NIO

Half Duplex (CSMA/CD) MAC & PLCA Interaction

Yong Kim, yong dot kim at nio dot io

V.03

What's my concern? ← Just a starting point…

- (**Half-duplex**) DTE that TX and experiences COL, will ignore associated RX, even if it is RX'ed with good FCS and valid (> min packet) length.
- Why do I think that?
  - DTE that transmit expects its own TX reflected back to RX.
  - No collision case: RX sees good frame, receives it ok, and throw it out (content == TX).
    - Test w/ MAC DA == MAC SA, or MAC DA = [broadcast] to verify this behavior.
    - Collision case: RX sees fragment (content == mixture) . Fragment discarded.
  - But wait… What does standard say?
    - RX independent of TX or TX with COL. (CL4). If RX'ed with good FCS and valid length, it is received.
    - MA_DATA.indication has no dependency to MAC_DATA.request (CL2)
    - So standard says every frame a DTE transmits, it also receives it (in half duplex mode), so every MA_DATA.request is reflected back (through a medium) to MA_DATA.indication (for MAC DA match).
    - This **CANNOT RIGHT**, but this **IS** right, according to the standard (802.3-2015 CL4, CL2).
    - For an example, every time an 802.1 bridge sends a broadcast frame onto an 802.3 segment, it should also be received and forwarded to all the other [VLAN] ports. ← Obviously this **IS WRONG**.
    - For an example, every time a station sends a broadcast frame, it must also receive the frame. ← this **SHOULD BE** wrong. I don't see transmitted ARP (broadcast) frame being reflected back onto my IP stack.

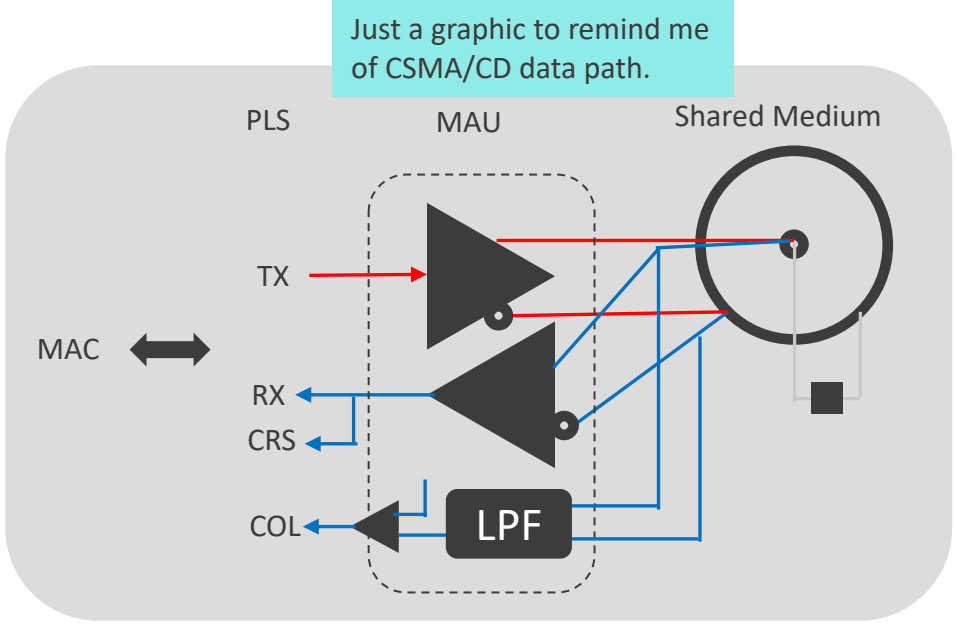
Contents

Assumptions

1. CSMA/CD (i.e. half-duplex) operates as half-duplex, i.e. either Transmit or Receive, but not allowed to do both at the same time.
2. Full-duplex operates as full-duplex, i.e. transmit and receive independently and at the same time.

<u>If you find yourself disagreeing (all of a sudden) with the statement #1 above, then yes, this presentation is about that. If you have no clue what I am saying, you will. So hang on.</u>

Some assumptions are so obvious, we actually miss stating it. Our bad!



Receive behavior during collision.

- Receive without contention (4.1.2.1.2) ← there is no reference to contention nor collision in this clause.
- Access interference and recovery (4.1.2.2)
  - [TX] deferral, network acquisition time (collision window), collision, jam, back-off.
  - [RX] received/decoded normally. “Fragmentary frames received during collisions are distinguished from valid transmissions by the MAC sublayer’s Receive Media Access Management component.” ← exact term not repeated. **Safe to assume fragment == length < min frame size*, and rx == length > min frame size.**

Note: Some may say this should be network acquisition time + jam + repeater enforcements, but not relevant to this presentation, so we will not get into that. Also the fragment length measured from preamble or from CRS – re: SFD detection sensitivity to length measurement – not relevant to this presentation, so we will not talk about that either.

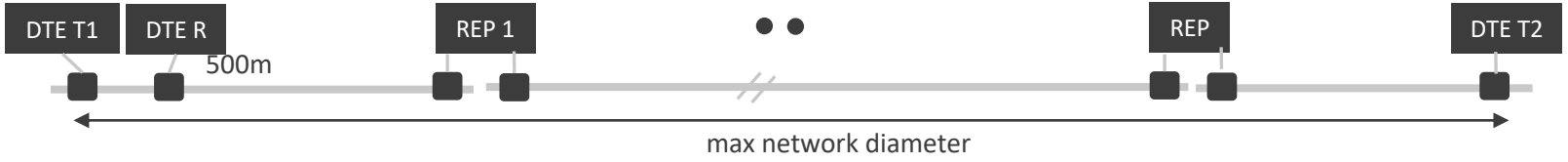
- Formal model (4.2) in text, pascal, and figure consistent with the above. So far so good.
- Let’s look at the following sub-clause and high-lighted text, just to see how far we go to ensure false reception of collision fragment as a good packet.

4.2.3.2.4 Collision detection and enforcement (half duplex mode only)

Collisions are detected by monitoring the collisionDetect signal provided by the Physical Layer. When a collision is detected during a packet transmission, the transmission is not terminated immediately. Instead, the transmission continues until additional bits specified by jamSize have been transmitted (counting from the time collisionDetect went on). This collision enforcement or jam guarantees that the duration of the collision is sufficient to ensure its detection by all transmitting stations on the network. The content of the jam is unspecified; it may be any fixed or variable pattern convenient to the Media Access implementation; however, the implementation shall not be intentionally designed to be the 32-bit CRC value corresponding to the (partial) packet transmitted prior to the jam.

Receive during collision. Just CSMA/CD stuff.

" …the implementation shall not be intentionally designed to be the 32-bit CRC value corresponding to the (partial) packet transmitted prior to the jam." WHY?



0. Setup – Max network diameter. DTE R is a receiver right next to the DTE T1.

1. DTE T1 sends. Just about DTE T1 signal arrives at DTE T2 (about 27 bit-time window), DTE T2 sends. Collision.

2. DTE T2 sends remaining 64 bit preamble and jams for 32 bits.

3. Repeaters repeats 96 bit (enforced) back toward DTE T1 direction, including the repeater 1.

4. Repeater 1 MAU signal is attenuated @ ~500 meters. Both DTE T1 and DTE R receivers are receiving DTE T1 signal. The only indication of the collision is bias voltage change (low pass filter, 1 vs more than 1 TX onto medium). AND DTE T1 has been transmitting near min packet length (~network acq time length).

5. If DTE T1 sends JAM that terminates with correct CRC value, DTE R (and others) may receive the frame ok, and this contains corrupted data (JAM) followed by good CRC value. DTE T1's receiver sees the same TX as RX as the DTE R.

6. Sending frames that terminates with good CRC values is bad. Std says "shall not"

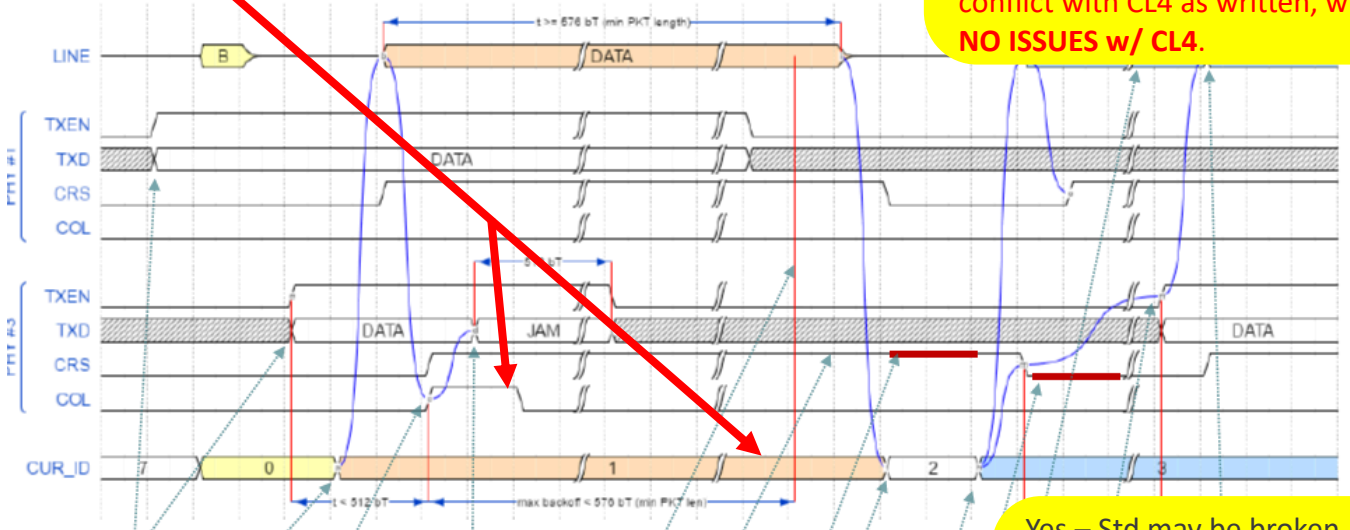
What does this all mean? 1) RX'ed during collision is not a good frame. Std goes out of its way to ensure it.

PLCA behavior

What's my issue here?
RX good frame while TX with COL @ MAC3

MAC3 upon COL *may not* store RX (assumes fragment). MAC3 TX will be tried again. BTW, FCS should not be trusted with large # of bit errors.

What does standard say? It says RX is completely independent of TX and COL. Just counts on frame length < min frame size. **The premise of PLCA**. PLCA forwards a good frame from MAC1 to MAC3 RX while forcing collision on MAC3. This does **NOT** conflict with CL4 as written, which is to say **NO ISSUES w/ CL4**.



Yes – Std may be broken. At least **incomplete** and assumed too much.

If half-duplex behaves this way, all 802.1 transparent bridges with Ethernet CSMA/CD is broken. There may be other broken stuff.

Bigger concern..

Use of CSMA/CD MAC with PLCA, when the premise of CS, and MA, and CD are not all there.

- PLCA takes over CD (as discussed prior), and manages CS to help with its cause.
- MA has issues too. Perhaps bigger issues.
  - Back-off and fair access and transmit opportunities with PLCA.

    A station that experienced COL (due do access), back-off timer force it into listen only mode for the back-off duration. Punishing a station that had a frame to transmit. For an example, if all had a frame to transmit, then every station (except for one) gets COL count 1 and back-off attempt. That one TX succeeds. Then all the remaining stations (doesn't matter it selected 0 or 1, after on normal RX, all timers expired) contend, and all but one (chosen by PLCA) gets COL count, now 2. PLCA round-robin access method aliasing with random back-off periods of respective stations. Which takes us to the capture effect.
  - Capture effect.

    Once a station successfully transmits, its back-off counter is zero, and it transmits IPG after carrier sense desertion. More contention there are, more likely that a station that transmits without COL will succeed with subsequent TX, because other stations are in greater back-off count. PLCA round-robin has little effect. You could argue that COL count may be artificially higher with PLCA under similar contention conditions, which would make the capture effect more pronounced (frankly, it does not need any help).

Capture Effect – yes, it is real.

- Ethernet Capture Effect in a nut-shell: Exponential back-off leads to self-adaptive use of channel
  - When a node succeeds, it transmits the next packet immediately.
  - Result: bursts of packets from single nodes.
- General description (not a bad one): https://en.wikipedia.org/wiki/Carrier-sense_multiple_access_with_collision_detection
- A better one (very good one) to really understand the issue (I always liked how Rich Seifert explains things). Go to directly to 3.3 of the reference if you are impatient sort: http://www.ethermanage.com/ethernet/pdf/techrept13.pdf
- And I have my own results on capture effect while I was developing Ethernet test equipment back in 1983. Two stations, max offered load, min pkt, 10 Mbps, single short segment. Capture effect where one gets to send versus the other had observed frequency of 2~ >300 seconds of only one sending vs the other, and capture effect was immediate, after ~ 1 second or less. This is what all the other discoverer found independently around the similar time frame.

- Why does it matter to PLCA?
  - Both the **Back-off** + **capture effect** makes the PLCA (as it stands) worst case access time to be worse than before, and before it was not bounded.

Summary & Conclusion

Summary

- PLCA counts on CL4. It works with CL4. YAY!... Wait.
- CL4, or CL2, or upper layer (Bridge relay or LLC or Network layer) has missing or incomplete specification WRT to RX during TX. SOMEWHERE there is an inconsistency WRT to this.
  - Ethernet **system** does NOT work the way CL2/CL4 states. Look at the bridge operation; Multicast DA TX; etc.
- "What PLCA is not a replacement of CSMA/CD → PLCA relies on it"
  - It's a bad idea to layer below CSMA/CD and rely on it. No way to fix fair-access deficiencies (capture, back-off)

Conclusion and possible solution

- PLCA did great job, working with published CL4.
- Automotive industry deserves full-duplex MAC, and fair access, AND deterministic latency, AND certainty that may come with its use.
- Much of the concepts in PLCA has analog (well, very loosely) in P2MP MAC in EPON, and it may be sensible to take PLCA above the MAC – in MAC control sublayer and preserve the access concept. ← possible solution.
- Inviting other ol'timers of 802.3 to help identify where the root cause of the inconsistency lies. And if appropriate, clarify it with a maintenance request of sorts to the right body.

Thank You