

GalaxyVend EXECUTIVE TO MDB INTERFACE

**v1.001 - STANDALONE
v1.003 – EXTERNAL
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. Keypress expansion module (external mode only).....	10
D. Keypress board connectors description.....	10
II. STANDALONE working mode.....	11
A. Bluetooth activation.....	11
B. Android application flow.....	11
III. External working mode (connected over USB to computer, PLC, or Raspberry Pi and compatible SBCs or over MQTT).....	15
A. Bill validator related commands.....	15
1. Check if the bill validator was initialized by the interface.....	15
2. Check if the bill validator was enabled.....	15
3. Get the bill validator configured bills values.....	16
4. Get the bill validator information.....	16
5. Get the bill validator settings.....	16
6. Reset bill validator.....	17
B. Coin related commands.....	18
1. Reset coin acceptor.....	18
2. Get total value of coins in tubes (for coin changers only).....	18
3. Dispense some coins (change) to the customer – obsolete, try to use COINAP command whenever the coin acceptor/changer supports it.....	19
4. Dispense some coins using MDB alternative payout method.....	19
5. Check if the coin acceptor/changer was initialized.....	19
6. Check if the coin acceptor/changer was enabled.....	20
7. Get last 10 coin acceptor/changer codes.....	20
8. Get the coin acceptor/changer configured coins values.....	20
9. Get the coin acceptor/changer information.....	21
10. Get the coin acceptor/changer settings.....	21
11. Get the token values.....	21
12. Set the token values.....	22
C. Cashless related commands.....	23
1. Reset cashless device.....	23
2. Request cashless current revalue limit.....	23
3. Request approval for a vend.....	23
4. Confirm a success vend to the cashless device.....	24
5. Report a vend failure to the cashless device.....	24
6. Report a cash sale to the cashless device.....	24
7. Send a revalue request (the customer's account amount refill).....	25
8. Get last 10 cashless device codes.....	25
9. Check if the cashless device was initialized by the interface.....	25
10. Check if the cashless device was enabled.....	26
11. Get the cashless device information.....	26
12. Get the cashless device settings.....	26
D. System and executive related commands.....	27
1. Interface reboot.....	27
2. Check if the interface is up and running.....	27
3. Read internal counters.....	27
4. Requesting interface settings.....	28

5. Sending interface settings.....	29
6. Sending some simulated cash credit to the VMC.....	30
7. Saving settings to non-volatile memory (persistent after reboot).....	30
8. Reset to factory default.....	30
9. Reading board serial number.....	30
10. Keypress (one key) – available only with the optional keypress module.....	31
11. Keypress (two successive keys) – available only with the optional keypress module.....	31
12. Keypress (three successive keys) – available only with the optional keypress module.....	32
13. Keypress (four successive keys) – available only with the optional keypress module.....	32
14. Check if the VMC is enabled or out of order.....	33
15. Check current available credits.....	33
16. Clear all current credit variables.....	33
17. Clear cash credit only.....	33
18. Resetting all MDB peripherals (payment systems).....	33
19. Enable all cash MDB peripheral.....	33
20. Disable all cash MDB peripheral.....	34
21. Enable all cash peripherals for a specific amount and time.....	34
22. Check peripherals' status.....	34
26. File system format.....	35
27. Product stock update.....	35
28. Complete stock update.....	35
29. Save all current stocks to non-volatile memory.....	35
30. Read current stocks (obsolete, do not use it, use CURRENTSTOCK? instead).....	36
31. Select product to sale.....	36
32. Cancel product selection.....	36
33. Select cash payment method.....	36
34. Select cashless #1 payment method.....	37
35. Select free vend payment method.....	37
36. Requesting RFID card available credit (only working with the MQTT over WiFi version, using our SaaS backend).....	37
37. Reading product parameters.....	38
38. Writing product parameters.....	39
39. Requesting complete stock status.....	40
40. Setting interface working flow (there are 2 working flows in external mode).....	40
41. Reading interface working flow (there are 2 working flows in external mode).....	40
IV. Unsolicited messages.....	41
A. Bill validator related messages.....	41
1. Bill validator just reset time exceeded.....	41
2. Bill validator setup time exceeded.....	41
3. Bill validator expansion ID time exceeded.....	41
4. Bill validator expansion ID with options time exceeded.....	41
5. Bill validator optional feature enable time exceeded.....	42
6. Bill validator with recycling support has been identified.....	42
7. Interface will try to enable the recycling support.....	42
8. Bill recycler setup time exceeded.....	42
9. Bill device has no support to recycler any known bills.....	42
10. Bill recycler enabling time exceeded.....	42
11. Bill recycler enabling failed.....	42
12. Bill recycler function successfully enabled.....	43
13. Bill recycler answered with a NAK on enable function.....	43
14. Bill recycler reading dispense status time exceeded.....	43
15. Bill recycler reading dispense status returned a NAK.....	43
16. Bill recycler reading dispense status returned an ACK.....	43

17. Bill recycler reading dispense status returned an ACK.....	43
18. Bill recycler remaining stock value.....	43
19. Bill dispensing command time exceed.....	44
20. Bill dispensing total value finished.....	44
21. Bill dispensing total value.....	44
22. Bill stacker status command time exceeded.....	44
23. Bill stacker status – stacker full.....	44
24. Bill stacker status – stacker not full.....	44
25. Bill is not ready.....	44
26. Bill validator/recycler failed to answer on poll command.....	45
27. Bill validator – one bill stacked.....	45
28. Bill in escrow position.....	45
29. Bill returned to customer.....	45
30. Bill received in recycler.....	45
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.....	46
34. A disabled bill was transferred from the recycler to cashbox.....	46
35. Bill validator is in normal condition.....	46
37. Bill validator have a defective motor.....	46
36. Bill validator have a defective sensor.....	46
37. Bill validator is busy.....	46
38. Bill validator ROM error.....	46
39. Bill validator jam.....	46
40. Bill validator was reset.....	46
41. Bill removed from bill validator.....	47
42. Bill validator cashbox has been removed.....	47
43. Bill validator has been disabled by your application or by an internal error.....	47
44. Bill validator has been rejected a bill.....	47
45. Bill removed after it was credited.....	47
46. A bill was inserted while the bill validator is deactivated.....	47
47. Recycler has received a change request.....	47
B. Cashless related unsolicited messages.....	48
1. Cash sale successfully reported to the cashless device.....	48
2. Cash sale reporting to the cashless device failed.....	48
3. Cashless device setup time exceeded.....	48
4. Cashless device max/min prices reporting time exceed.....	48
5. Cashless device poll time exceeded.....	48
6. Cashless device expansion request ID time exceeded.....	48
7. Cashless device expansion enable options time exceeded.....	49
8. Cashless device have Always Idle support and it will be enabled.....	49
9. Cashless device writing date/time exceeded.....	49
10. Cashless device sent a display message.....	49
11. Cashless device sent a BEGIN SESSION message.....	49
12. Cashless device sent a VEND APPROVED message.....	49
13. Cashless device sent a VEND DENIED message.....	50
14. Cashless device sent an END SESSION message.....	50
15. Cashless device sent a CANCELED message.....	50
16. Cashless device is ready.....	50
17. Cashless device returned a malfunction error.....	50
18. Cashless device returned COMMAND OUT OF SEQUENCE message.....	50
19. Cashless device sent a REVALUE APPROVED message.....	50
20. Cashless device sent a REVALUE DENIED message.....	50

21. Cashless device sent a REVALUE LIMIT message.....	51
22. Cashless device sent a DATE/TIME request message.....	51
23. Interface successfully sent date/time command to the cashless device.....	51
24. Interface failed sending date/time command to the cashless device.....	51
25. Interface successfully enabled the cashless device.....	51
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
11. Coin acceptor/changer has failed to dispense all or some of the required coins.....	54
12. Coin acceptor/changer is reporting a manual coin dispense.....	54
13. Coin acceptor/changer received a token.....	54
14. Coin acceptor/changer received a coin.....	54
15. Coin acceptor/changer detected a slug.....	55
16. Coin acceptor/changer is in normal condition.....	55
17. Coin acceptor/changer received a change request.....	55
18. Coin acceptor/changer received a coin that was not credited.....	55
19. Coin acceptor/changer has a defective tube sensor.....	55
20. Coin acceptor/changer detected a double arrival.....	55
21. Coin changer detected an acceptor disconnection.....	55
22. Coin acceptor/changer detected a tube jam.....	56
23. Coin acceptor/changer detected an internal ROM error.....	56
24. Coin acceptor/changer detected a routing error.....	56
25. Coin acceptor/changer detected reset condition.....	56
26. Coin acceptor/changer detected a coin jam.....	56
27. Coin acceptor/changer detected the removal of a credited coin.....	56
D. Executive and system messages.....	57
1. Vending machine is not answering.....	57
2. CRC error on message from the vending machine.....	57
3. The ID of the selected product (PRICE HOLDING mode).....	57
4. Machine is in idle status (active).....	58
5. Machine is in out order status (inactive).....	58
6. The interface successfully sent the credit.....	58
7. The interface failed sending credit to the machine.....	58
8. Last dispensing action success.....	58
9. Last dispensing action failed.....	58
10. Last vend approval success.....	58
11. Last vend approval failure.....	58
12. RFID reader SN received.....	58
13. RFID card removed.....	59
14. RFID card reading failed.....	59
E. Display helper messages.....	60
1. Go to idle (main) screen.....	60
2. Coin change is not available at this point.....	60
3. Wait for the coin change to be dispensed.....	60

