Gap_count v.s. Bus Reset

Yasumasa Hasegawa
FUJIFILM Microdevices
yasumasa@ffm.fujifilm.co.jp

Gap_count reset to 3F

- So far, 1995-PHYs and old C-code (Draft1.3) resets the gap_count every two bus resets.
- “Two bus resets” meant “Two invokes of reset_start_actions()”.

Problem with Short Bus Reset

- Renard Ulrey pointed out that one short bus reset will be “reflected” at 1995 PHY and results in two consecutive resets (a short and a long).
- Short-reset-savvy PHY ends up resetting its gap_count to 3F.

“Follow-up Bus Reset”

- The draft requires a software to issue a bus reset after setting gap_count so that next bus reset (may will be caused by new connection) can be “the second” bus reset. (“9.21 Gap count optimization”)
- As a result, the follow-up bus reset may end up two consecutive resets and resets short-reset-savvy PHY’s gap_count to 3F!

C-code Fix

- PHYDOGS fixed the old C-code and now it is in the Draft 1.4.
- New C-code (Draft1.4) resets the gap_count every two bus “configurations”.

```
If (!bus_initialize_active) {
   bus_initialize_active = TRUE;//until end of self-ID phase
   if (gap_count_reset_disable)
      gap_count_reset_disable = FALSE;
   else
      gap_count = 0x3F;
}
```

Legacy issue

- Bill Duckwall presented “Connection Debouncing” and “Arbitrated Short Resets” at the TA meeting in January 1996.
- Most PHY vendors implemented “debounce” and “short reset” in their 1995 silicon.
- Gap_count v.s. short reset issue still remains and only software can solve the problem.

Recommended procedure

1. Set gap_count then issue an isbr.
2. Collect self-ID and examine.
3. If any inconsistencies exist, set gap_count then should **not** issue a bus reset.
4. After that, if it detects bus reset, it may be new connection. Examine gap_counts in collected self-ID and issue a bus reset if necessary.

Suggested change in “9.21 Gap count optimization”

“A node that transmits a PHY configuration packet with the T bit set to one shall initiate a bus reset as soon as possible after the PHY configuration has been sent and examine collected `gap_count` values. This is essential so that the `gap_count_reset_disable` variables at all node(s) is cleared to FALSE. If, however, any of the `gap_count` remains 63, this may be due to some legacy PHYs which reset its `gap_count` to 63 before completing bus configuration. If this is the case, a node that sets `gap_count` by transmitting a PHY configuration packet should not initiate a bus reset for the diagnostic purpose. Instead, the node should initiate a bus reset after observing a incoming bus reset. Without these precautions, the addition of a new node to the bus could result in different values of gap_count at different nodes and resultant unpredictable arbitration behavior.”