

GalaxyVend self service universal controller

v1.008

2026-02-10

Table of Contents

I. Introduction.....	7
II. Hardware.....	8
A. Board overview.....	8
B. Core board connectors and buttons description.....	8
C. Self service expansion board.....	9
D. Self service expansion board connectors description.....	9
F. Transaction flows.....	11
1. General transaction flow.....	11
2. Transaction flow specific for access control and timer application.....	11
III. Configuration software.....	12
A. Main screen.....	12
B. Self service mode action selection screen.....	12
C. Vending machine parameters configuration.....	13
D. Products configuration.....	14
1. Disabled product.....	14
2. Simple tokens dispenser.....	15
3. Subscription token dispensing.....	16
4. Internal relay timer.....	17
5. Subscription recharging product.....	18
6. Outer relay board.....	19
7. Ramp relay product (not available for this device).....	20
8. Key press product (not available for this device).....	21
9. Change return product.....	22
E. Stocks control.....	23
III. External control commands.....	24
A. Bill validator related commands.....	24
1. Check if the bill validator was initialized by the interface.....	24
2. Check if the bill validator was enabled.....	24
3. Get the bill validator configured bills values.....	25
4. Get the bill validator information.....	25
5. Get the bill validator settings.....	25
6. Reset bill validator.....	26
B. Coin related commands.....	27
1. Reset coin acceptor.....	27
2. Get total value of coins in tubes (for coin changers only).....	27
3. Dispense some coins (change) to the customer – obsolete, try to use COINAP command whenever the coin acceptor/changer supports it.....	28
4. Dispense some coins using MDB alternative payout method.....	28
5. Check if the coin acceptor/changer was initiali34. Select cash payment method.....	28
6. Check if the coin acceptor/changer was enabled.....	29
7. Get last 10 coin acceptor/changer codes.....	29
8. Get the coin acceptor/changer configured coins values.....	29
9. Get the coin acceptor/changer information.....	30
10. Get the coin acceptor/changer settings.....	30
11. Get the token values.....	30
12. Set the token values.....	31
C. Cashless related commands.....	32
1. Reset cashless device.....	32
2. Request cashless current revalue limit.....	32
3. Request approval for a vend.....	32

4. Confirm a success vend to the cashless device.....	33
5. Report a vend failure to the cashless device.....	33
6. Report a cash sale to the cashless device.....	33
7. Send a revalue request (the customer's account amount refill).....	34
8. Get last 10 cashless device codes.....	34
9. Check if the cashless device was initialized by the interface.....	34
10. Check if the cashless device was enabled.....	35
11. Get the cashless device information.....	35
12. Get the cashless device settings.....	35
D. System and transaction related commands.....	36
1. Interface reboot.....	36
2. Check if the interface is up and running.....	36
3. Read internal counters.....	36
4. Read internal counters.....	37
5. Requesting controller settings.....	38
6. Sending controller settings.....	39
7. Sending converter settings.....	39
8. Saving settings to non-volatile memory (persistent after reboot).....	40
9. Reset to factory default.....	40
10. Reading board serial number.....	40
11. Check current available credits.....	40
12. Clear all current credit variables.....	40
13. Clear cash credit only.....	41
14. Resetting all MDB peripherals (payment systems).....	41
15. Enable all cash MDB peripheral.....	41
15. Disable all cash MDB peripheral.....	41
16. Enable all cash peripherals for a specific amount and time.....	42
17. Check peripherals' status.....	42
18. File system format.....	43
19. Product stock update.....	43
20. Complete stock update.....	43
21. Save all current stocks to non-volatile memory.....	43
22. Read current stocks (obsolete, do not use it, use CURRENTSTOCK? instead).....	44
23. Select product to sale.....	44
24. Cancel product selection.....	44
25. Select cash payment method.....	44
26. Select cashless #1 payment method.....	45
27. Select free vend payment method.....	45
28. Reading product parameters.....	46
29. Writing product parameters.....	47
30. Requesting complete stock status.....	48
31. Set RFID reader password (key-A) – only with the optional (sold separately) RFID reader	48
32. Get RFID reader password (key-A) – only with the optional (sold separately) RFID reader	48
33. Enable the RFID reader, using the password you set before with RDRPASS().....	48
34. Get last read card serial number.....	49
35. Erase last read card serial number.....	49
IV. Unsolicited messages.....	50
A. Bill validator related messages.....	50
1. Bill validator just reset time exceeded.....	50
2. Bill validator setup time exceeded.....	50
3. Bill validator expansion ID time exceeded.....	50

4. Bill validator expansion ID with options time exceeded.....	50
5. Bill validator optional feature enable time exceeded.....	51
6. Bill validator with recycling support has been identified.....	51
7. Interface will try to enable the recycling support.....	51
8. Bill recycler setup time exceeded.....	51
9. Bill device has no support to recycler any known bills.....	51
10. Bill recycler enabling time exceeded.....	51
11. Bill recycler enabling failed.....	51
12. Bill recycler function successfully enabled.....	52
13. Bill recycler answered with a NAK on enable function.....	52
14. Bill recycler reading dispense status time exceeded.....	52
15. Bill recycler reading dispense status returned a NAK.....	52
16. Bill recycler reading dispense status returned an ACK.....	52
17. Bill recycler reading dispense status returned an ACK.....	52
18. Bill recycler remaining stock value.....	52
19. Bill dispensing command time exceed.....	53
20. Bill dispensing command time exceed.....	53
21. Bill dispensing command time exceed.....	53
22. Bill stacker status command time exceeded.....	53
23. Bill stacker status – stacker full.....	53
24. Bill stacker status – stacker not full.....	53
25. Bill is not ready.....	53
26. Bill validator/recycler failed to answer on poll command.....	54
27. Bill validator – one bill stacked.....	54
28. Bill in escrow position.....	54
29. Bill returned to customer.....	54
30. Bill received in recycler.....	54
31. A disabled bill was rejected.....	54
32. A bill was manually loaded to recycler.....	54
33. A disabled bill was manually dispensed from the recycler.....	55
34. A disabled bill was transferred from the recycler to cashbox.....	55
35. Bill validator is in normal condition.....	55
37. Bill validator have a defective motor.....	55
36. Bill validator have a defective sensor.....	55
37. Bill validator is busy.....	55
38. Bill validator ROM error.....	55
39. Bill validator jam.....	55
40. Bill validator was reset.....	55
41. Bill removed from bill validator.....	56
42. Bill validaor cashbox has been removed.....	56
43. Bill validaor has been disabled by your application or by an internal error.....	56
44. Bill validator has been rejected a bill.....	56
45. Bill removed after it was credited.....	56
46. A bill was inserted while the bill validator is deactivated.....	56
47. Recycler has received a change request.....	56
B. Cashless related unsolicited messages.....	57
1. Cash sale successfully reported to the cashless device.....	57
2. Cash sale reporting to the cashless device failed.....	57
3. Cashless device setup time exceeded.....	57
4. Cashless device max/min prices reporting time exceed.....	57
5. Cashless device poll time exceed.....	57
6. Cashless device expansion request ID time exceed.....	57
7. Cashless device expansion enable options time exceed.....	58

