ANNEX K HIGH-FREQUENCY CARRIER-SENSE MULTIPLE-ACCESS PROTOCOL (Optional)

K.1 INTRODUCTION

This Annex specifies a High-Frequency Carrier-Sense Multiple Access (CSMA) with Collision Avoidance protocol [1] for STANAG 5066 in multi-node single-frequency networks.

The HF CSMA protocol introduces no new DPDU types to the STANAG 5066 catalogue. There is no explicit peer-to-peer communication required by the CSMA protocol. CSMA media access control relies on a node’s local information of HF channel activity and inferences of the activity — or lack thereof — of other nodes.

In order to be effective, the options used in this annex and the values for timers need to be set consistently for all nodes on the network.

Annex K is Optional. However if CSMA or any other form of Listen Before Transmit is implemented with STANAG 5066, it **shall** conform to Annex K.

Section K.2 of this Annex presents an overview of the protocol and provides a definition of terms. Details of the protocol, its state diagram, and parameter values are specified in section K.3

K.1.1 Changes in This Edition

The functional differences between this specification and Edition 3 are set out in Section K.5~~K.5K.5~~.

The key change relative to Edition 3 is to add some new timer values, that enable better performance on a network with a small number of nodes.

K.2 OVERVIEW: CARRIER-SENSE MULTIPLE-ACCESS PROTOCOLS

Definitions and management concepts are introduced below prior to detailed specification of the protocol in later sections.

K.2.1 Background

The protocol specified in Edition 3 was based on a presentation to the High-Frequency Industry Association in 2002 by Robert McFarland of Rockwell Collins., “Collision Avoidance using STANAG 5066 in a Network Environment”, with additional input from Michael Stringer of Harris Corporation. This used a Jitter mechanism so that many nodes on a network could reduce the chance of collision. Standardizing this enables multi-vendor deployments.

The update for Edition 4 uses work from Isode specified in the paper “Slotted Option for STANAG 5066 Annex K”. This introduced the following improvements:

1. An optional “slotted” mechanism where each node is assigned a slot. This gives faster switching and higher resilience for small networks.

   a. The “jitter” mechanism remains an option for larger networks, although modern large networks will generally use ALE rather than fixed frequency.

2. Optimized switching when just two nodes are communicating. This removes any residual benefits of operating without CSMA (Annex K) or WTRP (Annex L).
3. Use of different timings when STANAG 5066 EOT is detected to enable faster switching than is possible with a single timer.
4. Optimizing repeat transmissions from a single node, which is important for non-ARQ bulk protocols such as ACP 142.

This combination leads to Annex K providing an alternative to WTRP (Annex L) which will generally provide better performance for a lightly loaded network.

K.2.2 **Definitions**

The following terms are used in the specification of the High-Frequency Carrier-Sense Multiple Access (CSMA) protocol.

K.2.2.1 Stations and Nodes

The terms “*station*” and “*node*” are used interchangeably to describe the communication entities on the shared HF channel.

K.2.2.2 Collisions

Channel collisions — or, simply, *collisions* — are simultaneous or overlapping transmissions by two or more nodes that interfere with each other, preventing reception by another node.

K.2.2.3 Listen-Before-Transmit (LBT)

Listen-before-transmit is the action a node takes to ensure that the channel is unoccupied and free of activity before it attempts to transmit.

K.2.2.4 DCD

DCD is the *Data-Carrier-Detect* signal provided by the communications equipment interface. A node senses that an HF radio carrier is present when DCD is *true* (or — using the nomenclature of some interfaces — asserted). Communications equipment that senses an idle channel will provide set DCD = *false* (noted in this annex as the !DCD signal). DCD does not always give a clear indication of when a transmission ends, due to channel fades. Note that DCD requires an interface from modem to STANAG 5066, which might need a crypto bypass.

In this specification DCD is a variable that may be accessed by the state machine. There are also two events used by the state machine:

- DCD_RAISE event when DCD transitions from false to true.
- DCD_FALL when DCD transitions from true to false

K.2.2.5 ~~VDCD~~EOT

~~*VDCD* is a *Virtual Data-Carrier-Detect* signal, derived from reception of valid DPDUs.~~ Observation of DPDU header's End-of-Transmission (EOT) field (defined in Annex C Section C.3.2.3) **shall** be used to extrapolate into the future and predict the time at which a channel will be idle. The EOT mechanism provides a robust mechanism to determine end of transmission. It is preferred to DCD, and ~~when VDCD is available~~ it enables faster switching.

There are two variables to support EOT:

- EOT is to record the EOT value received.
- EOT_ACTIVE records that an EOT has been received for the current transmission, and enables DCD transitions to be ignored. It is initialized to false.