4. Coin change process finished.....	60
5. Sending credit to the machine.....	60
6. Sending a keypress to the machine.....	60
7. Sending dispensing approval to the machine.....	60
8. Product successfully dispensed.....	61
9. Product dispensing failed.....	61
10. Card payment failure.....	61
11. Card payment authorized.....	61
12. Cash payment failure.....	61
13. Cash payment authorized.....	61
14. Display a question about payment method.....	61
15. The product is not available.....	62
16. Information to insert cash.....	62
17. Information to tap credit card.....	62
18. Current credit.....	62
19. Received free vend command.....	62
20. Display last read RFID subscription available amount (only if connected to our SaaS backend).....	62

I. Introduction

This device was designed for an easy conversion of Executive machines to modern MDB payment systems. It can be also used to connect a computer, a PLC or a Raspberry Pi (and compatible SBC) to an Executive vending machine, in order to manage sales and other functions. It is driven by a simple ASCII protocol that can be easily implemented in any user application.

Firmware options:

- STANDALONE (automatic conversion, used to connect MDB payment systems to Executive vending machines, with built-in business logic).
- External (external control of MDB payment systems and Executive vending machine actions – over USB or WiFi with MQTT). You need to develop your own application that contains the business logic and running that application on a device externally connected to our interface through an USB-B cable.
- External /w keypress simulator board (external control of MDB payment systems and Executive machine actions, with built-in simulator that can be connected to vending machines' mechanical keyboard input, allowing automatic product selection – over USB or WiFi with MQTT). You need to develop your own application that holds business logic and running that application on a device externally connected to our interface through an USB-B cable. Used for vending machines retrofitting/refurbishing or other functions such as telemetry, stock management, etc.

Package content:

1. For STANDALONE and external:

- GALAXYVEND MDB EXECUTIVE CONVERTER board;
- Executive cable to connect to Executive VMC.

2. External /w keypress simulator board:

- GALAXYVEND MDB EXECUTIVE CONVERTER board;
- GALAXYVEND KEYPRESS SIMULATOR board;
- Executive cable to connect to Executive VMC.

NOTE!!! - this device can handle one MDB bill validator/recycler, one MDB coin acceptor/changer and one MDB cashless device (Level 2 or Level 3 with always idle cashless device) in STANDALONE mode and an additional secondary cashless on external versions. **Also, it is not supporting MDB Level 1 cashless devices and there is no plan to add support for Level 1 on this device.**

During its activity, the interface keeps some internal counters (total number of bills validated, total value of validated bills, coin, cashless transactions, product sales in PRICE HOLDING mode, 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 by the user application. For example, you may constantly poll for counters to check if you missed some bills or coins.

The interface is able to work with both Executive modes (prices set on the vending machine or "PRICE HOLDING" - with the prices stored on the interface). If you need to collect sales statistics with your external application, "PRICE HOLDING" mode is recommended. The interface is able to store up to 64 product individual counters. The counters can be read over USB interface, by using simple ASCII commands detailed later in this document.

To use external versions, if you need to communicate over USB and you are using Windows OS, you need to install the driver you can [DOWNLOAD HERE](#). Drivers are not required for Linux or Android OS, since they are already included with the kernels.

II. Hardware

A. Core board overview

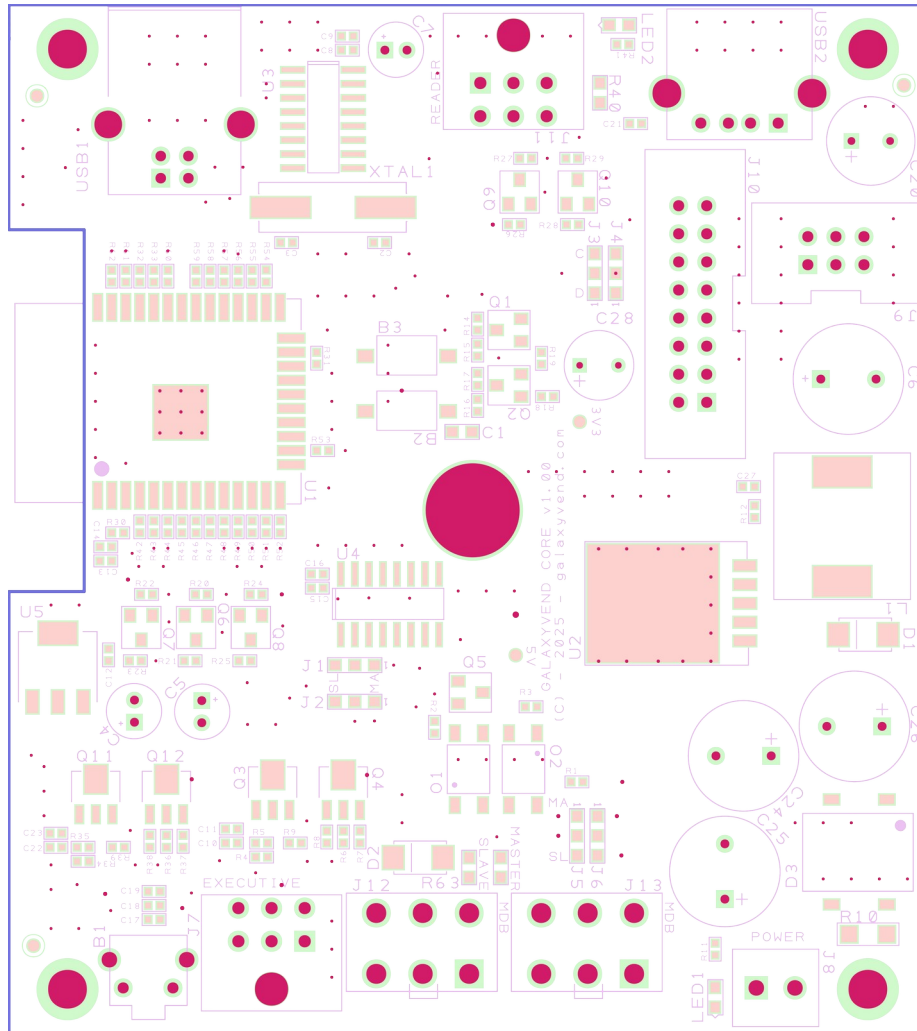


Figure 1: Core board

B. Core board connectors and buttons description

1. J7 – Executive connector – we are supplying a connecting cable with Executive connector for this device.

- PIN #1 (squared pin) – MDB POWER (out) – this pin is tied to the MDB power for future applications where Executive interface is used to connect additional modules instead of a vending machine. Not used in this configuration.
- PIN #2 – GND;
- PIN #3 – MDB POWER (out);
- PIN #4 – GND;
- PIN #5 – TX to VMC (out);
- PIN #6 – RX from VMC (IN).

2. J12 and J13 – MDB connectors – you can use any (or both) connectors to attach MDB peripherals (payment systems).

- 3. B1** – button to enable Bluetooth communication for STANDALONE version, in order to connect to the Android configuring application. To enable Bluetooth, apply power while keeping this button pressed for 15sec.
- 4. J8** - POWER (in) – this connector is used to apply necessary power for MDB peripherals. Polarity insensitive, you can apply any needed power that allows your peripheral to work (9 to 32VDC or 7 to 24VAC). Do not exceed the maximum recommended voltage for each type (DC or AC).
- 5. USB1** – USB-B connector for firmware update and USB communication on external versions (when you want to connect to a PC, SBC or Tablet).
- 6. J11** – RFID reader interface (sold separately – proprietary model – available only external versions). You can use the RFID reader (MIFARE Classic) to add serial number read function to your application.
- 7. USB2** – USB-A connector, to power external devices (maximum current 1.2A).