8. Cashless device have Always Idle support and it will be enabled.....	58
9. Cashless device writing date/time exceed.....	58
10. Cashless device sent a display message.....	58
11. Cashless device sent a BEGIN SESSION message.....	58
12. Cashless device sent a VEND APPROVED message.....	58
13. Cashless device sent a VEND DENIED message.....	59
14. Cashless device sent an END SESSION message.....	59
15. Cashless device sent a CANCELED message.....	59
16. Cashless device is ready.....	59
17. Cashless device returned a malfunction error.....	59
18. Cashless device returned COMMAND OUT OF SEQUENCE message.....	59
19. Cashless device sent a REVALUE APPROVED message.....	59
20. Cashless device sent a REVALUE DENIED message.....	59
21. Cashless device sent a REVALUE LIMIT message.....	60
22. Cashless device sent a DATE/TIME request message.....	60
23. Interface successfully sent date/time command to the cashless device.....	60
24. Interface failed sending date/time command to the cashless device.....	60
25. Interface successfully enabled the cashless device.....	60
26. Interface successfully enabled the cashless device.....	60
C. Coin related unsolicited messaged.....	61
1. Coin acceptor/changer just reset waiting time exceeded.....	61
2. Coin acceptor/changer setup time exceeded.....	61
3. Coin acceptor/changer expansion identification time exceeded.....	61
4. Coin acceptor/changer feature enable time exceeded.....	61
5. Coin acceptor/changer tube status time exceeded.....	61
6. Coin acceptor/changer is not ready for the issued command.....	62
7. Coin acceptor/changer tube status.....	62
8. Coin acceptor/changer poll time exceeded.....	62
9. Coin acceptor/changer is busy dispensing coins.....	62
10. Coin acceptor/changer temporarily unable to dispense coins.....	62
11. Coin acceptor/changer has failed to dispense all or some of the required coins.....	63
12. Coin acceptor/changer is reporting a manual coin dispense.....	63
13. Coin acceptor/changer received a token.....	63
14. Coin acceptor/changer received a coin.....	63
15. Coin acceptor/changer detected a slug.....	64
16. Coin acceptor/changer is in normal condition.....	64
17. Coin acceptor/changer received a change request.....	64
18. Coin acceptor/changer received a coin that was not credited.....	64
19. Coin acceptor/changer has a defective tube sensor.....	64
20. Coin acceptor/changer detected a double arrival.....	64
21. Coin changer detected an acceptor disconnection.....	64
22. Coin acceptor/changer detected a tube jam.....	65
23. Coin acceptor/changer detected an internal ROM error.....	65
24. Coin acceptor/changer detected a routing error.....	65
25. Coin acceptor/changer detected reset condition.....	65
26. Coin acceptor/changer detected a coin jam.....	65
27. Coin acceptor/changer detected the removal of a credited coin.....	65
D. System and reader related messages.....	66
1. RFID reader SN received.....	66
2. RFID card removed.....	66
3. RFID card reading failed.....	66
E. Display helper messages.....	67
1. Go to idle (main) screen.....	67

2. Coin change is not available at this point.....	67
3. Wait for the coin change to be dispensed.....	67
4. Coin change process finished.....	67
5. Sending a keypress to the machine.....	67
6. Sending dispensing approval to the machine.....	67
7. Product successfully dispensed.....	68
8. Product dispensing failed.....	68
9. Card payment failure.....	68
10. Card payment authorized.....	68
11. Cash payment failure.....	68
12. Cash payment authorized.....	68
13. Display a question about payment method.....	68
14. The product is not available.....	69
15. Information to insert cash.....	69
16. Information to tap credit card.....	69
17. Current credit.....	69
18. Received free vend command.....	69
19. Display last readed RFID subscription available amount (only if connected to our SaaS backend).....	70

I. Introduction

This device was designed to work as a self service ending machine controller. It can be used to build vending machines with one or more of the following functions:

- token dispenser/change machine;
- RFID subscription tokens dispenser;
- RFID subscription tokens recharging (only the online version, and only with the optional RFID reader – sold separately);
- timer with small relay output (can be also used for access control – connected to a turnstile, for example);
- access control with up to 8 outputs (with the optional external relay board attached).

The device, also have communication functions, to connect to our SaaS backend, if you want to develop a remote control and reporting application (JSON messages over MQTT websocket).

The device can be also connected with an Android tablet over the USB and used with our Android application to add touchscreen and graphics functions. The device can also work simultaneously works with touchscreen and physical buttons. Adding an Android tablet, with the proper development, allows usage of all 64 products that can be configured.

Standard package content:

- GalaxyVend self service universal controller;
- cable for buttons.

NOTE: this version is only working with 8 products (either in the physical buttons or Android touch configuration).

If you want to use all 64 available products, you need to develop your own external application or contact us for development.

II. Hardware

A. Board overview

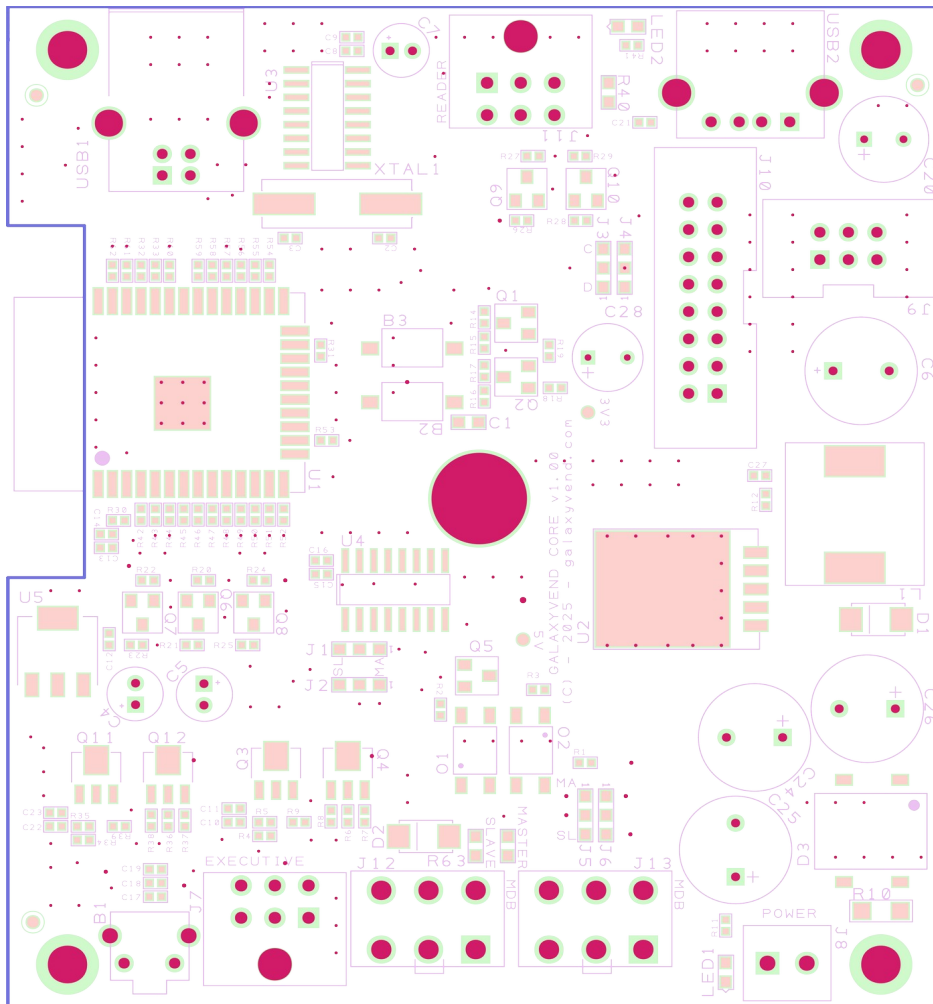


Figure 1: Core board

B. Core board connectors and buttons description

1. **B1** – multi purpose button – it sends the message “BTTNSMALLPRESSED” when pressed and “BTTNSMAPPRELEASED” when released. Your application can use this button to activate some tasks (for example, enter application service mode when pressed for 3 seconds).
2. **J7** – **Executive connector** – not used for this device.
3. **J12 and J13** – **MDB connectors** – connectors to attach MDB peripherals (payment systems). You can use any of the connectors (or both, if some of your peripherals do not have an “Y” cable to cascade them).
4. **J8** – **POWER INPUT** – power connector. For 12VDC or 24VDC (depending on your peripherals specification – MDB and ccTalk – for ccTalk there is an optional power connector on the expansion board).
5. **USB1** – USB-B connector for firmware update and for your app USB communication.
6. **J11** – **RFID reader** – connector to attach the optional (sold separately) RFID (MIFARE Classic) reader.
7. **USB2** – Connector to power external 5VDC devices (max. 1.2A, do not exceed this current)

C. Self service expansion board

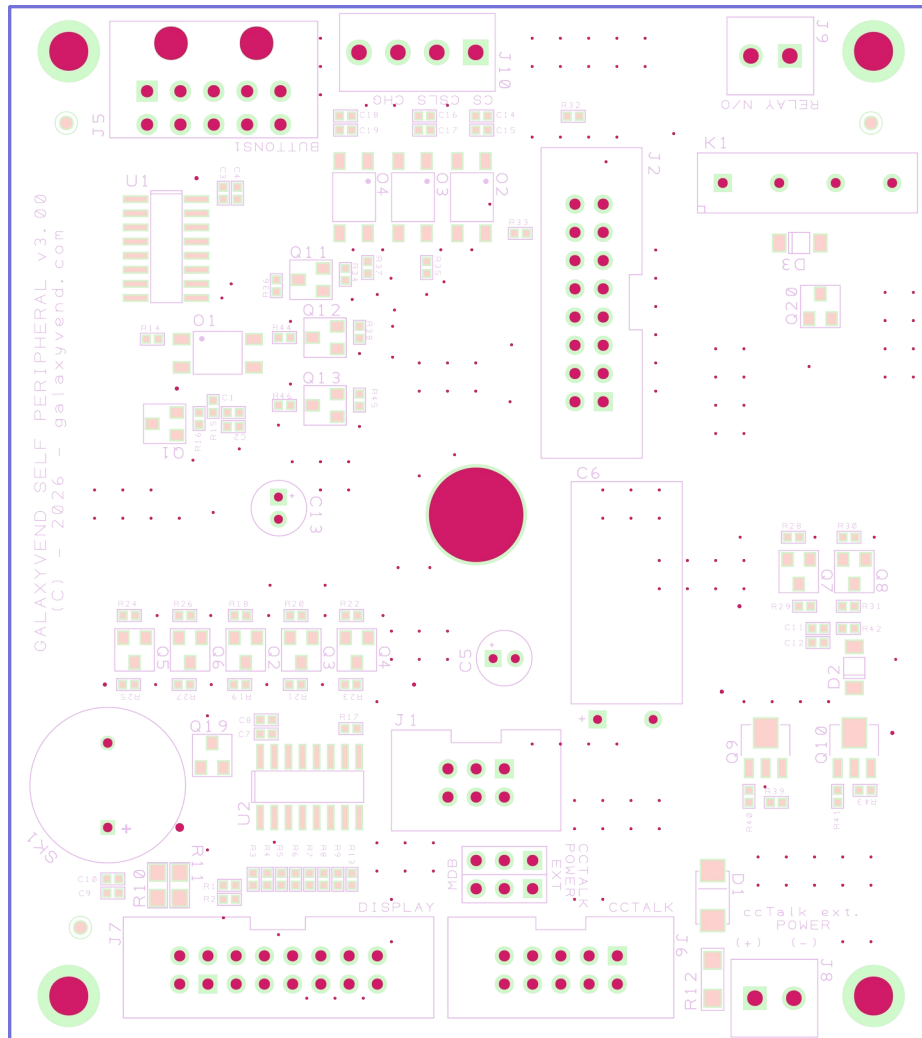


Figure 2: Self service expansion board

D. Self service expansion board connectors description

1. J5 – buttons connector – here you can connect up to 8 physical buttons (SPST – N/O) to select the product. Buttons are connected between each button pin and GND, active low.

- pin #1 – (squared pin) – button #1
- pin #2 – button #2
- pin #3 – button #3
- pin #4 – button #4
- pin #5 – button #5
- pin #6 – button #6
- pin #7 – button #7
- pin #8 – button #8
- pin #9 and pin #10 - GND

2. J10 – inputs for cash, cashless and change buttons (SPST - N/O)

- pin #1 – (squared pin) – cash payment method selection button
- pin #2 – cashless payment method selection button
- pin #3 – change button
- pin #4 – GND

3. J9 – relay output – dry contacts relay, maximum 100mA/100V – use external relay for higher current requirements.

4. J7 – alphanumerical display connector

5. J6 – ccTalk hoppers connector

- pin #1 – (squared pin) – ccTalk data
- pin #2 – N/C
- pin #3 – N/C
- pin #4 – GND
- pin #5 – N/C
- pin #6 – N/C
- pin #7 – ccTalk power
- pin #8 – GND
- pin #9 – N/C
- pin #10 – ccTalk power

6. external ccTalk power – used to power ccTalk hoppers if the main power is not suitable for the hoppers (for example, if you apply 24VDC to the main board to power the MDB peripherals, but the hoppers only accept 12VDC). Connection an external power supply for ccTalk requires ccTalk power selector jumpers to be moved to the “ccTalk power ext” position. The devices comes configured to power the ccTalk devices from the core board power input (ccTalk power selector jumpers on MDB position).

F. Transaction flows

This controller has its own business logic and can be externally controlled only for configuration and few actions (like selecting a product, selecting payment method, etc.)

1. General transaction flow

1. The customer is selecting the product (using the physical keys or the touchscreen – if the optional Android tablet is connected).
2. The controller ask about the payment method, if the product is available (no previous errors and there is enough stock).
3. The customer selects the desired payment method (physical keys or touchscreen) – cash or card.
4. The corresponding payment systems are activated.
5. The customer inserts the money or taps the card.
6. If the payment succeeded, the product is dispensed.
7. If the payment failed, there will be a message that informs the customer about this.
8. If the product dispensing failed, there will also be a message that informs the customer about this. The controller automatically inhibits all products requiring the peripherals used for product delivery. For example, if a product requires the hopper with address #3, all products requiring hopper #3 for delivery, will be disabled and a message will be displayed every time a customer selects a disabled product.

2. Transaction flow specific for access control and timer application

This particular flow applies if the first product is set as an internal relay product for applications like:

- access control – turnstiles, doors, etc.;
- kiddie rides timer;
- self service timer;
- other access control or self service systems requiring a payment in order to activate an electric circuit for a configurable amount of time.

In this case, selecting the product is no longer necessary, the transaction flow starts directly with the payment method selection. You only need to connect 2 buttons – one for cash payments and one for card payments. Optionally, you can connect a third button, in order to trigger the change giver after transactions.

1. The customer selects the desired payment method (physical keys or touchscreen) – cash or card.
2. The corresponding payment systems are activated.
3. The customer inserts the money or taps the card.
4. If the payment succeeded, the product is dispensed.
5. If the payment failed, there will be a message that informs the customer about this.
6. If the product dispensing failed, there will also be a message that informs the customer about this. The controller automatically inhibits all products requiring the peripherals used for product delivery. For example, if a product requires the hopper with address #3, all products requiring hopper #3 for delivery, will be disabled and a message will be displayed every time a customer selects a disabled product.

II. Configuration software

We are providing an Android application to connect and change controller's working parameters (prices, product prices, quantities, etc.)

It requires an Android device with OTG function and an USB-B cable matching your OTG connector type. Connect the cable between the device and the Android device before starting the application.

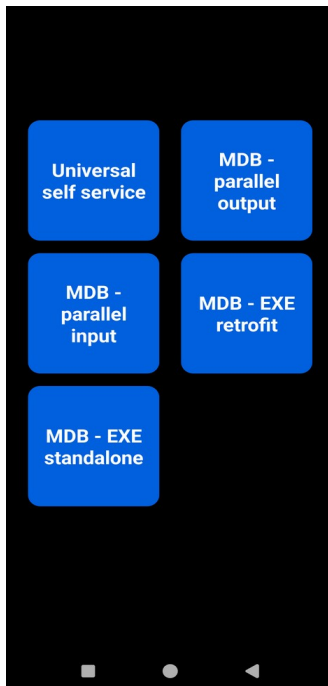


Figure 3: Main screen

A. Main screen

This application is used for several product in our portfolio so, in order to manage this product, in the main screen you need to select “Universal self service”.

This will load the self service mode action selection screen.

Do not use other option, since the products are slightly different and configure a device using the wrong device type, may lead to unexpected result or it may even brick the device and require it to be send back to us to unlock.

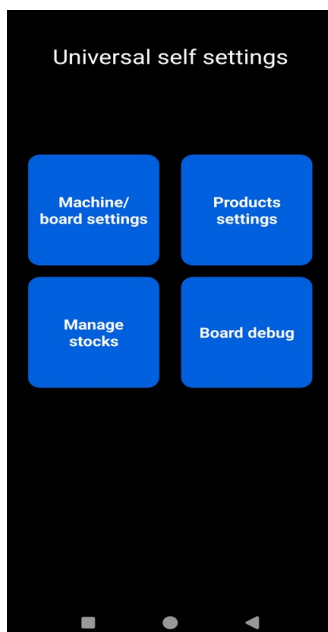


Figure 4: Self service mode action selection screen

B. Self service mode action selection screen

- “**Machine settings**” button selects the vending machine parameters configuration screen
- “**Products settings**” button selects the products parameters configuration screen
- “**Manage stock**” button selects the stocks management screen
- “**Board debug**” function is not available in this version

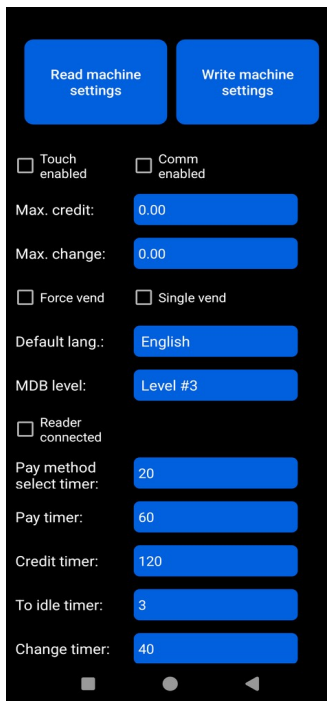


Figure 5: Vending machine configuration screen

C. Vending machine parameters configuration

To configure vending machine parameters, start by pressing “Read machine settings” button and wait until all settings are received. All fields will be filled with the current configuration.

Fields description:

- **“Touch enabled”** - not used for this device, changing has no effect, the machine will be able to use both mechanical buttons and touchscreen (if the optional Android tablet is attached) simultaneously.

- **“Comm enabled”** – if checked, the device will use the saved communication configuration to connect to our SaaS backend, where you will be able to connect via MQTT (websocket with JSON messages – optional subscription fee required). Do not enable this function unless you have a subscription and you are sure the WiFi and backend parameters are correctly set. Otherwise, the device may act erratically.

- **“Max. credit”** – the maximum credit machine accepts for cash payments.

- **“Max. change”** - the maximum allowed change value (after transaction)

- **“Force vend”** – if enabled, it forces a success or failed transaction in order to return change to the customer. This way, you avoid machine usage as a change machine that will very fast empty the coin changer.

- **“Single vend”** – not used with this device, it will always work as a multi-vend vending machine.

- **“Default lang.”** - selects the vending machine messages language. The Android configuring application is only available in English. If you need to add a specific language messages, you can contact us.

- **“MDB level”**- selects the MDB level configuration for cashless devices. For the moment, the device is only supporting Level 3 with Always Idle mechanism active.

- **“Reader connected”** – if enabled, the controller will try to initialize and activate the attached RFID reader (sold separately).

- **“Pay method select timer”** – is the time (in seconds) from the product selection, until the controller returns to the main screen (idle screen), if the customer is not selecting a payment method.

- **“Pay timer”** – is the time (in seconds) that the controller waits the credit accumulates, enough to cover the selected product price. This timer is updated every time an event occurs on the cash payment systems (for example, it is extended with another X seconds when a coin or a bill are accepted or rejected – if there are 10 seconds remaining and a coin arrives, the time is extended with another “pay timer” value).

- **“Credit timer”** – is the time (in seconds) until the remaining credit (after a transaction) expires if it is not used.

- **“To idle timer”** – the time (in seconds), until the controller returns to the main screen (idle screen) after a transaction.

- **“Change timer”** - the time (in seconds) the controller waits for a success change return task. If this time exceeds and the coin changer did not finished returning all the credit, the change task is considered as a failure and the credit remains on the screen.

D. Products configuration

The application only enables the necessary field for the selected product type.

In order to read a product current configuration, you need to select the product number in the top combo and wait for the information to be retrieved from the controller. Depending on the product type, the product configuration fields will be enabled/disabled and then filled with the current values.

After modifying a product, you need to press the “Write product settings” in order to send the modified values to the controller. After you finish modification to all the products you need to modify, you need to press “Save all products to flash” button, in order to save them to the non-volatile memory. Do not press this button after each product modification, since after saving to non-volatile memory, the controller reboots. You need to only press once, when all your products’ modifications are finished. **You may need to scroll to reach the end of parameters list and the buttons.**

Product #1: [None]

Product type: Disabled #0

Product default stock: 0

Product current stock: 0

Product price: 0.00

Quantity #1: 1

Quantity #2: 0

Hopper #1 address: Address #3

Hopper #2 address: Address #3

Hopper #1 mode: Auto #0

Hopper #2 mode: Auto #0

Relay board address: None

Relay number: None

Main relay/Keypress 1: None

Extra relay/Keypress 2: None

Extra relay/Keypress 3: None

Extra relay/Keypress 4: None

Keypress duration (ms): 500

Keypress space (ms): 1000

Delivery timeout (sec): 30

Write product settings

Save all products to flash

Figure 6: Disabled product configuration

1. Disabled product

Select “Disabled #0” to disable a specific product. All fields will be disabled. It does not matter if some fields are keeping some values, the product will be disabled and selection it in the future will display the product not available message.

The screenshot shows a configuration screen for a 'Tokens #1' product. The interface is organized into two main columns. The left column contains fields for 'Product #1', 'Product type' (set to 'Tokens #1'), 'Product default stock' (0), 'Product current stock' (0), 'Product price' (0.00), 'Quantity #1' (1), 'Quantity #2' (0), 'Hopper #1 address' (Address #3), 'Hopper #2 address' (Address #3), 'Hopper #1 mode' (Auto #0), 'Hopper #2 mode' (Auto #0), 'Relay board address' (None), 'Relay number' (None), and 'Main relay/Keypress 1' (None). The right column contains fields for 'Extra relay/Keypress 2' (None), 'Extra relay/Keypress 3' (None), 'Extra relay/Keypress 4' (None), 'Keypress duration (ms)' (500), 'Keypress space (ms)' (1000), and 'Delivery timeout (sec)' (30). At the bottom right, there are two buttons: 'Write product settings' and 'Save all products to flash'.

Figure 7: Tokens dispensing product

2. Simple tokens dispenser

Select “Tokens #1” in order to define the product as a tokens dispensing product. It is possible to set this product type to dispense from one or two hoppers (for example, 3 tokens from hopper with address #3 and 1 token from hopper with address #4).

- **“Product default stock”** – represents the number of products that are present in the machine when the machine is fully loaded. If you set this value to 0, then an infinite number of product can be sold (recommended for this product type). The dispensing will be disabled when the hopper is empty, anyway.
- **“Product price”** – is the price you want to receive for the product.
- **“Quantity 1”** - is the number of tokens you want to dispense from the hopper #1. If the quantity is set to 0, then it will not try to dispense from hopper #1.
- **“Quantity 2”** - is the number of tokens you want to dispense from the hopper #2. If the quantity is set to 0, then it will not try to dispense from hopper

#2.

- **“Hopper #1 address”** – is the hopper address where the “Quantity 1” will be delivered from
- **“Hopper #2 address”** – is the hopper address where the “Quantity 2” will be delivered from.
- **“Hopper #1 mode”** – is the hopper working mode
- **“Hopper #2 mode”** – is the hopper working mode

Having the option to set different working modes, allows you to use mixed hopper models or even manufacturers.

- **“Delivery timeout”** – is the time that controller waits for complete tokens dispensing. If the dispensing is not finished during this time, the product delivered is considered a failure and the credit is kept for another transaction. Do not set this timeout too short!

Product #1

Product type: Subscription tag #2

Product default stock: 0

Product current stock: 0

Product price: 0.00

Quantity #1: 1

Quantity #2: 0

Hopper #1 address: Address #3

Hopper #2 address: Address #3

Hopper #1 mode: Auto #0

Hopper #2 mode: Auto #0

Relay board address: None

Relay number: None

Main relay/Keypress 1: None

Extra relay/Keypress 2: None

Extra relay/Keypress 3: None

Extra relay/Keypress 4: None

Keypress duration (ms): 500

Keypress space (ms): 1000

Delivery timeout (sec): 30

Write product settings

Save all products to flash

Figure 8: Subscription token dispensing product

3. Subscription token dispensing

Select “Subscription tag #2” in order to define the product as a subscription token dispensing product. It is possible to set this product type to dispense from one hopper.

- **“Product default stock”** – represents the number of products that are present in the machine when the machine is fully loaded. If you set this value to 0, then an infinite number of product can be sold (recommended for this product type). The dispensing will be disabled when the hopper is empty, anyway.

- **“Product price”** – is the price you want to receive for the product.

- **“Quantity 1”** - is the number of subscription tokens you want to dispense from the hopper #1 (usually 1). If the quantity is set to 0, then it will not try to dispense from hopper #1.

- **“Hopper #1 address”** – is the hopper address where the “Quantity 1” will be delivered from

- **“Hopper #1 mode”** – is the hopper working mode

- **“Delivery timeout”** – is the time that controller waits for complete tokens dispensing. If the dispensing is not finished during this time, the product delivered is considered a failure and the credit is kept for another transaction. Do not set this timeout too short!

Product #1: [Product #1]

Product type: Inner board relay #3

Product default stock: 0

Product current stock: 0

Product price: 0.00

Quantity #1: 1

Quantity #2: 0

Hopper #1 address: Address #3

Hopper #2 address: Address #3

Hopper #1 mode: Auto #0

Hopper #2 mode: Auto #0

Relay board address: None

Relay number: None

Main relay/Keypress 1: None

Extra relay/Keypress 2: None

Extra relay/Keypress 3: None

Extra relay/Keypress 4: None

Keypress duration (ms): 500

Keypress space (ms): 1000

Delivery timeout (sec): 30

Write product settings

Save all products to flash

Figure 9: Board relay timer product

4. Internal relay timer

Select “Inner board relay #3” in order to define the product as timer

- **“Product default stock”** – represents the number of products that are present in the machine when the machine is fully loaded. If you set this value to 0, then an infinite number of product can be sold (recommended for this product type). The dispensing will be disabled when the hopper is empty, anyway.

- **“Product price”** – is the price you want to receive for the product.

- **“Quantity 1”** - is the number of seconds that the relay remains energized. A countdown timer will be displayed until it reaches 0.

- **“Delivery timeout”** – it should be higher than the Quantity 1

Product #1: Subscription recharge #4

Product type: Subscription recharge #4

Product default stock: 0

Product current stock: 0

Product price: 0.00

Quantity #1: 1

Quantity #2: 0

Hopper #1 address: Address #3

Hopper #2 address: Address #3

Hopper #1 mode: Auto #0

Hopper #2 mode: Auto #0

Relay board address: None

Relay number: None

Main relay/Kevoress 1: None

Extra relay/Kevoress 2: None

Extra relay/Kevoress 3: None

Extra relay/Kevoress 4: None

Keypress duration (ms): 500

Keypress space (ms): 1000

Delivery timeout (sec): 30

Write product settings

Save all products to flash

Figure 10: Tokens dispensing product

5. Subscription recharging product

Select “Subscription recharge #1” in order to define the product as a recharging. It allows to add credit on customer’s account. Requires WiFi connection, the optional (sold separately) RFID reader and a subscription to our SaaS backed.

- **“Product price”** – is the price you want to receive for the product.

- **“Quantity 1”** – is the value that should be charged to customer’s account. You may offer some bonuses (for example, for a 5.00EUR price, you can recharge with 5.50EUR)

- **“Delivery timeout”** – is the time that controller waits for complete recharging success response from the SaaS backend. If the recharge is not finished during this time, the product delivered is considered a failure and the credit is kept for another transaction. Do not set this timeout too short to allow the customer to tap the tag to the reader and to allow the server to return a response.

Figure 11: Outer relay board

6. Outer relay board

Select “Outer board relay” in order to define the product as an output relays board controller.

- **“Product default stock”** – represents the number of products that are present in the machine when the machine is fully loaded. If you set this value to 0, then an infinite number of product can be sold (recommended for this product type). The dispensing will be disabled when the hopper is empty, anyway.

- **“Product price”** – is the price you want to receive for the product.

- **“Quantity 1”** - is the number of seconds that the relay remains energized in order to activate the external equipment

- **“Relay board address”** – is board’s address (1-8).

- **“Relay number”** – is the relay that will be energized for this product

- **“Delivery timeout”** – the timeout for the external relay board to respond.

Figure 12: Ramp relay product

7. Ramp relay product (not available for this device)

This product is only available for GalaxyVend MDB-master converter, with the additional relay board installed converter in external mode, with our ramp self service application on an Android tablet.
WARNING! Do not select this product type for this !!!

One or more (up to 4) relays can be energized for each product. This allows to control multiple equipments for one product, for example pumps and valves according to controlled system's needs.

- **"Product price"** – is the price you want to receive for the product.
- **"Quantity 1"** - is the number of seconds that the relay remains energized in order to activate the external equipment.
- **"Main relay/keypress 1"**- is the relay that will be always energized for this product.
- **"Extra relay/keypress 2"** – is second relay that will be energized together with the main relay (if needed)
- **"Extra relay/keypress 3"** – is third relay that will

be energized together with the main relay (if needed)

- **"Extra relay/keypress 4"** – is fourth relay that will be energized together with the main relay (if needed)
- **"Delivery timeout"** – the timeout for the external relay board to respond.

Product #1

Product type: Key press #7

Product default stock: 0

Product current stock: 0

Product price: 0.00

Quantity #1: 1

Quantity #2: 0

Hopper #1 address: Address #3

Hopper #2 address: Address #3

Hopper #1 mode: Auto #0

Hopper #2 mode: Auto #0

Relay board address: None

Relay number: None

Main relay/Keypress 1: None

Extra relay/Keypress 2: None

Extra relay/Keypress 3: None

Extra relay/Keypress 4: None

Keypress duration (ms): 500

Keypress space (ms): 1000

Delivery timeout (sec): 30

Write product settings

Save all products to flash

Figure 13: Key press product

8. Key press product (not available for this device)

This product is only available for GalaxyVend MDB-Executive converter (with the additional relay board installed), in external mode, with our retrofit **WARNING! Do not select this product type for this device!!!**

The device can sequentially emulate key press for up to

Select “Key press #7” for this product

- “**Product default stock**” – represents the number of products that are present in the machine when the machine is fully loaded. If you set this value to 0, then an infinite number of product can be sold (recommended for this product type). The dispensing will be disabled when the hopper is empty, anyway.

- “**Product price**” – is the price you want to receive for the product.

- “**Quantity 1**” – is the number of products that will be dispensed when this product is selected and paid (for internal stock purposes, product disable when no stock available, etc.).

- “**Main relay/keypress 1**”- represents the pins that will be tied together for the first key press
- “**Extra relay/keypress 2**” – represents the pins that will be tied together for the second key press
- “**Extra relay/keypress 3**” – represents the pins that will be tied together for the third key press
- “**Extra relay/keypress 4**” – represents the pins that will be tied together for the fourth key press
- “**Keypress duration**” – represents the time the 2 pins will be tied together for a key press emulation
- “**Keypress space**” – represents the time between two successive key presses.
- “**Delivery timeout**” – the timeout for the external relay board to respond.

The screenshot shows a configuration screen for a product type 'Change giving #8'. The interface is divided into several sections with various input fields and buttons.

Field	Value
Product #1	Change giving #8
Product type	Change giving #8
Product default stock	0
Product current stock	0
Product price	0.00
Quantity #1	1
Quantity #2	0
Hopper #1 address	Address #3
Hopper #2 address	Address #3
Hopper #1 mode	Auto #0
Hopper #2 mode	Auto #0
Relay board address	None
Relay number	None
Main relay/Kevoress 1	None
Extra relay/Kevoress 2	None
Extra relay/Kevoress 3	None
Extra relay/Kevoress 4	None
Keypress duration (ms)	500
Keypress space (ms)	1000
Delivery timeout (sec)	30

Buttons at the bottom right: Write product settings, Save all products to flash.

Figure 14: Change giving

9. Change return product

This product should always be programmed for product ID 9 (or less than 9 if you want to use other product button and you are not installing the physical change button), since the change button has a fixed product ID 9.

There are no configurable parameters for this product, just set it as “Change giving #8” and make sure you have an attached coin changer. The change timeout is configured in “Machine settings” section.

