followed by - CSLS<X>VNDREQFAIL - CSLS<X>VNDSUCCFAIL or - CSLS<X>VNDFAILFAIL, depending on the issued command.

30. Cashless X basket functions are not active when a normal function was called

Message	Description
CSLS<X>BASKETMODEDISABLED	- This message is received if you are trying to use the basket VNDREQ, VNDSUCC or VNDFAIL functions when the basket functions are not enabled, because basket VEND REQUEST, VEND SUCCESS and VEND FAILURE have extra parameters. The message will be immediately followed by - CSLS<X>VNDREQFAIL - CSLS<X>VNDSUCCFAIL or - CSLS<X>VNDFAILFAIL, depending on the issued command.

D. RFID reader related unsolicited messages

1. RFID reader SN received

Message	Description
RDRBEGIN(a1)	- An RFID card was identified and the SN received - a1 – identified card SN

2. RFID card removed

Message	Description
RDRLEAVING	- RFID card moved away from the RFID reader

3. RFID card reading failed

Message	Description
RDRREADFAILED	- The card in front of the reader could not be used/read failure

Appendix IV – Cashless basket transaction example

This section will describe a simple basket transaction for the cashless #1 device (if the device is supporting this function).

A. Prerequisites

1. Send command CSLS1BASKET(1) – to activate interface's basket support for cashless #1
2. Send command SAVE SETTINGS – to save the parameter in the non-volatile memory
3. Send command SYSRESET to reboot the interface
4. After receiving CSLS1READY unsolicited message, send CSLS1ENABLE.

Apply steps 3. and 4. every time your application starts.

B. Cashless basket transaction

We will assume that the basket sale will include the following products:

- Product #1 – 1.50EUR;
- Product #2 – 1.20EUR;
- Product #3 – 2.20EUR.

1. Send command CSLS1VNDREQ(490,3,0);
2. Wait for CSLS1VNDREQOK;
3. Wait until the customer is presenting the card and authorization is performed - getting CSLS1VNDAPP(A,B,C) message
4. If product #1 is dispensed, send command CSLS1VNDSUCC(1,150,2,0)
- 4'. if product #1 dispensing failed, send command CSLS1VNDFAIL(1,150,2,0)
5. If product #2 is dispensed, send command CSLS1VNDSUCC(2,120,1,0)
- 5'. if product #2 dispensing failed, send command CSLS1VNDFAIL(2,120,1,0)
6. If product #3 is dispensed, send command CSLS1VNDSUCC(3,220,0,0)
- 6'. if product #1 dispensing failed, send command CSLS1VNDFAIL(3,220,0,0)

If the sale was based on a cashless session previously started, the cashless device will close the session after 6. or 6'.

NOTES: