DATE: 29 April 2023
NAME: David Law
AFFILIATION: Hewlett Packard Enterprise
E-MAIL: dlaw@hpe.com

REQUESTED REVISION:
STANDARD: IEEE Std 802.3-2022
CLAUSE NUMBER: 99.4.7.1
CLAUSE TITLE: State diagram conventions

PROPOSED REVISION TEXT:
Change the first sentence of the second paragraph of subclause 99.4.7.1 'State diagram conventions' that reads:

'The notation used in the state diagrams follows the conventions of 21.5.'

to read:

'The notation used in the state diagrams follows the conventions of 21.5 with the one exception that functions called within a state have to complete before the exit conditions from the state are evaluated.'.

RATIONALE FOR REVISION:

IEEE 802.3-2022 subclause 99.4.7.1 'State diagram conventions' says that
'The notation used in the state diagrams follows the conventions of 21.5.' Subclause 21.5.1 'Actions inside state blocks' says that
'The actions inside a state block execute instantaneously. Actions inside state blocks are atomic (i.e., uninterruptible).' and 'After performing all the actions listed in a state block one time, the state block then continuously evaluates its exit conditions until one is satisfied, at which point control passes through a transition arrow to the next block.'.

There is, however, a disconnect between this state diagram notation, and the time taken to complete the functions called inside some of the states in Clause 99. Take the example of the START_PREAMBLE state in Figure 99-5 'Transmit Processing state diagram'. On entry to the state, the rTX_DATA function will be called with the parameter pTX_DATA. This will generate eight rPLS_DATA.request primitives based on eight pPLS_DATA.request primitives, in summary, each bit will be sourced from the pMAC and passed to the RS.

While the rTX_DATA function call itself could conceptually be instantaneous, as required for an action in a state by the state diagram convention, the function itself will take a finite time to complete as the rate at which rPLS_DATA.request will be serviced by the RS will be determined by the bit transmit time.

The state diagram convention however states that after executing the action inside a state, the exit conditions are evaluated until one is
satisfied, at which point control passes to that state. In this case the
exit conditions !SFD_DET and $FD_DET$ are evaluated, and since only the
first byte of preamble has been sourced from the pMAC, the exit
conditions !SFD_DET will be true. As a result, the START_PREAMBLE state
will be re-entered instantaneously and the rTX_DATA function will be
called again. This will be before the RS has had a chance to service the
rPLS_DATA.requests from the previous call.

IMPACT ON EXISTING NETWORKS:
None. This is just a clarification of the operation of the state diagram
operation. Existing implementations must wait until functions have been
completed, if they did not they would not operate correctly.