Gap Count Analysis for the P1394a Bus

Jerry Hauck
Zayante, Inc.

1 Purpose and Scope

This paper analyzes the constraints placed on the gap count variable by the collection of PHY timing parameters and proper operation of cable arbitration. This paper addresses certain ballot comments submitted against Draft 2.0 of the P1394a standard that suggested the gap count derivation outlined in clause C.2 did not properly scale for allowable larger values of PHY_DELAY and/or longer cables.

Four well known limiting corner cases for gap count are examined in an effort to find the minimum allowable gap count for a given topology. Both the table method and pinging method of determining the optimal gap count are explored. Finally, recommended corrections and improvements to Draft 2.0 are offered at the conclusion.

It is important to note that this analysis assumes that PHY_DELAY can never exceed the maximum published in the PHY register set. However, corner conditions have been identified in which it is theoretically possible to have PHY_DELAY temporarily exceed the maximum published delay when repeating minimally spaced packets. ***Although not a rigorous proof, this phenomena is ignored for this analysis on the basis that it is presumed to be statistically insignificant.***

2 Credits

The topic, derivation, and very format of this document were suggested and or borrowed from an excellent paper prepared by Jim Skidmore of Texas Instruments titled *Analysis of Gap Count Settings for the IEEE-1394 Bus* and dated 6/18/98. Additional guidance was sought from an analysis prepared again by Jim Skidmore in response to an e-mail exchange on the P1394a reflector with the subject *ARB_DELAY and GAP_COUNT* submitted on 7/18/97. Jim personally assisted in the preparation of this paper through diligent review and verification.

Dave LaFollette's original work on gap count optimization through PHY pinging (submitted to the P1394a editor on 12/12/97 for inclusion into Annex C of the P194a 2.0 draft) was revised to reflect the changes to the underlying gap count limits. Dave assisted with thorough review and much patience.

3 Intervening Path Model

The path between any two given PHYs can be represented as a daisy chain connection of the two devices with zero or more intervening, or repeating, PHYs. Figure 1 illustrates such a path between two nodes, X & Y, and denotes the reference points required for a full analysis.

Figure 1: Intervening Path Model

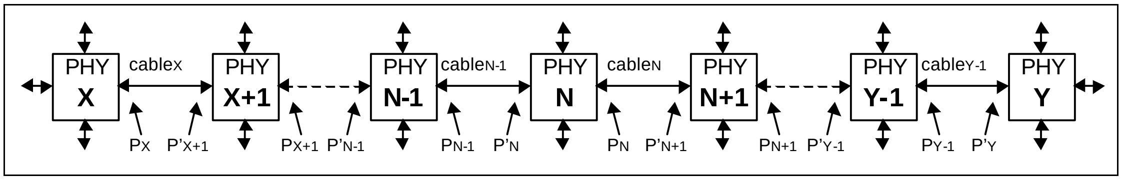

Table 1: Variable Definitions

$ARB\_RESPONSE\_DELAY_n^{P_n \rightarrow P_n'}$	Delay in propagating arbitration indication received from port P_n of PHY n to port P'_n of PHY n.
$BASERATE_n$	Fundamental operating frequency of PHY n.
$cable\_delay_n$	One-way flight time of arbitration and data signals through $cable_n$. The flight-time is assumed to be constant from one transmission to the next and symmetric.
$DATA\_END\_TIME_n^{P_n}$	Length of DATA_END transmitted on port P_n of PHY n.
$PHY\_DELAY_n^{P_n' \rightarrow P_n}$	Time from receipt of first data bit at port P'_n of PHY n to re-transmission of same bit at port P_n of PHY n.
$RESPONSE\_TIME_n^{P_n'}$	Idle time at port P'_n of PHY n between the reception of a inbound packet and the associated outbound arbitration indication for the subsequent packet intended to occur within the same isochronous interval or asynchronous subaction.

4 Minimum Subaction Timings

For any given topology, the gap count must be set such that an iso or ack gap observed/generated at one PHY isn't falsely interpreted as a subaction gap by another PHY in the network. Ack/Iso gaps are known to be at their largest nearest the PHY that originated the last packet. To ensure that the most recent originating PHY doesn't interrupt a subaction or isochronous interval with asynchronous arbitration, its subaction_gap timeout must be greater than the largest IDLE which can legally occur within a subaction or isochronous interval. Figure 2 illustrates the case in which PHY X originated the most recent packet and PHY Y is responding (either with an ack or the next isochronous arbitration/packet).

Figure 2: Ack/Iso Gap Preservation

```
            t0       t1   t2                                  t7
PX    --< DP | Packet | DE >---------------------------------< ARB |
PX+1  -----< DP | Packet | DE >----------------------------< ARB |
PN-1  --------< DP | Packet | DE >-----------------------< ARB |
PN    -----------< DP | Packet | DE >------------------< ARB |
PN+1  --------------< DP | Packet | DE >-------------< ARB |
PY-1  -----------------< DP | Packet | DE >--------< ARB |
P'Y   --------------------< DP | Packet | DE >---< ARB |
                               t3        t4   t5  t6
```

For all topologies, the idle time observed at point P_x must not exceed the subaction gap detection time:

(1) $$Idle_{\max}^{P_X} < subaction\_gap_{\min}^{P_X}$$

The idle time at point P_x can be determined by examining the sequence of time events in the network. All timing events are referenced to the external bus (as opposed to some internal point in the PHY).

- t_0 First bit of packet sent at point P_x
- t_1 Last bit of packet sent at point P_x, DATA_END begins. t_1 follows t_0 by the length of the packet timed in PHY X's clock domain.
- t_2 DATA_END concludes at point P_x, IDLE begins. t_2 follows t_1 by $DATA\_END\_TIME_X^{P_X}$
- t_3 First bit of packet received at point P'_Y. t_3 follows t_0 by all intervening cable_delay and PHY_DELAY instances.
- t_4 Last bit of packet received at point P'_Y. t_4 follows t_3 by the length of the packet timed in PHY Y-1's clock domain.
- t_5 DATA_END concludes at point P'_Y, gap begins. t_5 follows t_4 by $DATA\_END\_TIME_{Y-1}^{P_{Y-1}}$
- t_6 PHY Y responds with ack packet, isoch packet, or isoch arbitration within $RESPONSE\_TIME_Y^{P'_Y}$ following t_5
- t_7 Arbitration indication arrives at point P_x. t_7 follows t_6 by the all intervening cable_delay and ARB_RESPONSE_DELAY instances.

(2) $$t_1 = t_0 + \frac{packet\_length}{packet\_speed \cdot BASERATE_X}$$

(3)
$$\begin{aligned} t_2 &= t_1 + DATA\_END\_TIME_X^{P_X} \\ &= t_0 + \frac{packet\_length}{packet\_speed \cdot BASERATE_X} + DATA\_END\_TIME_X^{P_X} \end{aligned}$$

(4)
$$t_3 = t_0 + cable\_delay_X + \sum_{n=X+1}^{Y-1} \left(cable\_delay_n + PHY\_DELAY_n^{P_n^{'} \to P_n} \right)$$

(5)
$$\begin{aligned} t_4 &= t_3 + \frac{packet\_length}{packet\_speed \cdot BASERATE_{Y-1}} \\ &= t_0 + cable\_delay_X + \sum_{n=X+1}^{Y-1} \left(cable\_delay_n + PHY\_DELAY_n^{P_n^{'} \to P_n} \right) + \\ &\quad \frac{packet\_length}{packet\_speed \cdot BASERATE_{Y-1}} \end{aligned}$$

(6)
$$\begin{aligned} t_5 &= t_4 + DATA\_END\_TIME_{Y-1}^{P_{Y-1}} \\ &= t_0 + cable\_delay_X + \sum_{n=X+1}^{Y-1} \left(cable\_delay_n + PHY\_DELAY_n^{P_n^{'} \to P_n} \right) + \\ &\quad \frac{packet\_length}{packet\_speed \cdot BASERATE_{Y-1}} + DATA\_END\_TIME_{Y-1}^{P_{Y-1}} \end{aligned}$$