C. Keypress expansion module (external mode only)

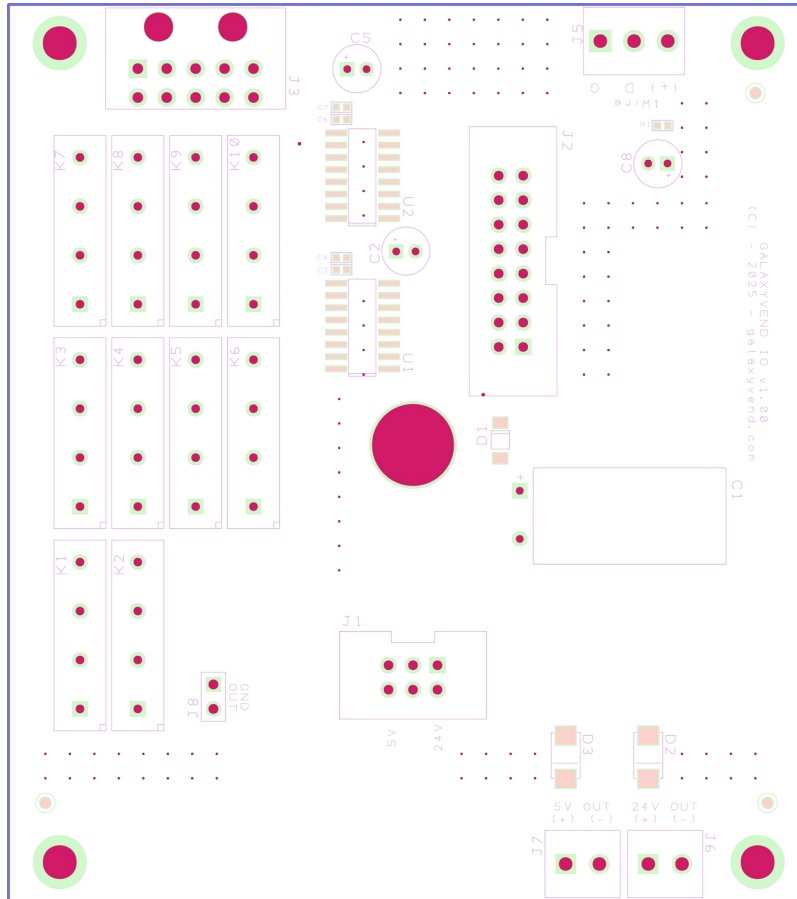


Figure 2: Keypress simulator

D. Keypress board connectors description

1. J3 – KEYPRESS (out) – this connector is used to attach to the machine’s mechanical keyboard interface. You will need to identify (reverse) your machine keypad matrix schematic (rows and columns). We are only supplying a cable that has a board compatible connector and free wires on the other and, to allow you to use the correct connector and wires combination for your specific vending machine. The KEYPRESS commands are energizing the relays 2-by-2, allowing you to put rows and columns together.

- PIN #1 – K1 – relay 1;
- PIN #2 – K2 – relay 2;
- PIN #3 – K3 – relay 3;
- PIN #4 – K4 – relay 4;
- PIN #5 – K5 – relay 5;
- PIN #6 – K6 – relay 6;
- PIN #7 – K7 – relay 7;
- PIN #8 – K8 – relay 8;
- PIN #9 – K9 – relay 9;
- PIN #3 – K10 – relay 10.

II. STANDALONE working mode

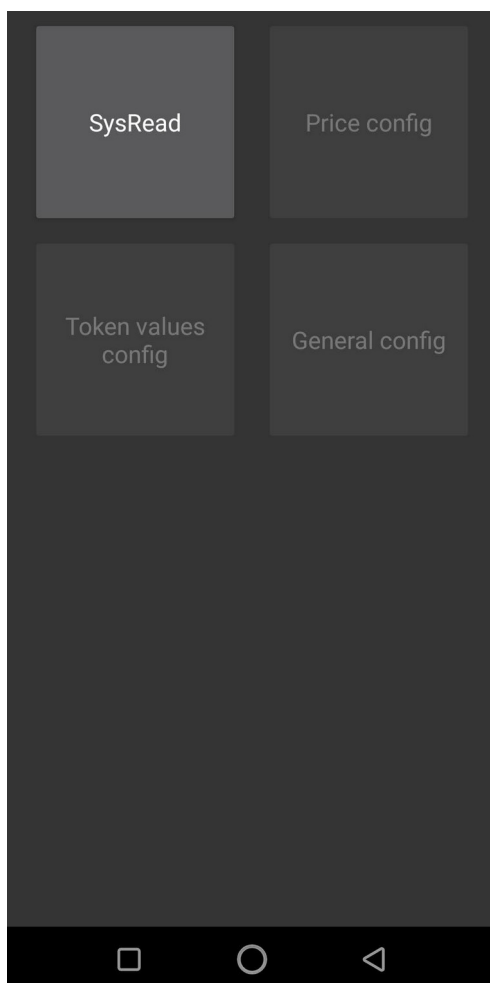
With this working mode, the device is a STANDALONE interface from Executive vending machine to MDB payment systems. You need to configure the working parameters using the Android application. Configuration is available by using an Android application only. There are no hidden menus and buttons on this device and the Android application is the easiest way to configure the device. The Android application is connecting over Bluetooth. The interface's Bluetooth must be activated for configuration and deactivated after the process is finished.

A. Bluetooth activation

To activate the Bluetooth communication, you need to keep B1 pressed on boot for about 15 seconds. Power up the device and using your Android device "Bluetooth" menu, pair with the PVEZEXE module in range. You will need to delete and follow the pairing procedure with any different GALAXYVEND EXECUTIVE TO MDB CONVERTER you may want to configure.

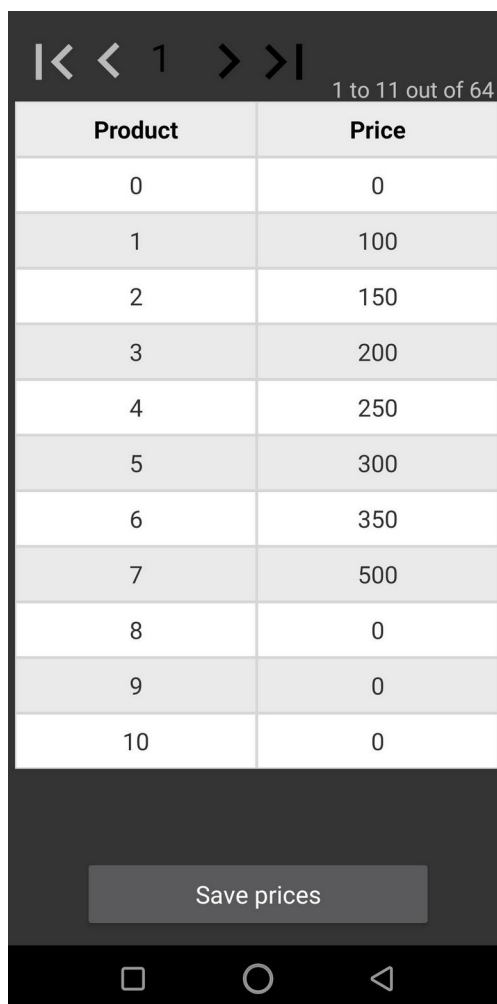
B. Android application flow

1. Start the application that will automatically try to connect to PVEZEXE device. If successful, you will get a message on your phone and the following screen will be displayed.



Press “SysRead” button, the application will read current interface settings. Data will be displayed and some short beeps can be heard. Please wait patiently while all data is retrieved.

Press “Back” button and go to “Price config” (for Executive PRICE HOLDING mode only).



Product	Price
0	0
1	100
2	150
3	200
4	250
5	300
6	350
7	500
8	0
9	0
10	0

Save prices

In the “Product” column there is the list containing products IDs. For each needed product ID you need to set the correct price (in cents) . For example, if you set the price to 150 for product ID #2, all VMC requests for this product will require a credit of 1.50 EUR. This option is only important for “Executive PRICE HOLDING” working mode. It is ignored if the prices are set on the vending machine. You need to make sure that the machine is set in “Executive PRICE HOLDING” mode and the products IDs are matching with the interface settings.

After setting the desired values, press “Save prices” button and wait for data to be saved on the device.

