Controlling the Z-Systems z-32.32r Digital Detangler Pro Routers

Inputs to the z-32.32r are designated as <u>sources</u> and outputs are designated as <u>destinations</u>. The routers have a 9-pin connector for RS422 control input and a 9-pin connector for looping the control through to multiple units. The only pins used are RXD+, RXD-, TXD+, TXD-, and GND; no ACK signals are used. The communication rate is 9600 baud, no parity, 8 data bits, 1 stop bit.

Protocol for z-32.32r

R = 4-bit router number (up to 16 routers may be controlled from a single controller. It is important to note that routers are numbered internally from 0 to 15. If you purchased a single router, its router number is 00)

X = 5-bit source address
Y = 5-bit destination address

Each router's configuration is defined by 32 pairs of settings (X and Y pairs). Settings are changed a single pair at a time. Therefore, changing a router's settings consists of sending it X and Y pairs. Querying the router as to its settings consists of sending it Y and asking for X. It is important to note that channels are numbered internally from 0 to 31. Input 1 is addressed as input 0, 2 as 1, etc.

A5h: Controller begins sequence of commands
5Ah: Controller ends sequence of commands
RRh: Controller tells router #R it will be receiving commands

20h: Controller is about to send a new X,Y pair (5 bits for X, 5 bits for Y)
000yyyyyh: Controller sends Y value
000xxxxxh: Controller sends X value

40h: Controller is about to query router for setting
000yyyyyh: Controller gives router Y address
000xxxxxh: Router gives back X address

80h: Controller asks router if it is alive
FFh: Router responds it is alive

When sending command 40h, router should respond with 000xxxxxh within 2 milliseconds afer receiving 000yyyyyh data from controller. When sending command 80h, router should respond within 1 millisecond.

<u>Here is an example communications transaction</u>:

Suppose we wish to tell router #4 to send source #6 to destination # 32 and source #10 to destination #3. This is the chain of events.

```
A5h     ; begin communication
33h     ; talk to router #3
20h     ; controller is about to send a new X    ,Y pair
1fh     ; destination address is sent first (1f hex, 00011111 binary, 31 decimal)
05h     ; source address is sent next
20h     ; controller is  about to send a new X  ,Y pair
02h     ; destination address is sent first
09h     ; sources address is sent next
5Ah     ; end communication
```

Notice that A5h and 5Ah "bracket" the communications and must be issued by the controller before addressing a different router. For example, suppose we wish to tell router #1 to send source #9 to destination #11 and then ask router #2 which source is connected to destination #4:

```
A5h
00h     (router 1)
20h
0Ah     (destination 11)
08h     (source 9)
5Ah
A5h
11h     (router 2)
40h
03h
router #2 sends back result as 000xxxxxh
5Ah
```