RS232 Interface for Coin Validators

WF-700-RELAY User Manual

Version 2.1-2010-09-13

1.0 Introduction

The WF-700-RELAY is an RS232 interface module for most pulse type coin acceptor or bill acceptor products. The WF-700-RELAY 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 WF-700-RELAY, 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 WF-700-RELAY with common computer languages such as Visual Basic / C++.

2.0 Compatibility

WF-700-RELAY is a successor of the WF-700 model and the updated version of WF-700B. WF-700-RELAY is 100% functionality compatible to WF-700B. 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 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 WF-700-RELAY 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 WF-700-RELAY at a periodic rate. This rate can be as fast as every 30 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.
WF-700-RELAY 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, WF-700-RELAY will put the acceptor into the Disabled mode. The acceptor will be enabled again once the communication resumes.

4.0 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 WF-700-RELAY does not receive a poll after an elapse time specified by the hardware jumper, it will:

1. Suspend (Disable) the acceptor.
2. 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.

Credit stack, Parallel Mode

WF-700-RELAY 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 WF-700-RELAY ram and to be released to the Master on future polls. When the 2nd level of stack is entered, WF-700-RELAY 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 WF-700-RELAY stack.

Pulse chain, Pulse Mode

WF-700-RELAY 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 WF-700-RELAY 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 WF-700 will output the inhibit to Lock the coin selector.

5.0 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 and the Checksum.

MSG TYPE and ACK # - One byte of Data

MSG Type - (Bits 4-6)

001 - for Master to WF-700-RELAY Message

010 - for WF-700-RELAY to Master Message

011 to 111 - reserved for future.

ACK # - (Bits 0-3)

00h or 01h

In the messages sent by Master, the Ack # is used to identify the message. The Ack # alternates between 00 and 01h. If the WF-700-RELAY 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 WF-700-RELAY, the Ack # number is set the same as in the Master message to

indicate the successfulness of the current message. If the WF-700-RELAY 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.

5.1 PC-to-WF-700-RELAY communications

Example: 02 08 10 7F 10 00 03 77

Data Fields:

BYTE 0

For WF-700-RELAY

Set to 00h – Disable acceptor

Otherwise – Enable acceptor

For future models with acceptor has programmable acceptance or escrow function.

Bit 0 - Enable 1st denomination channel
Bit 1 - Enable 2nd denomination channel
Bit 2 - Enable 3rd denomination channel
Bit 3 - Enable 4th denomination channel
Bit 4 - Enable 5th denomination channel
Bit 5 - Enable 6th denomination channel
Bit 6 - Enable 7th denomination channel

Set to 00h – Disable acceptor

BYTE 1

For WF-700-RELAY

Set to 10h

For future models

Bit 0 - Reserved (set to 0)
Bit 1 - Not used (set to 0)
Bit 2 - Not used (set to 0)
Bit 3 - Not used (set to 0)
Bit 4 - Reserved (set to 1)
Bit 5 – Escrow Stack (= 1 causes bill to be stacked)
Bit 6 – Escrow Return (= 1 causes bill to be returned)

BYTE 2

For all WF-700-RELAY models

Set to 00h

5.2 WF-700-RELAY-to-PC communications

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

Data Fields:

BYTE 0

For WF-700-RELAY

Set to 01h – Nothing to report

Set to 10h – Credit was accepted

Bit 0 - Idling (= 1 if WF-700-RELAY have nothing to report)
Bit 1 - Reserved
Bit 2 - Reserved
Bit 3 - Reserved
Bit 4 - Stacked (= 1 if a credit was accepted)
Bit 5 - Reserved
Bit 6 – Reserved

For future models
Bit 0 - Idling (= 1 if nothing to report)
Bit 1 - Reserved
Bit 2 - Escrowed (= 1 if a banknote is in Escrow)
Bit 3 - Reserved
Bit 4 - Stacked (= 1 if a banknote was accepted)
Bit 5 - Reserved
Bit 6 - Returned (=1 if a banknote was returned)

BYTE 1
For all WF-700-RELAY models
Set to 10h

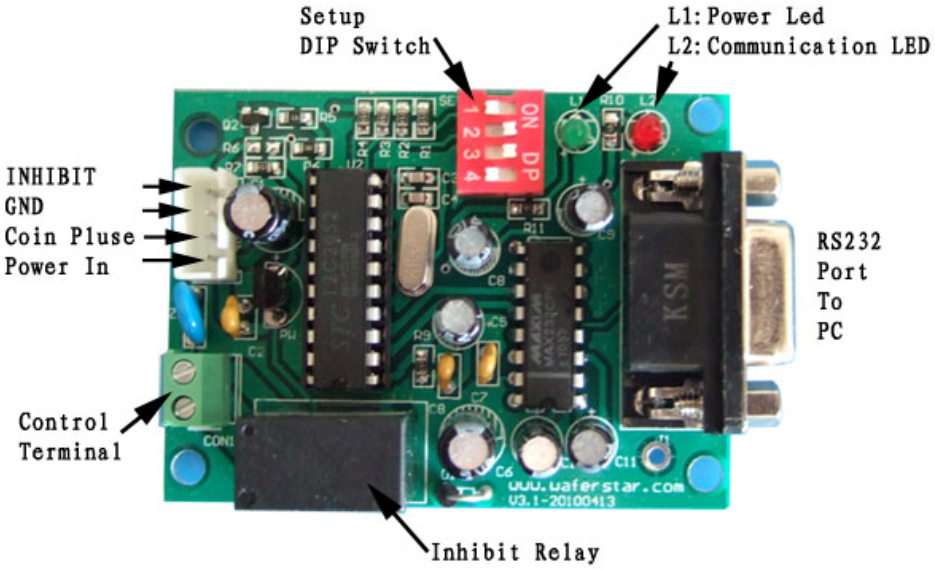
BYTE 2
For all WF-700-RELAY models
Bit 0 - Power up (= 1 if WF-700-RELAY experienced a reset since the last poll)
Bit 1 - Invalid command (= 1 if invalid command received)
Bit 2 - Failure (= 1 if acceptor has failed or cashbox full)
Bit 3-5 Credit Channel Field
000 = None
001 = 1st credit channel type
010 = 2nd credit channel type
011 = 3rd credit channel type
100 = 4th credit channel type
101 = 5th credit channel type
110 = 6th credit channel type, Pulse channel (WF-700 use this channel type)
111 = Reserved
Bit 6- Reserved (set to 0)

BYTE 3, BYTE 4

Credit Left in the stack

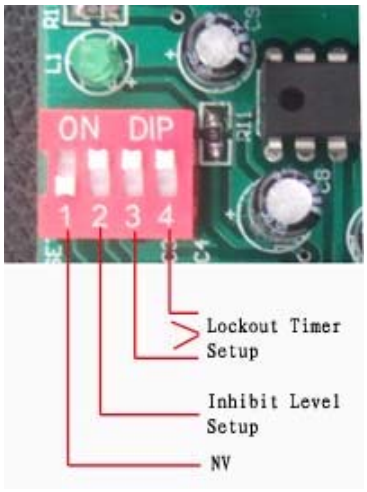
BYTE 5
Set to 01h

6.0 ELECTRICAL HOOKUP



6.1 Default DIP SWITCH Settings

(DIP Switch 1 is the simulator for the problem of the coin acceptor)



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

6.2 Inhibit line input

We have two way to inhibit the coin acceptor, for the simple coin acceptor without the inhibit line ,then you can use the relay to control the power supply of the coin acceptor.

And if have the inhibit line,then you can use the inhibit line to inhibit the coin acceptor.

6.3 Sample Wire Diagram Map

NV