The “Token values” option will set the tokens values, if the interface is connected with an MDB coin acceptor that receives tokens. The configuration depends on your coin acceptor settings and they must match (for example, set on channel #1 150 if your coin acceptor is sending a “token received” message on channel #1 for 1.50EUR tokens. Push “save token values” when you are done setting the token values, then hit “Back button to go back to the main screen.

The “General config” option allows you to set some general interface settings. Please use caution with these settings; incorrect configuration may render the controller inoperable.

The screenshot shows a mobile application interface for configuring general settings. It features a list of configuration items, each with a label in a grey box and a corresponding value in a text input field. The items are: SN (ATM000000001), ESF (10), EDP (4), EMC (1000), EMV (1), EMCH (500), EFV (1), EVT (50), and EPH (1). Below these fields is a large grey button labeled "Commit changes". At the very bottom of the screen, there are three standard Android navigation icons: a square, a circle, and a triangle.

Label	Value
SN	ATM000000001
ESF	10
EDP	4
EMC	1000
EMV	1
EMCH	500
EFV	1
EVT	50
EPH	1

Commit changes

- **SN** is the interface serial number, that will be also periodically show on interface’s display. This string should be always 12 characters length, any other length will make the interface to act unpredictable. You can leave this parameter as default, unless you are using an MDB sniffer/telemetry that will send this value to the back-end for machines management.

- **ESF** – is the value for Executive scaling factor. All set credit and product prices are sent between the interface and the Executive vending machine multiplied/divided by this scaling factor. For most of the currencies, you may use a scaling factor of 10, which gives you a credit/price resolution of 10cents. For example, if your last digit will be always 0 (1.50, 2.30, 1.10, etc.), you will use a scaling factor of 10. If your last digit is significant (1.55, 2.34, 1.12, etc.), you will need to use a scaling factor of 1. Please note that the maximum credit you can send to the machine will vary, depending on the scaling factor. The maximum value you can send to the machine is 0xFFFF (65535). That means that for a scaling factor of 1, you will be able to send a maximum credit of 655.35 with 2 decimals settings on the interface and the machine and a

maximum credit of 655.30 in the same conditions, with the scaling factor being 10. Also, the maximum credit amount depends on your decimal point settings on the machine and on the interface. Please note that, if the machine has a scaling factor setting option in its configuration menu, the value set on the machine and the value set on the interface must match. Otherwise, the credit value on the display will be incorrect.

- **EDP** – is the value for Executive decimal point. The value is binary sent and represents the position of the decimal point (for example, to have no decimals, the value should be 0, to have one decimal, the value should be 2 and to have two decimals, the value should be 4). You need to take care that the value set on the interface should match with the value set on the vending machine. Otherwise, you may get wrong values on the VMC display.

- **EMC** – is the value (cents) of the maximum cash credit accepted by the interface for transactions. When the customer has inserted money corresponding with this set amount, all connected MDB payment systems will be inhibited. After a purchase, if the available credit falls below this value, the payment systems will be automatically enabled by the interface. If the bill is supporting escrow function, this will be automatically activated and the bills that will lead to a higher value will be returned to the customer.

- **EMV** – is the current set for multivend/single vend option. If this value is set to 0, the interface will work in single vend mode (it will automatically return change after each sale). If this value is set to 1, the interface will work in multivend mode, the remaining credit is kept on the machine to allow another selection, after each sale. To get the change, the customer must press the escrow lever on changer (usually the mechanical change button on the machine will push the lever or will activate a small motor that will push the lever).

- **EMCH** – is the value for the maximum change allowed to be returned to the customer after a transaction. If the remaining credit will be higher than this value after a transaction, the interface will not automatically return change (in single vend mode) and will not manually dispense change (in multivend mode). The remaining credit is kept on the machine to allow another selection. This option is usually used to prevent using the machine as a change machine (for example if the customer is inserting bills or coins to the maximum credit value and is selecting a low price product in order to obtain coins).

- **EFV** – is the force vend option for the interface. If this value is set to 1, the machine will not return change if the machine did not perform a vend. If the value is set to 0, the machine will return change in any situation where a credit is available. This parameter is used to avoid customers using the machine as a change machine (for example, inserting a bill or a bigger value coin, to get coins/smaller coins). If the value is set to 0, the machine will allow the usage as a change machine. There is one exception: when the machine is reporting a vend failed, the customer will be able to get the inserted amount as coins, if does not want to select other product, due to machine malfunction.

- **EVT** – is the vend timeout (seconds). During a vend stage (product dispensing), if the interface will not receive a “vend success” or a “vend failed” answer from the machine. This may happen due to a machine temporary malfunction, for example. If the timeout interval has passed, the interface will return to the idle mode and the credit will be kept for future product selections. Since this is a backup method to avoid machine going out of order for not normal preparation duration, you need to make sure this time is covering the longest preparation time. For example, if the the hot drink preparation duration is 40 seconds, make sure you set this value to at least 60 seconds, because sometime the preparation duration can be longer (for example if one of the machine pump is out of parameters and cannot fill the tanks in a proper time).

- **EPH** – is the working mode setting. If this value is set to 1, the interface will work in PRICE HOLDING mode and it will use its internal price list. If this value is set to 0, the interface is expecting prices from the machine. Make sure your machine and the interface settings are matching, otherwise, strange behavior is expected (wrong price calculation and credit subtracting).

After modifying any of the above settings, push the label in front of the field to save it to the device. When finishing all modifications, push “Commit changes” to save all settings in the NV flash memory.

Remove the “Bluetooth enable” connector from the DB9 interface and restart the device when all settings are done to allow the interface load and set the new configuration.

III. External working mode (connected over USB to computer, PLC, or Raspberry Pi and compatible SBCs or over MQTT)

This working mode can be used for vending machines upgrade or remote credit loading. For example, you can use a Raspberry Pi (or compatible SBC) to send credit to the Executive vending machine, to receive and display credit information on a big screen, to display selected product name and price, to display the vend result (failed or succeeded), to send counters to a server, to receive remote settings and credit, etc. It may be a good start for a vending machines retrofit activity. There are a lot of commands for the interface and a lot of messages coming from the interface, in order to achieve this goal. **NOTE: For this working mode, set your vending machine to PRICE HOLDING.**

Below you can find the commands used over USB. To get information for MQTT messages format using our SaaS backend, please contact us.

A. Bill validator related commands

1. 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.

2. 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).

3. 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
---	--

4. 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.

5. 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

6. 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)	- 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)

- BILLNOTINITED	- The bill validator was not previously initialized and the information is not available
-----------------	--

7. Reset bill validator

Command	
BILLRESET	This command will reset the bill validator. The interface will automatically initialize the bill validator again and it will automatically enable it if the vending machine is up and running. 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 - BILLRESETOK	- 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.

B. Coin related commands

1. Reset coin acceptor

Command	
COINRESET	This command will reset the coin acceptor/changer. The interface will automatically initialize the coin acceptor/changer again and it will automatically enable if the vending machine is up and running. After issuing this command, you will receive some unsolicited messages while the interface is initializing the coin acceptor/changer. You may receive the following messages: - COINREADY - COINOK Please check the unsolicited messages information below in the “Interface unsolicited messages” section.
Possible answers	
- COINRESETFAIL - COINRESETOK	- 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.

2. 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) - COINTBSTATUSOK - COINTBSTATUSFAIL	- 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.

3. 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 - CHANGEREMAINING(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.)

4. 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 - CHANGEREMAINING(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.)

5. Check if the coin acceptor/changer was initialized

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).

6. 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

7. 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.

8. 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.

9. 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. Command - PAYCASH This command selects cash payment for the latest PRODSELECT(x,0) command. Possible answers - PAYCASHOK - NOSELECTEDPRODUCT - Command received and successfully processed. - There was no previously selected product for sale.

10. 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

11. 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)	- 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.
- COINNOTINITED	- The coin acceptor/changer was not previously initialized and the information is not available

12. 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.

C. Cashless related commands

For all cashless commands, “x” can be 1 or 2, depending on the cashless device address (0x10 or 0x60)

1. Reset cashless device

Command	
CSLSxRESET	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: - CSLSxREADY - CSLSxOK Please check the unsolicited messages information below in the “Interface unsolicited messages” section.
Possible answers	
- CSLSxRESETFAIL - CSLSxRESETOK	- If the cashless device could not be reset when the CSLSxRESET 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. Request cashless current revalue limit

Command	
CSLSxREVALLIMITREQ?	This command will read the current revalue limit.
Possible answers	
- CSLSxREVALLIMIT(X) - CSLSxNOSESSION - CSLSxNOREVALSUPPORT - CSLSxREVALLIMITFAIL	- X is the maximum revalue amount accepted by the cashless device for further CSLSxREVALREQ (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 excuted.

3. Request approval for a vend

Command	
CSLSxVNDREQ(A,B)	This command will request a vend approval from the cashless device - A is the scaled price (16bit value maximum) - B is the item ID/selection number (16bit value maximum)
Possible answers	
- CSLSxNOSESSION - CSLSxVNDREQOK - CSLSxVNDREQFAIL	- 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.

4. Confirm a success vend to the cashless device

Command	
CSLSxVNDSUCC(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	
- CSLSxNOSESSION - CSLSxVNDSUCCOK - CSLSxVNDSUCCFAIL	- 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.

5. Report a vend failure to the cashless device

Command	
CSLSxVNNDFAIL	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	
- CSLSxNOSESSION - CSLSxVNDFAILOK - CSLSxVNDFAILFAIL	- 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.

6. Report a cash sale to the cashless device

Command	
CSLSxCASHSALE(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	
- CSLSxNOCASHSALESUPPORT - CSLSxCASHSALEOK - CSLSxCASHSALEFAIL	- 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 sub-command. - If the interface successfully received and parsed the command. - If the interface was not able to successfully receive and parse the command.

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

Command	
CSLSxREVALREQ(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	
- CSLSxNOREVALSUPPORT	- 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)
- CSLSxNOSESSION	- If there is no cashless session opened, the cashless device will not be able to load any amount to customer's account.
- CSLSxREVALOVER	- The specified amount exceeds the cashless maximum revalue capacity for the current session.
- CSLSxREVALREQOK	- If the interface successfully received and parsed the command.
- CSLSxREVALREQFAIL	- If the interface was not able to successfully receive and parse the command.

8. Get last 10 cashless device codes

Command	
CSLSxSTATUS?	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	
- CSLSxSTATUS(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.

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

Command	
CSLSxINITED?	This command will check if the cashless device was initialized by the interface after power-up or after issuing CSLSxRESET command
Possible answers	
- CSLSxINITEDOK - CSLSxNOTINITED	- The cashless device was successfully initialized - The cashless device was not initialized (missing or not initialized, yet).

10. Check if the cashless device was enabled

Command	
CSLSxACTIVE?	This command will check if the cashless was previously activated by a CSLSxENABLE.
Possible answers	
- CSLSxACTIVEOK - CSLSxNOTACTIVE	- The cashless device is currently enabled - The cashless device is not currently enabled

11. Get the cashless device information

Command	
CSLSxINFOREQ?	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	
- CSLSxINFOREQ(A,B,C) - CSLSxNOTINITED	- 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

12. Get the cashless device settings

Command	
CSLSxSETTINGS?	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	
- CSLSxSETTINGS(A,B,C,D,E,F) - CSLSxNOTINITED	- 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 its 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

D. System and executive related commands

1. Interface reboot

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

2. 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.

3. 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(a1,...,a19)	- a1 – N/A - a2 – N/A - a3 – N/A - a4 – N/A - a5 – N/A - a6 – N/A - a7 – total number of received bills - a8 – total value of received bills - a9 – total number of rejected bills - a10 – total number of received coins - a11 – total value of received coins - a12 – total number of rejected coins - a13 – total number of cashless #1 transactions - a14 – total value of cashless #1 transactions - a15 – total number of cashless #2 transactions - a16 – total value of cashless #2 transactions - a17 – total number of received tokens - a18 – total value of received tokens - a19 – total value of coin change returned to customers

4. Requesting interface settings

Command	
- VMCSETTINGS?	This command requests the board settings
Possible answers	
- VMCSETTINGS(a1, ..., a23)	- a1 – N/A - a2 – N/A - a3 – N/A - a4 – Maximum allowed cash credit - a5 – Maximum allowed change amount - a6 – N/A - a7 – N/A - a8 – Force vend (1 = enable, 0 = disable) - a9 – N/A - a10 – N/A - a11 – N/A - a12 – single vend (1 = enable, 0 = disable) - a13 – cashless feature level (2 = MDB Level 2, 3 = MDB Level 3) - a14 – N/A - a15 – N/A - a16 – N/A - a17 – RFID reader attached (1 = reader present, 0 = reader not used) - a18 – RFID reader SN read sector (default 0, unless you store your own information on the first 4 bytes of other sector) - a19 – payment method timeout – not internally used by the device, provided for the user app, if needed (timeout for payment method selection - seconds) - a20 – payment timeout – not internally used by the device, provided for the user app, if needed (timeout for user to perform the payment - seconds) - a21 – credit timeout – cash credit expiration time (seconds) - a22 – to idle timeout – time (seconds) from the last success/failure message to return to the main screen. - a23 – change timeout – time (seconds) to successfully return change. If this timeout exceeds, it is considered that the change operation failed.

5. Sending interface settings

Command	
- VMCSETTINGS(a1, ..., a23)	This command sets the board settings: - a1 – N/A - a2 – N/A - a3 – N/A - a4 – Maximum allowed cash credit - a5 – Maximum allowed change amount - a6 – N/A - a7 – N/A - a8 – Force vend (1 = enable, 0 = disable) - a9 – N/A - a10 – N/A - a11 – N/A - a12 – single vend (1 = enable, 0 = disable) - a13 – cashless feature level (2 = MDB Level 2, 3 = MDB Level 3) - a14 – N/A - a15 – N/A - a16 – N/A - a17 – RFID reader attached (1 = reader present, 0 = reader not used) - a18 – RFID reader SN read sector (default 0, unless you store your own information on the first 4 bytes of other sector) - a19 – payment method timeout – not internally used by the device, provided for the user app, if needed (timeout for payment method selection - seconds) - a20 – payment timeout – not internally used by the device, provided for the user app, if needed (timeout for user to perform the payment - seconds) - a21 – credit timeout – cash credit expiration time (seconds) - a22 – to idle timeout – time (seconds) from the last success/failure message to return to the main screen. - a23 – change timeout – time (seconds) to successfully return change. If this timeout exceeds, it is considered that the change operation failed. You need to send command SAVESETTINGS after issuing this command, to save the settings in the non-volatile memory. It is also recommended to send SYSRESET command so the device reboots and performs a full settings reload.
Possible answers	
- VMCSETTINGSOK	- command correctly executed.
- VMCSETTINGSFAIL	- command execution failed (parameters error).

6. Sending some simulated cash credit to the VMC

Command	
- EXECREDIT(a1, ...,a4)	This command will send some simulated cash credit to the vending machine. - a1 is the scaled value for the credit you need to send to the machine. For example, to send 1.10 credit to the vending machine, you need to issue the command EXECREDIT(110,1,4,0) - a2 is the scaling factor (1, 10 or 100) - a3 is the decimal places number (if == 0 – no decimals, if == 2 – one decimal, if == 4, 2 decimals)
Possible answers	
- EXECREDITOK	- The command were successfully received by the interface

7. Saving settings to non-volatile memory (persistent after reboot)

Command	
- SAVESETTINGS	This command will save all settings to the non-volatile memory. It is recommended to use this command after altering any of the board's settings. It is also recommended to send SYSRESET command after saving settings, device reboots and reloads all the parameters.
Possible answers	
- SAVESETTINGSOK - SAVESETTINGSFAIL	- command correctly executed. - command execution failed.

8. Reset to factory default

Command	
- FACTORYRESET	This command will reset all settings, counters and stock. Important!!! If you are using custom MQTT and connection settings, your device will return to its defaults and needs reconfiguring over USB, since it will not reconnect with the custom settings. Do not issue this command remotely over MQTT.
Possible answers	
- FACTORYRESETOK - FACTORYRESETFAIL	- command correctly executed. - command execution failed.

9. Reading board serial number

Command	
- SNREAD?	This command will interrogate board's serial number
Possible answers	
- SNREAD(xxxxxxxxxxxx)	- command correctly executed, serial number returned.

10. Keypress (one key) – available only with the optional keypress module

Command	
- KEYPRESS1(a1, a2, a3)	This command sends the command to put 2 keyboard pins together (similar with a key pressed on machine's matrix mechanical keyboard). - a1 – first of the 2 pins you want to tie together - a2 – second of the 2 pins you want to tie together - a3 – time to keep the key pressed (milliseconds) – minimum 500
Possible answers	
- KEYPRESSOK - KEYPRESSFAIL	- command correctly executed. - command execution failed.

11. Keypress (two successive keys) – available only with the optional keypress module

Command	
- KEYPRESS2(a1, ..., a6)	This command sends the command to put 2 keyboard pins together (similar with a key pressed on machine's matrix mechanical keyboard). First key pressed: - a1 – first of the 2 pins you want to tie together - a2 – second of the 2 pins you want to tie together - a3 – time to keep the key pressed (milliseconds) – minimum 500 Second key pressed: - a4 – first of the 2 pins you want to tie together - a5 – second of the 2 pins you want to tie together - a6 – time to keep the key pressed (milliseconds) – minimum 500
Possible answers	
- KEYPRESSOK - KEYPRESSFAIL	- command correctly executed. - command execution failed.

12. Keypress (three successive keys) – available only with the optional keypress module

Command	
- KEYPRESS3(a1, ..., a9)	This command sends the command to put 2 keyboard pins together (similar with a key pressed on machine's matrix mechanical keyboard). First key pressed: - a1 – first of the 2 pins you want to tie together - a2 – second of the 2 pins you want to tie together - a3 – time to keep the key pressed (milliseconds) – minimum 500 Second key pressed: - a4 – first of the 2 pins you want to tie together - a5 – second of the 2 pins you want to tie together - a6 – time to keep the key pressed (milliseconds) – minimum 500 Third key pressed: - a7 – first of the 2 pins you want to tie together - a8 – second of the 2 pins you want to tie together - a9 – time to keep the key pressed (milliseconds) – minimum 500
Possible answers	
- KEYPRESSOK - KEYPRESSFAIL	- command correctly executed. - command execution failed.

13. Keypress (four successive keys) – available only with the optional keypress module

Command	
- KEYPRESS4(a1, ..., a12)	This command sends the command to put 2 keyboard pins together (similar with a key pressed on machine's matrix mechanical keyboard). First key pressed: - a1 – first of the 2 pins you want to tie together - a2 – second of the 2 pins you want to tie together - a3 – time to keep the key pressed (milliseconds) – minimum 500 Second key pressed: - a4 – first of the 2 pins you want to tie together - a5 – second of the 2 pins you want to tie together - a6 – time to keep the key pressed (milliseconds) – minimum 500 Third key pressed: - a7 – first of the 2 pins you want to tie together - a8 – second of the 2 pins you want to tie together - a9 – time to keep the key pressed (milliseconds) – minimum 500 Forth key pressed: - a10 – first of the 2 pins you want to tie together - a11 – second of the 2 pins you want to tie together - a12 – time to keep the key pressed (milliseconds) – minimum 500
Possible answers	
- KEYPRESSOK - KEYPRESSFAIL	- command correctly executed. - command execution failed.

14. Check if the VMC is enabled or out of order

Command	
- EXEVMCACTIVE?	This command interrogates the Executive machine status.
Possible answers	
- EXEVMCACTIVE(0) - EXEVMCACTIVE(1)	- machine is not ready for selling products (possible out of order). - machine is ready for selling products.

15. Check current available credits

Command	
- SHOWCREDIT?	This command interrogates the current available credits.
Possible answers	
- CREDITS(a1, ..., a6)	- a1 – total credit - a2 – cash credit - a3 – cashless #1 available credit - a4 – cashless #1 authorized (approved) amount - a5 – cashless #2 available credit - a6 – cashless #2 authorized (approved) amount

16. Clear all current credit variables

Command	
- CREDITDELETE	This command clears all current credit variables described at 17.
Possible answers	
- CREDITDELETEOK	- command successfully executed

17. Clear cash credit only

Command	
- CLEARCREDIT	This command clears only cash credit variables
Possible answers	
- DELETECASHCREDITOK	- command successfully executed

18. Resetting all MDB peripherals (payment systems)

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

19. Enable all cash MDB peripheral

Command	
- CASHENABLE	This command enables all attached cash peripherals
Possible answers	
- CASHENABLEOK	- command successfully executed

20. Disable all cash MDB peripheral

Command	
- CASHDISABLE	This command disables all attached cash peripherals
Possible answers	
- CASHDISABLEOK	- command successfully executed

21. 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.

22. Check peripherals' status

Command	
- VMCSTATUS?	This command checks board's and peripherals status.
Possible answers	
- VMCSTATUS(a1, ..., a32)	This message is also received as an unsolicited message every time a board or peripheral status changed - a1 – if == 1, coin acceptor/changer connected and initialized, if == 0, coin acceptor not available. - a2 – if == 1, bill validator connected and initialized, if == 0, bill validator not available. - a3 – if == 1, cashless #1 connected and initialized, if == 0, cashless #1 not available. - a4 – if == 1, cashless #2 connected and initialized, if == 0, cashless #2 not available. - a5 – N/A - a6 – N/A - a7 – if == 1, RFID reader connected and initialized, waiting for reading cards, if == 0, reader not available. - a8 – a32 – N/A

26. File system format

Command	
- FSFORMAT	This command formats the non-volatile memory file system. Important!!! Do not issue this command remotely (over WiFi, while connected to MQTT backend). This command erases all custom settings and returns to the factory settings. The device may not reconnect to MQTT backend if the factory settings are different than your custom settings configured over the USB.
Possible answers	
- FSFORMATSTART - FSFORMATDONE	- Command received, execution started - Command executed and completed. Device will automatically reboot after FSFORMAT command.

27. Product stock update

Command	
- PRODREFILL(a1, a2)	This command updates the stock for a specific product. - a1 – the product ID (selection number) - a2 – the product current stock Please not that, when you update stock product-by-product, you need to issue command SAVESTOCKS after updating the latest product in order to save the new stock updates to the non-volatile memory.
Possible answers	
- PRODREFILLOK - PRODREFILLFAIL	- Command received, successfully executed - Command execution failed

28. Complete stock update

Command	
- FULLREFILL	This command updates the stock for all products, settings the current stock for each product to the value of that product default stock (see below – products settings). Current stocks are automatically saved to the non-volatile memory, no need to issue SAVESTOCKS command.
Possible answers	
- FULLREFILLOK - FULLREFILLFAIL	- Command received, successfully executed - Command execution failed

29. Save all current stocks to non-volatile memory

Command	
- SAVESTOCKS	This command saves all current stocks to the non-volatile memory. Issue this command after finishing to selectively update current stocks for a part of your products and not for all products with FULLREFILL.
Possible answers	
- SAVESTOCKSOK - SAVESTOCKSFALL	- Command received, successfully executed - Command execution failed, you may neet to use FSFORMAT and reconfigure your device.

30. Read current stocks (obsolete, do not use it, use CURRENTSTOCK? instead)

Command	
- READSTOCKS?	This command displays current stocks for all products – obsolete, do not use it, it may lead to unexpected behaviors.
Possible answers	
- READSTOKSOK(a1, ..., a64)	- Command received, successfully executed - a1 to a64 – current stock for the product id in position (a1 for product selection number 1, a2 for product selection number 2, etc.)

31. Select product to sale

Command	
- PRODSELECT(a1, a2)	This command is starting a product sale. - a1 – product id (selection number) - a2 – payment method (0 – asks back HOWDOYOU PAY, 1 – pay cash, 2 – pay cashless)
Possible answers	
- PRODAVAILABLE - SALEACTIVE - MACHINENOTREADY - NOTACTIVEPRODUCT - NOENOUGHSTOCK	- Product is available (enough stock, no previous errors on delivery). - A product selection command was previously issued and the transaction is still active. - The machine is not ready for any sale. - The product is disabled (product type == 0) - Selected product has not enough stock for purchases. Update the current stock with PRODREFILL() if necessary.

32. Cancel product selection

Command	
- PRODCANCEL	This command cancels a previous PRODSELECT() command
Possible answers	
- PRODCANCELOK	- Command received and successfully processed

33. Select cash payment method

Command	
- PAYCASH	This command selects cash payment for the latest PRODSELECT(x,0) command.
Possible answers	
- PAYCASHOK - PAYCASHFAIL - NOSELECTEDPRODUCT	- Command received and successfully processed. - Command execution failed. - There was no previously selected product for sale.

34. Select cashless #1 payment method

Command	
- PAYCARD(1)	This command selects cashless #1 payment for the latest PRODSELECT(x,0) command.
Possible answers	
- PAYCARDOK(1) - NOSELECTEDPRODUCT - CSLS1NOTREADY - CSLS1NOSESSION - PAYCARDFAIL(1)	- Command received and successfully processed. - There was no previously selected product for sale. - Cashless #1 is not prepared for payments. - Cashless #1 is configured in MDB Level 2 mode, but the session was not opened previously. - Command execution failed.