E. Stocks control

Stock #1	0	Stock #57	0
Stock #2	0	Stock #58	0
Stock #3	0	Stock #59	0
Stock #4	0	Stock #60	0
Stock #5	0	Stock #61	0
Stock #6	0	Stock #62	0
Stock #7	0	Stock #63	0
Stock #8	0	Stock #64	0
Stock #9	0		
Stock #10	0		
Stock #11	0		
Stock #12	0		
Stock #13	0		
Stock #14	0		
Stock #15	0		
Stock #16	0		
Stock #17	0		
Stock #18	0		

Read stocks Write stocks

Full refill

Figure 15: Current stock update

This screen allows you to set current stocks after each machine refilling.

You have the possibility to send a complete refill (button “Full refill”) that will set the current stock for each product to the value of default stock for that product or partial refill product by product (button “Write stocks”).

To complete a partial machine refill, you need to scroll to the end of the screen first, press “Read stocks” button and wait for a complete stocks reading.

After that, you set the the current stock value for any product you need.

At the end, press “Write stocks” and wait for finish.

III. External control commands

These commands are used in order to handle controller from an external device, over USB or over MQTT (WiFi required, connection through our SaaS backend, monthly fee based service).

This working mode can be used to add external control. Usually, it is not necessary to use low level commands such as BILLENABLE or COINDISABLE. You may want to use GETCASH() function instead, that will automatically handle cash payment, with a timeout.

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. You 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 BILLENABLE 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)	- 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.
- BILLNOTINITED	- 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)	- 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)
- BILLNOTINITED	- 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 vendign 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 initialized34. Select cash payment method

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. Select cash payment method 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. - Not available

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 executed.

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 it's own display; - b3 – if set, the cashless device is supporting cash sale reporting - The bill cashless device was not previously initialized and the information is not available

D. System and transaction 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 – total number of dispensed tokens - hopper addr #3 - a2 - total number of tokens dispensed – hopper addr #4 - a3 – total number of subscription tokens dispensed – hopper addr #5 - a4 – total value of subscription recharging - a5 – total number of products sold with internal relay - a6 – total value of products sold with internal relay - 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 fo 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. 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 (in cents: 12800 = 128.00EUR) - a9 – total number of rejected bills - a10 – total number of received coins - a11 – total value of received coins (in cents: 500 = 50.00EUR) - a12 – total number of rejected coin - a13 – total number of cashless #1 transactions - a14 – total value of cashless #1 transactions (in cents) - a15 – total number of cashless #2 transactions - a16 – total value of cashless #2 transactions (in cents) - a17 – total number of received tokens - a18 – total value of received tokens (in cents) - a19 – total value of returned change

5. Requesting controller 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's considered that the change operation failed.

6. Sending controller 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's 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 - VMCSETTINGSFAIL	- command correctly executed. - command execution failed (parameters error).

7. Sending converter settings

Command	
- COUNTERSRESET	This command is resetting all internal counters. The device automatically reboots after issuing this command.
Possible answers	
- COUNTERSRESETOK	- command correctly executed.

8. 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.

9. 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 it's 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.

10. Reading board serial number

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

11. 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

12. Clear all current credit variables

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

13. Clear cash credit only

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

14. 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

15. Enable all cash MDB peripheral

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

15. Disable all cash MDB peripheral

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

16. 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.

17. 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, bil 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

18. 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.

19. 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

20. 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

21. 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 - SAVESTOCKSFAIL	- Command received, successfully executed - Command execution failed, you may need to use FSFORMAT and reconfigure your device.

22. 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.)

23. 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.

24. Cancel product selection

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

25. 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.

26. 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.

27. 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.

28. 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.

29. 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.

30. 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.

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

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

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

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

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

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

34. Get last read card serial number

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

35. Erase last read card serial number

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

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 command time exceed

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

21. Bill dispensing command time exceed

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

22. Bill stacker status command time exceeded

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

23. Bill stacker status – stacker full

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

24. Bill stacker status – stacker not full

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

25. Bill is not ready

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

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

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

27. Bill validator – one bill stacked

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

28. Bill in escrow position

Message	Description
BILLESCROW(X) – 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 it's flow, maximum credit, etc

29. Bill returned to customer

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

30. Bill received in recycler

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

31. A disabled bill was rejected

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

32. A bill was manually loaded to recycler

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

33. A disabled bill was manually dispensed from the recycler

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

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

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

35. Bill validator is in normal condition

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

37. Bill validator have a defective motor

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

36. Bill validator have a defective sensor

Message	Description
BILLSENSORFAIL	- Bill validator encountered one of it's 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 validaor cashbox has been removed

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

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

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

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 exceed

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 exceed

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 exceed

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 exceed

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, it's value depends on the cashless device and you can find more information in it's 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 it's 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 it's front panel). - A is the scaled coin type value - B is the total number of manually dispensed coins - C is the total number of coins remaining in tubes for the A type value

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. System and reader related messages

11. RFID reader SN received

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

2. RFID card removed

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

3. RFID card reading failed

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

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 a keypress to the machine

Message	Description
DISPLAYSENDKEYPRESS	- Sending a keypress to the machine

6. Sending dispensing approval to the machine

Message	Description
DISPLAYSENDVENDAPP	- Sending dispensing approval to the machine

7. 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 - 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

8. Product dispensing failed

Message	Description
DISPLAYVNDFAIL	- Product dispensing failed

9. Card payment failure

Message	Description
DISPLAYCSLSPAYFAIL	- Card payment not authorized

10. Card payment authorized

Message	Description
DISPLAYCSLSPAYSUCCESS	- Card payment authorized

11. Cash payment failure

Message	Description
DISPLAYCASHPAYFAIL	- Cash payment not authorized

12. Cash payment authorized

Message	Description
DISPLAYCASHPAYSUCCESS	- Cash payment authorized

13. Display a question about payment method

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

14. The product is not available

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

15. Information to insert cash

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

16. Information to tap credit card

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

17. 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

18. Received free vend command

Message	Description
DISPLAYFREEVENDSUCCESS	- Received free vend command

19. Display last readed 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