IEEE P802.11  
Wireless LANs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Resolution of CID 7085 for D5.0 | | | | |
| Date: 2016-04-27 | | | | |
| Author(s): | | | | |
| Name | Affiliation | Address | Phone | email |
| Graham SMITH | SR Technologies | Davie, FL, USA. | 916 799 9563 | gsmith@srtrl.com |

Abstract

Resolutions for CID 7085 from D5

Green indicates material agreed to in the group,

yellow material to be discussed, red material rejected by the group and

cyan material not to be overlooked.

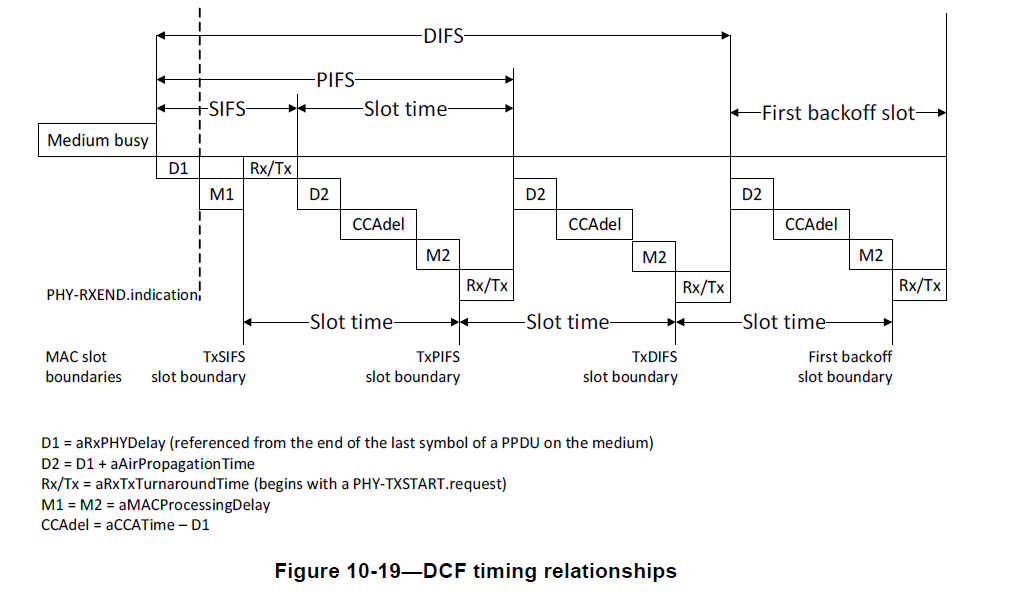
The “Final” view should be selected in Word.

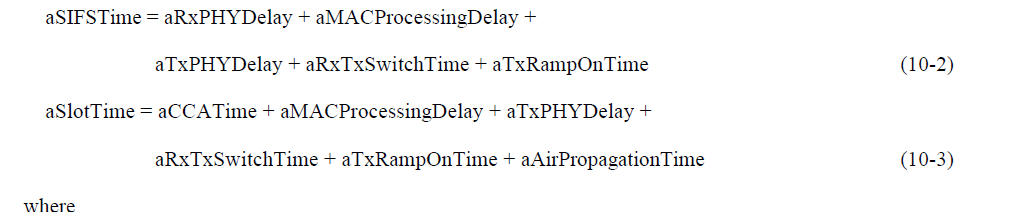


**Discussion**

Also note that CIDs 7086, 7087 and 7088 are related to this.

This is tied up with my comments on these related diagrams.





Now as far as the diagram is concerned the equations are right, but clearly, they are not correct.

For example, all these times are very small and in no way will add up to be equal to SIFS or SlotTime.

In another comment I point out that this diagram is completely misleading and in fact wrong. SIFS and SlotTime are fixed in the Standard, but all these other bits are not. The only criteria is that all these bit times must completed in a time less than SIFS or SlotTime.

Note that these values have no fixed values at all and are all implementation dependent.

*“The STA may employ any non-negative value for each of the parameters:*

* *aRxPHYDelay*
* *aMACProcessingDelay*
* *aRxTxTurnaroundTime*
* *aTxPHYDelay”*

But if these times are to mean anything, they are not just a number to be picked out of the air. Only one missing is *aCCATime,* what is this? First of all it is “implementation dependent” (in all PHY characteristic Tables).

“… the maximum time (in microseconds) that the CCA mechanism has available to detect the start of a valid IEEE Std 802.11 transmission…”

Again, no fixed value and in practice very short. Why the fixation with implementation switch over times? Thus simply confuses what the basic timing for DCF is.

How, therefore, can the equations 10-2 and 10-3 be correct? Obviously they cannot. There is no criteria that these arbitrary values must add up to be equal to SIFS or TimeSlot, which are fixed values.

In addition what about the Rx/Tx time, this is shown in every slot in the diagram. The STA only uses this when it actually is ready to transmit, and that is when the backoff timer has reached 0. As the diagram is drawn, the idea is that the STA waits SIFS minus Rx/Tx then calculates the backoff timer value. This is also not true. The wait period is DIFS. Also in the diagram the medium is busy at the beginning hence a random backoff slot must be calculated. So the STA must wait DIFS then backoff.

So the commenter is right but also the diagram needs to be changed.

Note also that the formulas do not even agree with the (old) diagram.

aSIFSTime is shown as D1 + M1 = aRxPHYDelay + aMACProcessingDelay ONLY

Why is a TXPHYDelay and aTXRampOn there as well as Rx/Tx

**We should have the formulas at least agree with the diagram, they do not at present.**

**Also note that**

aSIFSTime = aRxPHYDelay + aMACProcessingDelay +

aTxPHYDelay + aRxTxSwitchTime + aTxRampOnTime (10-2)

Diagram has

aSIFSTime = aRxPHYDelay + aMACProcessingDelay + (aRxTxSwitchTime)

If we have aRxTxSwitchTime then OK adding aTxPHYDelay + aTxRampOnTime, but this should not be in aSIFSTime.

Comments raised at first presentation, March 14, were along the lines of

The indeterminant terms are not properly described and relate back to when they probably had significant values. We should describe their use better.

Why? This is implementation stuff, the important point is to describe how DCF works, not how the STA changes modes – this is all implementation stuff and not directly pertaining to the basic idea of how DCF works. All it tends to do is muddy the water.. At the moment the diagram is totally misleading and an effort to make it more aligned with the real timing, as requested in previous discussions, was not even considered on merit.

**Resolution CID 7085**

REVISED

P1297.39 and 1297.44

aSIFSTime >= aRxPHYDelay + aMACProcessingDelay (10-2)

aSlotTime >= aCCATime + aMACProcessingDelay + aTxPHYDelay +

aRxTxSwitchTime + aTxRampOnTime + aAirPropagationTime (10-3)

AND

**New Figure 10-19—DCF timing relationships**



*(Aside - Now the diagram shows a backoff situation and the timings make some sense and “agree” with the formula and text)*