

# 上海威佛微电子有限公司.

上海市浦东新区东方路8号良丰大厦23E Tel.: (0086)21 6845-8945 Fax: (0086)21 5045-4820

## MDB硬币器,纸币接收器以及MDB刷卡器等 MDB设备RS-232通信PC转接盒

Model: MDB-RS232 (新版)

### 技术规格

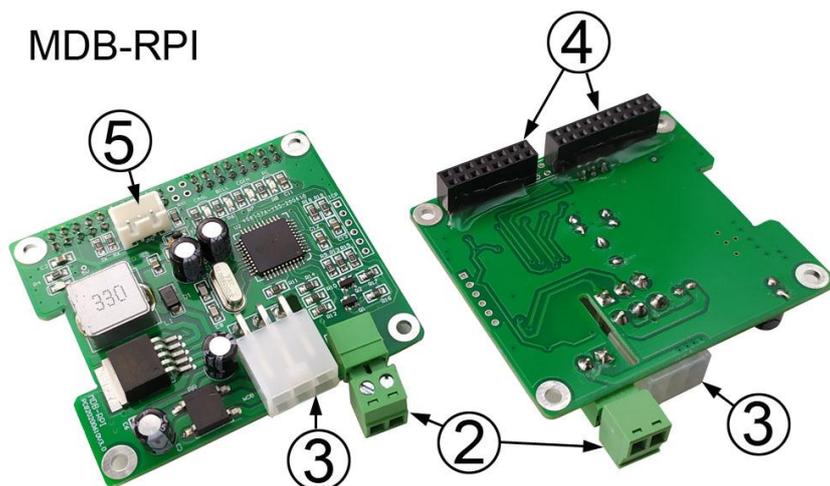
通信格式.	(主机(PC)向MDB-RS232发送指令为HEX格式, MDB-RS232发给PC的格式为ASCII)
波特率	9600BPS
起始位	1
数据位数	8
奇偶校验	NO
停止位	1
硬件规格.	
供电电压	最小 20 VDC (特殊情况下, 可以根据外围设备需要的供电电压进行选择) 正常供电范围: Nominal 24...34 VDC (或者24...30VAC) 静态电流 < 0.05A 在接收纸币的时候电流会有 0.5A 持续大概 1 秒
数据线	标准电脑串口通信线
规格尺寸	大约 80 x 50 mm
重量	大约 0.1kg





图中:

- (1) RS232串口插头 (如果用户设备没有串口, 则需要增加一个串口转USB的适配器)
- (2) DC24V电源输入接头, 不需要区分正负极
- (3) 为MDB接头, 直接和同样MDB接口的纸币器, 硬币器以及刷卡器相接即可



图中:

- (2) DC24V电源输入接头, 不需要区分正负极
- (3) 为MDB接头, 直接和同样MDB接口的纸币器, 硬币器以及刷卡器相接即可
- (4) 树莓派GPIO插座, 直接插到树莓派上, 通过RXD, TXD进行通信。同时通过5V, GND给树莓派供电
- (5) 为串口插头, 可以通过购买样品时附送的TTL转RS232适配器, 连接到电脑串口进行调试

关于树莓派版本 MDB-RS232 的测试:

- 因为标准的测试工具软件是 windows 系统下运行的, 因此我们能提供了一个工具, 可以把树莓派版本 MDB-RS232 通过 TTL 转 USB 适配器连接到 Windows 电脑 USB 口进行测试
- USB 适配器的芯片采用 SILICON 的 CP2102。驱动下载地址: [CP210x USB to UART Bridge VCP Drivers - Silicon Labs \(silabs.com\)](http://www.silabs.com/CP210x_USB_to_UART_Bridge_VCP_Drivers)
- 树莓派版本的 MDB-RS232 是直接插到树莓派的 GPIO 总线上的, 使用了总线上 RXD, TXD, GND 和 5V (其中 5V 引脚是 MDB 板子向树莓派供电 5V, 因此树莓派主板不再需要单独供电)
- 在 Windows 软件测试熟悉协议后, 就可以插到树莓派 GPIO 上, 进行软件编程和测试了

## 安装MDB-RS232并连接到PC:

MDB-RS232 的安装相对很简单，只有三个接口需要连接，包括电源接口，MDB 接口，RS232 通信接口。盒子上的 RS232 串口可以直接或者通过串口延长线连接 MDB 盒子到电脑后面的 COM 口。然后在盒子上会发现有两个 6-pin Molex 接头（一公一母），只要分别接上 MDB 接口纸币接收器，硬币器等 MDB 设备，最后给 MDB 转接盒上电，供电电压一般使用直流 DC24V，即可完成安装。

安装后，一定要弄清楚电脑的串口的串口号，如果是通过 USB 转串口方式得到的串口，可以在 WINDOWS 的“设备管理器”里看具体生成的虚拟串口号。

如果是刚开始通过测试软件测试，则注意，在“Serial Setting”那里，要打开对应的串口号，同时软件的“mode”那里要勾选上“HEX”。（软件默认是“Text”，这里要注意）

## 软件及通信说明:

MDB-RS232 不需要主机进行 POLL 查询操作，它会自动完成和 MDB 设备的通信和管理，并且在得到有效 MDB 数据的时候自动通过串口发送给电脑。

MDB 盒子主动发送给电脑的数据是按照 ASCII 方式，第一个字节是设备的 ID，比如 30 XX，前面的 30 就表示是纸币接收器的数据。如果收到的数据是 08 XX，则 08 就表示是硬币器发送到 PC 的数据。具体的指令含义，可以查询 MDB V4.2 指令协议。

主机在需要的时候，也可以主动向相应的设备发送指令，比如查询状态，使能纸币器，禁止硬币器等等。这个时候，设备会立即做出回复（注意：回复的数据前面是不带设备设别代号的）

MDB-RS232 的设计采用的是支付及实时数据优先，所以主机发送下去的命令，有可能因为设备繁忙等原因，得不到回复，一般回复的延迟时间不超过 1 秒。具体设计是如果超过 2 秒没有得到数据，就可以进行重发。

在进行用户软件开发之前，建议用户使用威佛的测试软件，对设备，MDB 协议数据进行一个完整的认识。设计 PC 程序的时候，可以使用事件中断的方式去截获串口上收到的 MDB 实时数据，电脑的 COM 口的配置是：9600-8-1-None。

### 指令示例:

比如硬币器的 SETUP 指令: **09**

我们发送 **09** 指令，主要是 HEX 的 09 数据，则设备通过 MDB-RS232 回复配置信息:

**03 11 56 05 01 00 03 01 02 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00**

比如我们发送纸币器使能指令: **0C FF FF FF FF**

发送数据 **0CFFFFFFF**，则纸币器通过 MDB-RS232 回复数据: **00**

也就是这些数据是设备给指令回复的原始数据（多字节数据最后一个字节是校验字节）

而设备实时数据，比如有硬币投入时候，硬币器主动上报: **08 51 00**

**51 00** 是硬币器的上报数据，MDB-RS232 在前面加上了设备 ID 08h，

比如纸币器被禁止了，则纸币器会报告故障数据: **30 09**

**09** 就表示纸币器被禁止，MDB-RS232 在报告的数据前加上设备 ID 30h

（从上面可以看出来，主机查询设备应答的数据是没有 ID 的。设备自动报告的数据是带有 ID 的。另外带有设备 ID 的主动报告数据，是不带校验字节的）

**MDB 工作简介**（以下仅仅为示例性描述，具体需要全面阅读 MDB 协议）

比如接收到有效纸币的 MDB 数据为：（一个字节） 1yyyyxxxx

yyy = 纸币的状态

000 = 纸币压入钱箱

001 = 纸币暂存器请求

010 = 纸币退回

011 = 未使用

100 = 禁止的纸币器拒收

xxxx = 纸币类型（按照美元方式做示例）

美元纸币器的类型有：

Type 0 = \$1

Type 2 = \$5

Type 4 = \$20

Type 1 = \$2

Type 3 = \$10

比如在投入纸币的时候，我们收到数据：30 82 09

30 表示数据来自纸币器；82 因为高位是 1，所以是纸币器的投币数据，后面的 2 就是接收到的纸币类型，就是\$5；后面的 09，是状态字节，对应协议确认是什么状态即可。

**纸币器主要控制指令：**

（硬件默认纸币器可以接收 1, 2, 5, 10, 20 纸币）

（US 纸币类型 – Bit 0 = \$1 Bit 1 = \$2 Bit 3 = \$5 Bit 4 = \$10 Bit 5 = \$20）

**(1) 控制接收纸币的类型：**

Bill Type 34h 4 个字节数据： Y1-Y4

Bill's Accepted（接收纸币的类型）

Y1-Y2 = 001Fh 设置可以接收的纸币

= 0000h 不接收纸币

Bill's held in Escrow（纸币可以进入暂存器(ESCROW)的类型）

Y3-Y4 = 001Fh 设置可以被暂存的纸币

= 0000h 都不启用暂存器，钱币直接进钱箱

发送 34H 指令，以及四个字节的参数 Y1-Y4 去更改接收纸币的类型和进入暂存器的纸币类型

比如

34001F001F 使能了暂存器 ESCROW

34001F001E 可以一元收钱，但是禁止了暂存器 ESCROW，收到的钱会直接进入钱箱

**(2) 纸币在暂存器中的控制指令：**

Escrow 35h 单字节 Y1

退回纸币 Y1 = 00h

接收纸币入钱箱 Y1 = 01h

发送 35H 指令，以及 Y1 参数去控制暂存器中的纸币

**(3) 钱箱的状态：**

Stacker

36h

纸币器回复： Z1-Z2

Byte1

Byte2

Fxxxxxxx

xxxxxxx

F=1 钱箱已满

Xxxxxxxxxxxxxxxxxx = 钱箱中的纸币的数量

发送 36H 指令，纸币器会返回两个字节的的状态数据 Z1-Z2

纸币接收器	
所有数据采用 HEX 方式	MDB 数据从纸币器到 PC
<b>Bill Accepted</b>	
\$1	30 80 09
\$2	30 81 09
\$5	30 82 09
\$10	30 83 09
\$20	30 84 09
<b>Bill Returned</b>	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
<b>Bill Held In Escrow</b>	
\$1	30 90 09
\$2	30 91 09
\$5	30 92 09
\$10	30 93 09
\$20	30 94 09
<b>Bill forcibly Removed</b>	
\$1	30 A1 09
\$2	30 A2 09
\$5	30 A3 09
\$10	30 A4 09
\$20	30 A5 09
<b>Bill Validator Status</b>	
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

**硬币器:**

收到硬币:

(Byte1)            (Byte 2)  
01yyxxxx        zzzzzzzz

yy = 硬币的位置状态

00: Cash Box    钱箱  
01: Tubes        钱管  
10: Not Used    未使用  
11: Reject       拒收

xxxx = 硬币的类型

zzzzzzzz = 当前接收到的硬币类型在钱管中的数量

手动操作退币

(Byte1)            (Byte 2)  
1yyyxxxx        zzzzzzzz

Yyy = 退出硬币的数量

xxxx = 退出硬币的类型

zzzzzzzz = 在硬币管子中的硬币的数量

硬币的类型有

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.

硬币器			
All values are in hex.	DATA RECEIVED FROM MDB AND SENT TO THE PC		
	Below Low Mark	Above Low Mark	Above High Mark
<b>Coin Inserted</b>			
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

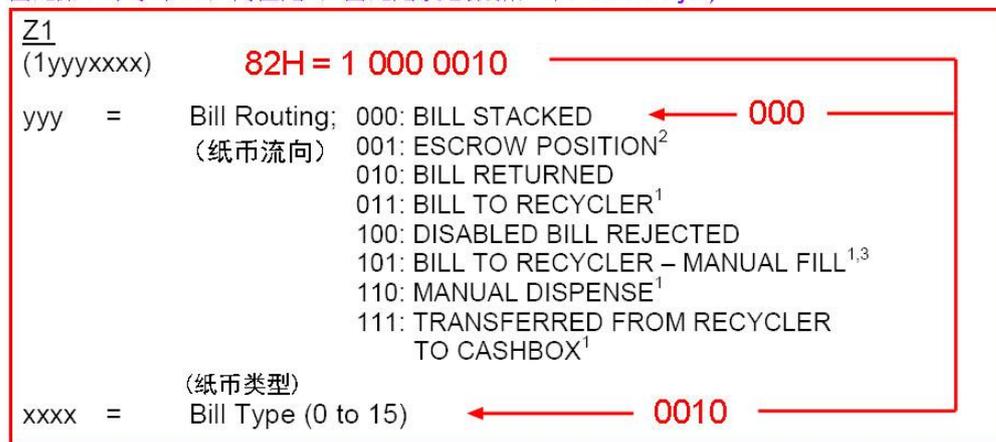
\$1 CANADIAN*	08 44 00		
\$2 CANADIAN*	08 45 00		
* Dollar coins are routed directly to the cash box			
<b>Coin Dispensed Manually</b>			
NICKEL	08 90 00	08 90 06	08 90 4C
DIME	08 91 00	08 91 08	08 91 6B
QUARTER	08 92 00	08 92 06	08 92 4B
QUARTER (1)	08 92 00	08 92 06	08 92 15
Coin Rejected			
NICKEL	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		

<b>MDB STATUS</b>	
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

\* 如何理解纸币器的货币数据?

当插入一个纸币后, 我们收到数据: 30 82 09 (MDB协议的p94 和p95页)

第一个字节: 30, 这个字节是设备ID, 代表这个数据来自纸币器  
 第二个字节: 82, 这个字节是有效数据 (不同的货币面值会有不同的数据)  
 我们把这个字节82分解成二进制数据比看一下数据的每个位, 即: 1000 0010  
 (最高位这个1, 表示这个是一个货币数据 "currency data", 因为其它状态数据, 高位都是0  
 因此第三个字节09, 高位是0, 因此是状态数据, 即: status byte)



第三个字节 09的含义, 我们可以对照MDB协议的第95页

(00001001) = Validator Disabled<sup>2</sup> - The validator has been disabled, by the VMC or because of internal conditions.

\* 如何理解硬币器的货币数据?

当我们投入一个硬币后, 收到数据: 08 52 09 (看MDB协议的P66和P67页)

第一个字节的数据: 08, 是设备ID, 表示数据来自硬币器  
 第二个字节: 52, 表示有效货币数据 (不同面值的硬币, 会有不同的数据)  
 我们把数据52分解为二进制数据: 0101 0010  
 (左数第七位是1, 就是: Coin Desoposited Data, 因为状态Status数据, 前三个位都是0.  
 投币数据由两个字节组成, 因此第三字节09, 就是钱管中该硬币的数量

Coins Dispensed Manually:

Z1      Z2  
 (1yyyxxxx) (zzzzzzzz)

yyy = 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 测试 MDB 刷卡器 (NAYAX, PAX 等)



重要提示：如果用户想使用 Nayax 读卡器进行测试，必须确保您的读卡器已经连接到 nayax 远程服务器系统。如果没有，那么 MDB 可能无法测试它。

### MDB 刷卡器有 6 个工作状态:

#### **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

### 指令列表:

1401	使能读卡器	1400	禁止读卡器
1402	取消读卡	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



## 开发过程中的重要说明:

1. 有时将命令发送到 mdb-rs232 时，但我没有得到任何回复。或者收到的是 FF 回复，我如何知道命令是否已成功发送到支付设备??

答：在 MDB-RS232 盒子设计中，当 PC 向其发送命令时，它会将命令转发到支付设备。但有时支付设备繁忙，例如插入纸币后纸币器正在收钱，支付设备无法及时回复盒子。所以 PC 无法得到回复，那么只需要稍后重新发送命令。当然，这不是问题，因为所有来自 PC 的命令都不是实时命令。

2. 我如何知道付款设备是启用还是禁用?

答：如果付款设备处于禁用状态，纸币器将始终向 PC 报告一次状态。它将向 PC 报告 30 09 或 30 03 09，并且对于大多数硬币器，如果是需要读取状态，用户可以使用 0F05 命令读取状态。所以当刚开机时，纸币器设备会处于“禁用”状态，会收到这样的状态报告。如果我们发送使能命令 34FFFF0000，纸币器将进入“已使能”状态，PC 将不会收到这样的状态报告。

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

3. 我是否可以使用 PLC 或安卓版作为主机连上 MDB-RS232 盒子?

答：没问题。实际上，用户可以使用任何具有 RS232 或 USB 端口的主控制器作为 VMC。当然，如果主控制器只有 USB 端口，那么将使用 USB 版本。并安装 USB 芯片 FT232RL 驱动程序以在主控制器中创建虚拟 com 端口。

4. 是否可以提供更多文档以方便应用程序的开发?

答：我们提供的 MDB 盒子是转发 MDB 支付设备上传到 PC 的数据。我们提供的文档只能使客户尽可能容易地了解发送和接收数据的过程以及数据的结构。至于主机的命令，以及

用户主机软件的设计，要求用户在详细了解支付设备和 MDB 协议后，设计控制流程。我们无法在客户特定的项目和软件设计方面提供帮助。

## 5. 关于硬币接收器的状态读取命令（0F05 命令读取状态）

并非所有硬币器都能上报完整的工作状态，尤其是旧版本，即使在 Level03 中，也无法提供正确的状态报告数据。通常开机后，应报 01 00 06 00 07（01/00 表示通电后）

再次读取状态应报告 06 00 06（上电后硬币接收器的初始状态为禁用状态）。

使用命令 **0CFFFFFFF** 使能后，读取状态，应该是报告 **03 00 03**

必须注意：有些硬币验证器总是用 030003 报告，无论处于什么状态，有些只用 000000 报告。在不遵循 MDB 协议的情况下，请咨询硬币器制造商。不是 MDB-RS232 盒子可以解决的问题

## 6. 如何使用 0F02 找零功能去找零硬币

按照已经示例去找零硬币：

✚ 首先从设置命令 09 读取配置数据

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

从这个配置数据中，可以看出能接收两种类型的硬币，硬币价值分别为 0.5\$ 和 1\$

✚ 然后使用 0A 检查管状态，哪种对应类型的硬币，钱管中有多少个可以找零

比如：00 00 **03 01** 00 00 00 00 00 00 00 00 00 00 00 00 00 04

那么两种类型的硬币分别是：3 枚 0.5\$ 和 1 枚 1\$

因此，在程序设计的收，必须不断检查 TUBE（钱管）的状态，以便实时知道可以支付多少硬币。

✚ 这样就可以使用找零指令

0F 02 01 找一个 0.5\$ 的硬币

0F 02 02 可以找一个 1\$ 和两个 0.5\$ 硬币（总金额是 2\$）

也就是必须要通过 0A 指令检查钱管的状态，才能去进行找零

（一定要注意，绝对不是自己放几个硬币到钱管里，就能找出几个，一定要通过 0A 指令，动态地检查钱管里硬币的数量）

## 7. 使用硬币器时，最初手动放入了一些硬币，但是当使用 0A 命令时，返回的数据值与硬币数量的变化不对应。

通常，硬币器的钱管，当里面硬币数量很少时，无法正确检测硬币的数量。也就是说，每个钱管都需要投入最少的硬币数量，才能确保硬币传感器可以检测到硬币。

此外，不要指望放入几个硬币并使用 0A 命令来准确检测几个硬币。0A 命令的正确用法是检测硬币器运行过程中的动态数值变化。例如，在操作过程中，如果放入硬币，0A 返回值将相应增加。当我们使用 PAYOUT 找零命令支付硬币时，我们也使用 0A 命令来确认更改成功和相应的硬币价值变化。

了解这一点可以更好地使用 0A 命令。

## 8. 之前版本 MDB-RS232 和 2020 版 MDB-RS232 有什么区别？

之前版本的 MDB-RS232 会主动向连接的 MDB 硬币器、纸币器和非现金设备发送启用命令，但 2020 版 MDB-RS232 不会去主动使能，只是轮询从属支付设备，VMC 将需要自己发送使能命令。因此，如果我们将硬币器和纸币器连接到 MDB 盒子并通电，即使 PC 没有连接，之前的 MDB-RS232 适配器会使能硬币器和纸币器，并准备接受硬币或纸币。但是 V2020 版本，开机后，所有支付设备都处于禁用状态。

### 9. 用户是否可以从 PC 发送轮询命令？

目前，我们的设计是通过 MDB-RS232 盒子完成 POLL 命令。不需要从 PC 发送 POLL 命令。PC 只需要处理剩余的命令并接收相应的数据即可。

### 10. 如果使用 MDB-RS232® 去测试纸币循环器（找零器）？

我们有一个文件可以下载后参考：

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

### 11. MDB-RS232 适配器是否支持年龄验证设备接口？

由于法律限制，某些产品通过自动售货机售卖时，需要检查客户年龄。年龄和规则因国家而异。比如在某些国家/地区与香烟或酒精产品有关。或者某些服务或产品内容可能会限制不同的年龄。MDB 标准协议里提供了这样的接口设备，我们也设计了专业版版本的 MDB-RS232 盒子，可以支持年龄验证接口。但并非所有 MDB 适配器都默认支持它。订购时需要咨询。

### 12. MDB 适配器可以支持两种无现金支付设备吗？

MDB-RS232 盒子启动后，它将对几个 MDB 设备进行轮询和检测，包括硬币器（地址：08H），纸币器（地址：30H），非现金支付设备 1（地址：10H）和非现金支付设备 2（地址：60H）。此外，我们特定的“PRO”版本还将支持 AGE 验证设备（68H）和 USD 设备（40H）。当然，大多数客户只需要购买标准设备。

需要注意的是，我们还有一个面板上带有三个 LED 的版本，也被许多客户使用，但仅支持硬币器，纸币器和非现金设备 01。

因此，如果您需要支持更多设备，请注意购买最新版本，或在订购前咨询客服。

本手册是简要的介绍了 MDB 转接适配器的使用和简要指令，具体请参考详细的纸币器，硬币器等 MDB 协议。

更丰富的指导文本我们会不定时更新或者发布在我们的产品网站上：

<http://www.mdb-rs232.com> 或者 <http://www.mdb-rs232.cn>

Email: [wafer@waferstar.com](mailto:wafer@waferstar.com)

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

Tel: 0086-21-68458945

Also accessible via our online skype service: wafer-service

Copyright 2014, WEFU MDB  
V2024M05-V2.1