35. Select free vend payment method

Command	
- PAYFREE	This command approves free vend dispensing for the latest PRODSELECT(x,0) command.
Possible answers	
- PAYCARDOK(1) - NOSELECTEDPRODUCT - CSLS1NOTREADY - CSLS1NOSESSION - PAYCARDFAIL(1)	- Command received and successfully processed. - There was no previously selected product for sale. - Cashless #1 is not prepared for payments. - Cashless #1 is configured in MDB Level 2 mode, but the session was not opened previously. - Command execution failed.

36. Requesting RFID card available credit (only working with the MQTT over WiFi version, using our SaaS backend)

Command	
- RDRCREDIT(<card_serial_number>)	This command sends a read available amount to the SaaS backend.
Possible answers	
- RDRCREDITOK - RDRCREDITFAIL After DB interrogation confirmation, the interface returns the available credit: - RDRCREDIT(<value>)	- Command received and successfully processed. - Command execution failed. - The value (in cents), available for the requested user account.

37. Reading product parameters

Command	
- GETPRODUCT(a1)	This command reads the complete settings of an indicated product: - a1 – the product ID (selection_number).
Possible answers	
- GETPRODUCTOK(a1, ..., a20)	- Command received and successfully processed. - a1 – Selected product ID. - a2 – N/A - a3 – Product default stock (when the machine is full for this product). If == 0 – unlimited stock (stock monitoring is disabled). - a4 – Product current stock. - a5 – Product price (in cents) – 245 = 2.45EUR - a6 – Product quantity #1 – quantity delivered for the selected product (for stock management purposes). This value is subtracted from the current stock for each successful sale. If default stock is set to 0, the stock management is disabled. - a7 – N/A - a8 – a13 – N/A - a14 – keypress combination #1 (works only with the keypress expansion board attached). HI byte is the first pin, LO byte is the second pin. Calculus example. To put pin #1 and pin #6 together on sale, the value will be $(1 * 256) + 6$. To put pin #2 and pin #7 together, the value will be $(2 * 256) + 7$. - a15 - keypress combination #2 - a16 - keypress combination #3 - a17 - keypress combination #4 - a18 – keypress length (milliseconds) - a19 – keypress space (milliseconds) - a20 – product delivery timeout (seconds). If the product is not successfully dispensed within the configured time, the sale is considered a failure (the cash credit remains on the machine or, the cashless credit is refunded).
- GETPRODUCTFAIL	- Command execution failed.

38. Writing product parameters

Command	
- SETPRODUCT(a1, ..., a19)	This command sends configuration for a product: - a1 – the product ID (selection number). - a2 – not used, it must always be 7. Do not change product type to other value than 7. - a3 – product default stock. - a4 – product price - a5 – product quantity (for stock management purposes, how many products are delivered for each sale) - a6 → a12 – N/A - a13 – keypress combination #1 (works only with the keypress expansion board attached). HI byte is the first pin, LO byte is the second pin. Calculus example. To put pin #1 and pin #6 together on sale, the value will be $(1 * 256) + 6$. To put pin #2 and pin #7 together, the value will be $(2 * 256) + 7$. - a14 - keypress combination #2 - a15 - keypress combination #3 - a16 - keypress combination #4 - a17 – keypress length (milliseconds) - a18 – keypress space (milliseconds) - a19 – product delivery timeout (seconds). If the product is not successfully dispensed within the configured time, the sale is considered a failure (the cash credit remains on the machine or, the cashless credit is refunded). This command cannot be used to update current stocks. Use PROREFILL or FULLREFILL for this purpose. Important!!! You need to send SAVESETTINGS command when you finished altering products' settings, in order to be saved to the non-volatile memory. It is also recommended to send command SYSRESET, in order to reboot the device and completely load the saved settings (including products settings). Very important!!! For N/A parameters, always set the same values you receive issuing command GETPRODUCT(). Some of them may be used for internal purposes and not available for your app to handle, but altering their values may lead to unexpected behaviors. So, before using SETPRODUCT() function, make sure you use GETPRODUCT() for the same product ID and store the N/A parameters that you MUST send with SETPRODUCT() with the exact same value. Also, please note that SETPRODUCT() is 1 parameter short, since it does not include current stock.
Possible answers	
- SETPRODUCTOK(a1) - SETPRODUCTFAIL	- Command received and successfully processed, a1 is the product ID parameters modification was operated. - Command execution failed.

39. Requesting complete stock status

Command	
- CURRENTSTOCK?	This command requests all current stocks, for all 64 available products. Please note that it is a slow method, since the current stock for each product is sent separately (it may take up to 30 seconds for all current stocks to be listed). A better solution is to use GETPRODUCT() function for each product you are interested in.
Possible answers	
- CURRENTSTOCK(a1, ..., a2)	- Command received and successfully processed. - a1 – product ID (selection number) - a2 – current stock
- CURRENTSTOCKFAIL	- Command execution failed.

40. Setting interface working flow (there are 2 working flows in external mode)

Command	
- FOREXTERNAL(a1)	This command sets the external working flow: - a1 == 0, the keypress functions are automatically executed if the accumulated credit matches product's price. - a1 == 1, the keypress functions are not automatically executed if the accumulated credit matches product's price. You need to issue KEYPRESSx() command to simulate customer's button presses to the mechanical keyboard interface. You need to send SAVESETTINGS after issuing this command and it is recommended to send SYSRESET command after saving the settings.
Possible answers	
- FOREXTERNALOK - FOREXTERNALFAIL	- Command received and successfully processed. - Command execution failed.

41. Reading interface working flow (there are 2 working flows in external mode)

Command	
- FOREXTERNAL?	This command sets the external working flow.
Possible answers	
- FOREXTERNAL(a1)	- Command received and successfully processed. - a1 == 0, the keypress functions are automatically executed if the accumulated credit matches product's price. - a1 == 1, the keypress functions are not automatically executed if the accumulated credit matches product's price. - Command execution failed.

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 or MQTT backend, 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.

15. 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.

18. 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 total value finished

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

21. Bill dispensing total value

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 its 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) – obsolete – not used	- 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 its 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.

37. Bill validator have a defective motor

Message	Description
BILLDEFMOTOR	- Bill validator encountered one of its motors failure

36. Bill validator have a defective sensor

Message	Description
BILLSENSORFAIL	- Bill validator encountered one of its sensors failure

37. Bill validator is busy

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

38. Bill validator ROM error

Message	Description
BILLROMERROR	- Bill validator encountered an internal ROM error

39. Bill validator jam

Message	Description
BILLJAM	- Bill validator encountered a bill jam error

40. Bill validator was reset

Message	Description
BILLRESET	- Bill validator has just been reset.

41. Bill removed from bill validator

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

42. Bill validator cashbox has been removed

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

43. Bill validator 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

44. 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.

45. Bill removed after it was credited

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

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

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

47. Recycler has received a change request

Message	Description
RECYCLERCHGREQUEST	- Recycler has received a change request

B. Cashless related unsolicited messages

1. Cash sale successfully reported to the cashless device

Message	Description
CSLSxCASHSALEOK	- A cash sale was successfully reported to the cashless device. "x" can be 1 or 2, depending on the cashless that have received the information.

2. Cash sale reporting to the cashless device failed

Message	Description
CSLSxCASHSALEFAIL	- A cash sale reporting to the cashless device failed.

3. Cashless device setup time exceeded

Message	Description
CSLSxSETUPEXCEED	- 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.

4. Cashless device max/min prices reporting time exceed

Message	Description
CSLSxMAXMINEXCEED	- 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.

5. Cashless device poll time exceeded

Message	Description
CSLSxPOLLEXCEED	- 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.

6. Cashless device expansion request ID time exceeded

Message	Description
CSLSxEXPREQIDEXCEED	- 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.
--	---

7. Cashless device expansion enable options time exceeded

Message	Description
CSLSxEXPENOPTXCEED	- 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.

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

Message	Description
CSLSxALWAYSIDLE	- The cashless device have Always Idle support and the interface will try to enable it

9. Cashless device writing date/time exceeded

Message	Description
CSLSxWRDTEXCEED	- 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.

