RS232/USB Interface for PULSE-PC
User Manual

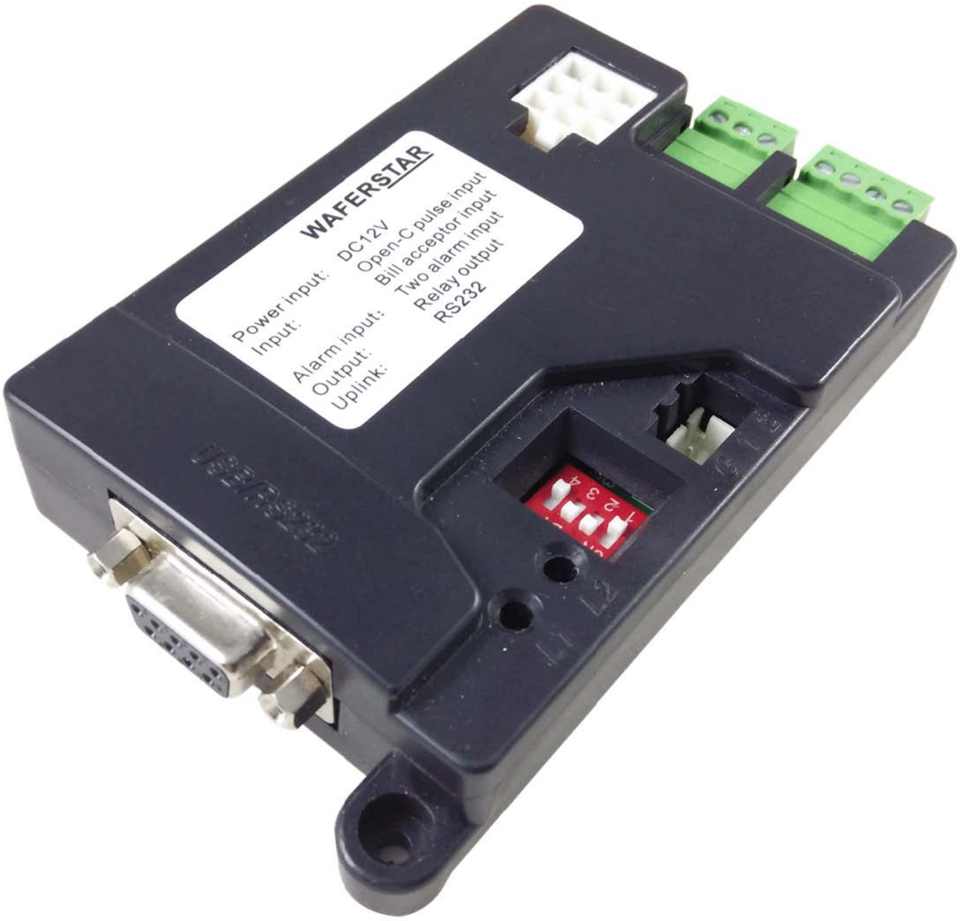
Version 2.2-2017-10-02

1.0 Introduction

The PULSE-PC is an RS232 interface module for most pulse type coin acceptor or bill acceptor products. The PULSE-PC has a built-in true RS232 port on its uplink communication port to interface with the serial port on a PC and a downlink port with inhibit line or inhibit relay to interface with validators. With the PULSE-PC, the end operator can treat all different kinds of validators as a standard RS232 device and can concentrate on the real operation of the validator rather than the tedious interface work of the hardware discrepancy. On the PC platform, the software developer can easily program the validators via the PULSE-PC with common computer languages such as Visual Basic / C++.

For USB version,that has a USB to RS232 chip on board to convert the RS232 to USB and build a virtual RS232 port in the PC. And user firstly need to install a driver in the PC for USB / RS232 chip. And the USB chip FT232RL driver download address:

http://www.ftdichip.com/Products/ICs/FT232R.htm



(Picture is the PULSE-PC RS232 Version)

2.0 Compatibility

PULSE-PC is a successor of the WF-700 model and the updated version of WF-700. PULSE-PC is 100% functionality compatible to WF-700. With the DIP Switch, a more versatile configuration can be setup to interface with over 90% of the most commonly used banknote / coin validators found in the market.

3.0 Features

- PULSE-PC has bill acceptor and coin acceptor credit pulse input. now the bill acceptor hardware interface is customized for ICT pulse bill acceptor.We also provide other customized service for other brand bill acceptor or coin acceptor.
- PULSE-PC has two alarm input port,that alarm is switch type input,that would be easily used for device status detection. when Alarm1 or Alarm2 is shorted to GND ,then alarm is triggered and the status changed data would be reported to the PC
- PULSE-PC has one relay output,that can be controlled with the COMMAND sent from PC
- WAFER Provide optional SDK and technical support.

4.0 Communications

This interface is a three wire interface using a Transmit line (TXD), a receive line (RXD) and a Common line. The Host machine is thought of as being the Master device. The PULSE-PC is the slave device that responds only to the polls (or requests) from the Master.In this polled system, the Master requests information from the PULSE-PC at a periodic rate. This rate can be as fast as every 100 milliseconds or as slow as the elapse time defined by the Lockout Timer. Faster rates are more desirable since overall system performance (in terms of bills/coins per minute accepted) will be higher at high poll rate system.

PULSE-PC has a “Lockout” Timer feature that it will disable the acceptor if the specified poll rate is not met. In another words, if the host system is too busy to keep up with the desired minimum poll rate, PULSE-PC will put the acceptors into the Disabled mode. The acceptor will be enabled again once the communication resumes.

5.0 GENERAL DATA

Baud Rate:	- 9600
Duplex:	- Half Duplex
Character Format:	1 start and 1 stop bit,,8 data bits (Bits 0 = sent first (LSB)

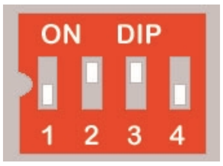
Lockout Timer

If the PULSE-PC does not receive a poll after an elapse time specified by the hardware jumper, it will:

- ✧ Suspend (Disable) the acceptor.
- ✧ Return any note being held in escrow (if applicable)

This Lockout Timer has a selection of 1 second ,10 second, 20 seconds or no lockout specified by the setting of a DIP Switch.

For normal ,we select the 1s:



Credit stack, Parallel Mode

PULSE-PC has a 6-level of credit stack in parallel mode. The stack is a FIFO type and holds up to 6 distinct credits and status information from the acceptor. On a slow polling system or whenever a communication loss occurs, credits issued by the acceptor are temporary stored in the PULSE-PC ram and to be released to the Master on future polls. When the 2nd level of stack is entered, PULSE-PC will disable the acceptor to avoid taking in any new tokens. Any credits already on the way sending from the acceptor will be piled up in the PULSE-PC stack.

Pulse chain, Pulse Mode

PULSE-PC has a pulse counter that can hold up to 261 pulses. Each pulse received from the acceptor is treated as an individual event. Thus, in another word, if a $20 dollar bill is accepted by a bill acceptor, 20 individual credits will be reported by the PULSE-PC in 20 consecutive events in its uplink communication. If the number of pulses is larger than 261, and the system is not polling fast enough to clear the stack, a stack overflow situation will occur. In such a case, the PULSE-PC will output the inhibit to Lock the coin selector.

6.0 MESSAGE FORMAT, UPLINK COMMUNICATION

Format: STX, Length, MSG Type and Ack #, Data Fields….., ETX, Checksum

Descriptions are as follows:

STX - 02h One byte indicating Start of message

LENGTH - One byte representation of the number of bytes in each message (binary), including the STX, ETX ,the Checksum and LENGTH

MSG TYPE and ACK # - One byte of Data

MSG Type - (Bits 4-6)

001 - for Master to PULSE-PC Message

010 - for PULSE-PC to Master Message

011 to 111 - reserved for future.

ACK # - (Bits 0-3)

00h

(So data from PC to PULSE-PC will be 10h and from PULSE-PC to PC will be 20h)

In the messages sent by Master, the Ack # is used to identify the message. The Ack # alternates between 00 and 01h. If the PULSE-PC receisves two consecutive messages with the same number, the second message is treated as a retransmission of the first message.

In the messages sent by PULSE-PC, the Ack # number is set the same as in the Master message to indicate the successfulness of the current message. If the PULSE-PC receives a message incorrectly (wrong checksum), the received message will be discarded and no message will be sent back.

DATA - The data portion of the message consists of the multiple data fields. We will discuss it in the section of DATA FIELDS.

ETX - 03h One byte indicating End of message.

CHECKSUM - One byte checksum. The checksum is calculated on all bytes except the STX, ETX, and checksum byte itself. The calculation is done by XORing the bytes.

(Data from PC to PULSE-PC ,checksum no need to be exactly calculated)

For example data:	02 0B 20 **01** 10 **01** 00 00 01 03 3A
Checksum	3A= 0B⊕20⊕01⊕10⊕01⊕00⊕00⊕01

6.1 PC-to- PULSE-PC communications

Example: 02 08 10 03 20 00 03 77

(Checksum data from PC to PULSE-PC , no need to be exactly calculated,so can always use 77h)

Data Fields (02 08 10 will be fixed data bytes):

BYTE 0

For PULSE-PC

Set to 00H – Disable Bill acceptor and coin acceptor

Set to 01H – Enable ICT Bill acceptor but Disable coin acceptor

Set to 02H – Enable coin acceptor,but Disable ICT Bill acceptor

Set to 03H – Enable both payment input (other data will also do this enable status)

Other data segment will be used when upgrading the device,Please consider follow-compatible design

BYTE 1

For PULSE-PC

Set to 10H - Relay Output is switched ON

Set to 20H - Relay Output is switched OFF

Other data segment will be used when upgrading the device,Please consider follow-compatible design

BYTE 2

For PULSE-PC models

Set to 00h

02 08 10 **00** 20 00 03 77	Inhibit both acceptors
02 08 10 **01** 20 00 03 77	Enable ICT acceptor
02 08 10 **02** 20 00 03 77	Enable pulse acceptor
02 08 10 **03 10** 00 03 77	Enable both acceptors and Switch on the Relay output

6.2 PULSE-PC-to-PC communications

Example: 02 0B 20 01 10 00 00 01 01 03 3A

Data Fields (02 0B 20 will be fixed data bytes):

BYTE 0

For PULSE-PC

Set to 01h – Nothing to report

Set to 10h – Credit was accepted

BYTE 1

For all PULSE-PC models

Set to 10h

BYTE 2

For all PULSE-PC models:

Bit 0 - Power up (= 1 if PULSE-PC experienced a reset since the last poll)

Bit 1 - Invalid command (= 1 if invalid command received)

Bit 2 - Failure (= 1 if acceptor has alarm input2 Report , for example, failed or cashbox full)

Bit 3-5 Credit Channel Field
000 = None
001 = 1st credit channel type
010 = 2nd credit channel type
011 = 3rd credit channel type
100 = 4th credit channel type
101 = 5th credit channel type (Bill acceptor credit report)
110 = 6th credit channel type, Pulse channel (Coin acceptor credit report, WF-700B in)
111 = 000 (Reserved)
Bit 6 - Failure (= 1 if acceptor has alarm input1 Report , for example, failed or cashbox full)
Bit 7 - Reserved (set to 0)

For example,pc will receive the following data:

02 0B 20 **01** 10 **01** 00 00 01 03 3A	The first power up, No data report
02 0B 20 **01** 10 00 00 00 01 03 3B	No data report
02 0B 20 10 10 **30** 00 00 01 03 1A	Coin acceptor payment report
02 0B 20 10 10 **28** 00 00 01 03 02	Bill acceptor payment report
02 0B 20 10 10 **04** 00 00 01 03 2E	Alarm input2 report
02 0B 20 10 10 **40** 00 00 01 03 6A	Alarm input1 report

BYTE 3, BYTE 4

Credit Left in the stack,that need to be reported

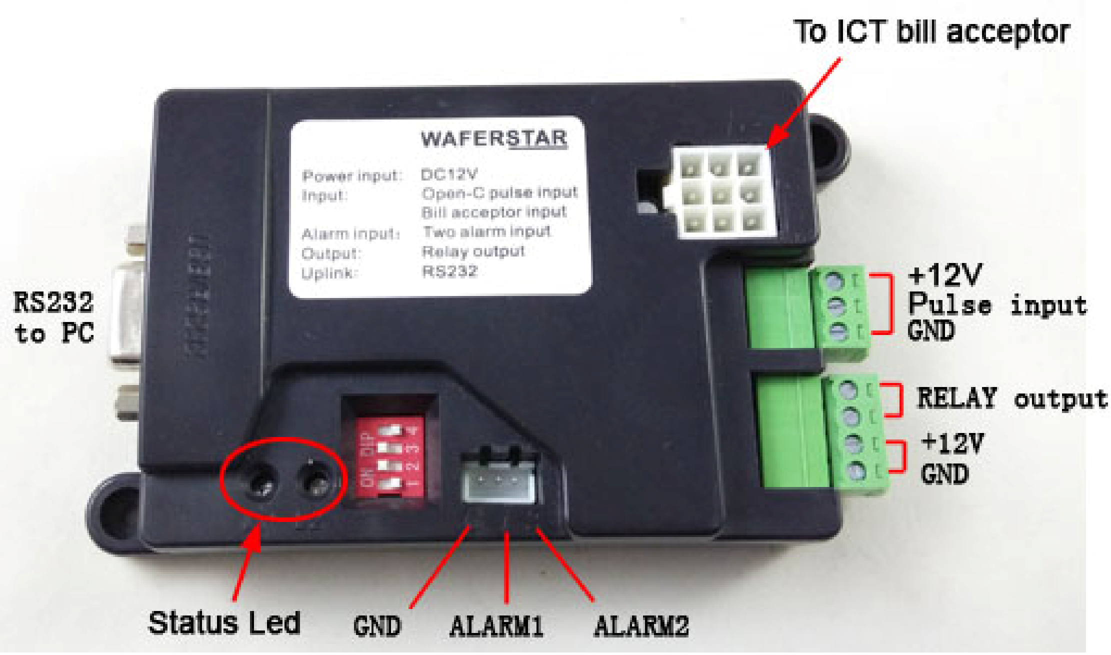
(If the adapter only received one pulse,then this would be 00h,because after the current report data,there is no credit left to report)

02 0B 20 10 10 **30** 00 00 01 03 1A	Coin acceptor payment report

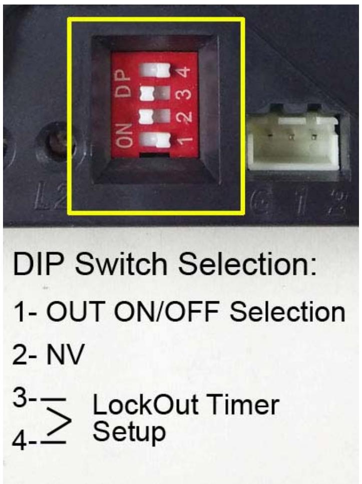
BYTE 5

Set to 01H

7.0 ELECTRICAL HOOKUP FOR PULSE-PC



7.1 Default DIP SWITCH Settings



DIP Switch Selection:

1- OUT ON/OFF Selection

2- NV

3- } LockOut Timer Setup

4-

Lockout timer DIP switch Configuration

4	3	Lockout Time
OFF	OFF	No Lockout
OFF	ON	1 second
ON	OFF	10 second
ON	ON	20 second

Inhibit Level Configuration

When DIP Switch 2 is ON, then Inhibit Level Output is LOW

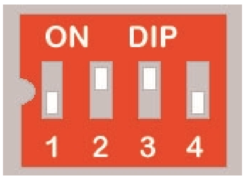
When DIP Switch 2 is OFF, then Inhibit Level Output is HIGH

For example:

Setting the DIP switch as the right picture:

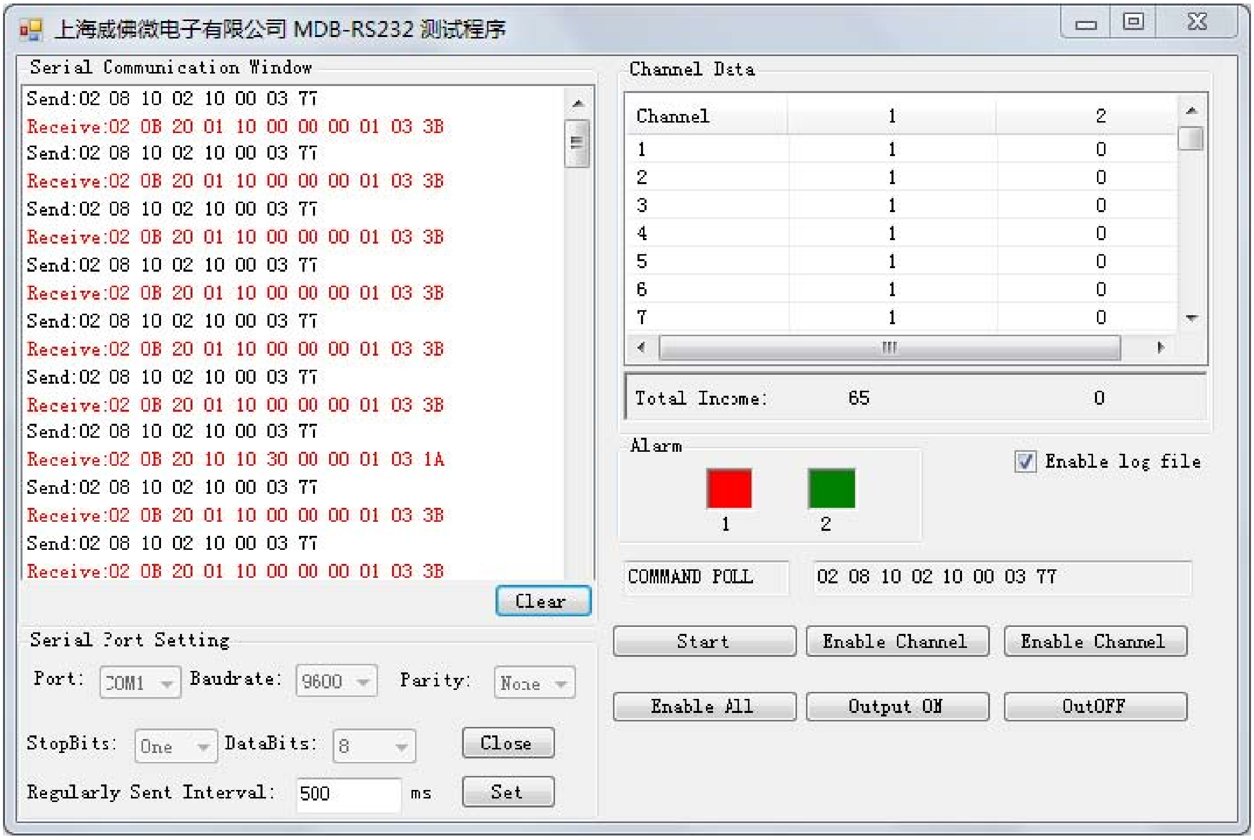
Lockout Time is: One second

Relay output is shorted when power on



7.2 WAFER Demo and test software

(This demo test software can be downloaded from wafer company website for fast development support.If you want to get the source code of this test software, please contact Wafer for order the SDK kits)


上海威佛微电子有限公司 MDB-RS232 测试程序

Serial Communication Window

```
Send:02 08 10 02 10 00 03 77
Receive:02 0B 20 01 10 00 00 00 01 03 3B
Send:02 08 10 02 10 00 03 77
Receive:02 0B 20 01 10 00 00 00 01 03 3B
Send:02 08 10 02 10 00 03 77
Receive:02 0B 20 01 10 00 00 00 01 03 3B
Send:02 08 10 02 10 00 03 77
Receive:02 0B 20 01 10 00 00 00 01 03 3B
Send:02 08 10 02 10 00 03 77
Receive:02 0B 20 01 10 00 00 00 01 03 3B
Send:02 08 10 02 10 00 03 77
Receive:02 0B 20 01 10 00 00 00 01 03 3B
Send:02 08 10 02 10 00 03 77
Receive:02 0B 20 10 10 30 00 00 01 03 1A
Send:02 08 10 02 10 00 03 77
Receive:02 0B 20 01 10 00 00 00 01 03 3B
Send:02 08 10 02 10 00 03 77
Receive:02 0B 20 01 10 00 00 00 01 03 3B
```

Serial Port Setting

Port: COM1 Baudrate: 9600 Parity: None

StopBits: One DataBits: 8

Regularly Sent Interval: 500 ms

Channel Data

Channel	1	2
1	1	0
2	1	0
3	1	0
4	1	0
5	1	0
6	1	0
7	1	0

Total Income: 65 0

Alarm

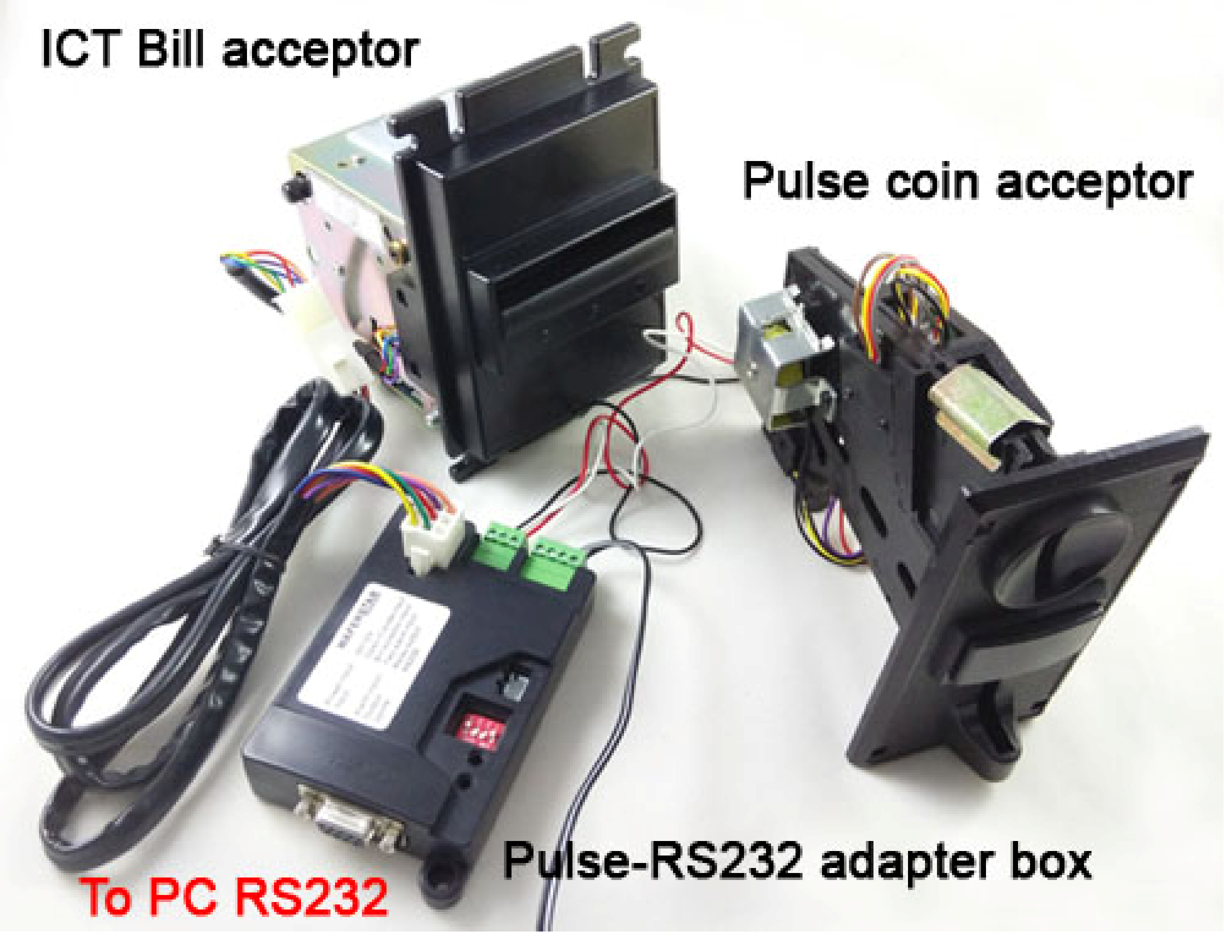
🟥 1 🟩 2

☑ Enable log file

COMMAND POLL 02 08 10 02 10 00 03 77

(Demo or SDK of this test software is only for sample,not for user's projects)

7.3 Sample Wire Diagram Map (USB version is the same connection)



7.4 Important for software development

✧ About the data received

When pc received the data bytes,sometimes will some interference bytes in front of the correct data,for example you should receive the " 02 0B 20 10 10 00 00 01 01 03 3A ",but in fact you maybe receive the following bytes in your com port:

FF 02 0B 20 10 10 00 00 01 01 03 3A

FE 02 0B 20 10 10 00 00 01 01 03 3A

FF FE 02 0B 20 10 10 00 00 01 01 03 3A

So must received the 02h and also followed with 0Bh, then will be determined as the correct data byte.

✧ Why I cann't input the coins after powered on the PULSE-PC adapter box?

(1) Check the power supply is DC12V and properly connected to the PULSE-PC box

(2) Even after powered on,if no PC polling data received,PULSE-PC box will inhibit both acceptors

(3) Don't put the coin acceptor on the desk, because for all the coin acceptors if the previous coin is blocked in the exit,then the acceptor will be inhibited and cann't insert more coins.so you must keep the exit smooth when you start to insert the coin.