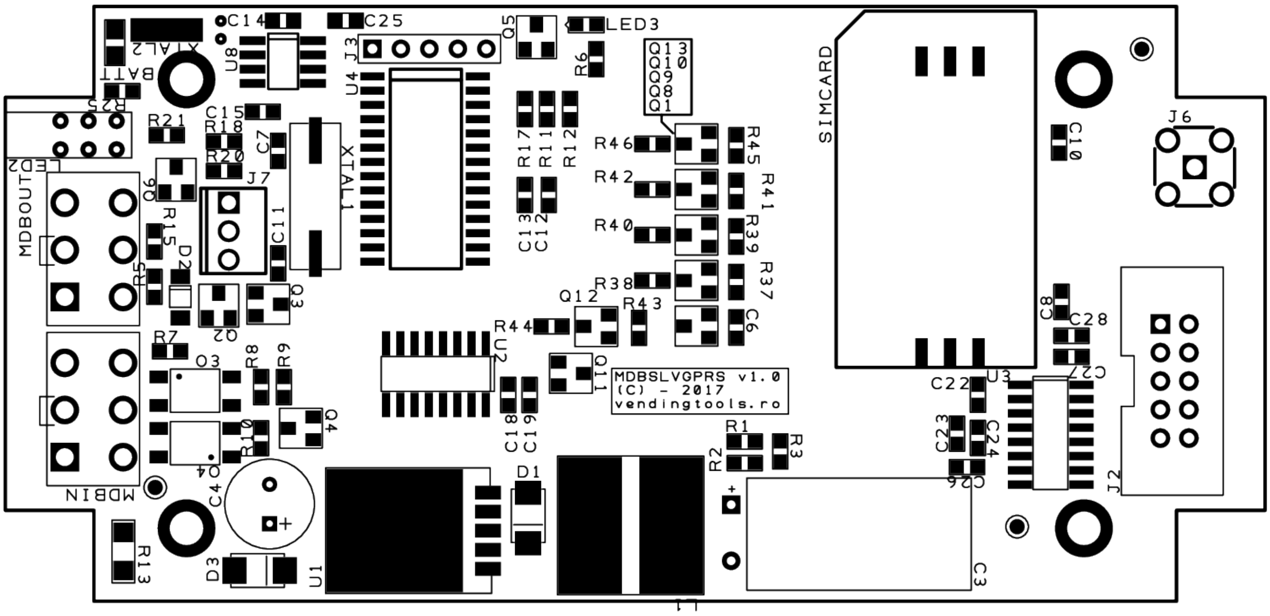
Interface MDB SLAVE to GPRS v26.09.2017 Quick reference

Table of Contents

I. Hardware overview



1. Power supply requirements

The INTERFACE is powered directly from the MDB bus, with a power consumption of 80mA in idle mode and 500mA on high data transfer stages, depending on the GSM signal quality, at a maximum 37VDC. You have to make sure the the MDB bus can support this power consumption and can supply DC. The device is not working with vending machines that are powering MDB peripherals with AC.

2. Connector description

- **<MDBIN> –** MDB input connector. It requires a pin to pin straight MDB cable.
- **<MDBOUT>** - Used to connect the MDB PERIPHERALS. This port was added to eliminate the need of an “Y” cable.
- **<J2>** - RS232 connector (interface for device configuration);
- **<J6>** - GSM antenna connector (SMA);
- **<J7>** - Buttons connector
    - PIN #1 - squared pin – button #1
    - PIN #2 – GND – common
    - PIN #3 – button #2

3. General description

You do not need to perform any settings on the INTERFACE, neither hardware or software.

This interface can be only used to act as a cashless device. You will be responsible to develop the server application, that will connect to the devices by using our gateway server.
Also, the interface is actively sniffing the MDB bus to catch the entire communication between VMC and the cash payment systems (bills and coins). There are some internal counters that are accumulating the cash inserted (coins and bills) and the change returned (coins). Those counters are accessible for reading by the server command (see reference below) and can be used for some sort of telemetry systems (there is also a reporting system that sends out on the serial port every selection on cash transactions – if the machine has the software implementation for MDB cashless cash vend messages).
The interface is managing the 9th bit and the time critical answers for the VMC and you only need to send some simple command.

This interface is using a configurable scaling factor using the local set-up application over the serial port.
It can be, also, remotely set (from the server) in single vend or multi vend.
The interface is emulating a Level 2 cashless device.

II. Communication gateway server

The device is connecting to the gateway server that will manage the low level communication protocol and convert it to a simple high level protocol consisting in text commands sent by a socket, with JSON answers. You need to develop your own wallet/card processing system that is connecting to our gateway in order to load credit on the machine, manage cashless sales and, if the machine supports the MDB cash sales reporting to the cashless device, to collect cash sales and stock. The gateway access is available for a small fixed fee. Your transactions and other data will not be stored on our gateway and are not available for reporting. Your server will be responsible for data collecting and reporting. Each device is identified by it's unique IMEI. All commands and answers are containing the IMEI to easily identify the destination/originating device.

There are 2 JSON messages categories, some of them are received from the gateway itself as a response to your commands and some of them are received from the device. Message's type is identified by the first variable (ServerMessage means answers from the gateway and DeviceMessage means an answer received by the gateway from the device and translated to the client).

1. Client login

Your server will act as a client to our gateway. For each device or devices group you will receive an user name and a password that will allow you to login on the gateway.

<table>
<tr><th><COMMAND></th><th><PARAMETERS></th></tr>
<tr><td>hello(<user_name>,<password>)</td><td>- <user_name> is your login user name, associated for a specific device or a device group;
- <password> is the password allocated for a specific user managing a device or a devices group.</td></tr>
<tr><th colspan="2">INTERFACE possible answers</th></tr>
<tr><td colspan="2">- {"ServerMessage": "ACK"}
- {"ServerMessage": "LoginRejected","Reason": "UnknownCommand"}
- {"ServerMessage": "LoginRejected","Reason": "ClientError"} – if your credentials are not valid
- {"ServerMessage": "LoginRejected","Reason": "SyntaxError"} – if your syntax is not correct
- {"ServerMessage": "LoginRejected","Reason": "DatabaseError"} – if the gateway encounters an internal database error
- {"ServerMessage": "LoginRejected","Reason": "UnknownOrInactive"} – if the user name is not in the database or it was inactivated for some reason</td></tr>
</table>

2. Request device status

<COMMAND>	<PARAMETERS>
requeststat(<IMEI>,null)	- <IMEI> is the IMEI of the device you want to access for status request. Each status request will receive a message from the server followed by a device message after the device answers to the command. - “null” is a field reserved for future use. It must be sent to keep syntax compatibility.
INTERFACE possible answers	
- {"ServerMessage": "NACK","Reason": "SyntaxError"} – if your command could not be interpreted by the gateway - {"ServerMessage": "NACK","Reason": "DeviceNotConnected"} – the device with the specified IMEI is not connected to the gateway - {"ServerMessage": "NACK","Reason": "AccessDenied"} – you do not have access to the specified IMEI with the current user used in the previous HELLO command. - {"ServerMessage": "ACK","DeviceID": "866104024186606","CommandSent": "RequestStat"} - {"DeviceMessage": "Status","DeviceID": "866104024186606","CoinCounter": 1830,"BillCounter": 6800, "ChangeCounter": 1300,"CashlessStatus": 10,"CashlessStatusHuman": “EnabledByVMC","ScalingFactor": 1, "DecimalPlaces": 2, "AcceptRevalue": True, AccepCashSaleReporting": True, ”ButtonsCounters": [18,3], "AvailableChange": 1530}	
Device message answer variables: - <DeviceMessage> means that the message was originated by the device; - <DeviceID> is the device’s IMEI - <CoinCounter> is the accumulated value of the coins accepted while the interface was connected to the VMC - <BillCounter> is the accumulated value of the bill accepted while the interface was connected to the VMC - <ChangeCounter> is the accumulated value of the coins returned to the customers after transactions while the interface was connected to the VMC - <CashlessStatus> is the current MDB numeric device status, depending on last command received from VMC - <CashlessStatusHuman> is the current MDB human readable device status, depending on last command received from VMC - <ScalingFactor> is the current scaling factor set on the device - <DecimalPlaces> is the current number of decimals set on the device - <AcceptRevalue> can be True or False and depends on options byte set by <setoptions> command (see this command details). If the parameter is set to accept revalue, if there is an opened cashless session and the customers is inserting some cahe into the machine, the VMC will report this and it ask the accept. You are required to send a VendApproved or a VendDenied command for each revalue request (please see <vendapproved> and <venddenied> for details). Some machines requires special setting in the VMC controller to support this function. Please check your vending machine’s configuration manual. - <AccepCashSaleReporting> can be True or False and depends on options byte set by <setoptions> command. If it is True each cash sale is reported to the cashless interface. Some machines requires special settings in the VMC controller to support this function and some other machines are not able to send this information after cash sales. Please check your machine’s configuration manual. - <ButtonsCounters> - each device has 2 inputs that can be used to connect some buttons (connected buttons should be NO). For each button press the corresponding counter will be incremented by 1. Your application will assign a function to each button If needed (for example, Button #1 for cash collect and Button #2 for products refill). The counters can have a value from 0 to 255 and they will rollover to 0 anytime the a button I pressed and it’s corresponding counter value is 255. - <AvailableChange> is the sum value of the coins present in the coin changer, available to be ejected as change after the cash transactions. Due to the MDB limitations regarding the number of the coins reported for each denomination, some changers are reporting a maximum of 255 coins for each of the coins’ value and the physical coins stock could be bigger. For example, if the changer has a 300 pcs stock for 0.50EUR coins, it will always report 255. Also, some changers could be set to keep a security stock (minimum coins number for each tube). Depending on the coin changer settings/firmware, it will report a number of coins including this security stock or not.	

3. Request status for all connected devices

This command will send a requeststat message to any connected device which is attached to your user.

<COMMAND>	<PARAMETERS>
requeststatall(null)	- “null” is a field reserved for future use. It must be sent to keep syntax compatibility. The gateway will answer with the status message when this is transmitted by the device.
INTERFACE possible answers	
Status messages when received from the devices	

4. Begin a cashless session

This command will send a begin session message to the vending machine. As a result, a cashless credit will appear on the machine’s display.

<COMMAND>	<PARAMETERS>
beginsession(<IMEI>,<credit_value>)	- <IMEI> is the IMEI of the device you want receive the message and start a session. - <credit_value> is the desired credit you need to rise on the machine. This value is scaled by using the device scaling factor. Depending on the machine’s firmware, some VMCs are accepting a value of 65535 for <credit_value>. According to MDB specifications, it the transmitted credit value is 0xFFFF, it means that the credit could not be determined. The machine will only display a request for the customer, to select a product. Some machines are not supporting this feature. Also, if you set the machine and the device for multivend, this feature will not be supported and can lead you to undesired VMC behavior. Use this feature only with single vend. NOTE: Many machines are not able to receive this command repeatedly. If you don’t get a correct ACK from the device in a reasonable time, before issuing this command again, you need to use the <vendcancel> command and wait for an ACK for this. In other case, the machine will disable the cashless device because of an “out of sequence” situation.
INTERFACE possible answers	
- {"ServerMessage": "NACK","Reason": "SyntaxError"} - {"ServerMessage": "NACK","Reason": "DeviceNotConnected"} - {"ServerMessage": "NACK","Reason": "AccessDenied"} - {"ServerMessage": "NACK","Reason": "NonNumericCredit"} - {"ServerMessage": "ACK","DeviceID": "866104024186606","CommandSent": "BeginSession"} - {"DeviceMessage": "ACK","DeviceID": "866104024186606”}	

5. Vend approved

This command will send a “vend approved” message to the vending machine, which is a result of a “vend request” coming from it. As a result, machine should start the product delivery.

<COMMAND>	<PARAMETERS>
vendapproved(<IMEI>,<approved_value>)	- <IMEI> is the IMEI of the device you want receive the message and approve the previous vend request. - <approved_value> is the approved value for the requested product. Using this variable you can approve a product dispense with it’s regular price or with a discount, if the <approved_value> is lower than the product’s regular price. In the case of multi vend environments, the remaining cashless credit on the VMC will be the difference between the previous credit and the one you are sending in this message. Try to avoid sending a value bigger than the existing credit available on the machine. Some machines are not able to handle negative credit values.
INTERFACE possible answers	
- {"ServerMessage": "NACK","Reason": "SyntaxError"} - {"ServerMessage": "NACK","Reason": "DeviceNotConnected"} - {"ServerMessage": "NACK","Reason": "AccessDenied"} - {"ServerMessage": "NACK","Reason": "NonNumericValue"} - {"ServerMessage": "ACK","DeviceID": "866104024186606","CommandSent": "VendAppoved"} - {"DeviceMessage": "ACK","DeviceID": "866104024186606”}	

6. Vend denied

This command will send a “vend denied” message to the vending machine, which is a result of a “vend request” coming from it. As a result, machine will display a specific message about the impossibility to sell using the cashless system.

<COMMAND>	<PARAMETERS>
venddenied(<IMEI>,null)	- <IMEI> is the IMEI of the device you want to receive and reject the current vend request. - “null” is a field reserved for future use. It must be sent to keep syntax compatibility.
INTERFACE possible answers	
- {"ServerMessage": "NACK","Reason": "SyntaxError"} - {"ServerMessage": "NACK","Reason": "DeviceNotConnected"} - {"ServerMessage": "NACK","Reason": "AccessDenied"} - {"ServerMessage": "ACK","DeviceID": "866104024186606","CommandSent": "VendDenied"} - {"DeviceMessage": "ACK","DeviceID": "866104024186606”}	

7. Vend cancel

This command will send a “vend cancel” message to the vending machine. As a result, machine will end the current cashless session and will clear the cashless credit. If there is some cash credit on the machine and this was not approved for revalue, it will remain as credit on the machine’s display.

<COMMAND>	<PARAMETERS>
vendcancel(<IMEI>,null)	- <IMEI> is the IMEI of the device you want to cancel the session. - “null” is a field reserved for future use. It must be sent to keep syntax compatibility.
INTERFACE possible answers	
- {"ServerMessage": "NACK","Reason": "SyntaxError"} - {"ServerMessage": "NACK","Reason": "DeviceNotConnected"} - {"ServerMessage": "NACK","Reason": "AccessDenied"} - {"ServerMessage": "ACK","DeviceID": "866104024186606","CommandSent": "VendCancel"} - {"DeviceMessage": "ACK","DeviceID": "866104024186606”}	

8. Revalue approved

This command will send a “revalue approved” message to the vending machine in a response to a “revalue request” coming from it. As a result, machine will clear the current existing cash credit and it will consider that the entire credit is a cashless one. You have to add the entire amount in the “revalue request” to the customer’s cashless account. You can disable the revalue function if this is not necessary for your application (if the customer’s account cannot be recharged with cash on the vending machine) by modifying the options byte. Please see “setoptions” command for details.

<COMMAND>	<PARAMETERS>
revalueapproved(<IMEI>,null)	- <IMEI> is the IMEI of the device you want to approve the revalue action. - “null” is a field reserved for future use. It must be sent to keep syntax compatibility.
INTERFACE possible answers	
- {"ServerMessage": "NACK","Reason": "SyntaxError"} - {"ServerMessage": "NACK","Reason": "DeviceNotConnected"} - {"ServerMessage": "NACK","Reason": "AccessDenied"} - {"ServerMessage": "ACK","DeviceID": "866104024186606","CommandSent": "RevalueApproved"} - {"DeviceMessage": "ACK","DeviceID": "866104024186606”}	

9. Reset

This command will send a reset message to the device. The device will resume from the cashless stage 0 (“just reset”) and it will force the VMC to perform the entire cashless initialization procedure. Also, the GPRS communication will be disconnected and reconnected again within a maximum 2 minutes interval. Use this command after any options byte modifications, allowing the VMC to reload the cashless configuration.

<COMMAND>	<PARAMETERS>
reset(<IMEI>,null)	- <IMEI> is the IMEI of the device you want to reset. - “null” is a field reserved for future use. It must be sent to keep syntax compatibility.
INTERFACE possible answers	
- {"ServerMessage": "NACK","Reason": "SyntaxError"} - {"ServerMessage": "NACK","Reason": "DeviceNotConnected"} - {"ServerMessage": "NACK","Reason": "AccessDenied"} - {"ServerMessage": "ACK","DeviceID": "866104024186606","CommandSent": "Reset"} - {"DeviceMessage": "ACK","DeviceID": "866104024186606”}	

10. Reading device memory informations

This command will send a request to read a non-volatile memory area. In the non-volatile memory area there are stored some configuration informations and some counters (including products and cash counters).

<COMMAND>	<PARAMETERS>
readmemory(<IMEI>, <start_address>,<block_length>)	- <IMEI> is the IMEI of the device you want to access for memory read. - <start_address> is the memory location where the read operation will start. - <block_length> is the length of the memory block you need to read. The maximum block length is 896bytes. If you are trying to read a larger block, please split it in multiple read commands. Otherwise you can get reading errors. Same reading errors could be encountered if you are trying to read beyond the 1024 address, which is the upper limit of the non-volatile memory. NOTE: Please check the memory map table to find the exact location of each information.
INTERFACE possible answers	
- {"ServerMessage": "NACK","Reason": "SyntaxError"} - {"ServerMessage": "NACK","Reason": "DeviceNotConnected"} - {"ServerMessage": "NACK","Reason": "AccessDenied"} - {"ServerMessage": "NACK","Reason": "NonNumericValue"} - {"ServerMessage": "ACK","DeviceID": "866104024186606","CommandSent": "ReadMemory"} - {"DeviceMessage": "ReadMemory","DeviceID": "866104024186606","StartAddress": 4,"DataLength": 4,"Data": [0,0,0,150]}	

11. Memory map table

Address (decimal)	Size (bytes)	Description
0	4	Coin credit – the accumulated value of all coins inserted by the customers in the vending machine. B_0 is the most significant byte and B_3 is the least significant byte of this counter. The counter cannot be remotely cleared.
4	4	Change credit – the accumulated value of the coins ejected as change at the end of each transaction. B_4 is the most significant byte and B_7 is the least significant byte of this counter. The counter cannot be remotely cleared.
8	1	Feature level – is the feature level reported by the cashless device to the vending machine. This location must be 0x02 all the time.
9	2	Country code – is the country code reported by the cashless device to the vending machine. B_9 is the most significant byte and B_{10} is the least significant byte.
11	1	Scaling factor – is the scaling factor reported by the cashless device to the vending machine and used to calculate all the money related values (cashless credit, cash sales value, revalue and vend approved values, etc.)
12	1	Decimal places – is the number of cashless supported decimals,reported to the vending machine.
14	1	Options byte – contains the supported vend options reported by the cashless device to the vending machine. This byte include the following settings: - b_0 – if this bit is set, then the cashless will report to the vending machine that is able to support revalue. This option may need some specific settings on the VMC, too. - b_1 – if this bit is set, then the cashless will report to the vending machine that is able to support multivend transactions. This option may need some specific settings on the VMC, too. - b_2 – not used, keep this bit cleared (0)

Address (decimal)	Size (bytes)	Description
		- b_3 – if this bit is set, then the cashless device will report to the vending machine that is able to receive VEND/CASH SALE subcommand. Some machine are not able to send this subcommand or may require some specific settings on the VMC, too. Keep this bit set in order to receive cash sales informations (selection number and selection price).
15	1	Communication options – is a byte that defines the communication behavior. Please keep this byte with fixed value (0x03). If you modify this address, you may experience some communications troubles.
16	100	Server FQDN – this block keeps the FQDN of the gateway server. Please do not modify this address. If this memory block contains wrong informations, the device will not connect for data reporting and cashless functions.
116	5	Server port – this block keeps the listening port on the gateway server. Please do not modify this address. If this memory block contains wrong informations, the device will not connect for data reporting and cashless functions.
121	100	APN name – this block keeps the name of the APN used for your GPRS connection. If the APN name is not correct, then your device will not connect to the gateway. For this information, please contact your GSM operator.
221	100	APN username – this block keeps the APN username received from your GSM operator.
321	100	APN password – this block keeps the APN password received from your GSM operator.
421	400	Product counters – this bloc contains the sales counter for each selection available on your vending machine. The maximum number of supported products is 100.
828	1	Button #1 counter
829	1	Button #2 counter
830	4	Bill credit is the accumulated value of all bills inserted by the customers in the vending machine. B_{830} is the most significant byte and B_{833} is the least significant byte of this counter.
834	189	Not used in this version

12. Set options byte

Thgis command will modify the options byte and will enable/disable some cashless features. Please see the memory map table for details about options byte (address 14 – options byte)

<COMMAND>	<PARAMETERS>
setoptions(<IMEI>,<options_byte>)	- <IMEI> is the IMEI of the device you want to reset. - <options_byte> is the vend options byte. You can modify this vales according to the details you can find in the memory map table – address 14).
INTERFACE possible answers	
- {"ServerMessage": "NACK","Reason": "SyntaxError"} - {"ServerMessage": "NACK","Reason": "DeviceNotConnected"} - {"ServerMessage": "NACK","Reason": "AccessDenied"} - {"ServerMessage": "NACK","Reason": "NonNumericOptions"} - {"ServerMessage": "ACK","DeviceID": "866104024186606","CommandSent": "SetOptions"} - {"DeviceMessage": "ACK","DeviceID": "866104024186606"}	

III. Unsolicited messages

Those messages will be transmitted by the gateway server every time the vending machine is notifying the cashless device. Your server should be able to receive those messages at any moment and react on them accordingly.

1. ACK

This message is sent by the device after a command is successfully received by the vending machine and the VMC is accepting it as a valid command.

Message
{"DeviceMessage": "ACK","DeviceID": "866104024186606"}
Details
When the device is receiving a specific command for the vending machine, it will try to send it as soon as possible. If the vending machine is accepting it as a valid command, the device will send back this message.

2. NACK

This message is sent by the device if a command send to the vending machine fails.

Message
{"DeviceMessage": "NACK","DeviceID": "866104024186606"}
Details
When the device is receiving a specific command for the vending machine, it will try to send it as soon as possible. If the vending machine is not accepting it as a valid command, the device will send back this message.

3. Status message

The gateway server has it’s own keep-alive mechanism, running in background for keeping the device on-line and to avoid GPRS disconnection. The keep-alive message is sent by the gateway to the device every 10 minutes.

Message
{"DeviceMessage": "Status","DeviceID": "866104024186606","CoinCounter": 1830,"BillCounter": 6800, "ChangeCounter": 1300,"CashlessStatus": 10,"CashlessStatusHuman": “EnabledByVMC","ScalingFactor": 1, "DecimalPlaces": 2, "AcceptRevalue": True, AcceptCashSaleReporting": True, ”ButtonsCounters": [18,3], "AvailableChange": 1530}
Details
For each keep-alive message, the device will send back a status message. Please see the “request status” command for details about this message content.

4. Vend request

This message is sent by the device if there is a on opened cashless session and the customer is selecting a product. After receiving this message, you need to issue a “vend approved” or a “vend denied” command.

Message
{"DeviceMessage": "VendRequest","DeviceID": "866104024186606","ItemPrice": 100,"ItemNumber": 6}
Details
The machine is notifying the cashless that the customer has selected the item number 6 with a price of 100 (EUR 1.00)

5. Vend canceled by the VMC

VMC can send this message if there is a “cancel” button that will end the session

Message
{"DeviceMessage": "VendCancel","DeviceID": "866104024186606"}
Details
This message is a cancel request from the customer and it is signaling that the current session was canceled for the device with IMEI “866104024186606”

6. Vend success

VMC can send this message if it considers that the transaction was a success and it is able or it will be able to deliver the product. Please not that some machines are sending this at the delivery/preparation beginning and, sometimes the success message is false positive.

Message
{"DeviceMessage": "VendSuccess","DeviceID": “866104024186606”,"ItemNumber": 6}
Details
This message will notify about a successful vend for the item #6 on the device with IMEI “866104024186606”

7. Vend failure

VMC can send this message if it considers that the transaction was not a success

Message
{"DeviceMessage": "VendFailure","DeviceID": “866104024186606”}
Details
This message will notify about a failure during a vend on the device with IMEI “866104024186606”

8. Session complete

This message is sent by the device every time the VMC is issuing an “end session” command to the cashless system. After receiving this message, to perform a cashless vend, you need to open a new session with “beginsession” command.

Message
{"DeviceMessage": "SessionComplete","DeviceID": “866104024186606”}
Details
VMC has closed the current session on the device with IMEI “866104024186606”

9. Cash sale

If VMC has cash sales reporting to the cashless device capabilities, will send this command after each successful cash transaction.

Message
{"DeviceMessage": "CashSale","DeviceID": "866104024186606","ItemPrice": 100,"ItemNumber": 4}
Details
This message will notify about a successful cash vend transaction on the device with IMEI “866104024186606”, for the product #4 with a price of EUR 1.00

10. Cashless disabled by VMC

This message is sent by the device every time the machine has sent a “disable” command to the cashless device.

Message
{"DeviceMessage": "CashlessDisabled","DeviceID": "866104024186606"}
Details
This message will notify that the device with IMEI “866104024186606” was disabled by the vending machine

11. Cashless cancel

This message is sent by the device every time the machine has sent a “cancel” command to the cashless device.

Message
{"DeviceMessage": "CashlessVMCCancel","DeviceID": "866104024186606"}
Details
This message will notify that the device with IMEI “866104024186606” has received a cancel command from VMC

12. Cashless enabled by VMC

This message is sent by the device every time the machine has sent an “enable” command to the cashless device.

Message
{"DeviceMessage": "CashlessEnabled","DeviceID": "866104024186606"}
Details
This message will notify that the device with IMEI “866104024186606” was enabled by the vending machine

13. Revalue request

This message is sent by the device every time the machine has sent a “revalue request” command to the cashless device.

Message
{"DeviceMessage": "RevalueRequest","DeviceID": "866104024186606","Value": 100}
Details
This message will notify that the device with IMEI “866104024186606” has received a “revalue request” command from VMC with a value of EUR 1.00

14. Button pressed

This message is sent by the device every time one of it’s buttons are pressed.

Message
{"DeviceMessage": "ButtonPressed","DeviceID": "866104024186606","ButtonNumber": 1,"ButtonCounter": 17'}
Details
This message will notify that one button was pressed on the device with IMEI “866104024186606”

IV. Using Python 3 configuration tool

If you need to modify the device’s current configuration (for example, change the country code, scaling factor, server, port, APN, APN username, APN password, etc.) we offer a Python 3 configuration tool which, connected over the RS232 serial port, will allow you to modify or to verify the configuration, by writing and reading the non-volatile memory.

You need to pay attention to the following device memory map and avoid altering some critical areas that can put the device in an idle/out of order state. For the most important settings there are specific commands that are writing in the designated memory locations. Try to avoid using raw writemem command if there is an available specific command. Writing in a wrong memory location will put the device in an “out of order” state.

Address	Details
0x0000 → 0x0003	Cash credit – this is an accumulated cash credit obtained, by sniffing cash payment systems commuinication
0x0004 → 0x0007	Change value – this is an accumulated change value counter, by sniffing cash payment systems communication
0x0008	Feature level – this is the cashless feature level. This address must be always 0x02, otherwise some problems could appear in the communication with the vending machine
0x0009 → 0x000A	Country code in a Hi → Lo format
0x000B	Scaling factor – this is the cashless scaling factor reported to the machine. You will use this to calculate the value sent to the machine (usually 1, but you can change to other needed values)
0x000C	Decimal places – this is the cashless decimal places value reported to the machine. Usually 2, but you can change to other needed value
0x000D	Not used in this version – reserved for future use
0x000E	Options byte - check 11. Set options byte on low level protocol description for accepted values
0x000F	Comm options – this value is reserved for further function and must always have value 0x03. If you change to other value, the device will not connect to the server.
0x0010 → 0x0073	Server name – is the server FQDN. The device will try to connect to this server. If the server name is wrong or not configured, the device will not connect.
0x0074 → 0x0078	Server port – is the port on the server that device will try to connect over TCP SOCKET.
0x0079 → 0x00DC	APN name – is the name of the APN used for GPRS connection. Please contact your GSM/GPRS service provider in order to obtain the proper information about APN name, otherwise, the device will not connect to the server.
0x00DD → 0x0140	APN username – is the username used for the APN on the GPRS connection. Please contact your GSM/GPRS service provider in order to obtain the proper information about APN username, otherwise, the device will not connect to the server.
0x0141 → 0x01A4	APN password – is the password used for the APN on the GPRS connection. Please contact your GSM/GPRS service provider in order to obtain the proper information about APN password, otherwise, the device will not connect to the server.
0x01A5 → 0x04C5	Selection counters – is an area where the sales counters are stored, 4 bytes for each selection number, maximum 200 distinct selections per machine.
0x04C6	Button 1 counter – is the counter incremented once on each button press. This counter will rollover on 0xFF.
0x04C7	Button 2 counter – is the counter incremented once on each button press. This counter will rollover on 0xFF.
0x04C8 → 0x0F37	Not used by the application, available for user. You can use this memory to store some informations by using direct low level write memory command and use this informations by reading them by using direct low level read memory command. You can store here any personal informations your application may need (location ID, machine type, etc.).

Connect the device to the computer, using te RS232 cable. Using a terminal window start the Python 3 configuration tool. In other terminal window, telnet to localhost on port 5126.

You will see a message about the application version and then you will be able to send and of the below commands.

1. Setting country code

<COMMAND>	<PARAMETERS>
countrycode(“N”)	“N” will be the new country code reported by the device to the vending machines.
INTERFACE answer	
- {"VMCResponse" : "ACK"} - ACK – command success - {"VMCResponse" : "NACK"} – NACK – command failed	

2. Setting scaling factor

<COMMAND>	<PARAMETERS>
scalingfactor(“N”)	“N” will be the new scaling factor reported by the device to the vending machines.
INTERFACE answer	
- {"VMCResponse" : "ACK"} - ACK – command success - {"VMCResponse" : "NACK"} – NACK – command failed	

3. Setting options byte

<COMMAND>	<PARAMETERS>
setoptions(“N”)	“N” will be the new options byte. See 11. Set options byte on the low level protocol section of this manual, for more informations.
INTERFACE answer	
- {"VMCResponse" : "ACK"} - ACK – command success - {"VMCResponse" : "NACK"} – NACK – command failed	

4. Setting decimal places

<COMMAND>	<PARAMETERS>
decimalplaces(“N”)	“N” will be the new decimal places value reported by the device to the vending machines.
INTERFACE answer	
- {"VMCResponse" : "ACK"} - ACK – command success - {"VMCResponse" : "NACK"} – NACK – command failed	

5. Reading memory

<COMMAND>	<PARAMETERS>
readmem(<address>,<length>)	<address> will be the start address for reading and <length> is the data length to read. Maximum <length> for one reading is 128 (0x80)
INTERFACE answer	
- JSON with a data array - {"VMCResponse" : "NACK"} – NACK – command failed	

6. Writing memory

<COMMAND>	<PARAMETERS>
writemem(<address>,<data$_1$>, …, <data$_n$ >)	“<address> will be the start address for write. Data bytes are given in decimal, separated by “comma”. Maximum 32 bytes are allowed for one write operation.
INTERFACE answer	
- {"VMCResponse" : "ACK"} - ACK – command success - {"VMCResponse" : "NACK"} – NACK – command failed	

7. Reading server name

<COMMAND>	<PARAMETERS>
readserver	This command reads the current server FQDN configured into the device’s non-volatile memory
INTERFACE answer	
- JSON containing server FQDN - {"VMCResponse" : "NACK"} – NACK – command failed	

8. Reading server listening port

<COMMAND>	<PARAMETERS>
readport	This command reads the current server port configured into the device’s non-volatile memory
INTERFACE answer	
- JSON containing server listening port - {"VMCResponse" : "NACK"} – NACK – command failed	

9. Reading APN name

<COMMAND>	<PARAMETERS>
readapnname	This command reads the current APN name configured into the device’s non-volatile memory
INTERFACE answer	
- JSON containing APN name - {"VMCResponse" : "NACK"} – NACK – command failed	

10. Reading APN user name

<COMMAND>	<PARAMETERS>
readapnuser	This command reads the current APN user name configured into the device’s non-volatile memory
INTERFACE answer	
- JSON containing APN username - {"VMCResponse" : "NACK"} – NACK – command failed	

11. Reading APN password

<COMMAND>	<PARAMETERS>
readapnpass	This command reads the current APN password configured into the device's non-volatile memory
INTERFACE answer	
- JSON containing APN password - {"VMCResponse" : "NACK"} – NACK – command failed	

12. Writing server name

<COMMAND>	<PARAMETERS>
writeserver(<server_name>)	<server_name> is the server FQDN
INTERFACE answer	
- {"VMCResponse" : "ACK"} - ACK – command success - {"VMCResponse" : "NACK"} – NACK – command failed	

13. Writing server port

<COMMAND>	<PARAMETERS>
writeport(<server_port>)	<server_port> is the server listening port
INTERFACE answer	
- {"VMCResponse" : "ACK"} - ACK – command success - {"VMCResponse" : "NACK"} – NACK – command failed	

14. Writing APN name

<COMMAND>	<PARAMETERS>
writeapnname(<APN_name>)	<APN_name> is the APN name provided by your GSM service provider
INTERFACE answer	
- {"VMCResponse" : "ACK"} - ACK – command success - {"VMCResponse" : "NACK"} – NACK – command failed	

15. Writing APN user name

<COMMAND>	<PARAMETERS>
writeapnuser(<APN_user>)	<APN_user> is the APN user name provided by your GSM service provider.
INTERFACE answer	
- {"VMCResponse" : "ACK"} - ACK – command success - {"VMCResponse" : "NACK"} – NACK – command failed	

16. Writing APN user password

<COMMAND>	<PARAMETERS>
writeapnpass(<APN_pass>)	<APN_pass> is the APN password provided by your GSM service provider.
INTERFACE answer	
- {"VMCResponse" : "ACK"} - ACK – command success - {"VMCResponse" : "NACK"} – NACK – command failed	

17. Writing device communication options

<COMMAND>	<PARAMETERS>
writecommoptions("N")	"N" contains communication options. For the moment, keep this value as 0x03. Changing to other value may put the device into "out of order" state and prevent it to connect to the server.
INTERFACE answer	
- {"VMCResponse" : "ACK"} - ACK – command success - {"VMCResponse" : "NACK"} – NACK – command failed	