Application Note

How to communicate with networked IBC readers using Multidrop Programming

Description

This document explains the use and programming of IBC readers when in the multidrop (network) mode. All Smart Slot, Smart Mag, and Magbar readers manufactured before June 1995 were available with multidrop firmware as an ordering option.

After June of 1995, all “J”, SA, and STA series readers with a serial interface can be programmed by the user to change the interface protocol from standard serial to a multidrop interface. This means that if you have an RS422-interface reader, you can turn the networking protocol on or off by programming the reader. Also, if you have an RS232 reader, you can also turn on the multidrop mode which will force the reader not to send data until it has been polled. This is a way of forcing an IBC reader not to send unsolicited data to a PC until you poll it.

What is Multidrop Mode?

Multidrop mode is a mode IBC readers can be placed in which allows them to be networked together on one communication line to a computer. Multidrop readers must have an RS422 interface if they are to be connected together on one line. Each reader is internally coded with an “address” which the computer uses to talk with each reader. Each message that the computer sends is prefixed by the address of the reader you wish to communicate with, so only that reader will respond to the message.

The IBC protocol firmware allows you to address up to 127 readers. You cannot, however, connect 127 readers on one communications line because this will cause problems with the line. In order to connect more than 32 readers, you have to use a “repeater” which is available from most companies selling RS422 convertors and other types of communications equipment.

Reader Addressing

Each reader has an internal “address” which the computer uses to talk with that reader. IBC supports 127 different addresses (0 thru 126). Address 127 is a reserved address which is used to send messages to all readers in the network. The internal address for each reader is re-programmable by the user. When your computer software sends a message to the readers, every reader in the network intercepts the message and then interprets the message to see if the message is for him (by checking the address). If so, the reader will respond to the computer that it has acknowledged the message, and also send data to the computer if the computer requested it in the message.

Communication Settings

The baud rate at which you talk with IBC readers in multidrop mode can be anywhere from 1200 baud to 19200 baud. This is a programmable option which you can change in the reader. You must, however, talk to the reader in 8-bit mode, that is, 8 data bits, with no parity and 1 stop bit. This is a requirement for proper multidrop operation with IBC readers.

Protocol

Communication to and from each reader is done with an ack/nak protocol as follows:

HOST COMPUTER		READER
“ADDRESS” + COMMAND	----->	
	<-----	ACK or NAK

The reader will always send an ACK if it recognizes the command and it is a valid command. If it is an invalid command, the reader will send a NAK. An ACK is a hexadecimal 06, decimal 06, CTRL-F. Nak is a hexidecimal 15, decimal 21, CTRL-U.

Commands which are sent to a reader that require the reader to return some sort of data, such as a barcode which has just been scanned, work the same way except that after the data is transmitted, the host computer must also ACK or NAK the reader as follows:

HOST COMPUTER		READER
“ADDRESS” + COMMAND	----->	
	<-----	ACK or NAK
	<-----	“DATA”
ACK or NAK	----->	

If the reader receives a NAK from the host computer, the reader will re-send it’s transmission. Any data which is in the reader which is waiting to be sent to the computer, is not deleted until the ACK is received from the host computer. The reader will therefore assume that the data has not been received by the host computer, until it has received the ACK in response to a transmission.

A good practice, therefore, when you are writing software to communicate with these readers is to send a few ACKs to the readers when your software starts. This will clear all of the readers’ communication buffers.

The turnaround time (the amount of time it takes the reader to respond to a command, or ACK/NAK) is very quick. For “J” series readers, this time is about 50us.

Reader Addresses

The reader address is a one-byte character which is constructed as follows:

```
bit     7   6   5   4   3   2   1   0
     ______________________________________
     | 1 | x | x | x | x | x | x | x |
     ______________________________________
```

NOTE that bit 7 is always a “one”. This is required otherwise the protocol will not work. This is also why the communications must be set to 8 data bits with no parity, otherwise bit # 7 could get overwritten by the parity bit.

“xxxxxxx” refers to the reader address that you want to talk to, therefore “10000000” refers to address 0, “11111110” refers to address 126.

Command and Data Structure

Each command which is sent to a reader from the host computer is braketed by an STX and ETX. The STX is a hexadecimal 02 or CTRL-B, the ETX is a hexadecimal 03, or CTRL-C. This is a requirement in order for the multidrop protocol to work.

To send the command “!10”, which will turn on the relay in the reader for 10 seconds, would therefore be sent as follows (to reader number 0):

chr(128) + STX + !10 + ETX

Data sent from a reader to the host computer also is bracketed by the STX and ETX.

If there is no data in the reader to send to the host computer, then the reader sends a NUL (hexadecimal 00).

Poll Command

In order to receive data which is in a reader, you must send the POLL command from the host computer to the reader. The POLL command is a “?”. Therefore, if you have two readers on the network - reader # 0 and reader # 1, then to poll them, you would send the commands chr(128) + STX + ? + ETX which will poll reader # 0, and then chr(129) + STX + ? + ETX which will poll reader # 1. Of course, you must wait after the first poll for the ACK from the reader, and then the reader response, and of course send an ACK back to reader # 0. Then you are ready to poll reader # 1. An example follows:

HOST COMPUTER		READER
128 + STX + ? + ETX	----->	
	<-----	ACK
	<-----	STX + NUL + ETX (NUL = no data)
ACK	----->	
129 + STX + ? + ETX	----->	
	<-----	ACK
	<-----	STX + NUL + ETX (NUL = no data)
ACK	----->	

If there is data to be sent from the reader to the host computer, then the above NUL would be replaced by the data being sent. Please note that there are some commands which are sent to a reader which do not return data, and there are some commands which return data. Detailed descriptions of the commands can be found in the respective product manuals. Also note that address 127 (the global address) will never respond with an ACK or NAK from a reader, because only one reader can talk on the network at a time. You therefore cannot poll unit # 127, and will never receive any data from sending a command to # 127.

Wiring

The IBC 422 protocol requires a 4-wire system, 2 wires for transmit and 2 wires for receive. Our protocol will not work on a 2-wire system.