

PICOVEND
Executive/MDB bridge

v11.11.2019

Table of Contents

I. Device overview.....	3
II. Hardware overview.....	4
1. Motherboard.....	4
2. Connector description.....	4
3. Power supply requirements.....	4
III. Low level communication protocol.....	5
A. Serial port settings.....	5
B. Low level protocol messages.....	5
1. MDB messages.....	6
2. Executive macros.....	7
Notes:.....	10

I. Device overview

This device was designed to offer a quick development support for cashless systems on Executive vending machines and using MDB payment systems. It acts like a bridge between any Executive machine and MDB payment systems (bill validator, coin acceptor/changer and a cashless device). The device is intended to handle all MDB and Executive low level tasks that are heavy or impossible to handle in a buffered serial communication (usually available on PC, Raspbery Pi or compatible SBCs and PLCs). The device handles 9bit to 8bit conversion, CRC calculation and ACK/NAK sending to MDB in a timely manner.

IMPORTANT!!! - The machine must be set on Executive, either price holding mode (prices are kept on the device not on the machine) or with prices on the machine. When the prices are held on the machine, you will not be able to obtain a total value of sold products, unless your application keeps the prices for each item, too.

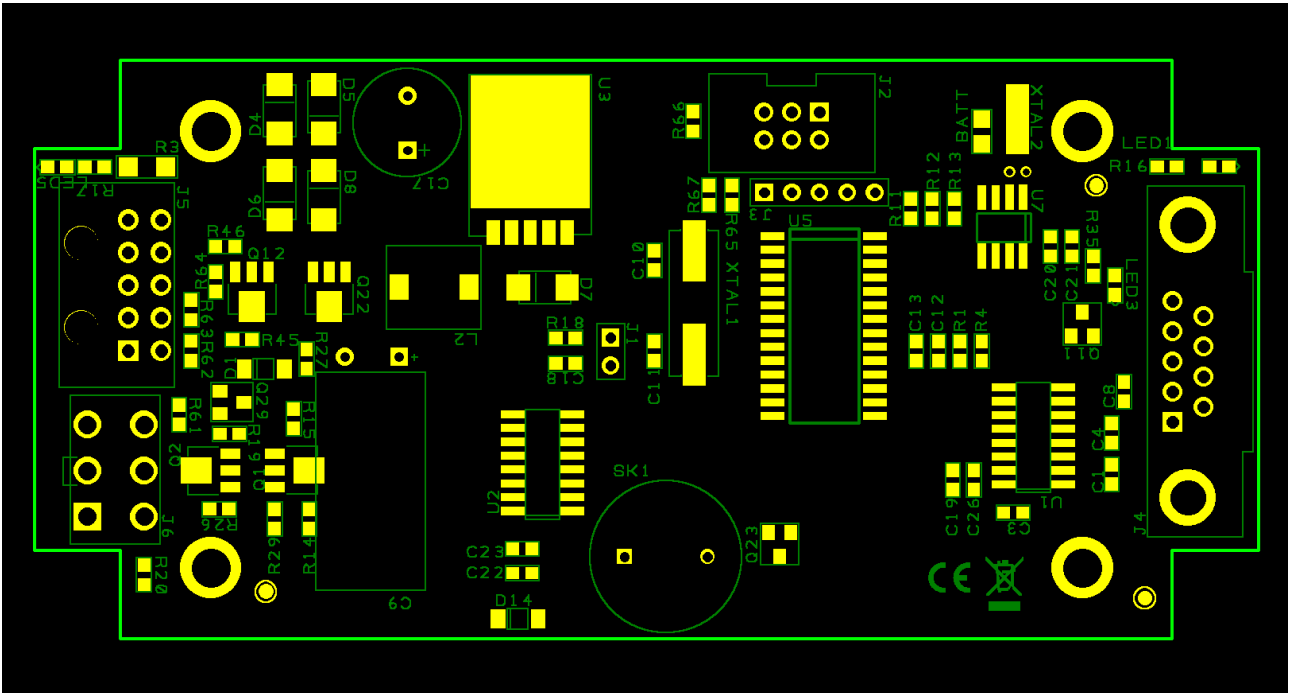
This device allows your application to send any MDB command to any peripheral (payment systems) and some macro to any Executive machine. This way, your application can manage any MDB payment systems combination, can be adjusted to handle non-standard peripherals and can be upgraded at any moment, to meet newer MDB features.

The bridge is the easiest way to either upgrade an Executive machine to MDB peripherals support, retrofit it by adding some nice features and to obtain sales and cash reports (by using Executive price holding mode on VMC). Also, it allows your application to implement dynamic prices, bundles and also capturing MDB peripherals errors (like bill ja, coin jam, etc.)

In Executive price holding mode, your application keeps products' prices and handle dispensing task accordingly. This way, prices can be remotely updated and adjusted.

II. Hardware overview.

1. Motherboard



2. Connector description

- J5 – Power + Executive connector – device comes with the corresponding cable.
- J6 – MDB master connector, for MDB peripheral connections.
- J4 – RS232 connector. This requires an RS232 straight (pin-to-pin) cable with DB9 female and DB9 male connector.

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

3. Power supply requirements

The PICOVEND can be powered with 24VDC/AC or 12VDC/AC, depending on your MDB PERIPHERALS. You have to make sure your MDB peripherals will not exceed 3A peaks, with a maximum 1A continuous current draining. If your peripherals require higher current, you need to power them using external circuits.

III. Low level communication protocol

A. Serial port settings

The low level communication protocol can be used over the Raspberry Pi serial interface and does not need any hardware or software flow control.

Communication parameters are:

- baud rate – 115200 (could be delivered with other baud rates upon request);
- data bits – 8;
- stop bits – 1;
- parity – None.

B. Low level protocol messages

Every message consists of a message header, message data and CRC. The CRC is calculated by following standard MDB rules, for both MDB messages and Executive macros. For MDB messages, the message must always encapsulate the entire MDB command, including it's CRC.

Every time a command or a response is sent, the other device should answer with ACK or NAK and your application should take all actions needed, according to device's answer.

ACK message is: 0xFC 0xFC 0xFC 0xFC 0xF0

NAK message is: 0xFD 0xFD 0xFD 0xFD 0xF4

As a convention for the protocol description tables, B0, B1, ... Bn represents "byte number" and b0, b1, ... bn represents "bit number". Also, we will refer to this interface as "device" or as "bridge"

1. MDB messages

MDB messages must always begin with 0x4D (“M”). If the first byte of the message begins with this byte, the interface will redirect it to MDB interface. Your application must handle correct CRC calculation for both MDB and device message. Please check MDB manual for peripheral specific commands set.

Examples:

a. To send bill validator poll, you need to send the following message:

0x4D 0x33 0x33 0xB3 where

- 0x4D is the message header that informs the bridge it should be sent to the MDB bus;
- 0x33 – bill validator poll
- 0x33 – bill validator message CRC
- 0xB3 – bridge message CRC.

CRC calculation is following the MDB rule:

$0x4D + 0x33 + 0x33 = 0xB3$

b. To send a vend request for the first cashless device, you need to send the following message:

0x4D 0x13 0x00 <item price – 2 bytes> <itemnumber – 2 bytes> <MDBCRC>
<BRIDGECRC>

For example, a vend request for item number 14 with a price of 1.50EUR on a cashless with scaling factor = 1 will be as follows:

0x4D 0x13 0x00 0x00 0x96 0x00 0x0E 0xB7 0xBB

- 0x4D – message header – it must be sent to MDB;
- 0x13 0x00 – MDB vend for cashless #1
- 0x00 0x96 – item price (150, the scaled value, according to MDB rules);
- 0x00 0x0E – item number (14);
- 0xB7 – encapsulated MDB message CRC ($0x13 + 0x96 + 0x0E = B7$)
- 0xBB – device message CRC ($0x4D + 0x13 + 0x96 + 0x0E + 0xB7 = 0x01BB$) and according to MDB rules, CRC is the lower byte of the sum = 0xBB.

Your application must take care of all MDB messages, their correct order, etc. according to MDB peripheral specific.

If the message sent to the device has a correct CRC, it will be sent to MDB and the bridge will return the raw message received from the peripheral (for example, 0x00 = ACK).

If the message sent to the device has not a correct CRC, you will receive an error message (0xFF).

2. Executive macros

Due to Executive protocol restrictions, basic messages to VMC are encapsulated in device's hardware and you only need to invoke them from time to time to allow your application follow the correct course. All messages for Executive bus must begin with 0x45 ("E") meaning that the bridge should analyze it and send the correct macro command to the VMC.

a. Executive status – is returning vending machine status. If the answer is 0x00, the machine is up and running. If the answer is 0x40, the machine is disabled, either due to a "dispense/prepare" condition, either because of an internal malfunction. Your application must constantly poll the machine using this message to keep it running and to take some decisions in the case of a malfunction (for example, disabling payment systems while the machine is out of order). This message format is fixed:

0x45 0x01 0x46

- 0x45 - "E" - this is an Executive message
- 0x01 – macro ID – Executive status poll
- 0x46 – message CRC (0x45 + 0x01 = 0x46)

b. Executive credit poll – is returning the price or the product ID for the selected product. If the machine is configured in the "Price holding" mode, it will return one byte containing the item number of the selected product. If the machine is configured with prices in the VMC, it will return one byte, containing scaled price of the selected product. Your application must constantly poll the machine using this message, only if there is some credit accumulated and previously sent to VMC. This message format is fixed:

0x45 0x02 0x47

- 0x45 - "E" - this is an Executive message
- 0x02 – macro ID – Executive credit poll
- 0x47 – message CRC (0x45 + 0x02 = 0x47)

c. Executive send credit – is sending a scaled credit to the machine. This command is sent to the device every time the accumulated credit is modified, wither by a cash or a cashless peripheral. The message format is:

0x45 0x03 <credit to send – 4 bytes> <scaling factor – 1 byte> <decimal places – 1 byte>
<exact change flag – 1 byte> <BRIDGECRC>

For example, sending 2.00EUR credit to the machine, with a scaling factor of 1, with 2 decimal places and with exact change flag set:

0x45 0x03 0x00 0x00 0x00 0xC8 0x04 0x01 0x12

- 0x45 - “E” - message to Executive bus
- 0x03 – Macro ID – Executive send credit
- 0x00 0x00 0x00 0xC8 – credit value (200 scaled by 1)
- 0x04 – 2 decimal places (according to Executive specifications, this byte has one bit set in the position where decimal point must be placed – in this case 0b00000100)
- 0x01 – Exact change flag set – this means that there is no change available in the changer device or not changed device attached to the MDB bus.
- 0x12 – message CRC (0x45 + 0xC8 + 0x04 + 0x01 = 0x112 and the lower byte is 0x12)

d. Executive vend – is sending a vend approval, following an Executive credit poll that returned a product value or ID. The format is fixed:

0x45 0x04 0x49

- 0x45 - “E” - message to Executive bus
- 0x04 – Macro ID – Executive vend
- 0x49 - Message CRC (0x45 + 0x04 = 0x49)

After sending this message, the device will stop responding while waiting the machine to deliver/prepare the selected product and returning the action result (vend success or vend failed).

e. Executive vend result – is the message for polling last vend result (vend success or vend failed). Your application must start polling the device with this macro immediately after sending an Executive vend message (sale approval). Message has fixed format:

0x45 0x05 0x4A

- 0x45 - “E” - Message to Executive bus
- 0x05 – Macro ID – Executive vend result
- 0x4A – Message CRC (0x45 + 0x04 = 0x4A)

During product preparation/delivery, the bridge will not answer to any command/macro since it is monitoring the machine for the vend result. You need to keep polling the bridge

with this command until there is an answer from it. The possible answers are: 0x00 – vend success or 0x80 – vend failed.

If a vend success occurs, your application is responsible to subtract product's value from the available credit and send the remaining credit value to the machine by using Executive send credit macro.

Notes: