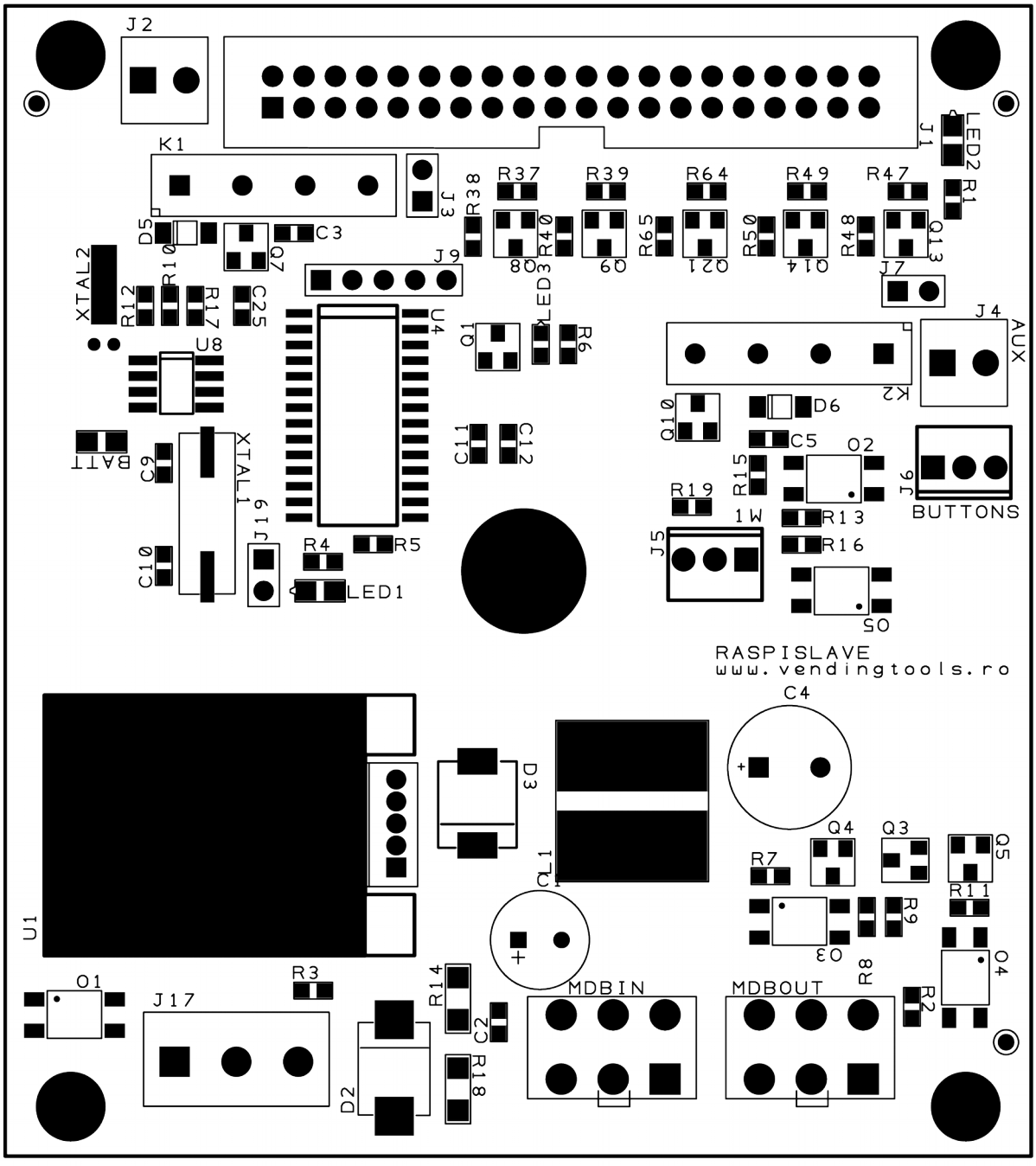
Interface MDB SLAVE to RASPBERRY Pi & Compatible SBC v11.10.2017 (with keyboard simulator optional module) Quick reference

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 at a maximum 37VDC. You have to make sure the the MDB bus can support

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.
- **<J1>** - Raspberry Pi connector – PIN #1 on this connector is the squared pin and must be connected to PIN #1 of Raspberry Pi.
- **<J2> -** 5VDC – same with Raspberry Pi power
- **<J4>** - AUX 5VDC controlled power for external 5VDC. This can be used to power Raspberry Pi peripherals that need power reset from time to time (3G modems, etc.)
- **<J6>** - Buttons connector
    - PIN #1 - squared pin – button #1
    - PIN #2 – GND – common
    - PIN #3 – button #2
- **<J17>** - External power connector
    - PIN #1 – squared pin - +24VDC
    - PIN #2 – GND
    - PIN #3 – POWER GOOD signal (+24VDC when power present).