10. Cashless device sent a display message

Message	Description
CSLSxDISPMSG(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

11. Cashless device sent a BEGIN SESSION message

Message	Description
CSLSxBEGIN(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

12. Cashless device sent a VEND APPROVED message

Message	Description
CSLSxVNDAPP(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)
--	--

13. Cashless device sent a VEND DENIED message

Message	Description
CSLSxVNDDEN	- The cashless device sent a VEND DENIED message to the interface

14. Cashless device sent an END SESSION message

Message	Description
CSLSxENDSESSION	- The cashless device sent an END SESSION message to the interface

15. Cashless device sent a CANCELED message

Message	Description
CSLSxCANCELED	- The cashless device sent a CANCELED message to the interface

16. Cashless device is ready

Message	Description
CSLSxREADY	- The cashless device was correctly initialized and is ready to be enabled.

17. Cashless device returned a malfunction error

Message	Description
CSLSxMALFUNCTION(X)	- The cashless device returned a malfunction message - X is the internal malfunction code, its value depends on the cashless device and you can find more information in its manual

18. Cashless device returned COMMAND OUT OF SEQUENCE message

Message	Description
CSLSxCMDOUTOFSEQ	- The cashless device returned a COMMAND OUT OF SEQUENCE message

19. Cashless device sent a REVALUE APPROVED message

Message	Description
CSLSxREVALAPP	- The cashless device returned a REVALUE APPROVED message

20. Cashless device sent a REVALUE DENIED message

Message	Description
CSLSxREVALDEN	- The cashless device returned a REVALUE DENIED message

21. Cashless device sent a REVALUE LIMIT message

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

22. Cashless device sent a DATE/TIME request message

Message	Description
CSLSxDTREQ	- The cashless device is requesting a date/time command to synchronize its internal RTC

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

Message	Description
CSLSxDTSENDOK	- The interface successfully sent date/time command to the cashless device.

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

Message	Description
CSLSxDTSENDFAIL	- The interface failed sending date/time command to the cashless device.

25. Interface successfully enabled the cashless device

Message	Description
CSLSxENABLED	- The interface successfully enabled the cashless device.

26. Interface successfully enabled the cashless device

Message	Description
CSLSxENABLED	- 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 its working stage. Your application should retry later.

11. 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.

12. 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 its 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

13. 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)

14. 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)

15. 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)

16. Coin acceptor/changer is in normal condition

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

17. 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.

18. 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.

19. Coin acceptor/changer has a defective tube sensor

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

20. 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).

21. Coin changer detected an acceptor disconnection

Message	Description
COINACCUNPL	- Coin changer detected an acceptor disconnection.

22. Coin acceptor/changer detected a tube jam

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

23. Coin acceptor/changer detected an internal ROM error

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

24. Coin acceptor/changer detected a routing error

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

25. Coin acceptor/changer detected reset condition

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

26. 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

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

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

D. Executive and system messages

1. Vending machine is not answering

Message	Description
EXENOANS	- Vending machine is not answering on STATUS message

2. CRC error on message from the vending machine

Message	Description
EXECRCERR	- Last message from the vending machine fails on CRC

3. The ID of the selected product (PRICE HOLDING mode)

Message	Description
EXESELECTION(a1, ..., a20)	- The customer has selected a product: - a1 – Selected product ID. - a2 – N/A - a3 – Product default stock (when the machine is full for this product). If == 0 – unlimited stock (stock monitoring is disabled). - a4 – Product current stock. - a5 – Product price (in cents) – 245 = 2.45EUR - a6 – Product quantity #1 – quantity delivered for the selected product (for stock management purposes). This value is subtracted from the current stock for each successful sale. If default stock is set to 0, the stock management is disabled. - a7 – N/A - a8 – a13 – N/A - a14 – keypress combination #1 (works only with the keypress expansion board attached). HI byte is the first pin, LO byte is the second pin. Calculus example. To put pin #1 and pin #6 together on sale, the value will be $(1 * 256) + 6$. To put pin #2 and pin #7 together, the value will be $(2 * 256) + 7$. - a15 - keypress combination #2 - a16 - keypress combination #3 - a17 - keypress combination #4 - a18 – keypress length (milliseconds) - a19 – keypress space (milliseconds) - a20 – product delivery timeout (seconds). If the product is not successfully dispensed within the configured time, the sale is considered a failure (the cash credit remains on the machine or, the cashless credit is refunded).

4. Machine is in idle status (active)

Message	Description
- EXENOINHIBIT - EXEVMCACTIVE(1)	- Any of this messages are indicating that the machine is ready to use.

5. Machine is in out order status (inactive)

Message	Description
- EXEINHIBIT - EXEVMCACTIVE(0)	- Any of this messages are indicating that the machine is ready to use.

6. The interface successfully sent the credit

Message	Description
EXECREDITOK	- The last credit send succeeded

7. The interface failed sending credit to the machine

Message	Description
EXECREDITFAIL	- The last credit sens command failed, interface will automatically retry sending it.

8. Last dispensing action success

Message	Description
EXEVENDSUCCESS(a1)	- The product was successfully dispensed - a1 – N/A for manual keypress, product ID for automatic external work flow.

9. Last dispensing action failed

Message	Description
EXEVENDFAIL(a1)	- The product dispensing failed -a1 – product ID (selection number)

10. Last vend approval success

Message	Description
EXEVENDAPPOK	- Vend approval received by the machine

11. Last vend approval failure

Message	Description
EXEVENDAPPFAIL	- Vend approval reception failed by the machine

12. RFID reader SN received

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

13. RFID card removed

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

14. RFID card reading failed

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

E. Display helper messages

These messages are coming from the interface during the automatic or semi-automatic external transactions, in order to help you adding in your app, the necessary messages to guide the customer and its actions.

1. Go to idle (main) screen

Message	Description
DISPLAYGOTOIDLE(a1)	- Go to idle (main) screen. - a1 – the total available credit

2. Coin change is not available at this point

Message	Description
DISPLAYNOCHANGE	- change mode is not available at this point.

3. Wait for the coin change to be dispensed

Message	Description
DISPLAYWAITCHANGE	- change returning in progress, wait for finish.

4. Coin change process finished

Message	Description
DISPLAYCHANGEFINISHED	- change returning finished

5. Sending credit to the machine

Message	Description
DISPLAYEXESENDCREDIT(a1)	- Sending credit to the vending machine - a1 = the credit value sent to the machine

6. Sending a keypress to the machine

Message	Description
DISPLAYSENDKEYPRESS	- Sending a keypress to the machine

7. Sending dispensing approval to the machine

Message	Description
DISPLAYSENDVENDAPP	- Sending dispensing approval to the machine

8. Product successfully dispensed

Message	Description
DISPLAYVNDSUCC(a1, ..., a6)	- Product successfully dispensed - a1 – product price (in cents) – 220 = 2.20EUR - a2 – product ID - a3 – product type – always 7 for this device - a4 – product quantity - a5 – N/A - a6 – payment result - 1 – cash OK - 2 – cash failed - 3 – cashless #1 OK - 4 – cashless #1 failed - 5 – cashless #2 OK - 6 – cashless #2 failed - 7 – N/A - 8 – N/A - 9 – free vend

9. Product dispensing failed

Message	Description
DISPLAYVNDFAIL	- Product dispensing failed

10. Card payment failure

Message	Description
DISPLAYCSLSPAYFAIL	- Card payment not authorized

11. Card payment authorized

Message	Description
DISPLAYCSLSPAYSUCCESS	- Card payment authorized

12. Cash payment failure

Message	Description
DISPLAYCASHPAYFAIL	- Cash payment not authorized

13. Cash payment authorized

Message	Description
DISPLAYCASHPAYSUCCESS	- Cash payment authorized

14. Display a question about payment method

Message	Description
DISPLAYHOWDOYOU PAY	- Asking the customer to select the payment method

15. The product is not available

Message	Description
DISPLAYPRODUCTNOTAVAILABLE	- The product is not available for some reason (no stock, error on last dispense activity, etc.)

16. Information to insert cash

Message	Description
DISPLAYINSERTCASH	- Asking the customer to insert cash in order to perform a payment.

17. Information to tap credit card

Message	Description
DISPLAYPRESENTCARD	- Asking the customer to tap the credit card

18. Current credit

Message	Description
DISPLAYCREDIT(a1, a2)	- Display the latest updated credit - a1 – current total credit - a2 – the amount needs to be inserted to complete the cash payment

19. Received free vend command

Message	Description
DISPLAYFREEVENDSUCCESS	- Received free vend command

20. Display last read RFID subscription available amount (only if connected to our SaaS backend)

Message	Description
DISPLAYSUBSOLD(a1)	- a1 is the available credit for last tapped subscription tag

Notes: