

Shanghai Wafer Microelectronics Co., Ltd.

23E, Liang Feng Building, NO.8, DongFang Road, Shanghai, China Tel.: (0086) 21 6845-8945 Fax: (0086)21 5045-4820

Платежное устройство MDB с интерфейсом RS232

Модель: MDB-RS232 (Новая версия коробки типа V3)

ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ

Формат связи:

Скорость передачи:	9600	Начальный бит:	1
Биты данных:	8	Паритет:	NO
Стоповый бит:	1	Данные ПК Отправить:	Hex
Получение данных ПК:	ASCII		

Спецификация оборудования.

Источник питания

Мин. 20 VDC (Нормальное рабочее напряжение 24 ... 30 переменного или постоянного тока)

Потребляет ток холостого хода

< 0.05A (Платежное устройство MDB не подключено)

Кабель

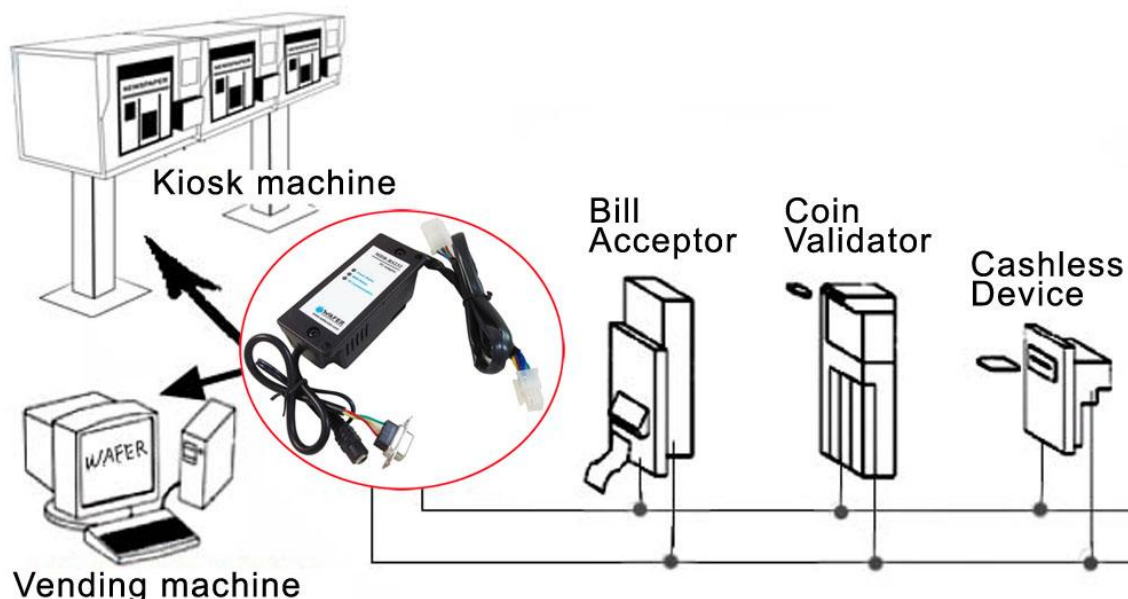
Разъем MDB, стандартный последовательный кабель RS232 и разъем питания DC2.1

Габаритные размеры ок. 100 x 42 mm (Не включать кабели)

Масса ок. 100 g

Установите MDB-RS232

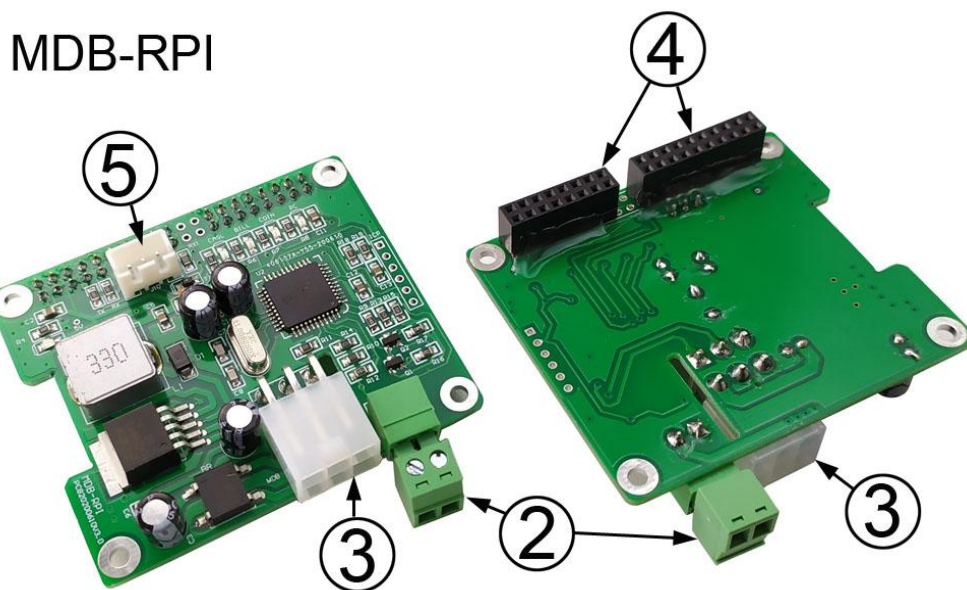
(VMC может быть компьютером (ПК), ПЛК или основной платой Android.)



Ниже представлено изображение двух типов MDB-RS232, одинаковой функции и протокола, единственная разница в том, что версия с тремя светодиодами активирует платежное устройство с самого начала, а версия с пятью светодиодами запускается со статусом отключения для платежного устройства, и пользователю необходимо использовать Команда enable для включения.



- (1) Разъем RS232 к COM-порту компьютера
- (2) Разъем питания для источника питания переменного / постоянного тока (обычно мы используем 24 В постоянного тока)
- (3) Разъем MDB для купюроприемника, валидатора монет или кард-ридера



- (2) Разъем питания для источника питания переменного / постоянного тока (обычно мы используем 24 В постоянного тока)
- (3) Разъем MDB для купюроприемника, валидатора монет или кард-ридера
- (4) GPIO для платы Raspberry Pi (используйте RXD, TXD, GND и выход 5 В для Raspberry Pi)
- (5) Последовательный порт к ПК (можно использовать этот порт для тестирования с помощью программного обеспечения ПК)

Установка относительно проста; для полноценного функционирования устройства необходимо выполнить всего три подключения. На переходной коробке MDB есть разъемы.

Один кабель подключается к источнику питания 24 В постоянного тока. (При подключении источника питания постоянного тока 24 В нет необходимости учитывать особую полярность. 6-контактный Molex подключается к устройствам MDB. Последний разъем представляет собой DB-9 и подключается к задней части компьютера или любой другой материнской плате с портом RS232. На задней панели компьютера должен быть открытый порт с надписью «SERIAL2» или «COM2». Если на вашем компьютере есть только порт USB, вам будет легко заказать адаптер RS232 / USB.

Для платы MDB-RPI имеет два последовательных порта для ПК, а также для платы GPIO Raspberry pi. Также он будет обеспечивать напряжение DC5V для платы Raspberry pi.

После подключения устройства, а затем включите питание с DC24V. Проверьте наличие светодиода «Состояние питания» на коробке (КРАСНЫЙ светодиод), указывающего, что с питанием все в порядке.

Если устройство MDB правильно подключено и работает правильно, светодиод «Состояние MDB» должен начать мигать.

На коробке MDB-RS232 также есть еще один светодиод «Связь с ПК», который будет мигать, если есть какие-либо данные, отправленные из коробки MDB-RS232 на ПК.

Программное обеспечение

MDB-RS232 отправляет информацию, сгенерированную устройством MDB, непосредственно на ПК через последовательный порт RS-232. Нет необходимости опрашивать каждое устройство MDB. Это делается с помощью MDB-RS232. По умолчанию MDB-RS232 опрашивает все известные устройства MDB. Затем MDB-RS232 отправляет данные на ПК, если есть активность. Информация, отправляемая на ПК, отправляется в виде байтов в «**hexadecimal**». Для данных активности первый отправленный байт - это идентификатор устройства. Например, 30 XX означает, что валидатор счетов отправил информацию. В то время как 08 XX означает, что монетоприемник отправил данные. Обратитесь к спецификации команд MDB V4.2, чтобы узнать о командах, специфичных для вашего устройства MDB.

Как использовать команду для связи с блоком адаптера MDB-RS232?

Отправленные данные и ответ совместимы с протоколом MDB.

Команда ПК: Просто используйте ту же команду, что и протокол MDB для VMC

ПК получил данные1:

Любые данные, отправленные обратно в соответствии с инструкциями запроса ПК, в том же формате, что и протокол MDB, нет необходимости добавлять первый байт идентификатора устройства

ПК получил данные2:

Любые данные об активности с платежного устройства будут отправлены на ПК автоматически, и первый байт - это идентификатор устройства.

Например, команда настройки монетоприемника: **09**

После отправки данных **09** с ПК, блок адаптера считывает и отправляет обратно информацию о настройке монетоприемника:

03 11 56 05 01 00 03 01 02 00 00 00 00 00 00 00 00 00 00 00 76

Например, команда включения акцептора купюры: **0C FF FF FF FF**

После отправки данных **0CFFFFFFF** с ПК, тогда блок адаптера отправит купюроприемнику и отправит обратно данные ответа валидатора: **00**

Выше ПК получил данные, в первом байте нет идентификатора устройства, но следующие данные будут иметь данные об активности от монетоприемника.

Например, монета принята, будет отправлена как:

"08 51 00" 51 00 - это данные об активности монетоприемника,

и когда данные передаются на ПК, MDB-RS232 добавляет первый байт идентификатора устройства 08h

Данные об активности купюроприемника, например, купюроприемник отключен, будут представлены как: **30 09**

09 - это данные об активности купюроприемника, а MDB-RS232 добавляет первый байт идентификатора устройства. 30h

(Просто отметьте: автоматически сообщаемые данные будут иметь байт идентификатора устройства)

Связь программного обеспечения MDB-RS232

Получать и передавать

Используйте событие связи, управляемое прерыванием, на соответствующем СОМ-порту. Настройки 9600-8-1-Нет.

Следует отметить еще раз

Любые данные, отправленные с ПК на блок MDB-RS232, которые должны использовать данные HEX

Любые данные, которые ПК получил от коробки MDB-RS232, то есть данные ASCII.

Поэтому, когда пользовательская тестовая команда с помощью инструмента MDB Дето или инструментов RS232, должна выбрать «шестнадцатеричный» режим для отправки данных.

Следующее содержимое на английском языке, пожалуйста, поймите его в соответствии с протоколом MDB.

MDB OPERATION INSTRUCTIONS:

For BILL VALIDATOR

Bills Accepted (Byte 1) 1yyyxxxx

yyy = Bill Routing

000 = Bill Stacked

001 = Escrow Request

010 = Bill Returned

011 = Not Used

100 = Disabled Bill Rejected

xxxx = Bill Type

The bill types are:

Type 0 = \$1

Type 2 = \$5

Type 4 = \$20

Type 1 = \$2

Type 3 = \$10

The software should have all of the bill types enabled, this will allow the user to set which type of bills to be accepted on the validator itself.

Bill Validator Operation Notes

-Firmware sets Bill Validator to accept 1, 2, 5, 10, 20 US bills by default

-Any commands to changed bills accepted or held in escrow will be set back to the firmware defaults upon a cycling of power or reset.

VMC Commands for Bill Validator

US Bills – Bit 0 = \$1 Bit 1 = \$2 Bit 3 = \$5 Bit 4 = \$10 Bit 5 = \$20

BILL'S ACCEPTED

Bill Type 34h 4bytes Y1-Y4

Bill's Accepted

Y1-Y2 = 001Fh for all US bills accepted
 = 0000h accept no bill's

For example:

Send the command: 3400030000h

Bill's held in Escrow

Y3-Y4 = 001Fh for all US bills held in escrow
 = 0000h for no bill's held in escrow

Send out 34h and then the 4 bytes Y1-Y4 to change bill's accepted and held in escrow.

For example:

Send the command: 34001F001F

Enable the ESCROW, can accept the \$1, But the coin go to the cash box directly

BILL'S IN ESCROW ACTION

Escrow	35h	1byte	Y1
Return bill	Y1 = 00h		
Stack bill	Y1 = 01h		

Send 35h and then Y1 to act on bill held in escrow

For example:

Send the command: 3500h to Return the bill

The bill acceptor will reply: 00 and also following two data bytes: 30 A3 to show the bill type returned

Send the command: 3501h to accept the bill to the bill stacker box

The bill acceptor will reply: 00 and also following three status data bytes: 30 83 09 to show the bill type returned

STACKER STATUS

Stacker	36h	response	Z1-Z2
Byte1	Byte2		
Fxxxxxxx	xxxxxxx		

F=1 Stacker Full

Xxxxxxxxxxxxxx = Number of bill's in stacker

Send out a 36h to the Bill Validator—It will respond with 2 bytes Z1-Z2

For example,if the bill stacker is not full and already stacked 3 pcs bills,then would reply 03 03

Enable the escrow to accept or return bills

(For the following test,user should have bill validator with Bill recycler mounted,Following command data is tested with ITL NV11 bill validator)

(1) Use the 34FFFFFFF or other 34 command to enable the escrow

(2) When user insert the bill, it will stop at the escrow and send the command 30 90 09 to PC and also later will keep to send 3009 to PC

30 that means data is from bill acceptor, 90 means a TYPE0 bill is accepted and stoped at the Escrow position (For different bills,maybe 91 or 92 or 93, just depend on the Bill Type accepted) and the last 09 means Validator is Disabled to wait the VMC to acceptor or return the bill

(3) PC send the command 3501 to accept the bill or 3500 to return the bill

BILL VALIDATOR	
All values are in hex	MDB data from Bill Validator to the PC
Bill Accepted	
\$1	30 80 09
\$2	30 81 09
\$5	30 82 09
\$10	30 83 09
\$20	30 84 09
Bill Returned	All valid bill types disabled in software
\$1	30 C0 09
\$2	30 C1 09
\$5	30 C2 09
\$10	30 C3 09
\$20	30 C4 09
Bill Held In Escrow	
\$1	30 90 09
\$2	30 91 09
\$5	30 92 09
\$10	30 93 09
\$20	30 94 09
Bill forcibly Removed	
\$1	30 A1 09
\$2	30 A2 09
\$5	30 A3 09
\$10	30 A4 09
\$20	30 A5 09
Bill Validator Status	
01	Defective Motor
02	Sensor Problem
03	Validator Busy
04	ROM Checksum Error
05	Validator Jammed
06	Validator was Reset
07	Bill Removed
08	Cash Box Out of Position
09	Unit Disabled
0A	Invalid Escrow Request
0B	Bill Rejected
010xxxxxx	Number of attempts to input a bill while validator is disabled
14	Bill not accepted either because the bill type is not enabled in

	the software or the bill was not recognized
--	---

For COIN ACCEPTOR

Coins Deposited:

(Byte1)	(Byte 2)
01yyxxxx	zzzzzzzz
yy = Coin Routing	
	00: Cash Box
	01: Tubes
	10: Not Used
	11: Reject

xxxx = Coin Type

zzzzzzzz = The number of coins in the tube for the type accepted.

Coins Dispensed Manually

(Byte1)	(Byte 2)
1yyyxxxx	zzzzzzzz
yyy = The number of coins dispensed	
xxxx = The coin type dispensed	
zzzzzzzz = The number of coins in the tube	

The coin types are:

Type 0 = 5c	Type 2 = 25c	Type 5 = \$2 Can.
Type 1 = 10c	Type 4 = \$1 Can.	

Note: The type of the coin is the same as the bit that needs to be set in the 'mdbCointype' routine in order to enable the acceptance, or distribution of that coin.

COIN ACCEPTOR			
All values are in hex.	DATA RECEIVED FROM MDB AND SENT TO THE PC		
	Below Low Mark	Above Low Mark	Above High Mark
Coin Inserted			
NICKEL	08 50 00	08 50 06	08 40 4C
DIME	08 51 00	08 51 08	08 41 6B
QUARTER	08 52 00	08 52 06	08 42 4B
QUARTER (1)	08 52 00	08 52 06	08 42 15
\$1 CANADIAN*	08 44 00		
\$2 CANADIAN*	08 45 00		
* Dollar coins are routed directly to the cash box			
Coin Dispensed Manually			
NICKEL	08 90 00	08 90 06	08 90 4C
DIME	08 91 00	08 91 08	08 91 6B
QUARTER	08 92 00	08 92 06	08 92 4B
QUARTER (1)	08 92 00	08 92 06	08 92 15
Coin Rejected			

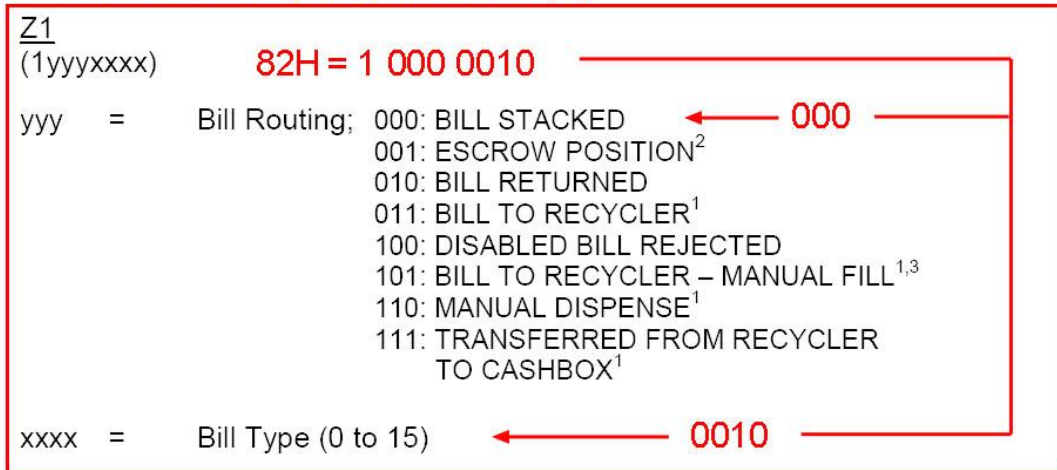
NICKEL	08 70 00	08 70 06	08 70 4C
DIME	08 71 00	08 71 08	08 71 6B
QUARTER	08 72 00	08 72 06	08 72 4B
QUARTER (1)	08 72 00	08 72 06	08 72 15
\$1 CANADIAN*	08 74 00		
\$2 CANADIAN*	08 75 00		

MDB STATUS	
01	Escrow Request
02	Changer Payout Busy
03	No Credit
04	Defective Tube Sensor
05	Double Arrival
06	Acceptor Unplugged
07	Tube Jam
08	ROM Checksum Error
09	Coin Routing Error
0A	Changer Busy
0B	Changer was Reset
0C	Coin Jam
21	Coin not recognized/slug. Returned
Upon startup one of these values below may be sent to the PC – These are the VMC Commands.	
08	Reset
09	Status
0A	Tube Status
0B	Poll
0C	Coin Type
0D	Dispense

* How to understand a currency data from bill validator ?

When insert a bill, we received: 30 82 09 (p94 and p95 in MDB file)

The first byte: 30, that means this data is from bill validator
 The second byte: 82 that is valid data (Different currency credit will be different data)
 We translate 82 into binary values to compare each bit.: 1000 0010
 (the first bit is 1, that is "currency" data. because other device "status" data, that first bit is 0 so that means the third byte: 09, that is status byte)



The third byte is 09, we can check the 95th page of the MDB protocol, compare it
 (00001001) = Validator Disabled² - The validator has been disabled, by the VMC or because of internal conditions.

* How to understand a credit data from coin validator ?

When insert a bill, we received: 08 52 09 (p66 and p67 in MDB file)

The first byte: 08, that means this data is from coin validator
 The second byte: 52 that is valid data (Different coin credit will be different data)
 We translate 52 into binary values to compare each bit.: 0101 0010
 (the second bit is 1, that is "Coin Despsited" data. because other device "status" data, that first three bits are all 0, so that means the third byte: 09, that is status byte)

Coins Dispensed Manually:

Z1 Z2
 (1yyyxxxx) (zzzzzzzz) **000**

yy = The number of coins dispensed.
 xxxx = The coin type dispensed (0 to 15)
 zzzzzzzz = The number of coins in the tube.

Coins Deposited:

Z1 Z2 **52 09H = 01 01 0010 00001001**
 (01yyxxxx) (zzzzzzzz)

yy = Coin routing. 00: CASH BOX
 01: TUBES ←
 10: NOT USED
 11: REJECT

(xxxx = Coin type deposited (0 to 15). ←

zzzzzzzz = The number of coins in the tube for the coin type accepted. ←

MDB-RS232 test with Nayax MDB cashless reader

Important note: if User want to test with Nayax card reader, must make sure that your card reader already connected to the nayax remote server system.that would be better to confirm with nayax servicer person. If not then MDB maybe can't test it.

MDB card reader has six working status:

1.Inactive:

After powered on or after a Reset command,then device will go to this state.User can use the SETUP command to active the nayax reader

2.Disable:

When card reader received the config data,will go to this "Disable" Status,Also If user send the "Disable" Command,also will go to this state.

3.Enable:

After the "Enable" command and Before detected a valid card,then stay in this state.

4.Session Idle:

When read a valid card,will go to "Session Idle" state to wait the "Vend request"

5.Vend request

When received the "Vend request" from the VMC,will go to vend status

6.Vend

Command List:

1401	Enable card reader	1400	Diable card reader
1402	Cancel card reader	1300000A0001	REQUEST VEND
1301	VEND CANCEL	13020001	VEND SUCCESS
130500640001	CASH SALE	1304	SESSION COMPLETE
1500	REQUEST REVALUE	1700	REQUEST ID
110001000000	Config the card reader	1101FFFF0000	Setup the max and min price

Steps for Testing with the card reader:

VMC: 110001000000 Config the card reader
 READER: 010109720102070D94 card reader reply
 VMC: 1101FFFF0000 Set the maximum and minimum price
 READER: no reply



Start to swipe the card



READER: 03FFFE Reading a valid card and wait the Vend selection
 VMC: 1300000A0001 VMC command to card reader after Vend selection
 READER: 00
 READER: 05000A Card reader send to VMC, confirm the good dispatch
 VMC: 13020001 VMC tell the card reader,already dispatch the selected goods
 READER: no reply
 VMC: 1304 end the section.
 READER: 07

Important questions during application:

1. Sometimes when send the command to the mdb-rs232, but I can't get any reply. How can I know if the command has already been sent successfully to the payment device?

Answer: In MDB-RS232 box design, when PC send the command to it, it will forward the command to payment device. But sometimes the payment device is busy, for example a bill is inserted and payment device can't reply the box in time. So PC can't get the reply. Then just need to resend the command later. Of course, this is not a problem, because ALL the command from PC, that is not a realtime command.

2. How can I know if the payment device is enabled or disabled ?

Answer: If the payment device is in disabled status, Bill acceptors will report the status to PC always once each POLL. It will report **30 09** or **30 03 09** to PC, And also for most coin acceptor, if need the status, user can read the status with **0F05** command.

So When the power is just turned on, the bill validator device will be in the "Disabled" state, and such a status report will be received. If we send the enable command **34FFFF000**, the bill validator will enter the "Enabled" state and PC will not receive such a status report.

Multi-Drop Bus / Internal Communication Protocol	
Status:	Bill Validator (Only)
(00000001) =	Defective Motor ³ - One of the motors has failed to perform its expected assignment.
(00000010) =	Sensor Problem ³ - One of the sensors has failed to provide its response.
03 (00000011) =	Validator Busy ² - The validator is busy and can not answer a detailed command right now.
(00000100) =	ROM Checksum Error ³ - The validators internal checksum does not match the calculated checksum.
(00000101) =	Validator Jammed ³ - A bill(s) has jammed in the acceptance path.
(00000110) =	Validator was reset ¹ - The validator has been reset since the last POLL.
(00000111) =	Bill Removed ¹ - A bill in the escrow position has been removed by an unknown means. A BILL RETURNED message should also be sent.
(00001000) =	Cash Box out of position ³ - The validator has detected the cash box to be open or removed.
09 (00001001) =	Validator Disabled ² - The validator has been disabled, by the VMC or because of internal conditions.
(00001010) =	Invalid Escrow request ¹ - An ESCROW command

3. If I can use a PLC or android board to connect it ?

Answer: That is no problem. Actually, user can use any main controller which has RS232 or USB port to work as the VMC. Of course, if the main controller only has the USB port, then will use a USB version. And install the USB Chip FT232RL driver to create the virtual com port in the main controller.

3. If can supply more documents to facilitate the development of applications?

Answer: The MDB box we provide is to forward the data uploaded to the PC by the MDB payment device. The documents we provide can only make it as easy as possible for customers to understand the process of sending and receiving data and the structure of the data. As for the commands of the host, the design of the user's host software requires the user to design the control process after having a detailed understanding of the payment device and the MDB protocol. We are not able to provide assistance in customer specific projects and software design.

4. About the Status reading command of coin acceptor (0F05 command to read the status)

Not all coin validators can report the full working status, especially the old version, even in Level03, also can't provide the proper status report data

Normally after powered on, it should report **01 00 06 00 07** (**01/00** means powering up)

Read the status again should report **06 00 06** (coin validator is disabled by default)

After use the command **0CFFFFFFF** to enable, then read status should report with **03 00 03**

Must be noted: Some coin validators always report with **030003**, no matter in what status and some only port with **000000**. In these cases where the MDB protocol is not followed, please consult the coin validator manufacturer. Not a problem that MDB-RS232 boxes can solve

5. How to use the 0F02 command to payout the coins

Following is the samples to payout the coins:

- ✦ Firstly from the setup command 09 to read the configuration data
03 11 56 **05 01** 00 03 **01 02** 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 76

From this configuration data, then two type coins, coin value is 0.5\$ and 1\$

- ✦ Then use the 0A to check the tube status, which type and how many pieces in the tube
For example: 00 00 **03 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 04**

then have two type of the coins and 03pcs 0.5\$ and 1pcs 1\$

So while the program is running, it must constantly check the status of the TUBE, so that it can know in real time how many coins can be paid out.

- ✦ Then you can use the payout command

0F 02 01 to payout one pcs 0.5\$ coin.

0F 02 02 to payout one pcs 1\$ coin or 2pcs 0.5\$ coin

That means you must have the coins in the tube and also that can be read by the 0A command, then can be payed out.

5. What is the difference between normal MDB-RS232 and the 2020 Version MDB-RS232?

Normal MDB-RS232 will send the enable command to the connected MDB coin acceptor, bill acceptor and the cashless device, but the 2020 Version MDB-RS232 will not do the job, just start to poll the slave payment devices and VMC will need to send these commands. So if we connect the coin and bill acceptors to MDB box and powered on, Even PC is not connected, Normal MDB-RS232 will enable the coin acceptor and bill acceptor and is ready to accept the coin or bills. But V2020 Version box, after powered on, all payment devices are in the disable status.

6. If user can send the poll command from the PC ?

At present, our design is to complete the POLL command by the MDB-RS232 box. The POLL command cannot be sent from the PC. The PC only needs to process the remaining commands

and receive the corresponding data.

7. How to use MDB-RS232® to test Bill recycler ?

We have a special file for bill recycler

<http://www.waferlife.com/downloads/How to use MDB Bill validator recycler.pdf>

8. Can the MDB-RS232 adapter support the age verification device interface? ?

Due to legal restrictions, a variety of products are only allowed to be vended via vending machines by checking the customers age. The age and the rules vary from country to country. This is i.e. related in some countries to cigarettes or alcoholic products. Some services or product contents may restrict a different age, related to the vending machine, this needs different ages to be checked within the same machine.

We also provide a specific MDB-RS232 box that can support age verification interface. But not all boxes support it by default. Need to consult when ordering.

How to order a sample to test ?

Answer : We can order the samples from online shopping store: <http://cnkiosk.aliexpress.com>

Wafer manufacture a series of the MDB payment adapters for Vending machines cashless payment system,wireless payment payment and Provide a rich solution for vending machine,kiosk machine or other self-service projects.

How can we connect the Pulse type coin acceptors or bill acceptors to PC ?

Answer: WAFER also have the PULSE-PC adapter box,that is used to connect the pulse type payment device to computer RS232 port.

We also can order the samples from: <http://cnkiosk.aliexpress.com>

How can we develop a cashless payment interface to existing vending machine?

Answer: WAFER also have the RS232-MDB adapter box,that is used to connect the RS232 interface PC or android main board or popular Raspberry pi board to vending machine,and then developer can use some simple command to act as a perfect cashless payment solution.

We also can order the samples from: <http://cnkiosk.aliexpress.com>

How can we develop a mobile payment solution for existing vending machine ?

Answer: WAFER have the 3G or 4G or LAN version payment module for selection.

We also can order the samples from: <http://cnkiosk.aliexpress.com>

Where can we get more further information and get the fast technical service ?

<http://www.waferlife.com/en/MDB2PC-PC2MDB.html>

How to have a customized MDB or vending payment board ?

Answer : WAFER can provide users with customized control boards in the field of self-service vending. Including time control boards, payment interface boards, coin refunders, lottery machines and other control boards.

Both pre-sales and after-sales can receive help and advice through our online skype technical support.

Email: wafer@waferstar.com

Web: <http://www.waferlife.com>

Tel: 0086-21-51870528

Online Service Skype: wafer-service

V2021-V9.2

Copyright waferstar