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 (card payments, SMS payments, etc.). If you need to act as a bill validator or coin acceptor or other MDB slave device, please contact us for software development.
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 read by the MDBCashlessGetStat 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.

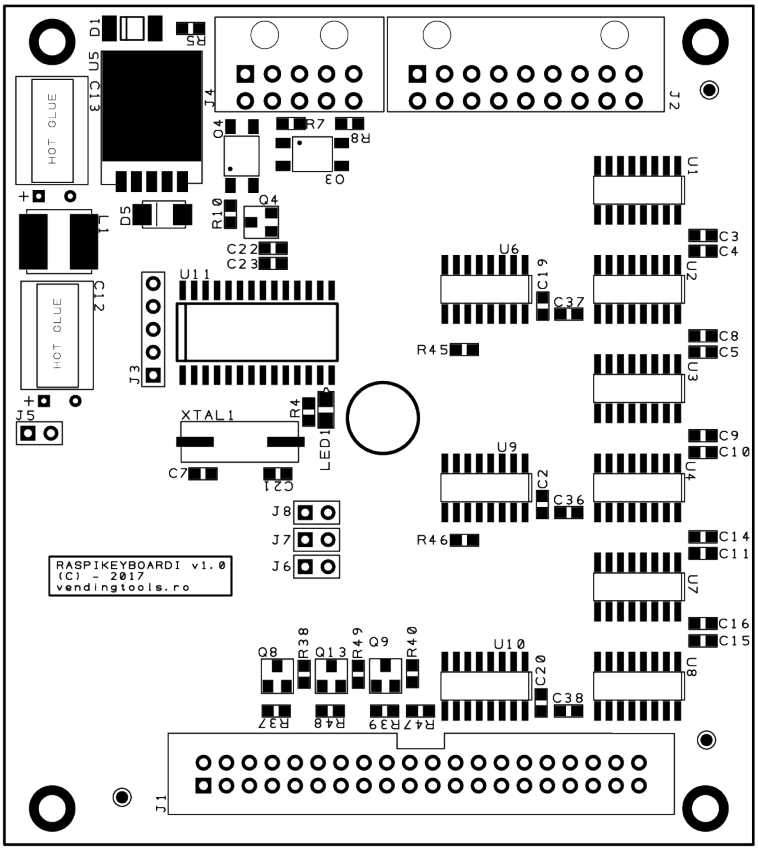
This interface is using a configurable scaling factor.
The interface is emulating a Level 2 cashless device.

4. GPIO map

Pin No.	Rpi function	RASPISLAVE	Pin No.	Rpi function	RASPISLAVE
1	3.3V	3.3V	2	5V	5V
3	GPIO2/SDA1/I2C	Not used	4	5V	5V
5	GPIO3/SCL1/I2C	Not used	6	GND	GND
7	GPIO4/GPCLK0	Not used	8	GPIO14/TXD	Serial TX
9	GND	GND	10	GPIO15/RXD	Serial RX
11	GPIO17	Not used	12	GPIO18/PCM_CLK	Not used
13	GPIO27	Not used	14	GND	GND
15	GPIO22	Not used	16	GPIO23	Not used
17	3.3V	3.3V	18	GPIO24	Not used
19	GPIO10/MOSI/SPI	KEYSIM*	20	GND	GND
21	GPIO9/MISO/SPI	KEYSIM*	22	GPIO25	Not used
23	GPIO11/SCLK/SPI	KEYSIM*	24	GPIO8/CE0/SPI	Not used
25	GND	GND	26	GPIO7/CE1/SPI	Not used
27	SDA0/I2C/ID EE	Not used	28	SCL0/I2C/IDEE	Not used
29	GPIO5/GPCLK1	Not used	30	GND	GND
31	GPIO6/GPCLK2	Power good	32	GPIO12/PWM0	Not used
33	GPIO13/PWM1	Not used	34	GND	GND
35	GPIO19/PCMFS/PWM1	Not used	36	GPIO16	MDB CTS
37	GPIO26	MDB RTS	38	GPIO20/PCMDIN	Not used
39	GND	GND	40	GPIO21/PCMDOUT	Not used

* KEYSIM pins are used only if the keyboard simulator board is connected to the system. Otherwise, those pins can be used by user's application. You need to manually disable those pins in the demo application if you want to use those pins for other purposes.

5. Optional keyboard simulator board



J1 – Raspberry PI connector

J2 – Matrix keyboard connector (pin 1 to 6 = columns 1 to 6 and pin 7 to 13 = rows from 1 to 7)

6. Keyboard registers truth table (with optional keyboard simulator board)

Col	Row	Register 2	Register 1	Register 0
1	1	0b00001000	0b00000000	0b00000000
1	2	0b00010000	0b00000000	0b00000000
1	3	0b00011000	0b00000000	0b00000000
1	4	0b00100000	0b00000000	0b00000000
1	5	0b00101000	0b00000000	0b00000000
1	6	0b00110000	0b00000000	0b00000000
1	7	0b00111000	0b00000000	0b00000000
2	1	0b00000001	0b00000000	0b00000000
2	2	0b00000010	0b00000000	0b00000000
2	3	0b00000011	0b00000000	0b00000000
2	4	0b00000100	0b00000000	0b00000000
2	5	0b00000101	0b00000000	0b00000000
2	6	0b00000110	0b00000000	0b00000000
2	7	0b00000111	0b00000000	0b00000000
3	1	0b00000000	0b00001000	0b00000000
3	2	0b00000000	0b00010000	0b00000000
3	3	0b00000000	0b00011000	0b00000000
3	4	0b00000000	0b00100000	0b00000000
3	5	0b00000000	0b00101000	0b00000000
3	6	0b00000000	0b00110000	0b00000000
3	7	0b00000000	0b00111000	0b00000000
4	1	0b00000000	0b00000001	0b00000000
4	2	0b00000000	0b00000010	0b00000000
4	3	0b00000000	0b00000011	0b00000000
4	4	0b00000000	0b00000100	0b00000000
4	5	0b00000000	0b00000101	0b00000000
4	6	0b00000000	0b00000110	0b00000000
4	7	0b00000000	0b00000111	0b00000000
5	1	0b00000000	0b00000000	0b00001000
5	2	0b00000000	0b00000000	0b00010000
5	3	0b00000000	0b00000000	0b00011000
5	4	0b00000000	0b00000000	0b00100000
5	5	0b00000000	0b00000000	0b00101000
5	6	0b00000000	0b00000000	0b00110000
5	7	0b00000000	0b00000000	0b00111000
6	1	0b00000000	0b00000000	0b00000001
6	2	0b00000000	0b00000000	0b00000010
6	3	0b00000000	0b00000000	0b00000011
6	4	0b00000000	0b00000000	0b00000100
6	5	0b00000000	0b00000000	0b00000101
6	6	0b00000000	0b00000000	0b00000110
6	7	0b00000000	0b00000000	0b00000111

II. Serial communication protocol

Scenario #1 - If you are using Raspbian on Raspberry Pi 3 and you don't need Bluetooth, you have to redirect the /dev/ttyAMA0 serial port back to the 40pin connector, using the following procedure:

Using your favorite text editor, open /boot/config.txt and add the following 2 lines:

```
dtoverlay=pi3-disable-bt
core_freq=250
enable_uart=1
```

Using your favorite text editor, open /boot/cmdline.txt, search for the following settings and delete them:

```
console=serial0,115200
console=ttyAMA0,115200
```

Reboot your Pi.
You will not able to use RPi Bluetooth with those settings.

Scenario #2 – If you are using Raspbian on Raspberry Pi 3 and you need also to use Bluetooth.

Using your favorite text editor, open /boot/config.txt and add the following 2 lines:

```
#dtoverlay=pi3-disable-bt
core_freq=250
enable_uart=1
```

Using your favorite text editor, open /boot/cmdline.txt, search for the following settings and delete them:

```
console=serial0,115200
console=ttyAMA0,115200
```

Reboot your Pi.
Modify the main .py demo file search for /dev/ttyAMA0 and change to /dev/serial0.
You will be able to use both Bluetooth and RASPISLAVE board.
The interface has a low level protocol and we are also offering a small daemon application (written with Python 3 and tested on Raspberry Pi 2 and 3 with PySerial versions 3.0.1, 3.2.4 and 3.3). You can use this daemon to connect your application simply by using a socket or as an example for your low level development, if you need to directly send commands to the serial port.
You must set the serial interface with the following parameters:
- baudrate – 115200bps;
- data bits – 8;
- stop bits – 1;
- parity – NONE;
- flow control – RTS/CTS.

ATENTION!!! - MDB CRC calculation
MDB CRC is the CRC calculated according to MDB specifications, by cumulating the value of messages byte from the first one to the last one and taking the low byte of the sum. For

example, if you send and MDBCashlessBeginSession to the interface, with a value of EUR1.50 (decimal 150) with media ID 0xF0F0F0F0:

- 0x03 0x00 0x96 0xF0 0xF0 0xF0 0xF0 0x00 – the CRC is calculated by the following formula:
- SUM = 0x03 + 0x00 + 0x96 + 0xF0 + 0xF0 + 0xF0 + 0xF0 + 0x00 = 0x0459
- CRC = LO(SUM) = 0x59

III. Low level communication protocol using direct serial connection

1. Cashless begin session

<COMMAND>	<PARAMETERS>
0x03	- Byte 1 – scaled value of available funds (hi) - Byte 2 – scaled value of available funds (lo) - Byte 3 → Byte 6 – media ID - Byte 7 – Payment type (according to MDB specifications) - Byte 8 → Byte 9 - Payment data (according to MDB specifications) - Byte 10 – MDB CRC
INTERFACE answer	
- 0xFC 0xFC 0xFC 0xFC 0xFC 0xFC - ACK – command success - 0xFD 0xFD 0xFD 0xFD 0xFD 0xFD – NACK – command failed	

2. Cashless cancel session

<COMMAND>	<PARAMETERS>
0x04	- Byte 1 – MDB CRC – always 0x04 for this command
INTERFACE answer	
- 0xFC 0xFC 0xFC 0xFC 0xFC 0xFC - ACK – command success - 0xFD 0xFD 0xFD 0xFD 0xFD 0xFD – NACK – command failed	

3. Vend denied

<COMMAND>	<PARAMETERS>
0x06	- Byte 1 – MDB CRC – always 0x06 for this command
INTERFACE answer	
- 0xFC 0xFC 0xFC 0xFC 0xFC 0xFC - ACK – command success - 0xFD 0xFD 0xFD 0xFD 0xFD 0xFD – NACK – command failed	

4. Vend approved

<COMMAND>	<PARAMETERS>
0x05	- Byte 1 – scaled value of the product price (hi) - Byte 2 – scaled value of the product price (lo) - Byte 3 – MDB CRC
INTERFACE answer	
- 0xFC 0xFC 0xFC 0xFC 0xFC 0xFC - ACK – command success - 0xFD 0xFD 0xFD 0xFD 0xFD 0xFD – NACK – command failed	

5. Revalue approved

<COMMAND>	<PARAMETERS>
0x0D	- Byte 1 – MDB CRC (0x0D)
INTERFACE answer	
- 0xFC 0xFC 0xFC 0xFC 0xFC 0xFC - ACK – command success - 0xFD 0xFD 0xFD 0xFD 0xFD 0xFD – NACK – command failed	

6. Revalue denied

<COMMAND>	<PARAMETERS>
0x0E	- Byte 1 – MDB CRC (0x0E)
INTERFACE answer	
- 0xFC 0xFC 0xFC 0xFC 0xFC 0xFC - ACK – command success - 0xFD 0xFD 0xFD 0xFD 0xFD 0xFD – NACK – command failed	

7. Get status

<COMMAND>	<PARAMETERS>
0xFB	Byte 1 – MDB CRC – always 0xFB for this command
INTERFACE answer	
- Byte 1 → Byte 4 – cash counter (non-volatile lifetime accumulated cash) - Byte 5 → Byte 8 – change counter (non-volatile lifetime accumulated change) - Byte 9 – cashless stage with one of the following values: - 0x00 – beginning - 0x01 – setup config data - 0x02 – max/min prices - 0x03 – just reset - 0x04 – vend request - 0x05 – vend cancel - 0x06 - vend success - 0x07 – vend failure - 0x08 – session complete - 0x09 – disabled - 0x0A – enabled - 0x0B – cancel from VMC - 0x0C – entry request - 0x0D – expansion request ID - 0x0E – revalue request - 0x0F – revalue limit request - Byte 10 – cashless current settings - scaling factor - Byte 11 – cashless current settings – decimal places - Byte 12 → Byte 13 – cashless current settings – country code - Byte 14 – cashless current settings – options byte - b0 – if set – the interface is reporting to the vending machine that it supports revalue options (recharge customer's account if a session is open and some cash is inserted by the customer) - b1 – not used - always 0 - b2 – not used – always 0 - b3 – if set – the interface accepts cash vending reporting from the vending machine (each coin/bill cashed from the customers is reported to the interface) - b4-b7 – not used – always 0 - Byte 15 – Button #1 counter (min. 0 – max 255 with rollover on 255) - Byte 16 – Button #2 counter (min. 0 – max 255 with rollover on 255) - Byte 17 → Byte 20 – available change in the coin changer (value). The interface captures the tube status messages from VMC to the coin changer and extracts the total value of the coins in the coin changer. - Byte 21 – MDB CRC	

8. Interface reset

<COMMAND>	<PARAMETERS>
0xFE	- Byte 1 – 0x01 - byte 2 – MDB CRC – always 0xFF for this command – will put the interface in 0 stage mode and force the VMC to initialize it again.
INTERFACE answer	
- 0xFC 0xFC 0xFC 0xFC 0xFC 0xFC - ACK – command success - 0xFD 0xFD 0xFD 0xFD 0xFD 0xFD – NACK – command failed	

9. Set country code

<COMMAND>	<PARAMETERS>
0xFE	- Byte 1 – 0x02 - Byte 2 – country code hi byte - Byte 3 – country code lo byte - Byte 4 – MDB CRC Requires Interface reset command after modifying country code
INTERFACE answer	
- 0xFC 0xFC 0xFC 0xFC 0xFC 0xFC - ACK – command success - 0xFD 0xFD 0xFD 0xFD 0xFD 0xFD – NACK – command failed	

10. Set scaling factor

<COMMAND>	<PARAMETERS>
0xFE	- Byte 1 – 0x03 - Byte 2 – scaling factor (usualy 10, but if your application requires 0.01 resolution, then you can set this parameter to 1 - Byte 3 – MDB CRC Requires Interface reset command after modifying scaling factor
INTERFACE answer	
- 0xFC 0xFC 0xFC 0xFC 0xFC 0xFC - ACK – command success - 0xFD 0xFD 0xFD 0xFD 0xFD 0xFD – NACK – command failed	

11. Set options byte

<COMMAND>	<PARAMETERS>
0xFE	- Byte 1 – 0x04 - Byte 2 – options byte with the following bit values • bit 0 – if set (1) then the interface will accept revalue instructions from the vending machine and send an unsolicited message with the revalue amount, for every coin or bill inserted by the customer while a session is opened. This depends, also, by the VMC settings and features. • bit 1 – reserved, must be 0 • bit 2 – reserved, must be 0 • bit 3 – if set (1) then the interface will report any cash sale (price and product number, by an unsolicited message). This depends, also, by the VMC settings and features. • Bit 4-7 – reserved, must be 0. - Byte 3 – MDB CRC Requires Interface reset command after modifying options byte
INTERFACE answer	
- 0xFC 0xFC 0xFC 0xFC 0xFC 0xFC - ACK – command success - 0xFD 0xFD 0xFD 0xFD 0xFD 0xFD – NACK – command failed	

12. Set decimal places

<COMMAND>	<PARAMETERS>
0xFE	- Byte 1 – 0x05 - Byte 2 – decimal places – it must match the settings on the vending machines - Byte 3 – MDB CRC Requires Interface reset command after modifying decimal places
INTERFACE answer	
- 0xFC 0xFC 0xFC 0xFC 0xFC 0xFC - ACK – command success - 0xFD 0xFD 0xFD 0xFD 0xFD 0xFD – NACK – command failed	

13. Set AUX 5VDC power ON/OFF

<COMMAND>	<PARAMETERS>
0xFE	- Byte 1 – 0xFE - Byte 2 - 0x07 - Byte 3 – AUX relay power mode (0x01 – relay is energized and the AUX power is active, 0x00 – relay is not energized and the AUX power is OFF) - Byte 4 – MDB CRC
INTERFACE answer	
- 0xFC 0xFC 0xFC 0xFC 0xFC 0xFC - ACK – command success - 0xFD 0xFD 0xFD 0xFD 0xFD 0xFD – NACK – command failed	

14. Set RTC (date and time on the board)

<COMMAND>	<PARAMETERS>
0xFE	- Byte 1 – 0xFE - Byte 2 - 0x06 - Byte 3 – hour (number – 0 to 23) - Byte 4 – minute (number – 0 to 59) - Byte 5 – second (number – 0 to 59) - Byte 6 – day (number – 1 to 31) - Byte 7 – month (number – 1 to 10) - Byte 8 – year (number – 0 to 99) - Byte 9 – day of the week (number 1 to 7) - Byte 10 – MDB CRC
INTERFACE answer	
- 0xFC 0xFC 0xFC 0xFC 0xFC 0xFC - ACK – command success - 0xFD 0xFD 0xFD 0xFD 0xFD 0xFD – NACK – command failed	

15. Get RTC (date and time from the board)

<COMMAND>	<PARAMETERS>
0xFE	- Byte 1 – 0xFE - Byte 2 - 0x0A - Byte 3 – MDB CRC
INTERFACE answer	
- Byte 1 – 0xFE - Byte 2 – 0x0A - Byte 3 – hour (number – 0 to 23) - Byte 4 – minute (number – 0 to 59) - Byte 5 – second (number – 0 to 59) - Byte 6 – day (number – 1 to 31) - Byte 7 – month (number – 1 to 10) - Byte 8 – year (number – 0 to 99) - Byte 9 – day of the week (number 1 to 7) - Byte 3 – MDB CRC	

16. Unsolicited message – cash received

<COMMAND>	<PARAMETERS>
None	None – this message comes anytime the VMC is receiving some cash
INTERFACE answer	
- Byte 0 – 0xFA - Byte 1 → Byte 4 – scaled value of received cash - Byte 5 – MDB CRC (bytes XOR from B0 to B4)	

17. Unsolicited message – cashless status

<COMMAND>	<PARAMETERS>
None	None – this message comes anytime the VMC is receiving some cash
INTERFACE answer	

- Byte 0 – 0x14
- Byte 1 – cashless status, with the following values:
  - 0x00 – cashless disabled
  - 0x01 – cashless enabled
  - 0x02 – cashless cancel
- Byte 2 – MDB CRC	

18. Unsolicited message – vend

<COMMAND>	<PARAMETERS>
None	None – this message comes anytime the VMC is receiving some cash
INTERFACE answer	

- Byte 0 – 0x13 - always
  - if Byte 1 == 0x00
  - Byte 2 → Byte 3 – scaled value of received cashless
  - Byte 4 → Byte 5 – item number (selection/line number)
  - Byte 6 – MDB CRC
  - if Byte 1 == 0x01 – vend cancel
  - Byte 2 – MDB CRC
  - if Byte 1 == 0x02 – vend success
  - Byte 2 – Byte 3 – Item ID
  - Byte 3 – MDB CRC
  - if Byte 1 == 0x03 – vend failure (the VMC could not deliver the product)
  - Byte 2 – MDB CRC
  - if Byte 1 == 0x04 – session complete
  - Byte 2 – MDB CRC
  - if Byte 1 == 0x05 – cash sale reported to cashless device
  - Byte 2 → Byte 3 – scaled value of received cashless
  - Byte 4 → Byte 5 – item number (selection/line number)
  - Byte 6 – MDB CRC	

19. Unsolicited message – revalue

<COMMAND>	<PARAMETERS>
None	None – this message comes anytime the VMC is receiving some cash and a session is opened. The cash amount will be reported and added to the total amount available on cashless support.
INTERFACE answer	
- Byte 0 – 0x15 - Byte 1 – 0x00 - Byte 2 – Scaled revalue amount (HI) - Byte 3 – Scaled revalue amount (LO) - Byte 4 – MDB CRC	

20. Unsolicited message – cash sale

<COMMAND>	<PARAMETERS>
None	None – this message comes anytime the VMC successfully dispensed a product using cash only.
INTERFACE answer	
- Byte 0 – 0x13 - Byte 1 – 0x05 - Byte 2 – Scaled product's price (HI) - Byte 3 – Scaled product's price (LO) - Byte 4 – Product number (HI) - Byte 5 – Product number (LO) - Byte 4 – MDB CRC	

21. Unsolicited message – button pressed

<COMMAND>	<PARAMETERS>
None	None – this message comes anytime one of the buttons on the interface is pressed. There could be also other switch (for example, door switch)
INTERFACE answer	
- Byte 0 – 0xFE - Byte 1 – 0x08 - Byte 2 – button number (0x01 for button 1 and 0x02 for button 2) – the last pressed button number - Byte 3 – button number counter (for each button there is an internal button counter – 0 at boot that is incremented on each button press and which will rollover to 0 when it reaches value 255). Keeping this value on your application you may know how many times the button were pressed since the last message. - Byte 4 – MDB CRC	

22. Unsolicited message – power status

<COMMAND>	<PARAMETERS>
None	None – this message comes only if you power the interface from external power supply with external battery backup that can send a power good signal. Also, the device will set Raspberry PI GPIO to low (0) in the case of power fail and to high (1) in the case the power is OK. You can use this signal to nicely shutdown the Raspberry PI on power failures, to avoid filesystem corruption.
INTERFACE answer	
- Byte 0 – 0xFE - Byte 1 – 0x09 - Byte 2 – Power status (only with external power supply, not for MDB powering) – 0x00 – power failure, 0x01 – power OK. - Byte 4 – MDB CRC	

IV. High level protocol, using Python 3 daemon

To use this mode, you need the following:

- install Python 3 on your Raspberry Pi (for Raspbian “sudo apt-get install python3”);
- install pip3 (for Raspbian “sudo apt-get install python3-pip”);
- install PySerial (“sudo pip3 install pyserial==3.0.1”);
- download and run the Python script from our website.
- open a new console and run telnet on localhost, port 5126 (“sudo telnet localhost 5126”)
- in the telnet window start sending commands to the device.

1. MDBCashlessBeginSession(AAA,BBB,CCC,DDD)

The command, sent on the socket, has the following syntax:

- **MDBCashlessBeginSession(AAA,BBB,CCC,DDD)**, where AAA is the value of the credit you need to send to MDB machine, scaled by scaling factor, BBB is the media ID (card ID an integer positive value), CCC is type ID (according to MDB speciffications, usually 0) and DDD is payment data (according to MDB specifications, usually 0). For example, if you need to send 1EUR, the AAA should be 100 and if you want to send 3.20EUR, you must set AAA to 320. This command is corresponding to the MDB low level message “Begin session”.

The daemon will respond with:

{"MDBCashlessBeginSession": "XXXX"} where XXXX should match NNNN value you have sent to the interface. If XXXX has a value of -1 (negative one), then there was an error in the communication between the interface end the vending machine. You must issue a cancel session command and retry.

2. MDBCashlessCancelSession

The command, sent on the socket, has the following syntax:

- **MDBCashlessCancelSession.** This command will reset the cashless session and the cashless interface will come back to idle mode. This command is corresponding to the MDB low level message “Session cancel request”.

The daemon will respond with:

{"MDBCashlessCancelSession": "0"} – for succes or

{"MDBCashlessCancelSession": "-1"} – for failure

3. MDBCashlessVendDenied

The command, sent on the socket, has the following syntax:

- **MDBCashlessVendDenied.** This command will deny cashless vend request that comes from the VMC. This command is corresponding to the MDB low level message “Vend denied”.

The daemon will respond with:

{"MDBCashlessVendDenied": "0"} – for success or

{"MDBCashlessVendDenied": "-1"} – for failure

4. MDBCashlessVendApproved(NNNN)

The command, sent on the socket, has the following syntax:

- **MDBCashlessVendApproved(NNNN).** This command will approve the vend for the previous vend request command issued by the VMC. The NNNN value has statistical functions, for VMC controller and you have to issue the right amount in response to VMC request. It is also scaled like on MDBCashlessBeginSession command. This command is corresponding to the MDB low level message “Vend approved”.

The daemon can respond with one or more messages:

{"VMCResponse" : "VendSuccess", "ProductID" : "10"} – this answer means that VMC succesfuly sold the product and you can substract the value in the vend request message from your customer’s account.

{"VMCResponse" : "SessionComplete"} – this answer is not solicited and can be sent by VMC or not, depending on it’s settings and/or software implementation.

5. MDBCashlessRevalueApproved

The command, sent on the socket, has the following syntax:

- **MDBCashlessRevalueApproved.** This command will confirm the revalue request that comes from the VMC. This command is corresponding to the MDB low level message “Revalue approved”.

The daemon will respond with:

{"MDBCashlessRevalueApproved": "0"} – for success or

{"MDBCashlessRevalueApproved": "-1"} – for failure

6. MDBCashlessRevalueDenied

The command, sent on the socket, has the following syntax:

- **MDBCashlessRevalueDenied.** This command will deny the revalue request that comes from the VMC. This command is corresponding to the MDB low level message “Revalue denied”.

The daemon will respond with:

{"MDBCashlessRevalueDenied": "0"} – for success or

{"MDBCashlessRevalueDenied": "-1"} – for failure

7. MDBCashlessGetStat

The command, sent on the socket, has the following syntax:

- **MDBCashlessGetStat.** This command will return some internal informations from the MDB-RS232 interface and has no corresponding low level MDB command.

Daemon response example:

{"MDBCashlessStatus":"10","MDBCashlessHuman":"Enabled by VMC","MDBCashSniff":"4120","MDBChangeSniff":"440","MDBCashlessScalingFactor":"1","MDBCashlessDecimalPlaces":"2","MDBCashlessCountryCode":"040","MDBCashlessCashSaleSubcmd":"True","MDBCashlessAcceptRevalue":"True","MDBAvailableChange":"0"}

Where:
- MDBCashlessStatus is a numeric value corresponding to the interface status;

- MDBCashlessHuman is a human readable information about the interface current status;

- MDBCashSniff is the total value of accumulated cash inserted into the payment systems, obtained by sniffing the MDB communication. This is non-resettable, it is the value since the first power-up of the interface and is available only for read, scaled by the scaling factor (for this example, total cash from the first power-up is 11.00;

- MDBChangeSniff is the total value of the change returned by the vending machine to the customers, also read-only and non-resettable, scaled by the scaling factor (for this example, the total change from the first power-up is 2.00.

- MDBAvailableChange is the total value of the coins available for change in the coin changer, if the VMC is requesting the tube status

8. MDBCashlessReset

The command, sent on the socket, has the following syntax:

- **MDBCashlessReset.** This command will put the cashless interface in a stage available on the power-up. The VMC will initialize the cashless again.

The daemon will respond with:

{"MDBCashlessReset" : “0"} – for success or

{"MDBCashlessReset" : “-1"} – for failure.

9. MDBCashlessSetCountryCode(N)

This command will set the country code according to MDB ISO. The response could be
{"MDBCashlessSetCountryCode": "N"} – for success

{"MDBCashlessSetCountryCode": "-1"} – for failure

10. MDBCashlessSetScalingFactor(N)

This command will set the scaling factor for cashless transactions. The response could be
{"MDBCashlessSetScaligFactor": "N"} – for success

{"MDBCashlessSetScaligFactor": "-1"} – for failure

This command needs an MDBCashlessReset command to be sent after a a success response. This will force the VMC to initialize the interface and set the new scaling factor according to the new settings. If you are not issuing a reset command, the credit and the prices will not be correct.

11. MDBCashlessSetOptions(N)

This command will set the scaling vend options. The response could be:
{"MDBCashlessSetOptions": "N"} – for success

{"MDBCashlessSetOptions": "-1"} – for failure

See the low level command “set options byte” on page 7 for details about corect values for “N”. Please be careful, since wrong options setting can have unexpected results (for example, not accepting revalue or not reporting cash sales). Also, for cash sales reporting and revalue option, the VMC may need extra settings on it’s configuration menu

This command needs an MDBCashlessReset command to be sent after a a success response. This will force the VMC to initialize the interface and set the new scaling factor according to the new settings. If you are not issuing a reset command, the credit and the prices will not be correct.

12. MDBCashlessSetDecimalPlaces(N)

This command will set the decimal places options. The response could be:
{"MDBCashlessSetOptions": "N"} – for success

{"MDBCashlessSetOptions": "-1"} – for failure

This command needs an MDBCashlessReset command to be sent after a a success response. This will force the VMC to initialize the interface and set the new scaling factor according to the new settings. If you are not issuing a reset command, the credit and the prices will not be correct.

13. MDBSetRTC(h,m,s,d,M,y,wd)

This command will set the RTC on the board that will keep time for situations when Raspberry Pi has no connectivity .

Parameters are:

- h – hour (0 – 23)
- m – minute (0 – 59)
- s – second (0 – 59)
- d – day in month (1 – 31)
- M – month (1 – 12)
- y – year (0 – 99)
- wd – day of week (1 – 7)

The response could be:
{"MDBSetRTC": "0"} – for success

{"MDBSetRTC": "-1"} – for failure

14. MDBGetRTC

This command will get the RTC from the board that will keep time for situations when Raspberry Pi has no connectivity .

The response is:

{"MDBGetRTC":"RTC","Hour":"17","Minute":"38","Second":"41","Day":"24","Month":"3", "Year":"17","WeekDay":"5"'

15. MDBSetAuxPower(N)

This command will turn ON/OFF the AUX 5VDC output on J4. The response could be:
{"MDBCashlessSetOptions": "N"} – for success

{"MDBCashlessSetOptions": "-1"} – for failure

If you set the value of “N” to 1, then the AUX 5VDC output will be enabled. If you set this value to 0, then the output will be disabled.

16. MDBKeyPress(column,row,delay)

This command will turn “press” the button on column <column> and on row <row> for a <delay> time, where delay is represented in seconds and can be below 1 (for example, 0.1 means 100ms)
{"DeviceMessage": "KeypressSent","Column": <column>,"Row": <row>} – for success

{"MDBKeyPress": "-1"} – for failure

17. Bye

The command, sent on the socket, has the following syntax:

- **Bye –** This command will stop the daemon, and this will terminate with exitcode 0.

There is no daemon answer for this command, the socket will be closed and the daemon will end it’s execution.

No other connections can be made on

17. Unsolicited messages

The daemon can send one or more of this messages, depending on machine’s configuration and/or software implementation:

{"CashInput" : "900"} – the value of the general cash counter. It is incremented and it will be sent on each cash input (coin and bill)

{"CashSale" : "Success", "ProductID" : "10","ProductPrice" : "100"} – this message will be received if the machine can report cash sales to a cashless device. This message will be sent on each cash sale, with the ID and the value of the selected product. The value is also scaled by scaling factor.

{"CashlessStatus" : "Enabled"} – this message will be received if the machine is sending “enable” command to the interface. Any command related to a session will be accepted by the VMC only when the interface was enabled.

{"CashlessStatus" : "Disabled"} – this message will be received if the machine is sending “disable” command to the interface. Any command related to a session will be ignored by the VMC after disabling the interface.

{"CashlessStatus" : "Cancel"} – this message will be received if the machine is sending “cancel” command to the interface. This means that the machine ends the transaction and

therefore, the interface has closed the current cashless session. Another session is required to rise a credit to the machine.

{"CasshlessOperation" : "CashlessRevalue", "Value" : "500"} – this message is sent by the interface every time a coin or a bill is accepted by the vending machine and a cashless session is opened. This can be used to recharge the cashless account. “Value” is the value of the last coin or bill inserted by the customer. When you want to stop recharging, simply close the session. The user’s application is responsible to accumulate and calculate the total credit as a result of a revalue transaction.

{"IOEvent":"Button","ButtonNumber":"1","ButtonCounter":"18"} – this message is sent by the interface every time a button is pressed. <ButtonNumber> can have values of 1 or 2 and <ButtonCounter> from 1 to 255, indicating how many times the button was pressed. The counter will be reset on every reboot.

{"IOEvent":"PowerStatus","Status":"PowerFailure”} - this message is sent by the interface every time the power failed (needs external power supply with positive power good signal).

{"IOEvent":"PowerStatus","Status":"PowerOK”} - this message is sent by the interface every time the power is restored (needs external power supply with positive power good signal)

NOTES: