Report from the 802.3 Maintenance MAC Interface Adhoc

Jeff Mandin
PMC-Sierra
Sept 2008

Background

1. 802.3-2005 describes two different interfaces to the MAC:

   a) MA_DATA.Request/Indication in Clause 2
      - This is the ISO/IEC/802.1 interface

   b) TransmitFrame()/ReceiveFrame() in Clause 4

2. In 802.3as it was determined that this “longstanding discrepancy” between the interface definitions should be remedied

   - “Service to humanity”
   - State diagram added to reconcile MA_DATA.Request with TransmitFrame()
   - MAC Control Clauses (31, 31B, 64) were modified to use the Clause 2 “service interface primitives” instead of the TransmitFrame()/ReceiveFrame() Pascal functions

Adhoc Charter

1. Maintenance Request 1196 noted that the changes made in 802.3as to figures 64-12 and 64-13 (Multipoint MAC Control) changed the behaviour of those state diagrams

    - TransmitFrame was assumed to include a delay in those diagrams
    - Replacing TransmitFrame with MA_DATA.Request removed the delay

2. The Adhoc was formed with the following agenda:

    – Determine whether it is the use of the revised MAC interface by MAC Control or the revised interface itself that needs to be fixed

    – Determine if there are similar problems elsewhere in the 802.3 spec resulting from the 802.3as MAC interface change

    – Assess the urgency of the problem

Summary of Adhoc Activity

1. Participants (Telecon and Reflector):

   Brad Booth
   Wael Diab
   Howard Frazier
   Bob Grow
   Glen Kramer
   David Law
   Eric Lynskey
   Jeff Mandin
   Shimon Muller
   Glenn Parsons
   Peter Scruton
   Pat Thaler

2. Telecons held on July 30, Aug 20, Aug 27 (7-8 participants each time)

3. Approximately 25 emails on Maintenance Reflector

4. Consensus was established on the major issues

Highlights of the adhoc discussions

1. Many basic aspects of the MAC interface and its history were discussed on the telecons and on the reflector.

   - In particular, various models for interaction between the MAC client and the MAC were discussed.

   - One point that can be stated safely is that the MAC interface as it has evolved includes compromises and is not perfect.

2. In most cases (eg. 802.1 standards) the MAC client does not concern itself with when the transmission of a particular frame completes.

   - One way of explaining this is that it is implicitly left to the implementation of the MAC client to ensure that a frame transmission is not started until the MAC is ready.

3. But in the MAC Control layer we have state diagrams (eg. PAUSE) which specifically address timing issues. Consequently what is usually left as “implicit” (or “by magic”) needs to be made explicit in MAC Control.

   - We will do this by employing state diagram variables that signify the implicit or “magic” awareness of frame transmission completion

Summary of proposed corrections to 802.3

1. Figure 31B-1 (PAUSE transmit state diagram) needs to be modified so that *Send Data Frame* and *Send Control Frame* states are not exited until frame transmission has completed.

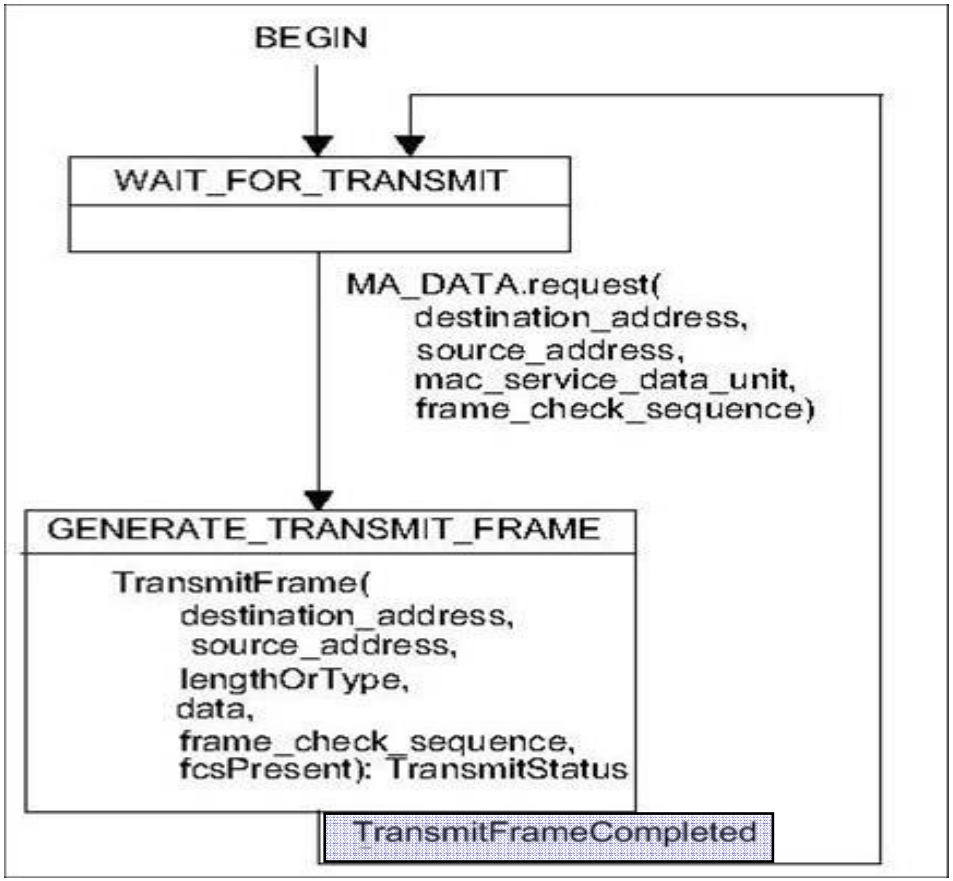
2. Though the intent of figure 4-6 (MAC client transmit interface state diagram) is clear, state diagram conventions in 21.7 are such that its interpretation is not unambiguous.

   - It is necessary to add an exit condition so that it is clear that the process remains in *Generate Transmit Frame* state until the TransmitFrame function completes

3. As with Figure 31B-1, Figures 64-12 and 64-13 need to be modified so that *Send Frame* And *Transmit Frame* states (respectively) are not exited until frame transmission has completed.

   - The adhoc did not specifically discuss the solution to these diagrams, but assumed that the resolution to PAUSE would apply to them as well

Revision #1 (to Figure 4-6)



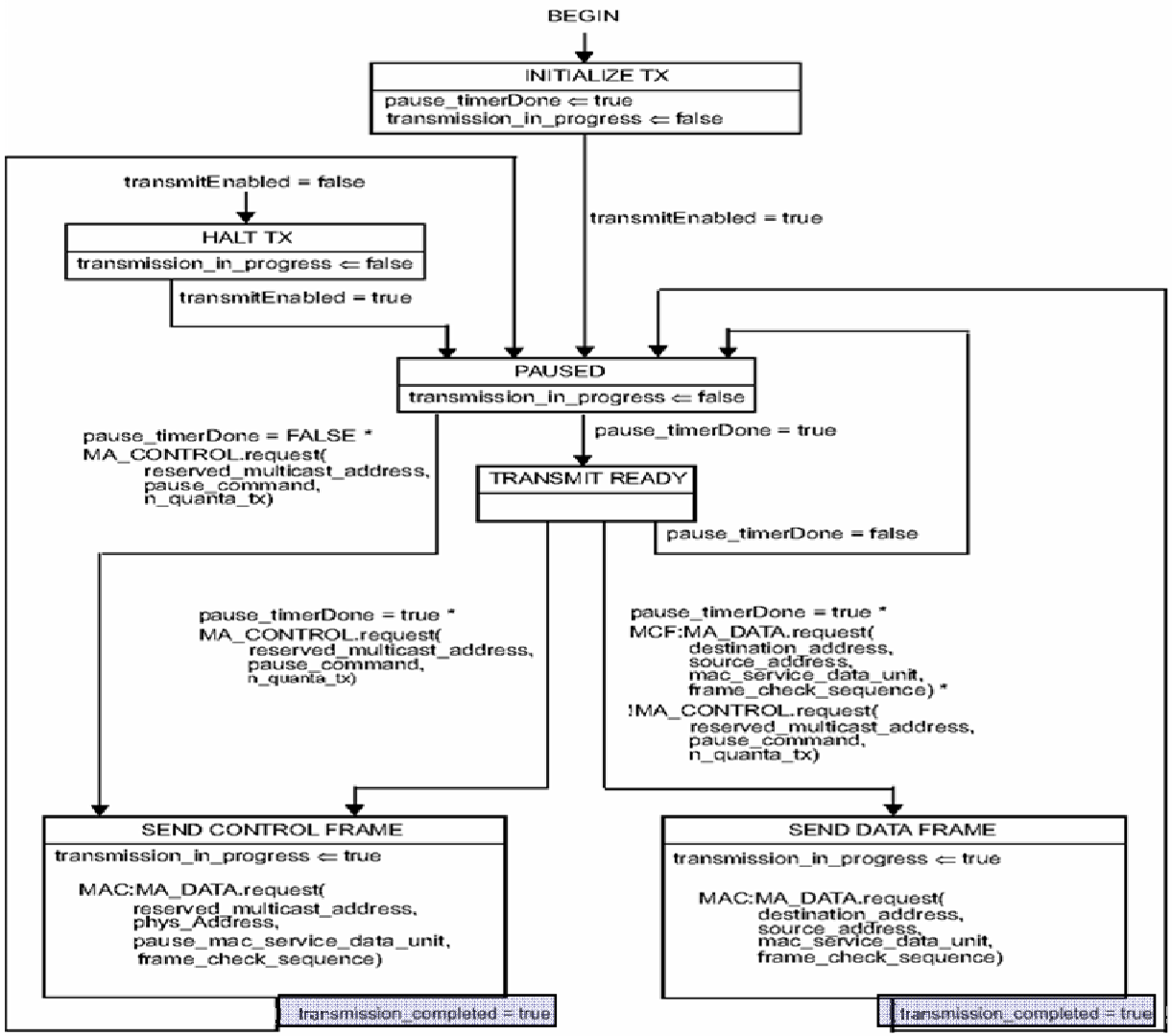
a) Change exit condition from UCT to new variable *TransmitFrameCompleted*

b) Add new variable to alphabetical list in 4.3.2.1.1 :

