

MDB-RS232 Commands for Controlling the Coin Validator

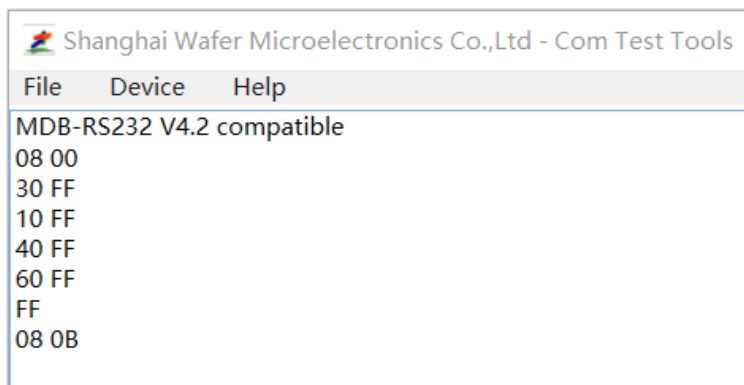
Command	HEX code	Description
RESET	08H	Command for changer to self-reset
SETUP	09H	Request for changer setup information
TUBE STATUS	0AH	Request for changer tube status
POLL	0BH	Request for changer activity status
COIN TYPE	0CH	Signifies coin types accepted and allowable coin dispensing. This command is followed by setup data. See command format section
DISPENSE	0DH	Command to dispense a coin type. Followed by coin type to dispense
EXPANSION	0FH	Command to allow addition of COMMAND features and future enhancements. Changers at feature level 2 do not support this command

- For the MDB-RS232 adapter, the adapter itself will perform the POLL (0BH)command to read the data and status of the coin validator. When the coin validator receives a coin or has a status that needs to be reported, it will send the corresponding data to the VMC board through the MDB-RS232 adapter in the corresponding time period. Therefore, the VMC board software engineer doesn't need to consider the POLL command and the specific timing, but only needs to send the RESET command when needed, The SETUP reads the configuration, and then enables the corresponding type of the coin through the COIN TYPE according to the SETUP data. Then, as long as there is a coin accepted or coin change, the TUBE STATUS command is used to query the status of the number of coins (data preparation for subsequent change commands)
- Change commands, generally as long as the 0F02 instruction is applicable, generally not applicable to the 0DH instruction

Next, we will briefly introduce the development command sequence of the coin validator:

For example, after the MDB-RS232 is powered on, we will receive the RESET success data (the following test data is only connected to the coin validator):

After the MDB-RS232 is powered on, it will actively send a 08H command to the coin machine to reset, and if the device exists, it will reply to 00 to confirm. When the device is RESET, 0B is reported. Therefore, when MDB-RS232 is used to control the payment device, we only need to judge whether the coin machine is working properly based on the data and status reported by MDB-RS232.



```

Shanghai Wafer Microelectronics Co.,Ltd - Com Test Tools
File Device Help
MDB-RS232 V4.2 compatible
08 00
30 FF
10 FF
40 FF
60 FF
FF
08 0B

```

- Next, we start to read the configuration data of the coin validator
 Send 09H (read the configuration parameters, the parameters of these replied byte are quite important and must be understood)
 Example of a reply: 031156050100030102000000000000000000000000000076
 Send 0F00 (read the device ID, these parameters have nothing to do with calculating the price, including the Device code,firmware version, manufacturer model, etc., we generally report these parameters to the server when we do DEX data reports, which is easy to manage)
 Example of a reply: 49435430303031323334353637383943434D362D545741204D444201000000000366
 Send 0F01 (Full command example: 0F0100000001) to enable the Optional Features in the 0F00 parameter description
 Example of a reply: 00
- During operation and when the coin data changes, the 0AH command is used to continuously detect the change of the quantity in the coin tubes
 0A
 Example of a reply: 00
 The first two bytes with 16 bits indicate which tube is full, and the last 16 bytes indicate the number of coins in the 16 tubes

Now, we have already completed the basic configuration and reading

- Next, we need to send the corresponding enabling command according to the type of coins we need to receive (or inhibit the receipt of coins), such as
 0CFFFFFFF (Once this command is sent, the coin validator can accept the coins)
 Example of a reply: 00
 0C0000000 (If change is in progress, or the VMC is under maintenance, then inhibit the coin validator)
 Example of a reply: 00

Next, just wait for MDB-RS232 to report the data when there is a coin deposit or when the coin status changes. Of course, the VMC engineer will still apply the 0A command every few seconds to read the status of the coin tube in real time. It can know the status of the coins in the coin tube at any time, providing an accurate basis for change.

Received coin data: 08 51 01 (We can look at the 0BH reply in the MDB protocol)
 Receive status change data: 08 01 (For example, An escrow lever activation has been detected)
 08 02 (找零 BUSY)

How can we tell the difference between coin data and status data?

The coin data is composed of two bytes, and the first byte must be greater than 40H. So we receive the data, starting with ID data 08, it is determined that it is the data of the coin validator, followed by 51, which is the coin data, and the decomposition into binary data is 0101 0001 00000001

Coins Deposited:

<u>Z1</u> (01yyxxxx)	=	<u>Z2</u> (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.

(According to the explanation of the screenshot of the MDB protocol, it is: 01 type of coin, enter the coin tube, and there is now 1 coin of this type in the coin tube 01)

- Next, let's introduce the following change cycle (payout)

For example, 0F0206 (0F02 is a fixed change command data, and the following 06 is the value of the currency to be paid out, of course, this currency value is the converted value with scaled and decimal)

Example of a reply: 00

During the change (payout) process of the coin validator, the status may be reported at the same time, such as BUSY, Tube Jam and other status

The following is a list of standard command sequences for change:

0A	Reply: 000000020002
	Read the status of the coin tubes to determine whether there is enough to be paidout
0C00000000	Reply: 00 Coin validator is disabled before giving change
0F0206	Reply: 00 Start sending payout command
0F04	Reply: 0000 Read the amount that has been changed, if it has not been found, it is 00 00
0F04	Reply: 00 Only one 00 is replied, which means that the change payout is finished.
0F03	Reply: 00 03 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 03
	Finally, check the Numbers of each coin type paid out
0CFFFFFFF	Reply: 00 Enable the coin validator again

Next, we introduce the Coin's monetary value calculation based on configurations

For example, with the 09H command, we read back the configuration data:

031156**0501****0003**01020076

That is, Coin Scaling Factor and Decimal Places is **0501** in red, and Coin Type Credit is 01 and 02 (0003 means that the last two digits of the coin can be received into the money tube)

Therefore,

1. The value of coin type 0 is 01: Multiply by the coin scale factor is $01 * 5 = 5$, and then divide by 10 (1 decimal place to the left), that is, so the coin's monetary value is 0.5
2. The value of coin type 0 is 02: Multiply by the coin scale factor is $02 * 5 = 10$, and then divide by 10 (1 decimal place to the left), that is, so the coin's monetary value is 1.0

(As can be seen here, our coin maker can receive a \$0.5 and a \$1 coin)

3. The Payout command **0F0206** in the above example where 06 is also converted to $06 * 5 = 30$, and then divided by 10, which is \$3.0 of change