(7)
$$\begin{aligned} t_6 &= t_5 + RESPONSE\_TIME_Y^{P_Y^{'}} \\ &= t_0 + cable\_delay_X + \sum_{n=X+1}^{Y-1} \left(cable\_delay_n + PHY\_DELAY_n^{P_n^{'} \to P_n} \right) + \\ &\quad \frac{packet\_length}{packet\_speed \cdot BASERATE_{Y-1}} + DATA\_END\_TIME_{Y-1}^{P_{Y-1}} + RESPONSE\_TIME_Y^{P_Y^{'}} \end{aligned}$$

(8)
$$\begin{aligned} t_7 &= t_6 + \sum_{n=X+1}^{Y-1} \left(cable\_delay_n + ARB\_RESPONSE\_DELAY_n^{P_n \to P_n^{'}} \right) + cable\_delay_X \\ &= t_0 + \sum_{n=X+1}^{Y-1} \left(2 \cdot cable\_delay_n + PHY\_DELAY_n^{P_n^{'} \to P_n} + ARB\_RESPONSE\_DELAY_n^{P_n \to P_n^{'}} \right) + \\ &\quad 2 \cdot cable\_delay_X + \frac{packet\_length}{packet\_speed \cdot BASERATE_{Y-1}} + DATA\_END\_TIME_{Y-1}^{P_{Y-1}} + \\ &\quad RESPONSE\_TIME_Y^{P_Y^{'}} \end{aligned}$$

Given t_0 through t_7 above, the Idle time seen at point P_x is given as:

$$
\begin{aligned}
Idle^{P_X} &= t_7 - t_2 \\
&= \sum_{n=X+1}^{Y-1}\left(2\cdot cable\_delay_n + PHY\_DELAY_n^{P_n^{'}\to P_n} + ARB\_RESPONSE\_DELAY_n^{P_n\to P_n^{'}}\right)+ \\
&\quad 2\cdot cable\_delay_X + RESPONSE\_TIME_Y^{P_Y^{'}} + \\
&\quad DATA\_END\_TIME_{Y-1}^{P_{Y-1}} - DATA\_END\_TIME_X^{P_X} + \\
&\quad \frac{packet\_length}{packet\_speed}\cdot\left(\frac{1}{BASERATE_{Y-1}} - \frac{1}{BASERATE_X}\right)
\end{aligned} \qquad (9)
$$

Let:

$$DE\_delta^{[P_{Y-1},P_X]} = DATA\_END\_TIME_{Y-1}^{P_{Y-1}} - DATA\_END\_TIME_X^{P_X} \qquad (10)$$

$$PPM\_delta^{[Y-1,X]} = \frac{packet\_length}{packet\_speed}\cdot\left(\frac{1}{BASERATE_{Y-1}} - \frac{1}{BASERATE_X}\right) \qquad (11)$$

$$
\begin{aligned}
Round\_Trip\_Delay^{[P_X \circlearrowleft P_Y]} = &\sum_{n=X+1}^{Y-1}\begin{pmatrix}2\cdot cable\_delay_n + PHY\_DELAY_n^{P_n^{'}\to P_n} + \\ ARB\_RESPONSE\_DELAY_n^{P_n\to P_n^{'}}\end{pmatrix}+ \\
&2\cdot cable\_delay_X
\end{aligned} \qquad (12)
$$

Then,

$$
\begin{aligned}
Idle^{P_X} &= Round\_Trip\_Delay^{[P_X \circlearrowleft P_Y]} + RESPONSE\_TIME_Y^{P_Y^{'}} + \\
&\quad DE\_delta^{[P_{Y-1},P_X]} + PPM\_delta^{[Y-1,X]}
\end{aligned} \qquad (13)
$$

Substituting into Equation (1), Ack and Iso gaps are preserved network-wide if and only if:

$$\begin{bmatrix}Round\_Trip\_Delay^{[P_X \circlearrowleft P_Y]} + RESPONSE\_TIME_Y^{P_Y^{'}} + \\ DE\_delta^{[P_{Y-1},P_X]} + PPM\_delta^{[Y-1,X]}\end{bmatrix}_{\max} < subaction\_gap_{\min}^{P_X} \qquad (14)$$

The minimum subaction_gap at point P_x isn't well known. IEEE1394-1995, in Table 4-33, defines the minimum subaction_gap timeout used at a PHY's internal state machines, not at the external interface. It has been argued that the internal and external representations of time may differ by as much as ARB_RESPONSE_DELAY when a PHY is counting elapsed time between an internally generated event and an externally received event. However, the ARB_RESPONSE_DELAY value for a particular PHY isn't generally known externally. Fortunately, the ARB_RESPONSE_DELAY value for a PHY whose FIFO is known to be empty is bounded by the worst case PHY_DELAY reported within the PHY register map. This suggests a realistic bound for the minimum subaction_gap referenced at point P_x:

$$subaction\_gap_{\min}^{P_X} \geq subaction\_gap_{\min}^{i_X} - PHY\_DELAY_{X,\max}^{P_X} \qquad (15)$$

where

(16) $$subaction\_gap_{\min}^{i_X} = \frac{27 + gap\_count \cdot 16}{BASERATE_{X,\max}}$$

Combing Equations (14), (15), and (16):

(17) $$\begin{bmatrix} Round\_Trip\_Delay^{[P_X \circlearrowleft P_Y]} + \\ RESPONSE\_TIME_Y^{P_Y'} + \\ DE\_delta^{[P_{Y-1}, P_X]} + \\ PPM\_delta^{[Y-1, X]} \end{bmatrix}_{\max} < \left[\frac{27 + gap\_count \cdot 16}{BASERATE_{X,\max}} - PHY\_DELAY_{X,\max}^{P_X} \right]$$

Solving for gap_count:

(18) $$gap\_count > \frac{BASERATE_{X,\max} \cdot \begin{bmatrix} Round\_Trip\_Delay_{\max}^{[P_X \circlearrowleft P_Y]} + \\ \begin{bmatrix} RESPONSE\_TIME_Y^{P_Y'} + \\ DE\_delta^{[P_{Y-1}, P_X]} + PPM\_delta^{[Y-1, X]} \end{bmatrix}_{\max} + \\ PHY\_DELAY_{X,\max}^{P_X} \end{bmatrix} - 27}{16}$$

Since RESPONSE_TIME, DE_delta, and PPM_delta are not independent parameters, the maximum of their sum is not accurately represented by the sum of their maximas. Finding a more accurate maximum for the combined quantity requires the identification of components of RESPONSE_TIME.

As specified in p1394a, RESPONSE_TIME includes the time a responding node takes to repeat the received packet and then drive a subsequent arbitration indication. (Note that by examination of the C code, RESPONSE_TIME is defined to include the time it takes to repeat a packet even if the PHY in question is a leaf node.) Figure 3 illustrates the sequence PHY Y will follow in responding to a received packet. i_Y denotes the timings as seen/interpreted by the PHY state machine. The figure is not to scale. (Note that P_Y can be any repeating port on PHY Y. Consequently, the timing constraints referenced to P_Y in the following analysis must hold worst case for any and all repeating ports.)

Figure 3: RESPONSE_TIME Sequence

```
              t3        t4                    t5        t6
P'Y:   --< DP | Packet |      DATA_END      >-------<    ARB

                                                t5a  t5b
iY:    ----------------< stop_tx_packet() >---< start_tx_packet()

                 t3'       t4'                    t5'
PY:    -----< DP | Packet |      DATA_END      >---<    ARB
```

Beginning with the first arrival of data at P'_Y (t_3), the elaborated timing sequence for RESPONSE_TIME is:

t_3 First bit of packet received at point P'_Y

t_3' First bit of packet repeated at point P_Y. t_3' lags t_3 by PHY_DELAY

t_4 Last bit of packet received at point P'_Y. t_4 follows t_3 by the length of the packet timed in PHY N's clock domain. DATA_END begins

t_4' Last bit of packet repeated at point P_Y. t_4' lags t_3' by the length of the packet timed in PHY Y's clock domain. The PHY begins "repeating" DATA_END

t_5 DATA_END concludes at point P'_Y. t_5 follows t_4 by $DATA\_END\_TIME_{Y-1}^{P_{Y-1}}$

t_{5a} stop_tx_packet() concludes at point i_Y and the state machines command the PHY ports to stop repeating DATA_END. t_{5a} leads t_5' by any transceiver delay.

t_5' DATA_END concludes at point P_Y. t_5' follows t_4' by $DATA\_END\_TIME_Y^{P_Y}$

t_{5b} start_tx_packet() commences at point i_Y and the state machines command the PHY ports to begin driving the first arbitration indication of any response. t_{5b} lags t_{5a} by an IDLE_GAP and an unspecified state machine delay herein called SM_DELAY.

t_6 PHY Y drives arbitration at points P'_Y. t_6 follows t_{5b} by any transceiver delay.

(19) $$t_{3'} = t_3 + PHY\_DELAY_Y^{P'_Y \to P_Y}$$

(20) $$\begin{aligned} t_{4'} &= t_{3'} + \frac{packet\_length}{packet\_speed \cdot BASERATE_Y} \\ &= t_3 + PHY\_DELAY_Y^{P'_Y \to P_Y} + \frac{packet\_length}{packet\_speed \cdot BASERATE_Y} \end{aligned}$$

(21) $$\begin{aligned} t_{5'} &= t_{4'} + DATA\_END\_TIME_Y^{P_Y} \\ &= t_3 + PHY\_DELAY_Y^{P'_Y \to P_Y} + \frac{packet\_length}{packet\_speed \cdot BASERATE_Y} + DATA\_END\_TIME_Y^{P_Y} \end{aligned}$$

(22) $$\begin{aligned} t_{5a} &= t_{5'} - transceiver\_delay_Y^{P_Y} \\ &= t_3 + PHY\_DELAY_Y^{P'_Y \to P_Y} + \frac{packet\_length}{packet\_speed \cdot BASERATE_Y} + DATA\_END\_TIME_Y^{P_Y} - \\ &\quad transceiver\_delay_Y^{P_Y} \end{aligned}$$

(23) $$\begin{aligned} t_{5b} &= t_{5a} + IDLE\_GAP_Y + SM\_DELAY_Y \\ &= t_3 + PHY\_DELAY_Y^{P'_Y \to P_Y} + \frac{packet\_length}{packet\_speed \cdot BASERATE_Y} + DATA\_END\_TIME_Y^{P_Y} + \\ &\quad IDLE\_GAP_Y + SM\_DELAY_Y - transceiver\_delay_Y^{P_Y} \end{aligned}$$

$$
\begin{aligned}
t_6 &= t_{5b} + transceiver\_delay_Y^{P_Y'} \\
&= t_3 + PHY\_DELAY_Y^{P_Y' \to P_Y} + \frac{packet\_length}{packet\_speed \cdot BASERATE_Y} + DATA\_END\_TIME_Y^{P_Y} + \\
&\quad IDLE\_GAP_Y + SM\_DELAY_Y + transceiver\_delay_Y^{P_Y'} - transceiver\_delay_Y^{P_Y}
\end{aligned} \tag{24}
$$

By definition,

$$RESPONSE\_TIME_Y^{P_Y'} = t_6 - t_5 \tag{25}$$

and through substitution,

$$
\begin{aligned}
RESPONSE\_TIME_Y^{P_Y'} = \; & PHY\_DELAY_Y^{P_Y' \to P_Y} + DE\_delta^{[P_Y, P_{Y-1}]} + PPM\_delta^{[Y,Y-1]} + \\
& IDLE\_GAP_Y + SM\_DELAY_Y + \\
& transceiver\_delay_Y^{P_Y'} - transceiver\_delay_Y^{P_Y}
\end{aligned} \tag{26}
$$

As such, the combination of RESPONSE_TIME, DE_delta, and PPM_delta from equation (18) can be represented as:

$$
\begin{aligned}
\begin{bmatrix} RESPONSE\_TIME_Y^{P_Y'} + \\ DE\_delta^{[P_{Y-1}, P_X]} + \\ PPM\_delta^{[Y-1,X]} \end{bmatrix}
&= \begin{bmatrix} PHY\_DELAY_Y^{P_Y' \to P_Y} + DE\_delta^{[P_Y, P_{Y-1}]} + PPM\_delta^{[Y,Y-1]} + \\ IDLE\_GAP_Y + SM\_DELAY_Y + transceiver\_delay_Y^{P_Y'} - \\ transceiver\_delay_Y^{P_Y} + DE\_delta^{[P_{Y-1}, P_X]} + PPM\_delta^{[Y-1,X]} \end{bmatrix} \\
&= \begin{bmatrix} PHY\_DELAY_Y^{P_Y' \to P_Y} + DE\_delta^{[P_Y, P_X]} + PPM\_delta^{[Y,X]} + \\ IDLE\_GAP_Y + SM\_DELAY_Y + transceiver\_delay_Y^{P_Y'} - \\ transceiver\_delay_Y^{P_Y} \end{bmatrix}
\end{aligned} \tag{27}
$$

Noting that if PHYs X and Y-1 both adhere to the same minimum timing requirement for DATA_END_TIME and maximum timing requirement for BASE_RATE, then

$$
\begin{aligned}
DE\_delta_{\max}^{[P_Y, P_X]} &= DE\_delta_{\max}^{[P_Y, P_{Y-1}]} \\
PPM\_delta_{\max}^{[Y,X]} &= PPM\_delta_{\max}^{[Y,Y-1]}
\end{aligned} \tag{28}
$$

The combined maximum can be rewritten as:

$$
\begin{bmatrix} RESPONSE\_TIME_Y^{P_Y'} + \\ DE\_delta^{[P_{Y-1}, P_X]} + \\ PPM\_delta^{[Y-1,X]} \end{bmatrix}_{\max}
= \begin{bmatrix} PHY\_DELAY_{Y,\max}^{P_Y' \to P_Y} + DE\_delta_{\max}^{[P_Y, P_{Y-1}]} + PPM\_delta_{\max}^{[Y,Y-1]} + \\ IDLE\_GAP_{Y,\max} + SM\_DELAY_{Y,\max} + \\ transceiver\_delay_{Y,\max}^{P_Y'} - transceiver\_delay_{Y,\min}^{P_Y} \end{bmatrix} \tag{29}
$$

Comparing to equation (26) allows

(30) $$\begin{bmatrix} RESPONSE\_TIME_{Y}^{P_Y'} + \\ DE\_delta^{[P_{Y-1}, P_X]} + \\ PPM\_delta^{[Y-1, X]} \end{bmatrix}_{\max} = RESPONSE\_TIME_{Y,\max}^{P_Y'}$$

Finally:

(31) $$gap\_count > \frac{BASERATE_{X,\max} \cdot \begin{bmatrix} Round\_Trip\_Delay_{\max}^{[P_X \circlearrowleft P_Y]} + \\ RESPONSE\_TIME_{Y,\max}^{P_Y'} + \\ PHY\_DELAY_{X,\max}^{P_X} \end{bmatrix} - 27}{16}$$

5 Minimum Arb Reset Timings

For any given topology, the gap count must be set such that subaction gaps observed/generated at one PHY aren't falsely interpreted as arb_reset gaps by another PHY in the network. Subaction gaps are known to be at their largest nearest the PHY that originated the last packet. To ensure that the most recent originating PHY doesn't begin a new fairness interval before all PHYs exit the current one, its arb_reset_gap timeout must be greater than the largest subaction_gap which can legally occur. Figure 4 illustrates the case in which PHY X originated the most recent packet and PHY Y is responding after a subaction gap with arbitration for the current fairness interval.

Figure 4: Subaction Gap Preservation

```
             t0        t1   t2                                t7
P_X    --< DP | Packet | DE >--------------------------------< ARB |
P_X+1  -----< DP | Packet | DE >----------------------------< ARB |
P_N-1  --------< DP | Packet | DE >------------------------< ARB |
P_N    -----------< DP | Packet | DE >--------------------< ARB |
P_N+1  --------------< DP | Packet | DE >----------------< ARB |
P_Y-1  -----------------< DP | Packet | DE >------------< ARB |
P'_Y   --------------------< DP | Packet | DE >---< ARB |
                                t3        t4   t5  t6
```

For all topologies, the idle time observed at point P_x must not exceed the arbitration reset gap detection time:

(32) $$Idle^{P_X}_{\max} < arb\_reset\_gap^{P_X}_{\min}$$

The analysis is identical to the case in which Ack and Iso gaps are preserved with the exception that PHY Y takes longer to respond to the trailing edge of DATA_END. Let PHY Y have a response time of subaction_response_time. Then,

(33) $$Idle^{P_X} = Round\_Trip\_Delay^{[P_X \circlearrowleft P_Y]} + subaction\_response\_time^{P'_Y}_{Y} + DE\_delta^{[P_{Y-1},P_X]} + PPM\_delta^{[Y-1,X]}$$

Substituting into Equation (32), subaction gaps are preserved network-wide if and only if:

(34) $$\begin{bmatrix} Round\_Trip\_Delay^{[P_X \circlearrowleft P_Y]} + subaction\_response\_time^{P'_Y}_{Y} + \\ DE\_delta^{[P_{Y-1},P_X]} + PPM\_delta^{[Y-1,X]} \end{bmatrix}_{\max} < arb\_reset\_gap^{P_X}_{\min}$$

The minimum arb_reset_gap at point P_x isn't well known. IEEE1394-1995, in Table 4-33, defines the minimum arb_reset_gap timeout used at a PHY's internal state machines, not at the external interface. It has been argued that the internal and external representations of time may differ by as much as ARB_RESPONSE_DELAY when a PHY is counting elapsed time between an internally generated event and an externally received event. However, the ARB_RESPONSE_DELAY value for a particular PHY isn't generally known externally. Fortunately, the ARB_RESPONSE_DELAY value for a PHY whose FIFO is known to be empty is bounded by the worst case PHY_DELAY reported within the PHY register map. This suggests a realistic bound for the minimum subaction_gap referenced at point P_x:

(35) $$arb\_reset\_gap^{P_X}_{\min} \ge arb\_reset\_gap^{i_X}_{\min} - PHY\_DELAY^{P_X}_{X,\max}$$

where

(36) $$arb\_reset\_gap_{\min}^{i_X} = \frac{51 + gap\_count \cdot 32}{BASERATE_{X,\max}}$$

The maximum subaction_response_time for PHY Y parallels the earlier dissection of RESPONSE_TIME. The timing sequence for subaction_response_time is identical to that of RESPONSE_TIME except that PHY Y, after concluding stop_tx_Packet(), must wait to detect a subaction gap and then wait an additional arb_delay before calling start_tx_packet(). Said differently, the idle period timed internally is a subaction gap plus arb_delay rather than an IDLE_GAP. Consequently, t_{5b} becomes:

(37) $$t_{5b} = t_{5a} + subaction\_gap^{i_Y} + arb\_delay^{i_Y} + SM\_DELAY_Y$$

and

(38) $$subaction\_response\_time_Y^{P_Y'} = RESPONSE\_TIME_Y^{P_Y'} - IDLE\_GAP_Y + subaction\_gap^{i_Y} + arb\_delay^{i_Y}$$

Substituting into Equation (34),

(39) $$\begin{bmatrix} Round\_Trip\_Delay^{[P_X \circlearrowleft P_Y]} + RESPONSE\_TIME_Y^{P_Y'} + \\ DE\_delta^{[P_{Y-1}, P_X]} + PPM\_delta^{[Y-1, X]} + \\ subaction\_gap^{i_Y} + arb\_delay^{i_Y} - IDLE\_GAP_Y \end{bmatrix}_{\max} < arb\_reset\_gap_{\min}^{P_X}$$

Again, RESPONSE_TIME, DE_delta, and PPM_delta are not independent parameters. As shown previously, if PHYs X and Y-1 adhere to the same timing constant limits, the explicit DE_Delta and PPM_delta terms can be subsumed within RESPONSE_TIME giving:

(40) $$\begin{bmatrix} Round\_Trip\_Delay_{\max}^{[P_X \circlearrowleft P_Y]} + RESPONSE\_TIME_{Y,\max}^{P_Y'} + \\ subaction\_gap_{\max}^{i_Y} + arb\_delay_{\max}^{i_Y} - MIN\_IDLE\_TIME_Y \end{bmatrix} < arb\_reset\_gap_{\min}^{P_X}$$

where

(41) $$subaction\_gap_{\max}^{i_Y} = \frac{29 + gap\_count \cdot 16}{BASERATE_{Y,\min}},$$

(42) $$arb\_delay_{\max}^{i_Y} = \frac{gap\_count \cdot 4}{BASERATE_{Y,\min}}$$

and

(43) $$IDLE\_GAP_{Y,\min} = MIN\_IDLE\_TIME_Y$$

Combining Equations (35), (36), (40), (41), and (42):

$$(44)\quad \left[\begin{array}{l} Round\_Trip\_Delay_{\max}^{[P_X \circlearrowleft P_Y]} + \\ RESPONSE\_TIME_{Y,\max}^{P_Y'} - \\ MIN\_IDLE\_TIME_Y + \\ \dfrac{29 + gap\_count \cdot 20}{BASERATE_{Y,\min}} \end{array}\right] < \left[\frac{51 + gap\_count \cdot 32}{BASERATE_{X,\max}} - PHY\_DELAY_{X,\max}^{P_X}\right]$$

Solving for gap_count:

$$(45)\quad gap\_count > \frac{BASERATE_{X,\max} \cdot \left[\begin{array}{l} Round\_Trip\_Delay_{\max}^{[P_X \circlearrowleft P_Y]} + \\ RESPONSE\_TIME_{Y,\max}^{P_Y'} - \\ MIN\_IDLE\_TIME_Y + \\ PHY\_DELAY_{X,\max}^{P_X} \end{array}\right] + 29 \cdot \dfrac{BASERATE_{X,\max}}{BASERATE_{Y,\min}} - 51}{32 - 20 \cdot \dfrac{BASERATE_{X,\max}}{BASERATE_{Y,\min}}}$$

6 Maximum Subaction Timings

For any given topology, the gap count must be set such that if a subaction gap is observed following an isochronous packet at one PHY, it is observed at all PHYs. The danger occurs when a subsequent arbitration indication is transmitted in the same direction as the previous data packet. Given that arbitration indications may propagate through intervening PHYs faster than data bits, gaps may be shortened as they are repeated. Figure 5 illustrates the case in which PHY X originates an isochronous packet, observes a subaction_gap, and begins to drive an arbitration indication.

Figure 5: Consistent Subaction Gap Detection

```
             t_0      t_1   t_2                  t_6
P_X     --< DP | Packet | DE >-------------------< ARB |
P_X+1   -----< DP | Packet | DE >-----------------< ARB |
P_N-1   --------< DP | Packet | DE >----------------< ARB |
P_N     -----------< DP | Packet | DE >---------------< ARB |
P_N+1   --------------< DP | Packet | DE >--------------< ARB |
P_Y-1   -----------------< DP | Packet | DE >-------------< ARB |
P'_Y    --------------------< DP | Packet | DE >------------< ARB |
                              t_3       t_4   t_5          t_7
```

For all topologies, the minimum idle time observed at point P'_Y must always exceed the maximum subaction gap detection time:

(46) $$Idle_{\min}^{P'_Y} > subaction\_gap_{\max}^{P'_Y}$$

The time events t_0 through t_5 are identical to the previous analyses. In this scenario, t_6 follows t_2 by the time it takes PHY X to time subaction_gap and arb_delay:

(47) $$\begin{aligned} t_6 &= t_2 + subaction\_gap^{P_X} + arb\_delay^{P_X} \\ &= t_0 + \frac{packet\_length}{packet\_speed \cdot BASERATE_X} + DATA\_END\_TIME_X^{P_X} + \\ &\quad subaction\_gap^{P_X} + arb\_delay^{P_X} \end{aligned}$$

The 1995 specification provides the timeouts used internally by the state machine. The externally observed timing requirements could differ (given possible mismatches in transceiver delay and state machines between the leading edge of IDLE and the leading edge of the subsequent arbitration indication). However, previous works have suggested any such delays could and should be well matched and that the external timing would follow the internal timing exactly. Consequently,

(48) $$subaction\_gap^{P_X} + arb\_delay^{P_X} = subaction\_gap^{i_X} + arb\_delay^{i_X}$$

T7 follows T6 by the time it takes the arbitration signal to propagate through the intervening PHYs and cables:

(49)
$$\begin{aligned} t_7 &= t_6 + \sum_{n=X+1}^{Y-1}\left(cable\_delay_n + ARB\_RESPONSE\_DELAY_n^{P'_n \to P_n}\right) + cable\_delay_X \\ &= t_0 + \frac{packet\_length}{packet\_speed \cdot BASERATE_X} + DATA\_END\_TIME_X^{P_X} + \\ &\quad subaction\_gap^{i_X} + arb\_delay^{i_X} + \\ &\quad \sum_{n=X+1}^{Y-1}\left(cable\_delay_n + ARB\_RESPONSE\_DELAY_n^{P'_n \to P_n}\right) + cable\_delay_X \end{aligned}$$

Given t_0 through t_7 above, the Idle time seen at point P'_Y is given as:

(50)
$$\begin{aligned} Idle^{P'_Y} &= t_7 - t_5 \\ &= subaction\_gap^{i_X} + arb\_delay^{i_X} - \\ &\quad \sum_{n=X+1}^{Y-1}\left(PHY\_DELAY_n^{P'_n \to P_n} - ARB\_RESPONSE\_DELAY_n^{P'_n \to P_n}\right) - \\ &\quad DE\_delta^{[P_{Y-1},P_X]} - PPM\_delta^{[Y-1,X]} \end{aligned}$$

Let

(51)
$$Data\_Arb\_Mismatch^{[P_X \to P_Y]} = \sum_{n=X+1}^{Y-1}\left(PHY\_DELAY_n^{P'_n \to P_n} - ARB\_RESPONSE\_DELAY_n^{P'_n \to P_n}\right)$$

Then,

(52)
$$\begin{aligned} Idle^{P'_Y} &= t_7 - t_5 \\ &= subaction\_gap^{i_X} + arb\_delay^{i_X} - Data\_Arb\_Mismatch^{[P_X \to P_Y]} - \\ &\quad DE\_delta^{[P_{Y-1},P_X]} - PPM\_delta^{[Y-1,X]} \end{aligned}$$

For the maximum subaction_gap detection time at point P'_Y, the 1995 standard again only specifies the internal state machine timeout values. Figure 6 provides the timing reference for relating the external gap detection times to the internal ones. The figure is not to scale.

Figure 6: Internal Gap Detection Sequence

```
                 t3        t4                    t5    t7
P'Y:    --< DP | Packet |       DATA_END        >---<    ARB

                                                   t5a  t7a
iY:     ----------------< stop_tx_packet() >----< ARB seen

                     t3'       t4'                   t5' t7'
PY:     -----< DP | Packet |       DATA_END        >---<    ARB
```

The elaborated timing sequence is identical to the case for RESPONSE_TIME through point t_5'. The remaining sequence is:

t_7 The arbitration indication launched by PHY X arrives at point P'_Y

t_{7a} The arbitration indication launched by PHY X arrives at point iY. t_{7a} lags t_7 by an unspecified arbitration detection time, herein termed ARB_DETECTION_TIME

The externally seen gap at point P'_Y is given as

(53) $$gap^{P'_Y} = t_7 - t_5$$

The corresponding internal gap at point iY is

(54) $$gap^{i_Y} = t_{7a} - t_{5a}$$

Given that

(55) $$t_{7a} = t_7 + ARB\_DETECTION\_TIME_Y^{P'_Y}$$

the external gap can be expressed as

(56)
$$\begin{aligned} gap^{P'_Y} &= t_7 - t_5 \\ &= t_{7a} - t_5 - ARB\_DETECTION\_TIME_Y^{P'_Y} \\ &= t_{7a} - t_{5a} + t_{5a} - t_5 - ARB\_DETECTION\_TIME_Y^{P'_Y} \\ &= gap^{i_Y} + t_{5a} - t_5 - ARB\_DETECTION\_TIME_Y^{P'_Y} \\ &= gap^{i_Y} + PHY\_DELAY_Y^{P'_Y \to P_Y} + DE\_delta^{[P_Y, P_{Y-1}]} + PPM\_delta^{[Y,Y-1]} - \\ &\quad transceiver\_delay_Y^{P_Y} - ARB\_DETECTION\_TIME_Y^{P'_Y} \end{aligned}$$

Consequently,

(57)
$$\begin{aligned} subaction\_gap^{P'_Y} = {} & subaction\_gap^{i_Y} + PHY\_DELAY_Y^{P'_Y \to P_Y} + \\ & DE\_delta^{[P_Y, P_{Y-1}]} + PPM\_delta^{[Y,Y-1]} - \\ & transceiver\_delay_Y^{P_Y} - ARB\_DETECTION\_TIME_Y^{P'_Y} \end{aligned}$$

Substituting (52) and (57) into (46) yields

(58)
$$\begin{bmatrix} subaction\_gap^{i_X} + arb\_delay^{i_X} - \\ Data\_Arb\_Mismatch^{[P_X \to P_Y]} - \\ DE\_delta^{[P_{Y-1}, P_X]} - PPM\_delta^{[Y-1,X]} \end{bmatrix} > \begin{bmatrix} subaction\_gap^{i_Y} + PHY\_DELAY_Y^{P'_Y \to P_Y} + \\ DE\_delta^{[P_Y, P_{Y-1}]} + PPM\_delta^{[Y,Y-1]} - \\ transceiver\_delay_Y^{P_Y} - ARB\_DETECTION\_TIME_Y^{P'_Y} \end{bmatrix}$$

The inequality holds generally if

$$(59)\quad \left[subaction\_gap^{i_X} + arb\_delay^{i_X}\right]_{\min} > \begin{bmatrix} subaction\_gap^{i_Y} + PHY\_DELAY_Y^{P_Y^{'} \to P_Y} + \\ DE\_delta^{[P_Y, P_{Y-1}]} + PPM\_delta^{[Y, Y-1]} + \\ DE\_delta^{[P_{Y-1}, P_X]} + PPM\_delta^{[Y-1, X]} + \\ Data\_Arb\_Mismatch^{[P_X \to P_Y]} - \\ transceiver\_delay_Y^{P_Y} - ARB\_DETECTION\_TIME_Y^{P_Y^{'}} \end{bmatrix}_{\max}$$

Combining the DE_Delta and PPM_delta terms gives:

$$(60)\quad \left[subaction\_gap^{i_X} + arb\_delay^{i_X}\right]_{\min} > \begin{bmatrix} subaction\_gap^{i_Y} + PHY\_DELAY_Y^{P_Y^{'} \to P_Y} + \\ DE\_delta^{[P_Y, P_X]} + PPM\_delta^{[Y, X]} + \\ Data\_Arb\_Mismatch^{[P_X \to P_Y]} - \\ transceiver\_delay_Y^{P_Y} - ARB\_DETECTION\_TIME_Y^{P_Y^{'}} \end{bmatrix}_{\max}$$

By assuming

$$(61)\quad DE\_delta^{[P_Y, P_X]} + PPM\_delta^{[Y, X]} \le transceiver\_delay_Y^{P_Y} + ARB\_DETECTION\_TIME_Y^{P_Y^{'}}$$

the constraining inequality can be further simplified to give

$$(62)\quad \left[subaction\_gap_{\min}^{i_X} + arb\_delay_{\min}^{i_X}\right] > \begin{bmatrix} subaction\_gap_{\max}^{i_Y} + PHY\_DELAY_{Y,\max}^{P_Y^{'} \to P_Y} + \\ Data\_Arb\_Mismatch_{\max}^{[P_X \to P_Y]} \end{bmatrix}$$

where

$$(63)\quad subaction\_gap_{\min}^{i_X} = \frac{27 + gap\_count \cdot 16}{BASERATE_{X,\max}}$$

$$(64)\quad arb\_delay_{\min}^{i_X} = \frac{gap\_count \cdot 4}{BASERATE_{X,\max}}$$

and

$$(65)\quad subaction\_gap_{\max}^{i_Y} = \frac{29 + gap\_count \cdot 16}{BASERATE_{Y,\min}}$$

Solving for gap count,

$$\text{(66)} \qquad gap\_count > \frac{BASERATE_{X,\max} \cdot \begin{bmatrix} PHY\_DELAY_{Y,\max}^{P_Y^{'} \rightarrow P_Y} + \\ Data\_Arb\_Mismatch_{\max}^{[P_X \rightarrow P_Y]} \end{bmatrix} + 29 \cdot \frac{BASERATE_{X,\max}}{BASERATE_{Y,\min}} - 27}{20 - 16 \cdot \frac{BASERATE_{X,\max}}{BASERATE_{Y,\min}}}$$

7 Maximum Arbitration Reset Timings

For any given topology, the gap count must be set such that if an arbitration reset gap is observed following an asynchronous packet at one PHY, it is observed at all PHYs. The danger occurs when a subsequent arbitration indication is transmitted in the same direction as the previous data packet. Given that arbitration indications may propagate through intervening PHYs faster than data bits, gaps may be shortened as they are repeated. Figure 5 illustrates the case in which PHY X originates an asynchronous packet, observes an arbitration reset gap, and begins to drive an arbitration indication.

Figure 7: Consistent Arbitration Reset Gap Detection

```
             t0       t1   t2                  t6
PX    --< DP | Packet | DE >------------------< ARB |
PX+1  -----< DP | Packet | DE >-----------------< ARB |
PN-1  --------< DP | Packet | DE >----------------< ARB |
PN    -----------< DP | Packet | DE >---------------< ARB |
PN+1  --------------< DP | Packet | DE >--------------< ARB |
PY-1  -----------------< DP | Packet | DE >-------------< ARB |
P'Y   --------------------< DP | Packet | DE >------------< ARB |
                              t3        t4   t5           t7
```

For all topologies, the minimum idle time observed at point P'_Y must always exceed the maximum arbitration reset gap detection time:

(67) $$Idle_{\min}^{P_Y'} > arb\_reset\_gap_{\max}^{P_Y'}$$

The time events t_0 through t_5 are identical to the previous analyses. In this scenario, t_6 follows t_2 by the time it takes PHY X to time arb_reset_gap and arb_delay:

(68) $$\begin{aligned} t_6 &= t_2 + arb\_reset\_gap^{P_X} + arb\_delay^{P_X} \\ &= t_0 + \frac{packet\_length}{packet\_speed \cdot BASERATE_X} + DATA\_END\_TIME_X^{P_X} + \\ &\quad arb\_reset\_gap^{P_X} + arb\_delay^{P_X} \end{aligned}$$

The 1995 specification provides the timeouts used internally by the state machine. The externally observed timing requirements could differ (given possible mismatches in transceiver delay and state machines between the leading edge of IDLE and the leading edge of the subsequent arbitration indication). However, previous works have suggested any such delays could and should be well matched and that the external timing would follow the internal timing exactly. Consequently,

(69) $$arb\_reset\_gap^{P_X} + arb\_delay^{P_X} = arb\_reset\_gap^{i_X} + arb\_delay^{i_X}$$

T7 follows T6 by the time it takes the arbitration signal to propagate through the intervening PHYs and cables:

(70)
$$
\begin{aligned}
t_7 &= t_6 + \sum_{n=X+1}^{Y-1}\left(cable\_delay_n + ARB\_RESPONSE\_DELAY_n^{P_n^{'}\to P_n}\right) + cable\_delay_X \\
&= t_0 + \frac{packet\_length}{packet\_speed \cdot BASERATE_X} + DATA\_END\_TIME_X^{P_X} + \\
&\quad arb\_reset\_gap^{i_X} + arb\_delay^{i_X} + \\
&\quad \sum_{n=X+1}^{Y-1}\left(cable\_delay_n + ARB\_RESPONSE\_DELAY_n^{P_n^{'}\to P_n}\right) + cable\_delay_X
\end{aligned}
$$

Given t_0 through t_7 above, the Idle time seen at point P'_Y is given as:

(71)
$$
\begin{aligned}
Idle^{P_Y^{'}} &= t_7 - t_5 \\
&= arb\_reset\_gap^{i_X} + arb\_delay^{i_X} - Data\_Arb\_Mismatch^{[P_X\to P_Y]} - \\
&\quad DE\_delta^{[P_{Y-1},P_X]} - PPM\_delta^{[Y-1,X]}
\end{aligned}
$$

For the maximum arbitration_reset_gap detection time at point P'_Y, equation (56) gives:

(72)
$$
\begin{aligned}
arb\_reset\_gap^{P_Y^{'}} = {} & arb\_reset\_gap^{i_Y} + PHY\_DELAY_Y^{P_Y^{'}\to P_Y} + \\
& DE\_delta^{[P_Y,P_{Y-1}]} + PPM\_delta^{[Y,Y-1]} - \\
& transceiver\_delay_Y^{P_Y} - ARB\_DETECTION\_TIME_Y^{P_Y^{'}}
\end{aligned}
$$

Substituting (71) and (72) into (67) yields

(73)
$$
\begin{bmatrix}
arb\_reset\_gap^{i_X} + arb\_delay^{i_X} - \\
Data\_Arb\_Mismatch^{[P_X\to P_Y]} - \\
DE\_delta^{[P_{Y-1},P_X]} - PPM\_delta^{[Y-1,X]}
\end{bmatrix}
>
\begin{bmatrix}
arb\_reset\_gap^{i_Y} + PHY\_DELAY_Y^{P_Y^{'}\to P_Y} + \\
DE\_delta^{[P_Y,P_{Y-1}]} + PPM\_delta^{[Y,Y-1]} - \\
transceiver\_delay_Y^{P_Y} - ARB\_DETECTION\_TIME_Y^{P_Y^{'}}
\end{bmatrix}
$$

The inequality holds generally if

(74)
$$
\left[arb\_reset\_gap^{i_X} + arb\_delay^{i_X}\right]_{\min} >
\begin{bmatrix}
arb\_reset\_gap^{i_Y} + PHY\_DELAY_Y^{P_Y^{'}\to P_Y} + \\
DE\_delta^{[P_Y,P_{Y-1}]} + PPM\_delta^{[Y,Y-1]} + \\
DE\_delta^{[P_{Y-1},P_X]} + PPM\_delta^{[Y-1,X]} + \\
Data\_Arb\_Mismatch^{[P_X\to P_Y]} - \\
transceiver\_delay_Y^{P_Y} - ARB\_DETECTION\_TIME_Y^{P_Y^{'}}
\end{bmatrix}_{\max}
$$

Combining the DE_Delta and PPM_delta terms gives:

$$(75)\quad \left[arb\_reset\_gap^{i_X} + arb\_delay^{i_X}\right]_{\min} > \begin{bmatrix} arb\_reset\_gap^{i_Y} + PHY\_DELAY_Y^{P_Y^{'} \to P_Y} + \\ DE\_delta^{[P_Y, P_X]} + PPM\_delta^{[Y,X]} + \\ Data\_Arb\_Mismatch^{[P_X \to P_Y]} - \\ transceiver\_delay_Y^{P_Y} - ARB\_DETECTION\_TIME_Y^{P_Y^{'}} \end{bmatrix}_{\max}$$

By requiring

$$(76)\quad DE\_delta^{[P_Y, P_X]} + PPM\_delta^{[Y,X]} \le transceiver\_delay_Y^{P_Y} + ARB\_DETECTION\_TIME_Y^{P_Y^{'}}{}_Y$$

the constraining inequality can be further simplified to give

$$(77)\quad \left[arb\_reset\_gap_{\min}^{i_X} + arb\_delay_{\min}^{i_X}\right] > \begin{bmatrix} arb\_reset\_gap_{\max}^{i_Y} + PHY\_DELAY_{Y,\max}^{P_Y^{'} \to P_Y} + \\ Data\_Arb\_Mismatch_{\max}^{[P_X \to P_Y]} \end{bmatrix}$$

where

$$(78)\quad arb\_reset\_gap_{\min}^{i_X} = \frac{51 + gap\_count \cdot 32}{BASERATE_{X,\max}}$$

$$(79)\quad arb\_delay_{\min}^{i_X} = \frac{gap\_count \cdot 4}{BASERATE_{X,\max}}$$

and

$$(80)\quad arb\_reset\_gap_{\max}^{i_Y} = \frac{53 + gap\_count \cdot 32}{BASERATE_{Y,\min}}$$

Solving for gap count,

$$(81)\quad gap\_count > \frac{BASERATE_{X,\max} \cdot \begin{bmatrix} PHY\_DELAY_{Y,\max}^{P_Y^{'} \to P_Y} + \\ Data\_Arb\_Mismatch_{\max}^{[P_X \to P_Y]} \end{bmatrix} + 53 \cdot \frac{BASERATE_{X,\max}}{BASERATE_{Y,\min}} - 51}{36 - 32 \cdot \frac{BASERATE_{X,\max}}{BASERATE_{Y,\min}}}$$

8 Combined Gap Count Limits and Minimum ARB_RESPONSE_DELAY

Equations (31), (45), (66) and (81) place a lower bound on gap count. Let:

(82) $$gap\_count_A = \frac{BASERATE_{X,\max} \cdot \begin{bmatrix} Round\_Trip\_Delay_{\max}^{[P_X \circlearrowleft P_Y]} + \\ RESPONSE\_TIME_{Y,\max}^{P_Y'} + \\ PHY\_DELAY_{X,\max}^{P_X} \end{bmatrix} - 27}{16}$$

(83) $$gap\_count_B = \frac{BASERATE_{X,\max} \cdot \begin{bmatrix} Round\_Trip\_Delay_{\max}^{[P_X \circlearrowleft P_Y]} + \\ RESPONSE\_TIME_{Y,\max}^{P_Y'} - \\ MIN\_IDLE\_TIME_Y + \\ PHY\_DELAY_{X,\max}^{P_X} \end{bmatrix} + 29 \cdot \frac{BASERATE_{X,\max}}{BASERATE_{Y,\min}} - 51}{32 - 20 \cdot \frac{BASERATE_{X,\max}}{BASERATE_{Y,\min}}}$$

(84) $$gap\_count_C = \frac{BASERATE_{X,\max} \cdot \begin{bmatrix} Data\_Arb\_Mismatch_{\max}^{[P_X \to P_Y]} + \\ PHY\_DELAY_{Y,\max}^{P_Y' \to P_Y} \end{bmatrix} + 29 \cdot \frac{BASERATE_{X,\max}}{BASERATE_{Y,\min}} - 27}{20 - 16 \cdot \frac{BASERATE_{X,\max}}{BASERATE_{Y,\min}}}$$

(85) $$gap\_count_D = \frac{BASERATE_{X,\max} \cdot \begin{bmatrix} Data\_Arb\_Mismatch_{\max}^{[P_X \to P_Y]} + \\ PHY\_DELAY_{Y,\max}^{P_Y' \to P_Y} \end{bmatrix} + 53 \cdot \frac{BASERATE_{X,\max}}{BASERATE_{Y,\min}} - 51}{36 - 32 \cdot \frac{BASERATE_{X,\max}}{BASERATE_{Y,\min}}}$$

Given the ratio of maximum to minimum BASERATE is always > 1 and that MIN_IDLE_TIME is ~40 ns, it is clear that:

(86) $$gap\_count_B > gap\_count_A$$

and

(87) $$gap\_count_D > gap\_count_C$$

To select an appropriate gap count for a given topology, both gap_count_B and gap_count_D must be calculated, rounded up to the next integer, and the maximum of the two results selected.

9 Table-Based Gap Count Selection

For IEEE1394-1995 style topologies (assumed to be limited to 4.5m cables and a worst case PHY_DELAY of 144 ns), a table can be constructed to provide the gap count setting as a function of hops. In constructing such a table, the constant values in Table 2 are assumed.

Table 2: PHY Timing Constants

Parameter	Minimum	Maximum
ARB_RESPONSE_DELAY[1]	PHY_DELAY(max) – 60 ns	PHY_DELAY(max)
BASERATE	98.294 mbps	98.314 mbps
cable_delay		22.725 ns
MIN_IDLE_TIME	40 ns	
PHY_DELAY		144 ns
RESPONSE_TIME		PHY_DELAY + 100 ns

The resulting gap count versus Cable Hops can then be calculated:

Table 3 : Gap Count as a function of hops

Hops	Gap Count
1	5
2	7
3	8
4	10
5	13
6	16
7	18
8	21
9	24
10	26
11	29
12	32
13	35
14	37
15	40
16	43
17	46
18	48
19	51
20	54
21	57
22	59
23	62

[1] Note that the values for ARB_RESPONSE_DELAY used for the gap count calculation don't hold generally. However, it can be argued that they hold in the limiting scenarios for gap count. See the later discussion regarding recommended changes to the draft.

10 Suggested Changes or Additions to Draft 2.0

1. Reference all normative timings to the external interface, not the internal state machines. The 1995 defined min/max values for subaction gap, arb_reset_gap, and arb_delay can still be used by the state machine, but external timings are required to aid compliance testing, debugging, etc. From the analysis presented within, the 1995 gap detection timings specified in Sub-clause 4.3.6 (Tables 4-22 and 4-34) should be replaced with those in the following table. (The reference numbers are included here only for easy correspondence to that analysis and should not be included in the draft.)

	minimum	ref	maximum	ref
subaction_gap detection	$\frac{27 + gap\_count \cdot 16}{BASERATE_{max}} - PHY\_DELAY_{max}$	(15) (16)	$\frac{29 + gap\_count \cdot 16}{BASERATE_{min}} + PHY\_DELAY_{max}$	(57) (61) (65)
arb_reset_gap detection	$\frac{51 + gap\_count \cdot 32}{BASERATE_{max}} - PHY\_DELAY_{max}$	(35) (36)	$\frac{53 + gap\_count \cdot 32}{BASERATE_{min}} + PHY\_DELAY_{max}$	(72) (76) (80)
gap between isoch packet and asynch arbitration at originating node (within current fairness interval)	$\frac{27 + gap\_count \cdot 20}{BASERATE_{max}}$	(48) (63) (64)	$\frac{29 + gap\_count \cdot 20}{BASERATE_{min}} + RESPONSE\_TIME_{max} - MIN\_IDLE\_TIME$	(38) (41) (42) (43)
idle before first arbitration of new fairness interval at originating node	$\frac{51 + gap\_count \cdot 36}{BASERATE_{max}}$	(69) (78) (79)		

2. ARB_RESPONSE_DELAY is a difficult parameter to characterize. Proper PHY operation requires that arb signals propagate at least as fast as the data bits, otherwise the arbitration indications could shorten as they are repeated through a network. This fact places a bound on the maximum ARB_RESPONSE_DELAY: ARB_RESPONSE_DELAY between two ports at a particular instant must always be less than or equal to the data repeat delay at the very same instant. Although the distinction is subtle, this is not the same as saying the maximum ARB_RESPONSE_DELAY is PHY_DELAY. (PHY_DELAY only applies to the first bit of a packet and is known to have some jitter from one repeat operation to the next. Consequently, requiring ARB_RESPONSE_DELAY <= PHY_DELAY doesn't force ARB_RESPONSE_DELAY to track the instantaneous PHY_DELAY nor does it allow ARB_RESPONSE_DELAY to track the data repeat time for the last bit of a packet which may actually exceed PHY_DELAY due to PPM drift.) Finally, the table approach to calculating gap_count_a and gap_count_b rely on ARB_RESPONSE_DELAY always being bounded by the maximum PHY_DELAY when determining the Round_Trip_Delay.

   The minimum ARB_RESPONSE_DELAY is only of significance when calculating Data_Arb_Mismatch as required by gap_count_c and gap_count_d. Ideally, Data_Arb_Mismatch should be a constant regardless of PHY_DELAY so that neither gap_count_c nor gap_count_d will begin to dominate the gap_count setting as PHY_DELAY increases. Consequently, the minimum ARB_RESPONSE_DELAY should track the instantaneous PHY_DELAY with some offset for margin. Simply specifying the min value as a function of PHY_DELAY is ambiguous, however, since PHY_DELAY can be easily confused with the max DELAY reported in the register map. (For example, with DELAY at 144 ns, it would be easy to assume a min of PHY_DELAY – 60 ns would be equivalent to 84 ns. But if the worst case first bit repeat delay was only 100 ns, arb signals repeating with a delay of 40 ns ought to be considered within spec even though the delay is < 84 ns.)

Consequently, specifying an upper and a lower bound for ARB_RESPONSE_DELAY is best done in the standard with words rather than values. The minimum and maximum values for ARB_RESPONSE_DELAY in Table 7-14 should be changed to “See comment” and the comment should include: Between all ordered pairs of ports, the PHY shall repeat arbitration line states at least as fast as clocked data, but not more than 60 ns faster than clocked data.

Perhaps a better approach would be to replace ARB_RESPONSE_DELAY with the parameter DELAY_MISMATCH which is defined in the comment column as “Between all ordered pairs of ports, the instantaneous repeat delay for data less the instantaneous repeat delay for arbitration line states.” Then, the minimum would be given as 0 ns and the maximum would be 60 ns.

For a table based calculation of Round_Trip_Delay, one could argue that either approach above allows the use of PHY_DELAY(max) for ARB_RESPONSE_DELAY. Since Round_Trip_Delay considers the arbitration repeat delay in the direction opposite to the original packet flow, the return arbitration indication of interest is known to arrive at the receive port when the PHY is idle (all caught up with nothing to repeat). At that point, the instantaneous PHY_DELAY is the same as the first data bit repeat delay which is bounded by PHY_DELAY(max). Since ARB_RESPONSE_DELAY is always bounded by the instantaneous PHY_DELAY, it to is bounded by PHY_DELAY(max) at the point the arbitration indication first arrives.

3. The minimum bound on PHY_DELAY is used by the bus manager when determining the round_trip_delay between leaf nodes that are *not* separated by the bus manager. The more precise the minimum bound, the more accurate the pinging calculation can be. Ideally then, the bound may want to scale with increasing PHY_DELAY. Alternatively, the lower bound could be calculated by examining the Delay field in the register map: if zero, the lower bound is assumed to be the fixed value specified (60 ns currently). If non-zero, the lower bound could then be determined by subtracting the jitter field (converted to ns) from the delay field (converted to ns).

4. In Table 6-1, the description of Delay should be updated to match the BRC accepted definition of PHY_DELAY (referenced to the 1st data bit, not any data bit) and to specify multiples of 1/BASE_RATE rather than 20 ns (which is a poor approximation to one SCLK). The description should read: “Worst-case repeat delay for the first data bit of a packet, expressed as 144 ns + 2*Delay/BASE_RATE.

5. The “Jitter” field was introduced to aid in selection of gap_count via pinging by describing the uncertainty found in any empirical measurement of Round_Trip_Delay. Since Round_Trip_Delay encompasses an “outbound” PHY_DELAY and a “return” ARB_RESPONSE_DELAY, the jitter term should capture uncertainty in both. However, the definition of jitter in Draft 2.0 fails to consider ARB_RESPONSE_DELAY. Consequently, the description of Jitter should be corrected. The needs of pinging can be met with the following description for Jitter: “Upper bound of the mean average of the worst case data repeat jitter (max/min variance) and the worst case arbitration repeat jitter (max/min variance), expressed as 2*(jitter + 1)/BASE_RATE.”

   Note that from the discussion on minimum PHY_DELAY, it may be desirable to require that if the delay field is non-zero, then the slowest first data bit repeat delay can be calculated by subtracting the jitter value from the delay value.

6. Clause C of the Draft requires substantial edits to conform to this analysis of gap counts and pinging.

	Reference	Requested Change
(a)	p. 163 last sentence	Reference to table 4-33 in 1995 standard should be replaced with reference to new normative detection timing proposed for P1394a above
(b)	p. 164 first parag. last sentence	"The only constraint is that gap count never be reduced to a value where some nodes perceive an arbitration reset gap while others observe a subaction gap." This is not accurate in that it doesn't capture all four constraining cases enumerated within this analysis. A better statement might be: "The only constraints are that the gap count never be reduced to a value where 1) all nodes don't consistently detect the end of a given isochronous period, 2) all nodes don't consistently detect the end of a given fairness interval, or 3) an asynchronous subaction is interrupted."
(c)	p. 164 2nd parag.	"The worst disparity between observed idle times occurs between whichever two nodes have the greatest round-trip delay for data transmission between them…" Based on the analysis within, this should be changed to: "The worst disparity between observed idle times can occur 1) between whichever two nodes have the greatest mismatch in one-way arbitration and data repeat delays, or 2) between whichever two nodes have the greatest round-trip delay for data transmission between them. The mismatch between data and arbitration repeating delay for each PHY is essentially 60 ns as specified by the PHY timing constants. Consequently, the total mismatch between two leaf nodes is given as: Data_Arb_Mismatch = 60 ns * the number of intervening PHYs According to IEEE Std 1394-1995, round-trip delay may be …
(d)	p. 164 Propagation time formulae	Remove $ARB\_RESPONSE\_DELAY_{max}$ from Propagation $time_{min}$; likewise, remove $ARB\_RESPONSE\_DELAY_{min}$ from Propagation $time_{max}$. The definition of RESPONSE_TIME has changed to include ARB_RESPONSE_DELAY since the time this section was originally written. Also, it may be wise to note that the summation of PHY jitter is exclusive of the leaf nodes. That is, the jitter is only summed for all intervening nodes, if any.
(e)	p. 164 last parag before NOTE	"The ping time, measured by link hardware, starts when the most significant bit of the ping packet is transferred from the link to the PHY and ends when a data prefix indication is signaled by the PHY." Actually, the ping time starts with the last bit of the packet sent to the PHY. This is the *least* significant bit rather than the most significant.

(f)	p. 165 last round-trip delay eq.	The last round-trip delay equation isn't correct in a few respects. Specifically, the maximum and minimums are reversed. Also, using the maximum for PHY delay b may be too aggressive. The equation should be: $RoundTripDelay_{(g,d)} = PropagationTime_{g,\max} + PropagationTime_{d,\max} + 2\times PhyDelay_{b,\max} - 2\times\left(PropagationTime_{b,\min} + 2\times PhyDelay_{b,\min}\right)$ where the appropriate maxima and minima are noted. The minimum PHY_DELAY will have to come from the timing constants, not the delay register.
(g)	pp. 165 next to last parag	Everything between the point beginning with "Once round-trip delays have been measured …" and ending with the exact gap count formula should be replaced with the following: For each *__ordered__* pair of leaf nodes X and Y, determine the round-trip delay from pinging and calculate the Data_Arb_Mismatch. Select the largest gap count yielded by the following formulas: $gap\_count = \dfrac{BASERATE_{\max}\cdot\left[Round\_Trip\_Delay_{\max} + RESPONSE\_TIME_{Y,\max} - MIN\_IDLE\_TIME + PHY\_DELAY_{X,\max}\right] + 29\cdot\dfrac{BASERATE_{\max}}{BASERATE_{\min}} - 51}{32 - 20\cdot\dfrac{BASERATE_{\max}}{BASERATE_{\min}}}$ $gap\_count = \dfrac{BASERATE_{\max}\cdot\left[Data\_Arb\_Mismatch_{\max} + PHY\_DELAY_{Y,\max}\right] + 53\cdot\dfrac{BASERATE_{\max}}{BASERATE_{\min}} - 51}{36 - 32\cdot\dfrac{BASERATE_{\max}}{BASERATE_{\min}}}$ Repeat for all ordered pairs of leaf nodes, keeping track of the largest gap count calculated. Round the resulting gap count up to the next largest integer; the resultant value may be transmitted in a PHY configuration packet to optimize Serial Bus performance.

<table>
<tr>
<td>(h)</td>
<td>p. 166
Table C-2</td>
<td>Replace Table C-2 with:
<table>
<tr><th>Hops</th><th>Gap Count</th></tr>
<tr><td>1</td><td>5</td></tr>
<tr><td>2</td><td>7</td></tr>
<tr><td>3</td><td>8</td></tr>
<tr><td>4</td><td>10</td></tr>
<tr><td>5</td><td>13</td></tr>
<tr><td>6</td><td>16</td></tr>
<tr><td>7</td><td>18</td></tr>
<tr><td>8</td><td>21</td></tr>
<tr><td>9</td><td>24</td></tr>
<tr><td>10</td><td>26</td></tr>
<tr><td>11</td><td>29</td></tr>
<tr><td>12</td><td>32</td></tr>
<tr><td>13</td><td>35</td></tr>
<tr><td>14</td><td>37</td></tr>
<tr><td>15</td><td>40</td></tr>
<tr><td>16</td><td>43</td></tr>
<tr><td>17</td><td>46</td></tr>
<tr><td>18</td><td>48</td></tr>
<tr><td>19</td><td>51</td></tr>
<tr><td>20</td><td>54</td></tr>
<tr><td>21</td><td>57</td></tr>
<tr><td>22</td><td>59</td></tr>
<tr><td>23</td><td>62</td></tr>
</table>
</td>
</tr>
</table>

7. Clause 3.3 should remove all algebraic equations since they do no match the existing Annex C or the suggested replacement for Annex C.

8. Table 5-17 needs to clarify the datum point used for assertions of indications on the PHY link interface. It isn't clear, for example, if "time from the assertion of Idle on Ctl[0:1] …" is referenced to the Ctl[0:1] lines or to the first SCLK edge at the PHY upon which Ctl[0:1] indicates Idle. The latter is preferred and recommended for BUS_TO_LINK_DELAY, DATA_PREFIX_TO_GRANT, and LINK_TO_BUS_DELAY.