TransmitFrameCompleted

Boolean that becomes true when the TransmitFrame function has finished all of its processing

Revision #2 (to Figure 31B-1)



a) Change exit conditions from UCT to condition based on new variable *transmission_completed*

b) Add new variable to **31B.3.2.2** :

transmission_completed

Boolean that becomes true when the data transmission resulting from the invocation of MAC:MA_DATA.Request is complete

Revision #3 pt. 1 (figure 64-12)

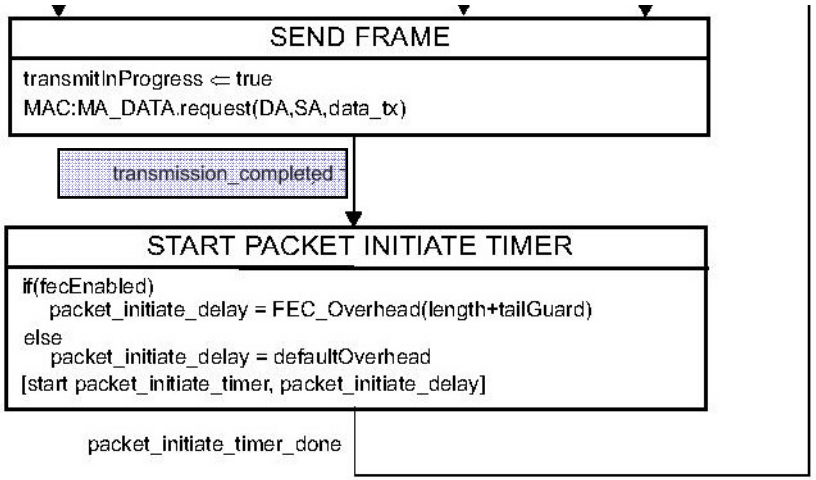


Figure 64–12—OLT Control Multiplexer state diagram

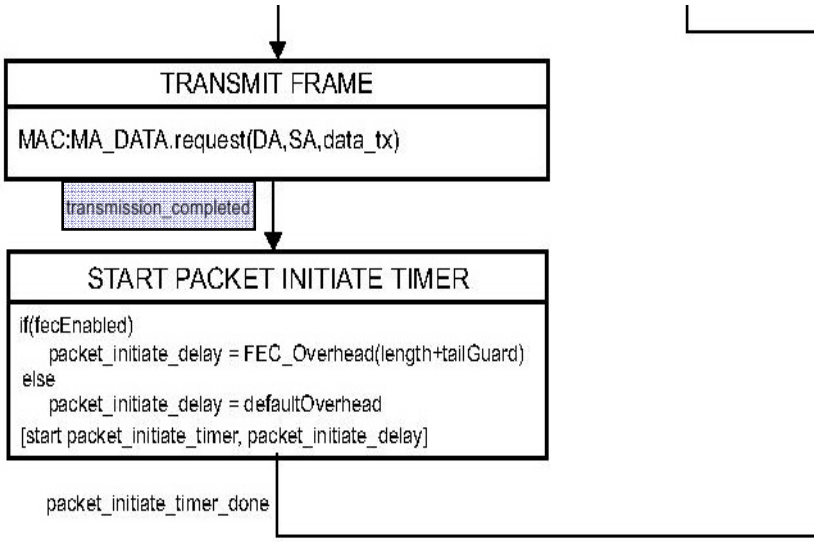
a) Change exit condition from UCT to new variable *transmission_completed*

b) Add new variable to alphabetical list in **64.2.2.3** :

transmission_completed

Boolean that becomes true when the data transmission resulting from the invocation of MAC:MA_DATA.Request is complete

Revision #3 pt. 2 (figure 64-13)



Instances of MAC data service interface:
MAC=interface to subordinate sublayer
MCI=interface to MAC Control multiplexer

Figure 64–13—ONU Control Multiplexer state diagram

- Change exit condition from UCT to new variable *transmission_completed*

Summary

1. The adhoc thoroughly reviewed the interface and interactions between the 802.3 MAC Control Layer and MAC

2. The adhoc recommends:
   - A clarification to a state diagram in the MAC layer
   - Minor corrections to 3 state diagrams in the MAC Control layer