

**PICOVEND EZ MASTER
(MDB master and ccTalk to
RS232/USB interface)
v2023-12-05**

Table of Contents

I. Introduction.....	7
II. Hardware.....	8
A. Board overview.....	8
B. Connectors description.....	8
C. Communication interfaces.....	10
III. Communication protocol.....	11
A. MDB master related commands and answers.....	11
B. Bill validator/recycler related commands.....	11
1. Enable bill validator.....	11
2. Selective bills enable.....	11
3. Disable bill validator.....	12
4. Reset bill validator.....	12
5. Approve bill acceptance while a bill is in escrow position.....	13
6. Reject bill while a bill is in escrow position.....	13
7. Get last 10 bill status codes.....	13
8. Check if the bill validator was initialized by the interface.....	13
9. Check if the bill validator was enabled.....	14
10. Get the bill validator configured bills values.....	14
11. Get the bill validator information.....	14
12. Get the bill validator settings.....	15
13. Get the bill recycler bill type values.....	15
14. Get the bills set for recycling by the user application.....	15
15. Set the bills set for recycling by the user application.....	16
16. Get tot total value of bills available for recycling.....	16
17. Dispense bills as a change to customer.....	17
18. Dispense bills as a change to customer.....	17
19. Get current bill stacker status.....	17
.....	17
.....	17
C. Coin acceptor/changer related commands.....	18
1. Enable coin acceptor/changer.....	18
2. Selective coins enable.....	18
3. Disable coin acceptor.....	19
4. Reset coin acceptor.....	19
5. Get total value of coins in tubes (for coin changers only).....	19
6. Dispense some coins (change) to the customer – obsolete, try to use COINAP command whenever the coin acceptor/changer supports it.....	20
7. Dispense some coins using MDB alternative payout method.....	20
8. Check if the coin acceptor/changer was initialized by the interface.....	20
9. Check if the coin acceptor/changer was enabled.....	21
10. Get last 10 coin acceptor/changer codes.....	21
11. Get the coin acceptor/changer configured coins values.....	21
12. Get the coin acceptor/changer information.....	22
13. Get the coin acceptor/changer settings.....	22
14. Get the token values.....	23
15. Set the token values.....	23
16. Get coins quantity for each type (only for devices purchased after 2023-12-05).....	23
D. Cashless related commands.....	24
1. Reset cashless device.....	24
2. Enable cashless device.....	24

3. Disable cashless device.....	24
4. Cancel current cashless activity.....	25
5. Request cashless current revalue limit.....	25
6. Request approval for a vend.....	25
.....	25
7. Confirm a success vend to the cashless device.....	26
8. Report a vend failure to the cashless device.....	26
9. Report a cash sale to the cashless device.....	26
10. Send a revalue request (the customer's account amount refill).....	27
11. Get last 10 cashless device codes.....	27
12. Check if the cashless device was initialized by the interface.....	27
13. Check if the cashless device was enabled.....	28
14. Get the cashless device information.....	28
15. Get the cashless device settings.....	28
16. Cashless force session complete.....	29
17. Request approval for a negative vend (only for devices purchased after 2023-12-05).....	29
.....	29
E. Interface (VMC) system related commands.....	30
1. Get VMC settings.....	30
2. Set VMC settings.....	30
3. Set VMC manufacturer code.....	30
4. Set VMC internal serial number.....	31
5. Set VMC internal model number.....	31
6. Read the last error.....	31
7. Clear the last error.....	31
8. Reset interface internal counters.....	32
9. Interface reboot.....	32
10. Check if the interface is up and running.....	32
11. Read internal counters.....	32
12. Save settings.....	33
13. Load settings.....	33
14. Factory reset.....	33
15. POLL command (only for devices purchased after 2023-12-05).....	34
.....	34
F. Relay related commands.....	35
1. Individual relay control.....	35
2. Batch relay control.....	35
3. Check relay status.....	36
4. Relay reset.....	36
G. ccTalk hoppers related commands.....	37
1. Set ccTalk debug information mode over USB.....	37
2. Get ccTalk debug information mode over USB.....	37
3. Check if the hopper is enabled.....	37
4. Hopper dispense tokens/coins.....	38
5. ccTalk hopper init.....	38
6. ccTalk hopper level sensors reading.....	39
H. USB to MQTT bridge.....	40
IV. Unsolicited messages.....	41
1. Power-up messages.....	41
2. Bill validator just reset time exceeded.....	42
3. Bill validator setup time exceeded.....	42
4. Bill validator expansion ID time exceeded.....	42
5. Bill validator expansion ID with options time exceeded.....	42

6. Bill validator optional feature enable time exceeded.....	42
7. Bill validator with recycling support has been identified.....	43
8. Interface will try to enable the recycling support.....	43
9. Bill recycler setup time exceeded.....	43
10. Bill device has no support to recycler any known bills.....	43
11. Bill recycler enabling time exceeded.....	43
12. Bill recycler enabling failed.....	43
13. Bill recycler function successfully enabled.....	43
14. Bill recycler answered with a NAK on enable function.....	44
15. Bill recycler reading dispense status time exceeded.....	44
16. Bill recycler reading dispense status returned a NAK.....	44
17. Bill recycler reading dispense status returned an ACK.....	44
18. Bill recycler reading dispense status returned an ACK.....	44
19. Bill recycler remaining stock value.....	44
20. Bill dispensing command time exceed.....	45
21. Bill dispensing command time exceed.....	45
22. Bill dispensing command time exceed.....	45
23. Bill stacker status command time exceeded.....	45
24. Bill stacker status – stacker full.....	45
25. Bill stacker status – stacker not full.....	45
26. Bill is not ready.....	45
27. Bill validator/recycler failed to answer on poll command.....	46
28. Bill validator – one bill stacked.....	46
29. Bill in escrow position.....	46
30. Bill returned to customer.....	46
31. Bill received in recycler.....	46
32. A disabled bill was rejected.....	46
33. A bill was manually loaded to recycler.....	47
34. A disabled bill was manually dispensed from the recycler.....	47
35. A disabled bill was transferred from the recycler to cashbox.....	47
36. Bill validator is in normal condition.....	47
37. Bill validator have a defective motor.....	47
38. Bill validator have a defective sensor.....	47
39. Bill validator is busy.....	47
40. Bill validator ROM error.....	47
40. Bill validator jam.....	48
41. Bill validator was reset.....	48
42. Bill removed from bill validator.....	48
43. Bill validaor cashbox has been removed.....	48
44. Bill validaor has been disabled by your application or by an internal error.....	48
45. Bill validator has been rejected a bill.....	48
46. Bill removed after it was credited.....	48
47. A bill was inserted while the bill validator is deactivated.....	48
48. Recycler has received a change request.....	48
49. Cash sale reported to the cashless device.....	49
50. Cash sale successfully reported to the cashless device.....	49
51. Cash sale reporting to the cashless device failed.....	49
52. Cashless device setup time exceeded.....	49
53. Cashless device max/min prices reporting time exceed.....	49
54. Cashless device poll time exceed.....	49
55. Cashless device expansion request ID time exceed.....	50
56. Cashless device expansion enable options time exceed.....	50
57. Cashless device have Always Idle support and it will be enabled.....	50

58. Cashless device writing date/time exceed.....	50
59. Cashless device sent a display message.....	50
60. Cashless device sent a BEGIN SESSION message.....	51
61. Cashless device sent a VEND APPROVED message.....	51
62. Cashless device sent a VEND DENIED message.....	51
63. Cashless device sent an END SESSION message.....	51
64. Cashless device sent a CANCELED message.....	51
65. Cashless device is ready.....	51
66. Cashless device returned a malfunction error.....	51
67. Cashless device returned COMMAND OUT OF SEQUENCE message.....	52
68. Cashless device sent a REVALUE APPROVED message.....	52
69. Cashless device sent a REVALUE DENIED message.....	52
70. Cashless device sent a REVALUE LIMIT message.....	52
71. Cashless device sent a DATE/TIME request message.....	52
72. Interface successfully sent date/time command to the cashless device.....	52
73. Interface failed sending date/time command to the cashless device.....	52
74. Interface successfully enabled the cashless device.....	52
75. Interface successfully enabled the cashless device.....	53
76. Coin acceptor/changer just reset waiting time exceeded.....	53
77. Coin acceptor/changer setup time exceeded.....	53
78. Coin acceptor/changer expansion identification time exceeded.....	53
79. Coin acceptor/changer feature enable time exceeded.....	53
80. Coin acceptor/changer tube status time exceeded.....	54
81. Coin acceptor/changer is not ready for the issued command.....	54
82. Coin acceptor/changer tube status.....	54
83. Coin acceptor/changer poll time exceeded.....	54
84. Coin acceptor/changer is busy dispensing coins.....	54
85. Coin acceptor/changer temporarily unable to dispense coins.....	55
86. Coin acceptor/changer has failed to dispense all or some of the required coins.....	55
87. Coin acceptor/changer is reporting a manual coin dispense.....	55
88. Coin acceptor/changer received a token.....	55
89. Coin acceptor/changer received a coin.....	56
90. Coin acceptor/changer detected a slug.....	56
91. Coin acceptor/changer is in normal condition.....	56
92. Coin acceptor/changer received a change request.....	56
93. Coin acceptor/changer received a coin that was not credited.....	56
94. Coin acceptor/changer has a defective tube sensor.....	56
95. Coin acceptor/changer detected a double arrival.....	57
96. Coin changer detected an acceptor disconnection.....	57
97. Coin acceptor/changer detected a tube jam.....	57
98. Coin acceptor/changer detected an internal ROM error.....	57
99. Coin acceptor/changer detected a routing error.....	57
100. Coin acceptor/changer detected reset condition.....	57
101. Coin acceptor/changer detected a coin jam.....	57
102. Coin acceptor/changer detected the removal of a credited coin.....	57
103. Hopper reset result.....	58
104. Hopper reset result.....	58
105. Hopper disabled.....	58
106. Hopper enabled.....	58
Appendix I – Bill related LAST ERROR messages.....	59
Appendix II – Coin related LAST ERROR messages.....	61
Appendix III – Cashless related LAST ERROR messages.....	63
NOTES:.....	65

I. Introduction

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

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

- a computer over USB or RS232;
- a single board computer (like Raspberry Pi, Orange Pi, etc.) over USB or RS232 serial interface;
- a PLC or other device with RS232 UART

Also, please pay attention to the part number while ordering, since there are 2 different firmware versions available (one for USB and one for RS232).

Connecting to a computer or a SBC by RS232 requires a straight DB9 male/DB9 female cable which is not sold by our company and which you can easily find on almost any hardware store. Also, you can use USB to RS232 adapters/cables.

The device is using a simple ASCII protocol over USB/RS232 interfaces. There is no need for you to have deep MDB knowledge, however, some knowledge about MDB will help you faster and better understand the interface functionality.

Standard package content:

- PICOVEND EZ MASTER board, no cables provided.

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

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

II. Hardware

A. Board overview

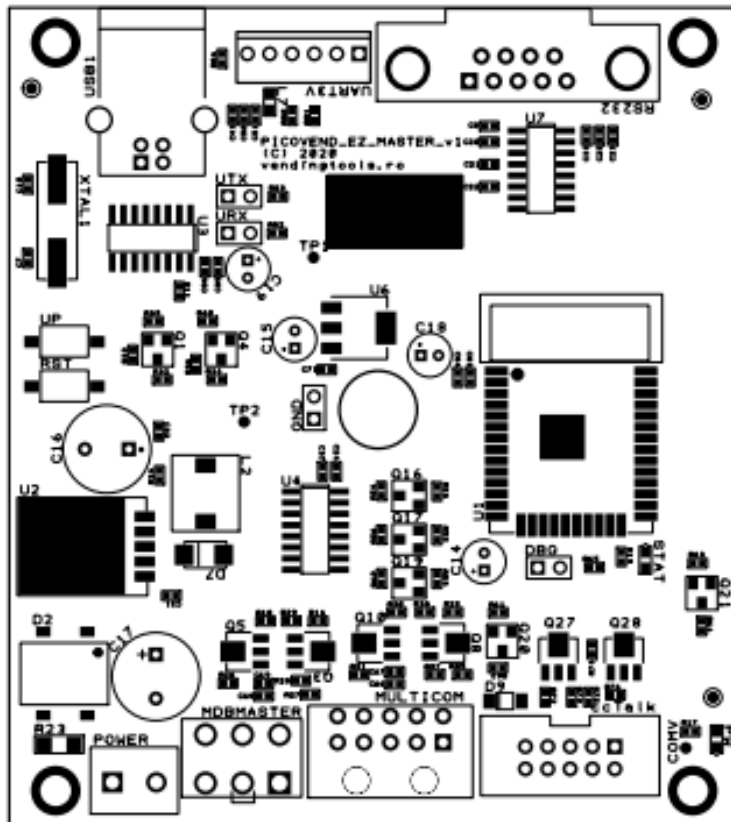


Figure 1 - Board overview

B. Connectors description

1. POWER – connect your external power supply to this connector in order to power the device and, also, the connected peripherals (MDB payment systems, ccTalk payment systems, etc.). You need to make sure your power supply is matching the connected MDB and ccTalk power requirements (voltage and current). The maximum momentary drained current simultaneously drained from MDB and ccTalk should not exceed 4A.

2. MDBMASTER – this connector allows the device interfacing with MDB peripherals (bill validator/recycler, coin acceptor/changer and cashless device);

3. MULTICOM – is a multipurpose interface connector, not used with this version. In future software versions it will offer Executive interface and ccTalk for non-standard connector peripherals.

4. ccTalk – this is the connector for standard 10pin ccTalk interfaces. It's pinout is the following:

- PIN #1 – ccTalk data;
- PIN #2 – N/C;
- PIN #3 – N/C;
- PIN #4 – GND;
- PIN #5 – N/C;
- PIN #6 – N/C;
- PIN #7 – VCC POWER;
- PIN #8 – GND;
- PIN #9 – N/C;
- PIN #10 – VCC POWER;

5. RS232 – this is the connector for RS232 interface (requires a special firmware to work on RS232, that will not support USB interface). This is a female connector and requires a DB9 male/DB9 female straight cable to connect to a computer (this cable is not shipped by our company, but can be easily found on almost any local hardware store).

6. UART3V – not provided in current version.

- PIN #1 – N/C;
- PIN #2 – GND;
- PIN #3 – TXD;
- PIN #4 – RXD;
- PIN #5 – 3.3V out (max 100mA);
- PIN #6 – 5V out (max 300mA).

7. USB1 connector to use on USB hosts – requires some drivers to install on host, to emulate a virtual serial interface (default baudrate is 115200).

C. Communication interfaces

Both RS232 and USB serial interfaces are using same communication parameters:

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

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

All commands must end with <CR> and <LF> (0x13, 0x10)

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

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

III. Communication protocol

A. MDB master related commands and answers

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

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

Answers are always upper case.

B. Bill validator/recycler related commands

1. Enable bill validator

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

2. Selective bills enable

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

3. Disable bill validator

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

4. Reset bill validator

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

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

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

6. Reject bill while a bill is in escrow position

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

7. 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.

8. 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).

9. Check if the bill validator was enabled

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

10. Get the bill validator configured bills values

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

11. Get the bill validator information

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

12. Get the bill validator settings

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

13. Get the bill recycler bill type values

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

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

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

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

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

16. Get tot total value of bills available for recycling

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

17. Dispense bills as a change to customer

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

18. Dispense bills as a change to customer

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

19. Get current bill stacker status

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

C. Coin acceptor/changer related commands

1. Enable coin acceptor/changer

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

2. Selective coins enable

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

3. Disable coin acceptor

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

4. Reset coin acceptor

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

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

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

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

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

7. Dispense some coins using MDB alternative payout method

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

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

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

9. Check if the coin acceptor/changer was enabled

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

10. Get last 10 coin acceptor/changer codes

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

11. Get the coin acceptor/changer configured coins values

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

12. Get the coin acceptor/changer information

Command	
COININFOREQ?	This command will read the coin acceptor/changer information for statistics and payment systems inventory tracking. This information is read during automatic coin acceptor/changer initialization phase, after a power-up or after issuing COINRESET command.
Possible answers	
- COININFOREQ(A,B,C)	- 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)
- COINNOTINITED	- The coin acceptor was not previously initialized and the information is not available

13. Get the coin acceptor/changer settings

Command	
COINSETTINGS?	This command will read the bill validator settings. This information is read during automatic bill validator initialization phase, after a power-up or after issuing COINRESET command.
Possible answers	
- COINSETTINGS(A,B,C,D,E,F,G,H,I,J,K,L,M,N,O,P,Q,R,S)	- 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.
- COINNOTINITED	- The coin acceptor/changer was not previously initialized and the information is not available

14. Get the token values

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

15. Set the token values

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

16. Get coins quantity for each type (only for devices purchased after 2023-12-05)

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

D. Cashless related commands

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

1. Reset cashless device

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

2. Enable cashless device

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

3. Disable cashless device

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

4. Cancel current cashless activity

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

5. Request cashless current revalue limit

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

6. Request approval for a vend

Command	
CSLS<X>VNDREQ(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	
- CSLS<X>NOSESSION	- You will receive this answer if you are requesting for a vend approval and the cashless device is Level 2 or Level 3 without Always Idle support and a cashless session is not opened.
- CSLS<X>VNDREQOK	- If the interface successfully received and parsed the command.
- CSLS<X>VNDREQFAIL	- If the interface was not able to successfully receive and parse the command.

7. Confirm a success vend to the cashless device

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

8. Report a vend failure to the cashless device

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

9. Report a cash sale to the cashless device

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

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

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

11. Get last 10 cashless device codes

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

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

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

13. Check if the cashless device was enabled

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

14. Get the cashless device information

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

15. Get the cashless device settings

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

16. Cashless force session complete

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

17. Request approval for a negative vend (only for devices purchased after 2023-12-05)

Command	
CSLS<X>NEGVNDREQ(A,B)	This command will request a negative 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	
- CSLS<X>NOSESSION - CSLS<X>NEGVNDREQOK - CSLS<X>NEGVNDREQFAIL	- You will receive this answer if you are requesting for a negative 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.

E. Interface (VMC) system related commands

1. Get VMC settings

Command	
VMCSETTINGS?	This command will read interface internal settings.
Possible answers	
- VMCSETTINGS(A,B,C,D)	<ul style="list-style-type: none">- A is the VMC configured feature level (this interface can only work as a Level 2 and level 3 VMC)- B is number of characters on display/columns (maximum 16). If this value is set to 0, the VMC will inform cashless devices that it is not supporting display messages.- C is the number of rows on display- D is the display type, according to MDB specifications, cashless display message section.

2. Set VMC settings

Command	
- VMCSETTINGS(A,B,C,D)	<p>This command will set interface internal settings</p> <ul style="list-style-type: none">- A is the VMC configured feature level (this interface can only work as a Level 2 and level 3 VMC)- B is number of characters on display/columns (maximum 16). If this value is set to 0, the VMC will inform cashless devices that it is not supporting display messages.- C is the number of rows on display- D is the display type, according to MDB specifications, cashless display message section.
Possible answers	
<ul style="list-style-type: none">- FTLVLERROR- VMCSETTINGSOK	<ul style="list-style-type: none">- The VMC feature level you mentioned in parameters is invalid.- Command correctly received and parsed.

3. Set VMC manufacturer code

Command	
- VMCSETMFCODE(AAA)	<p>This command will set interface internal manufacturer code that it is reporting to cashless device during automated initialization phase.</p> <ul style="list-style-type: none">- AAA is a fixed length, 3 characters (ASCII) value
Possible answers	
<ul style="list-style-type: none">- VMCSETMFCODEERR1- VMCSETMFCODEOK	<ul style="list-style-type: none">- The VMC manufacturer code length you mentioned as a parameter is invalid.- Command correctly received and parsed.

4. Set VMC internal serial number

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

5. Set VMC internal model number

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

6. Read the last error

Command	
- LASTERROR?	This command will read the last error value. You can use this command to obtain some additional error codes after a command fails. You can find details about last error codes in the Appendix I, II and III
Possible answers	
- LASTERROR(ERR_CODE)	- The interface will return last known error code.

7. Clear the last error

Command	
- CLEARLASTERROR	This command will clear the last error value. You may use this to clear last error code variable in order to keep it up to date. After issuing this command, the LASTERROR? command will read LASTERROR(NOERR).
Possible answers	
- CLEARLASTERROROK	- The interface correctly received and parsed the command.

8. Reset interface internal counters

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

9. Interface reboot

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

10. 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.

11. Read internal counters

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

12. Save settings

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

13. Load settings

Command	
- LOADSETTINGS	This command will force loading settings. It is also automatically executed on power-up.
Possible answers	
- LOADSETTINGSOK	- If the interface failed to save settings to non-volatile memory. This command will return the results of the following commands, together: - VMCSETTINGS? - VMCINFOREQ? - RECYCLERSETBILLS? - TOKENVALUES? Also, it will return VMCSWVER(A,B) – the internal software version, major an minor release
- DEFAULTSETTINGS	- If the settings file is not available.
- LOADSETTINGSFAIL	- If the settings file could not be read.

14. Factory reset

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

15. POLL command (only for devices purchased after 2023-12-05)

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

F. Relay related commands

This set of commands is working only when an optional PICOVEND EZ ESP 8X slave relay board is connected on the MDB interface of PICOVEND EZ MASTER interface. Please check the optional PICOVEND EZ ESP 8X board manual for its usage/specifications. Mainly, this board is used to control up to 8 external circuits, being equipped with 8 relays. It can receive relay control commands over USB or over MDB (being an MDB slave device). Each relay can be addressed individually, or you can address all relays at one time, and each relay can be energized for a desired amount of time, between 1 and 65535 seconds. Up to 8 relay boards can be controlled by this interface (a total of 64 relays).

1. Individual relay control

Command	
- RELAY(A,B,C)	<p>This command will energize one relay for a specified amount of time.</p> <ul style="list-style-type: none"> - A – is the relay board module address, set by MYADDRESS command over USB and saved with SAVESETTINGS command after. - B – is the relay number (1-8) - C – is the time for the relay to be energized, in seconds (1-65535). The board will automatically de-energize the relay after the specified time passed. To force the relay de-energize at any moment, you need to specify 0 for this parameter <p>Example: RELAY(1,3,45) – will energize the relay number 3 (OUT3 on the board) for a specified time of 45 seconds</p>
Possible answers	
- RELAYOK - RELAYFAIL	<ul style="list-style-type: none"> - If the command was successfully received and executed. - If the command was not successfully received and executed.

2. Batch relay control

Command	
- RELAYALL(A,B,C,D,E,F,G,H,I)	<p>This command will energize/de-energize all slave board relays with a single command</p> <ul style="list-style-type: none"> - A – is the relay board module address, set by MYADDRESS command over USB and saved with SAVESETTINGS command after. - B-I – is the time for relay 1-8 (OUT1-OUT8). The board will automatically de-energize the relays after the specified time passed. To force the relay de-energize at any moment, you need to specify 0 for the parameter on the desired relays. <p>Example: RELAYALL(1,10,60,0,0,0,0,0,0) – will energize OUT1 relay for 10 seconds and OUT2 relay for 60 seconds, all the other relays will be de-energized.</p>
Possible answers	
- RELAYALLOK - RELAYALLFAIL	<ul style="list-style-type: none"> - If the command was successfully received and executed. - If the command was not successfully received and executed.

3. Check relay status

Command	
- RELAYSTATUS(A)	This command will check the relays status and will return the time that each relay will be still energized. You can use this command to check if some of the relays are energized and for how long. - A is the relay board module address, set by MYADDRESS command over USB and saved with SAVESETTINGS command after.
Possible answers	
- RELAYSTATUS(A,B,C,D,E,F,G,H,I)	- If the command was successfully received and executed. - A is the relay board module address, set by MYADDRESS command over USB and saved with SAVESETTINGS command after.
- RELAYSTATUSFAIL	- B-I – is the time in seconds that relays (OUT1-OUT8) will be still energized. If the returned value is 0, the corresponding relay is de-energized. - If the command was not successfully received and executed.

4. Relay reset

Command	
- RELAYRESET(A)	This command will reset all relays time to 0 and de-energize them - A - is the relay board module address, set by MYADDRESS command over USB and saved with SAVESETTINGS command after.
Possible answers	
- RELAYRESETOK	- If the command was successfully received and executed.
- RELAYRESETFAIL	- If the command was not successfully received and executed.

G. ccTalk hoppers related commands

This command set is related to attached ccTalk hoppers (if any). The device can handle up to 4 hoppers (addresses 3-6). You need to set your hoppers addresses to one of the accepted one (3, 4, 5 or 6). The interface is working only with non-encrypted ccTalk hoppers.

1. Set ccTalk debug information mode over USB

Command	
- CCTDEBUG(A)	This command is used to enable or disable the debug messages over the USB interface if they are needed for ccTalk hoppers diagnostic. - A can have 2 values (0 – debug messages are not enabled – default, and 1 – debug messages are enabled). You need to use SAVESETTINGS command to make configuration persistent over reboot/power down events.
Possible answers	
- CCTDEBUGOK - CCTDEBUGFAIL	- If the command was successfully received and executed. - If the command was not successfully received and executed.

2. Get ccTalk debug information mode over USB

Command	
- CCTDEBUG?	This command is used to check the debug messages mode over the USB interface if they are needed for ccTalk hoppers diagnostic.
Possible answers	
- CCTDEBUG(0) - CCTDEBUG(1)	- If the debug is currently disabled - If the debug is currently enabled

3. Check if the hopper is enabled

Command	
- CCTHENABLED(A)	This command is used to check if a specific hopper is enabled. - A is the hopper address (accepted values 3-6)
Possible answers	
- CCTHENABLED(A,B)	The answer has 2 parameters: - A is the chopper address you requested - B is the enable flag (the requested hopper is enabled if this parameter is 1 and is not enabled if this parameter is 0). All detected hoppers are initialized and enabled on boot. If, during normal operation, some communication lost event occurs, the unresponsive hoppers are automatically disabled. At any moment you can force hopper re-initialization using command CCTHINIT(A) (see details below).

4. Hopper dispense tokens/coins

Command	
- CCTHDISPENSE(A,B,C)	<p>This command is used to dispense some tokens/coins</p> <ul style="list-style-type: none"> - A is the hopper address (accepted values are 3,4,5 and 6) depending on the number of connected hoppers and their configured address. - B is the number of tokens/coins you need to dispense. - C is the dispense mode and it can have the following values: <ul style="list-style-type: none"> • 0 – auto mode – the interface will try to detect if the hopper is using a cipher key for dispensing operation or not, it should work with most of the hoppers; • 1 – cipher mode – the interface will try to dispense tokens/coins using a cipher key – use this only if auto mode is not working with your hopper; • 2 – no cipher mode – the interface will try to dispense tokens/coins not using a cipher key – use this only if auto mode is not working with your hopper.
Possible answers	
- CCTHDISPENSEOK(A,B,C)	- This answer is a confirmation that the dispense command was ACK-ed by the addressed hopper. Parameters are identical with the above.
- CCTHFULLPAID(D,E)	- This answer is a confirmation that the hopper has dispensed all the requested tokens/coins (D – the hopper address, E – the number of dispensed tokens/coins).
- CCTHPARTPAID(F,G,H)	- This answer is a partially delivered message (or none). If the hopper could not dispense all the requested tokens/coins, it returns this message (F – the hopper address, G – the number of the dispensed tokens/coins, H – the number of tokens that could not be dispensed, from the total dispensing number issued by the dispense command).

5. ccTalk hopper init

Command	
- CCTHINIT(A)	<p>This command is used to force a hopper initialization. It may be useful to re-initialize hoppers on some moments.</p> <ul style="list-style-type: none"> - A is the hopper address (accepted values are 3, 4, 5 and 6).
Possible answers	
- CCTHINITOK(A)	<ul style="list-style-type: none"> - If the command was successfully received and executed. - A is the hopper address you requested

6. ccTalk hopper level sensors reading

Command	
- CCTHHILOSTAT(A)	This command is used to force a sensor set reading on the hopper level. - A is the address of the hopper you need to check (accepted values are 3, 4, 5 and 6).
Possible answers	
- CCTHHILOSTATFAIL(A) - CCTHILOSTAT(A,B,C,D,E)	- If the command returns no valid value or a hopper with the mentioned address is not connected <ul style="list-style-type: none">• A is the hopper address you requested - If the command was successfully executed <ul style="list-style-type: none">• A is the hopper address you requested• B high level sensor presence flag. If 1, the hopper is equipped with a high level sensor, if 0, the hopper has no high level sensor.• C high level flag. If 1, there are enough tokens/coins in the hopper, over the high level sensor.• D low level sensor presence flag. If 1, the hopper is equipped with a low level sensor, if 0, the hopper has no low level sensor.• E low level flag. If 1, then there are no (or little) tokens/coins in the hopper. If 0, there are some tokens/coins in the hopper, over the low level sensor.

H. USB to MQTT bridge

This is a small application that can be executed in background, in order to simplify the interface integration. Using this application you don't need to handle the serial communication and you can use a simple MQTT broker (like mosquitto) with MQTT libraries available for your preferred programming language.

For example, if you are using Raspberry Pi, all you need is to install and configure mosquitto, run our Java USB to MQTT bridge and subscribe and publish to the correct topic. The USB to MQTT bridge application can be also used in networked environments, for example, if your project is hosted on a local server with an MQTT broker, simplifying the integration. You can also create network based applications (for example, if you build a kiosks network, the business logic may reside on the backend and the kiosks are only used as an interface with the user, making the whole system easily to update and manage).

Click [DOWNLOAD](#) to get the application.

This application requires Java (JRE) and is executed in console/terminal without graphic interface so it can be used on headless projects as well by starting it on boot.

To launch the application you need to run the following command:

```
java -jar picovend_ez_master_mqtt_v1.jar <params>
```

<params> must include at least the following:

--mqttthost=<your_broker_IP_or_FQDN> - the application will try to connect to this host

--mqttport=<your_broker_port> - the application will try to connect on this port

--mqttuser=<broker_username> - the MQTT broker user name

--mqtpass=<broker_pass> - the MQTT broker password (does not accept spaces)

--serialport=<device> - the application will try to connect to the device using this serial port (on Linux it may be /dev/ttyUSB<x>, on Windows it may be COM<x>).

There is an optional parameter that will enable the debug information to console:

--debug=1

IV. Unsolicited messages

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

1. Power-up messages

Those messages are sent on interface power-up or reboot (SYSRESET command)

Message	Description
MDBMASTERSTART	- This message comes out on power-up
INITFSOK(A,B,C)	- A is the file system initialization mode - B is the file system used bytes - C is the file system total capacity (bytes)
CNTRINIT	- This message only occurs after using COUNTERSRESET command
CNTR(A,B,C,D,E,F,G,H,I,J,K,L)	- It is the counters vector, please check on CNTR? command for details.
VMCSETTINGS(A,B,C,D)	- It is the VMC settings vector, please check on VMCSETTINGS? command for details.
VMCINFOREQ(A,B,C)	- It is the VMC info vector, please check on VMCINFOREQ? command for details
VMCSWVER(A,B)	- It is the interface software version, A is the major release version and B is the minor release version
RECYCLERSETBILLS(A,B,C,D,E,F,G,H,I,J,K,L,M,N,O,P)	- It is the recycler info vector, please check RECYCLERSETBILLS? command for details
TOKENVALUES(A,B,C,D,E,F,G,H,I,J,K,L,M,N,O,P)	- It is the tokens values vector, please check TOKENVALUES? command for details
CNTRWRERR	- May appear on boot after using COUNTERSRESET command, if the counters file could not be initialized. This is a fatal error and the device will not work properly.
CNTRRDERR	- May appear on boot if the counters file is corrupted. You may try to use COUNTERSRESET and a reboot to create a fresh counters file
LOADSETTINGSOK	- This ends the configuration auto loading messages batch

2. 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.

3. 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.

4. 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.

5. 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.

6. 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.

7. Bill validator with recycling support has been identified

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

8. 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

9. 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.

10. 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.

11. 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.

12. Bill recycler enabling failed

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

13. Bill recycler function successfully enabled

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

14. 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.

15. Bill recycler reading dispense status time exceeded

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

16. Bill recycler reading dispense status returned a NAK

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

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 reading dispense status returned an ACK

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

19. Bill recycler remaining stock value

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

20. 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.

21. Bill dispensing command time exceed

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

22. Bill dispensing command time exceed

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

23. 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.

24. 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.

25. 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.

26. 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.

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

Message	Description
BILLPOLLEXCEED	<ul style="list-style-type: none">- 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.

28. Bill validator – one bill stacked

Message	Description
BILLSTACKED(A,B,C)	<ul style="list-style-type: none">- 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)

29. Bill in escrow position

Message	Description
BILLESCROW(X)	<ul style="list-style-type: none">- 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

30. Bill returned to customer

Message	Description
BILLRETURNED(X)	<ul style="list-style-type: none">- The bill in escrow position returned to customer- X is the scaled value of the returned bill.

31. Bill received in recycler

Message	Description
BILLTORECYCLER(X)	<ul style="list-style-type: none">- A bill was received and stored in the recycling box- X is the scaled value of the stored bill.

32. A disabled bill was rejected

Message	Description
BILLDISREJ(X)	<ul style="list-style-type: none">- A bill was rejected because it was previously disabled by the user application.- X is the scaled value of the rejected bill.

33. 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.

34. 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.

35. 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.

36. 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

38. Bill validator have a defective sensor

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

39. Bill validator is busy

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

40. Bill validator ROM error

Message	Description
BILLROMERROR	- Bill validator encountered an internal ROM error

40. Bill validator jam

Message	Description
BILLJAM	- Bill validator encountered a bill jam error

41. Bill validator was reset

Message	Description
BILLRESET	- Bill validator has just been reset.

42. Bill removed from bill validator

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

43. Bill validaor cashbox has been removed

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

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

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

45. Bill validator has been rejected a bill

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

46. Bill removed after it was credited

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

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

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

48. Recycler has received a change request

Message	Description
RECYCLERCHGREQUEST	- Recycler has received a change request

49. Cash sale reported to the cashless device

Message	Description
CSLS<X>CASHSALE(A,B)	<ul style="list-style-type: none">- Cash sale was reported to the cashless device- A is the item price- B is the item ID/selection number

50. Cash sale successfully reported to the cashless device

Message	Description
CSLS<X>CASHSALEOK	<ul style="list-style-type: none">- A cash sale was successfully reported to the cashless device.

51. Cash sale reporting to the cashless device failed

Message	Description
CSHS<X>CASHSALEFAIL	<ul style="list-style-type: none">- A cash sale reporting to the cashless device failed.

52. Cashless device setup time exceeded

Message	Description
CSLS<X>SETUPEXCEED	<ul style="list-style-type: none">- 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.

53. Cashless device max/min prices reporting time exceed

Message	Description
CSLS<X>MAXMINEXCEED	<ul style="list-style-type: none">- 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.

54. Cashless device poll time exceed

Message	Description
CSLS<X>POLLEXCEED	<ul style="list-style-type: none">- 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.

55. Cashless device expansion request ID time exceed

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

56. Cashless device expansion enable options time exceed

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

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

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

58. Cashless device writing date/time exceed

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

59. Cashless device sent a display message

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

60. Cashless device sent a BEGIN SESSION message

Message	Description
CSLS<X>BEGIN(A,B,C)	<ul style="list-style-type: none">- 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

61. Cashless device sent a VEND APPROVED message

Message	Description
CSLS<X>VNDAPP(A,B,C)	<ul style="list-style-type: none">- 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)

62. Cashless device sent a VEND DENIED message

Message	Description
CSLS<X>VNDDEN	<ul style="list-style-type: none">- The cashless device sent a VEND DENIED message to the interface

63. Cashless device sent an END SESSION message

Message	Description
CSLS<X>ENDSESSION	<ul style="list-style-type: none">- The cashless device sent an END SESSION message to the interface

64. Cashless device sent a CANCELED message

Message	Description
CSLS<X>CANCELED	<ul style="list-style-type: none">- The cashless device sent a CANCELED message to the interface

65. Cashless device is ready

Message	Description
CSLS<X>READY	<ul style="list-style-type: none">- The cashless device was correctly initialized and is ready to be enabled.

66. Cashless device returned a malfunction error

Message	Description
CSLS<X>MALFUNCTION(X)	<ul style="list-style-type: none">- 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

67. Cashless device returned COMMAND OUT OF SEQUENCE message

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

68. Cashless device sent a REVALUE APPROVED message

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

69. Cashless device sent a REVALUE DENIED message

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

70. Cashless device sent a REVALUE LIMIT message

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

71. Cashless device sent a DATE/TIME request message

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

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

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

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

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

74. Interface successfully enabled the cashless device

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

75. Interface successfully enabled the cashless device

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

76. 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.

77. 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.

78. 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.

79. 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.

80. 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.

81. 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

82. 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.

83. 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.

84. 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).

85. Coin acceptor/changer temporarily unable to dispense coins

Message	Description
COINCHGNOTNOW	<ul style="list-style-type: none">- The coin acceptor/changer is temporarily unable to dispense coins due to it's working stage. Your application should retry later.

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

Message	Description
CHANGEREMAINING(X)	<ul style="list-style-type: none">- 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.

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

Message	Description
COINMANDISP(A,B,C)	<ul style="list-style-type: none">- 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

88. Coin acceptor/changer received a token

Message	Description
TOKENIN(A,B,C,D)	<ul style="list-style-type: none">- 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)

89. Coin acceptor/changer received a coin

Message	Description
COININ(A,B,C,D,E)	<ul style="list-style-type: none">- 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)

90. Coin acceptor/changer detected a slug

Message	Description
COINSLUG(A,B)	<ul style="list-style-type: none">- 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)

91. Coin acceptor/changer is in normal condition

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

92. 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.

93. 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.

94. Coin acceptor/changer has a defective tube sensor

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

95. 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).

96. Coin changer detected an acceptor disconnection

Message	Description
COINACCUNPL	- Coin changer detected an acceptor disconnection.

97. Coin acceptor/changer detected a tube jam

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

98. Coin acceptor/changer detected an internal ROM error

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

99. Coin acceptor/changer detected a routing error

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

100. Coin acceptor/changer detected reset condition

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

101. 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

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

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

103. Hopper reset result

Message	Description
CCTHRESET(X)	- Hopper with address X answered to reset command

104. Hopper reset result

Message	Description
CCTHRESETFAIL(X)	- Hopper with address X failed to answer on reset command

105. Hopper disabled

Message	Description
CCTHDISABLED(X)	- Hopper with address X failed to initialize and will not be available for coin dispensing

106. Hopper enabled

Message	Description
CCTHENABLED(X)	- Hopper with address X successfully initialized and enabled

Appendix I – Bill related LAST ERROR messages

Those are the codes you can read using LASTERROR? command after the interface has been returned a bill validator/recycler related error or fail message

CODE	DESCRIPTION
ERRB001	No answer on bill reset
ERRB002	NAK on bill reset
ERRB003	Unknown answer on bill reset
ERRB004	CRC error on bill reset
ERRB005	Unknown error on bill reset
ERRB006	No answer while polling bill for JUST RESET
ERRB007	NAK while polling bill for JUST RESET
ERRB008	Unknown answer on polling bill for JUST RESET
ERRB009	CRC error on polling bill for JUST RESET
ERRB010	No answer on bill SETUP
ERRB011	Bill answer length error on bill SETUP
ERRB012	CRC error on bill SETUP
ERRB013	No answer on bill EXPANSION IDENTIFICATION
ERRB014	Bill answer length error on bill EXPANSION IDENTIFICATOIN
ERRB015	CRC error on bill EXPANSION IDENTIFICATION
ERRB016	No answer on bill EXPANSION IDENTIFICATION with options
ERRB017	Bill answer length error on bill EXPANSION IDENTIFICATOIN with options
ERRB018	CRC error on bill EXPANSION IDENTIFICATION with options
ERRB019	No answer on bill FEATURE ENABLE
ERRB020	NAK on bill FEATURE ENABLE
ERRB021	CRC error on bill FEATURE ENABLE
ERRB022	No answer on bill RECYCLER SETUP
ERRB023	NAK on bill RECYCLER SETUP
ERRB024	CRC error on bill RECYCLER SETUP
ERRB025	No answer on bill RECYCLER ENABLE
ERRB026	CRC error on bill RECYCLER ENABLE
ERRB027	No answer on bill RECYCLER DISPENSE STATUS
ERRB028	CRC error on bill RECYCLER DISPENSE STATUS
ERRB029	No answer on bill RECYCLER DISPENSE STATUS
ERRB030	CRC error on bill RECYCLER DISPENSE STATUS
ERRB031	No answer on bill DISPENSE VALUE
ERRB032	CRC error on bill RECYCLER ENABLE
ERRB033	No answer on bill PAYOUT VALUE POLL
ERRB034	CRC error on bill PAYOUT VALUE
ERRB035	No answer on bill STACKER

CODE	DESCRIPTION
ERRB036	NAK on bill STACKER
ERRB037	CRC error on bill STAKER
ERRB038	No answer on manual bill STACKER
ERRB039	Incorrect answer length on bill STACKER
ERRB040	CRC error on bill STACKER
ERRB041	No answer on bill POLL
ERRB042	CRC error on bill POLL
ERRB043	No answer on bill ENABLE
ERRB044	NAK on bill ENABLE
ERRB045	Unknown answer on bill ENABLE
ERRB046	CRC error on bill ENABLE
EERB047	Bill not in correct stage for this command
ERRB048	No answer on bill SELECTIVE ENABLE
ERRB049	NAK on bill SELECTIVE ENABLE
ERRB050	Unknown answer on bill SELECTIVE ENABLE
ERRB051	CRC error on bill SELECTIVE ENABLE
ERRB052	No answer on bill DISABLE
ERRB053	NAK on bill DISABLE
ERRB054	Unknown answer on bill DISABLE
ERRB055	CRC error on bill DISABLE
ERRB056	No answer on bill ACCEPT
ERRB057	NAK on bill ACCEPT
ERRB058	Unknown answer on bill ACCEPT
ERRB059	CRC error on bill ACCEPT
ERRB060	No answer on bill REJECT
ERRB061	NAK on bill REJECT
ERRB062	Unknown answer on bill REJECT
ERRB063	CRC error on bill RESET
ERRB064	Bill already enabled
ERRB065	Bill already disabled

Appendix II – Coin related LAST ERROR messages

Those are the codes you can read using LASTERROR? command after the interface has been returned a coin acceptor/changer related or fail message

CODE	DESCRIPTION
ERRC001	No answer on coin RESET
ERRC002	NAK on coin RESET
ERRC003	Unknown answer on coin RESET
ERRC004	CRC error on coin RESET
ERRC005	No answer polling coin for JUST RESET
ERRC006	NAK on coin polling for JUST RESET
ERRC007	Unknown answer on coin poll for JUST RESET
ERRC008	CRC error on coin poll for JUST RESET
ERRC009	No answer on coin SETU
ERRC010	NAK on coin SETUP
ERRC011	Unknown answer on coin SETUP
ERRC012	CRC error on coin SETU
ERRC013	No answer on coin EXPANSION IDENTIFICATION
ERRC014	NAK on coin EXPANSION IDENTIFICATION
ERRC015	Unknown answer on coin EXPANSION IDENTIFICATION
ERRC016	CRC error on coin EXPANSION IDENTIFICATION
ERRC017	No answer on coin FEATURE ENABLE
ERRC018	NAK on coin FEATURE ENABLE
ERRC019	CRC error on coin FEATURE ENABLE
ERRC020	No answer on coin TUBE STATUS
ERRC021	ACK/NAK only on coin TUBE STATUS
ERRC022	Incorrect answer length on coin TUBE STATUS
ERRC023	CRC error on coin TUBE STATUS
ERRC024	Unknown error on coin TUBE STATUS
ERRC025	No answer on coin TUBE STATUS
ERRC026	ACK/NAK only on coin TUBE STATUS
ERRC027	Incorrect answer length on coin TUBE STATUS
ERRC028	CRC error on coin TUBE STATUS
ERRC029	Unknown error on coin TUBE STATUS
ERRC030	No answer on coin ENABL
ERRC031	NAK on coin ENABLE
ERRC032	Unknown answer on coin ENABLE
ERRC033	CRC error on coin ENABLE
ERRC034	Coin not in correct stage for this command
ERRC035	No answer on coin SELECTIVE ENABLE

CODE	DESCRIPTION
ERRC036	NAK on coin SELECTIVE ENABLE
ERRC037	Unknown answer on coin SELECTIVE ANSWER
ERRC038	CRC error on coin SELECTIVE ANSWER
ERRC039	No answer on COIN DISABLE
ERRC040	NAK on coin DISABLE
ERRC041	Unknown answer on coin DISABLE
ERRC042	CRC error on coin DISABLE
ERRC043	No answer on coin EJECT
ERRC044	NAK on coin EJECT
ERRC045	Unknown answer on coin EJECT
ERRC046	CRC error on coin EJECT
EERB047	No answer on coin AP STATUS
ERRC048	ACK only answer on coin AP STATUS
ERRC049	NAK only answer on coin AP STATUS
ERRC050	Unknown answer on coin AP STATUS
ERRC051	CRC error on coin AP STATUS
ERRC052	No answer on coin AP EJECT
ERRC053	Unknown answer on coin AP EJECT
ERRC054	CRC error on coin AP EJECT
ERRC055	Failed to get tube status on coin AP
ERRC056	Coin already enabled
ERRC057	Coin already disabled

Appendix III – Cashless related LAST ERROR messages

Those are the codes you can read using LASTERROR? command after the interface has been returned a cashless device related or fail message

CODE	DESCRIPTION
ERRS001	NAK on CASHLESS RESET
ERRS002	CRC ERROR on CASHLESS RESET answer
ERRS003	Unknown error on CASHLESS RESET
ERRS004	No answer on CASHLESS SETUP
ERRS005	Cashless SETUP NAK
ERRS006	CRC error on CASHLESS SETUP
ERRS007	No answer on MAX/MIN prices
ERRS008	NAK on MAX/MIN prices
ERRS009	CRC error on MAX/MIN prices
ERRS010	Cashless not initied or not enabled
ERRS011	No answer on cashless vend request
ERRS012	NAK on vend request
ERRS013	CRC error on answer to vend request
ERRS014	Cashless not initied or not enabled
ERRS015	No answer on vend cancel
ERRS016	NAK on vend cancel
ERRS017	CRC error on response to vend cancel
ERRS018	Cashless not initied or not enabled
ERRS019	No answer on vend success
ERRS020	NAK on vend success
ERRS021	CRC error on vand success
ERRS022	Cashless not initied or not enabled
ERRS023	No answer on cashless vend failure
ERRS024	NAK on vend failure
ERRS025	CRC error on vend failure
ERRS026	Cashless not initer or not enabled
ERRS027	No answer on cashless session complet
ERRS028	NAK on cashless session complete
ERRS029	CRC error on cashless session complete
ERRS030	Cashless not initied or not enabled
ERRS031	No answer on cashless cash sale
ERRS032	NAK on cashless cash sale
ERRS033	CRC error on cashless cash sale
ERRS034	Cannot disable cashless during session
ERRS035	No answer on cashless disable

CODE	DESCRIPTION
ERRS036	Unknown answer on cashless disable
ERRS037	CRC error on cashless disable
ERRS038	No answer on cashless enable
ERRS039	Unknown answer on cashless enable
ERRS040	RC error on cashless enable
ERRS041	Cashless not initied or not enabled
ERRS042	No answer on cashless cancel
ERRS043	NAK error on cashless cance
ERRS044	CRC error on cashless cance
ERRS045	Cashless not initied or not enabled
ERRS046	No answer on cashless revalue request
EERB047	NAK on cashless revalue request
ERRS048	CRC error on cashless revalue request
ERRS049	Cashless not initied or not enabled
ERRS050	No answer on cashless revalue limit request
ERRS051	NAK on cashless revalue limit request
ERRS052	CRC error on cashless revalue limit request
ERRS053	No answer on cashless expansion ID reques
ERRS054	NAK on cashless expansion ID request
ERRS055	CRC error on cashless expansino ID request
ERRS056	No answer on cashless enable options
ERRS057	NAK on cashless enable options
ERRS058	CRC error on cashless enable options
ERRS059	No answer on cashless write time/date
ERRS060	NAK on cashless write time/date
ERRS061	CRC error on cashless write time/date
ERRS062	Cashless already enabled
ERRS063	Cshless already disabled

NOTES: