

1. 1394B Speed Signaling

This is a revised proposal for a speed signaling technique. This revision incorporates the following changes:

I have removed the facility for S200 speed signals to be revised to S400 when packets are forwarded from a DS to Beta port. This reflects what I sense to be the outcome of the small amount of discussion on this subject, i.e. that a receiving DS port ought to be sure of the speed signal value before forwarding the speed signal. This also seems to be the recommendation of Jim Skidmore's paper to 1394A.

I have re-used the SPEEDDB code (previously used to allow S200 to S400 revision) as an end delimiter for all speed signals. This means that as well as the speed signal being constant length at all port speeds for a given packet speed, the detection time in a receiving beta port will also be constant (as we do not need to wait for a non-speed code to know that the speed signal is over).

Changes from the September 97 proposal are marked by change bars.

Definitions:

Beta mode port: a 1394B port that is using 8B10B coding, not DS coding.

DS port: a 1394-1995 or 1394B port that is using Data Strobe coding, not 8B10B coding.

Character: a 10 bit codeword

Operating speed: the rate at which data is transmitted from a beta mode port.

symbol: a numerical value representing a 1394B control state

padded data: a stream of data characters at the port operating speed containing data from a lower speed packet.

packet prefix: the combination of data prefix and speed signals that precede a packet.

1.1 Introduction

A beta mode port is required to transport packets at all 1394 rates up to and including the operating speed of that particular port. 1394B must therefore provide a mechanism for signaling the speed of a packet during the data prefix. In particular, when a beta mode port is repeating a packet that has been received on a Data/Strobe (DS) port at S100, S200 or S400, the common mode speed signaling voltage received on the DS port must be represented in the stream of 10 bit characters transmitted on the beta mode port.

1.2 Requirements

The requirements for the 1394B speed signaling mechanism are as follows:

- For a given packet speed, the time duration of the speed signal should be constant regardless of the operating speed of the beta mode port through which the packet is being forwarded. This ensures that when a packet is being forwarded from one beta mode port to another, but the beta mode ports are operating at different speeds, then no buffering of data is required to allow for the speed signal being stretched in time.
- For a given beta mode port operating speed, the duration of the speed signal should be less than or equal to the time taken to assemble and transmit one data character. This maintains a constant delay between the speed signal and the start of data as a packet propagates across a bus, without data buffering.

1.3 Proposal

The proposed speed signals are detailed in the table below:

Table 1-1—

Port operating speed	Packet speed					
	S100	S200	S400	S800	S1600	S3200
S100	-					
S200	-	S_b				
S400	-	$S_a S_b$	S_b			
S800	-	$S_a S_a S_a S_b$	$S_a S_b$	S_b		
S1600	-	$S_a S_a S_a S_a S_a S_a S_b$	$S_a S_a S_a S_b$	$S_a S_b$	S_b	
S3200	-	$S_a S_a S_a S_a S_a S_a S_a S_a S_a S_a S_a S_a S_b$	$S_a S_a S_a S_a S_a S_a S_b$	$S_a S_a S_a S_b$	$S_a S_b$	S_b
speed signal duration (nsecs)	0	40	20	10	5	2.5

Notes:

- 1) For any particular packet speed, the speed signal has constant duration at all port speeds. At all packet speeds other than S100, the number of speed symbols used at a particular port operating speed is equal to the number of symbols transmitted per byte of packet data at that port operating speed.
- 2) S_a represents the control state SPEEDa. S_b represents the control state SPEEDb. It is proposed that these two states replace the current SPEED control state, and that each is mapped to a single control symbol independent of the disparity. i.e. control state mapping is changed to:

Table 1-2—Control State Mapping

Control state	Control symbol SRQP	
	rd>0	rd<0
SPEEDa	0100	
SPEEDb	1000	

- 3) S_b serves as an end delimiter for any speed signal.

1.4 General packet format

In general a beta mode port shall prefix any packet with a combination of data prefix and speed signals as shown below:

.....<data prefix symbols><speed symbols><data prefix symbols><padded data>.....

The <> delimiters indicate elements of the packet prefix that may be optional at some (or all) combination of operating speed and packet speed, as described below. The [] delimiters indicate mandatory elements of the packet prefix.

1.5 S100 packet speeds

No speed signal is sent for an S100 packet at any operating speed. Since there is no common mode speed signal used on DS ports for S100, it would be impossible to forward a speed signal on a beta mode port until data started. At this point it would then be necessary to buffer data for the length of the speed signal. The packet prefix for a S100 packet forwarded on any beta mode port shall be:

.....[data prefix symbols][padded data].....

The duration of the data prefix symbols shall be greater than or equal to the minimum duration of DATA_PREFIX required on a DS port, as specified in the 1394-1995 standard i.e. 120nsecs.

Note: this requirement ensures that if the packet is subsequently forwarded to a DS port, the DS port will be able to generate a minimum length DATA_PREFIX without buffering data.

1.6 S200 & S400 packets

The packet prefix for S200 and S400 packets forwarded on any beta mode port shall be:

.....<data prefix symbols>[speed symbols][data prefix symbols]<padded data>.....

The duration of the data prefix symbols and speed symbols shall be greater than or equal to the minimum duration of DATA_PREFIX required on a DS port, as specified in the 1394-1995 standard i.e. 120nsecs.

The time between completion of the final speed symbol and completion of the first byte of padded data shall be greater than or equal to 140nsecs.

Note: this requirement ensures that if the packet is subsequently forwarded to a DS port, the DS port will be able to generate a minimum length speed signal followed by a delay of DATA_PREFIX_TIME before starting data, without buffering data.

1.7 S800 and higher speed packets

Packets at rates greater than or equal to S800 will never be forwarded on a DS port. The packet prefix rules for such packets are therefore different, and much shorter, than for lower rate packets.

The packet prefix for S800, S1600 and S3200 packets forwarded on any beta mode port shall be:

.....[speed symbols][padded data | data prefix symbols].....

For packet speeds less than or equal to the port operating speed, the beta mode port shall transmit the specified speed signal followed immediately by padded data.

In the case of a packet speed that is greater than the port operating speed, the speed signal (if any) transmitted by the beta mode port shall be the speed signal defined for the packet speed that corresponds to the port operating speed (i.e. a single occurrence of SPEEDa). This speed signal shall be followed immediately by data prefix symbols.

1.8 Examples

1.8.1 Example 1

An S200 packet forwarded through a beta mode port operating at S800:

....[DP][DP][DP][DP][DP][Sa][Sa][Sa][Sb][DP][DP][DP] [DP][DP][padded data]
 > 20ns <

Same packet forwarded from beta mode at S800 to beta mode at S400:

....[DP][DP][DP][Sa][Sb][DP] [DP][padded data]
 > 20ns <

1.8.2 Example 2

An S400 packet forwarded from DS port through an S400 beta mode port:

....[DP][DP][Sb][DP][DP] [DP][padded data]
 > 20ns <

Same packet forwarded from S400 beta mode port through an S800 beta mode port:

....[DP][DP][DP][DP][Sa][Sb][DP][DP][DP][DP] [DP][DP][padded data]
 > 20ns <

1.8.3 Example 3

An S800 packet transmitted through an S1600 beta mode port:

....[Sa][Sb][padded data]
 > 10ns <

Same packet forwarded through an S800 beta mode port:

....[Sb][padded data]
 > 10ns <

1.9 Speed Signal Detection

A Beta mode port shall receive all of a speed signal before indicating the speed of the subsequent packet to other ports and the Link layer. The speed signal shall be considered complete when a SPEEDB code has been received. Valid speed signals shall be used to determine the speed of the subsequent packet according to table [tbd].

Table 1-3—

Received speed signal	Port operating speed					
	S100	S200	S400	S800	S1600	S3200
none	S100	S100	S100	S100	S100	S100
S _b	-	S200	S400	S800	S1600	S3200
S _a S _b	-		S200	S400	S800	S1600
S _a S _a S _a S _b	-			S200	S400	S800
S _a S _a S _a S _a S _a S _a S _b	-				S200	S400
S _a S _a S _a S _a S _a S _a S _a S _a S _a S _a S _b	-					S200

A speed signal shall only be considered valid if it meets the following criteria:

- a) All SPEEDA and SPEEDB codes are received consecutively without interruption.
- b) The pattern of SPEEDA and SPEEDB codes is found in Table [tbd].

If an invalid speed signal is received the port shall not pass a packet speed indication to other ports and shall discard the subsequent packet.

The following algorithm determines the received speed signal (do not attempt to compile !):

```
pkt_speed=100; //default packet speed is S100

if(rx_code==SPEEDA) {
    speed_ratio=1;
    speed_sig_done=0;
    while(~speed_sig_done) {
        wait_event(new_rx_code);
        switch(rx_code) {
            case SPEEDA:
                speed_ratio++; //for every SPEEDA increment the speed ratio...
                break;
            case SPEEDB:
                speed_ratio++;
                speed_sig_done=1; //...until the SPEEDB is received
                break;
            default :
                speed_sig_done=1; //If something other than a SPEEDA or SPEEDB is received
                speed_sig_error=1; //during the speed signal then an error has occurred, which
                speed_ratio=port_speed/100; // results in S100 packet speed
        }
    }
    pkt_speed=port_speed/speed_ratio; //actual speed is a function of the port speed
}

else if(rx_code=SPEEDB) {
    pkt_speed=port_speed; //a lone SPEEDB means the packet is at the port speed.
    speed_sig_done=1;
}
```