Configuration Timeout Issue

I would like to raise some issues, ask some questions, and propose a solution.

In IEEE-1394-1995 it states:

CONFIG_TIMEOUT = 166.6 usec - 166.9 usec
MAX_ARB_STATE_TIME = 166.8 usec - 166.9 usec

```
state = T0;
if (arb_timer >= CONFIG_TIMEOUT) && (child_count < NPORT - 1)
        {
                signal PH_STATE.ind (CONFIG_TIMEOUT);
                state = T0;
        }
```

autonomous action:

```
if (MAX_ARB_STATE_TIME)
        {
                arb_state_timeout = TRUE;
        }
```

transistion: All:R0b

```
if (PH_CONT.req(bus_reset)|| connection_state_change()||arb_state_timeout)
        {
                PH_STATE.ind(BUS_RESET_START);
                initiated_reset = TRUE;
                state = R0;
        }
```

First of all CONFIG_TIMEOUT and MAX_ARB_STATE_TIME overlap. So the action is undefined:

a) do a bus_reset
b) do a config_timeout
c) do both

Question:
Does MAX_ARB_STATE_TIME apply to state T0?

To answer my own question, I think the Standard intended the following:
MAX_ARB_STATE_TIME does not apply to state T0 and looping at T0 was

the intended.

P1394a clarifies the time overlap issue by changing the parameters:

CONFIG_TIMEOUT = 166.6 usec - 166.9 usec
MAX_ARB_STATE_TIME = 200 usec - 400 usec

However, p1394a confuses the issues by not filling in all the blanks.

I will explain:

Section 6.1 (PHY register map) of p1394a draft 1.4 (page 71) there is the following statement:

" When any one of the Loop, Pwr_fail, Timeout, or Port_event bits transitions from zero to one, PHY_interrupt shall be set to one. If the link is active, PHY_interrupt is reported as S[3] in a PHY status transfer, as specified by clause 5.5; other wise a PHY interrupt shall cause LinkOn to be asserted."

- I interpret from the above statement, "bits transitions from zero to one, PHY_interrupt shall be set to one" and the fact that no clear condition is mentioned that the PHY will only interrupt once per condition change. I also interpret that each condition will cause its own interrupt; even if other condition exists.
- (side note: why would you turn the link on when you have a Pwr_fail condition??? Also, to turn the link on without proper bus manager control for any of the above conditions may bring all cable powered nodes down.)

Section 7.10.3.2 (Tree identify) of p1394a draft 1.4(page 107) is the same as IEEE-1394-1995. See –1995 explanation above. It does not support Section 6.1s explanation.

Section 7.10.3.2.2 (Tree ID actions and conditions) of p1394a draft 1.4 (page 109) is the same as IEEE-1394-1995 and is missing this scenario completely.

Suggested fixes:

Section 6.1 Clarified

When any one of the following events occur (CONFIG_TIMEOUT, CABLE_POWER_FAIL, ARB_STATE_TIMEOUT, or PORT_EVENT) the PHY shall transition the following bits from zero to one (loop, Pwr_fail, Timeout, or Port_events respectfully) and PHY_interrupt shall be set to one. Reassertion of PHY_interrupt shall only take place if a new event (CONFIG_TIMEOUT, CABLE_POWER_FAIL, ARB_STATE_TIMEOUT, or PORT_EVENT) occurs.

With this description two new PHY event indications (PH_EVENT.indication) need to be defined:

ARB_STATE_TIMEOUT and PORT_EVENT

If the intent is to add these PH_EVENT.indications to IEEE-1394-1995 it should be stated as such. Since the PHY/Link interface is a standardized implementation NOT a requirement, I think that new requests, indications, and confirmations defined in this section should either be optional or defined in a non-implementation dependant area of P1394a.

Section 7.10.3.2 and 7.10.3.2.2 Clarified

Add the following to the R0 and T0:T0 states:

```
state = R0;
        loop = FALSE;

state = T0;
if ((arb_timer >= CONFIG_TIMEOUT) && (child_count < NPORT - 1)) && !loop)
        {
                signal PH_STATE.ind (CONFIG_TIMEOUT);
                loop = TRUE;
                state = T0;
        }
        else if
        {
                state = T0;
        }
```

I hope this clarifies the issues and starts the discussion.