When an EOT value is received in a D_PDU, the event EOT_RECEIVED is made available to the state machine and the EOT value set to the value from the D_PDU. The variable EOT_ACTIVE is set to true.

K.2.2.6 CSMA Enable/Disable

Formatted: Font: (Default) Arial, 12 pt, Bold, Not Italic

Formatted: Font: Bold

Formatted: Heading 3, Indent: Hanging: 1.95 cm

This specification allows CSMA to be enabled or disabled, with the disabled state represented by the OFFLINE state in the state machine. The operator can generate two events in support of this:

- START. This will enable CSMA.
- STOP. This will disable CSMA.

Formatted: No bullets or numbering

K.2.2.7 Queued Data

Formatted: Font: (Default) Arial, 12 pt, Bold, Not Italic

Formatted: Heading 3

When there is no data to transmit, the state machine sits in SENSE state and waits for data to arrive. QUEUED_DATA is a variable available to the state machine which is set to true when data is queued. There are two events used in the state machine in support of this:

- DATA_QUEUE_EMPTY is raised when the last queued data is transmitted and the queue is emptied.
- DATA_QUEUE_NOT_EMPTY is raised when data is queued in a previously empty queue.

Formatted: Outline numbered + Level: 5 + Numbering Style: Bullet + Aligned at: 0.81 cm + Indent at: 1.43 cm

~~K.2.2.6~~K.2.2.8 Contention Interval

The *contention interval* is the period of time during which nodes attempt to access the shared HF channel.

~~K.2.2.7~~K.2.2.9 Collision Avoidance

Collision Avoidance is a strategy that nodes use during the contention interval to increase the probability that some contending node successfully accesses the channel.

~~K.2.2.8~~K.2.2.10 Node States

A node may be in one of the following states as it executes the CSMA protocol:

- *Offline* state (OFFLINE) — the *offline* state is a state in which a station acts as if it were physically offline, i.e., it can neither transmit nor receive;
- *Sensing* State (SENSE) — the *sensing* state is a state in which a station monitors channel activity (using the DCD or ~~VDCD signals~~EOT), waiting for it to become idle.
- *Listen-before-Transmit-Wait* State (LBT_WAIT) — the *Listen-before-Transmit-Wait* state is a state in which a station waits to determine if another node has transmitted on the idle channel.
- *Contention-Wait* State (CONT_WAIT) — the *contention-wait* state is a state in which the station waits a random time before it may transmit. Nodes wait a random time during this state while continuing to sense channel activity to avoid collisions.
- ~~*Linking*~~ *Transmission* State (LINKING) — the ~~*Linking*~~ *Transmission* state is a state in which a station can transmit data.

~~K.2.2.9~~K.2.2.11 Timers

The following timers control operation of the CSMA protocol:

- LBT~~_~~_WAIT_TIMER – the LBT~~_~~_WAIT_TIMER controls the ~~maximum length of~~ time a node will wait on an idle channel before transitioning to the contention-wait state. Its value depends on whether idle channel is detected by DCD or EOT.~~. This value is determined from a number of parameters.~~
- CONT_WAIT_TIMER – the CONT_WAIT_TIMER controls the length of time a node (i.e., a contending node) waits to access the channel before it starts to transmit.

~~K.2.2.10~~K.2.2.12 Scalar Control Parameters

The following scalar parameters control operation of the CSMA protocol:

Formatted: Font: (Default) Arial, 12 pt

Formatted: List Paragraph, Indent: Hanging: 0.62 cm, Outline numbered + Level: 5 + Numbering Style: Bullet + Aligned at: 0.81 cm + Indent at: 1.43 cm

- ~~LBT_WAIT_TIMER_VALUE – the waiting time set for the LBT_WAIT_TIMER determined from a number of parameters;~~
- LBT_WAIT_TIMER_VALUE_DCD – the waiting time set for the LBT~~–~~_WAIT_TIMER when end of transmission is detected by DCD falling ~~when end of transmission is determined by DCD~~;
- LBT_WAIT_TIMER_VALUE_~~VDCD~~ EOT – the waiting time set for the LBT~~–~~_WAIT_TIMER ~~–GC~~ when end of transmission is determined by ~~VDCD~~EOT. This value is added to the time determined by EOT to ensure safe transition;
- LBT_WAIT_TIMER_VALUE_SELF – the waiting time set for the LBT_WAIT_TIMER when node has finished transmitting. This enables a node to transmit again without another node transmitting.
- CONT_WAIT_TIMER_VALUE – the waiting time set for the CONT_WAIT_TIMER; this is a computed value that depends on other parameters and whether Jitter or Slotted is used.;
- NUM_CONT_SLOTS – the number of slots defined. For slotted, this **shall** be greater than or equal to the number of nodes on the network;
- CONT_SLOT_WIDTH – the duration of each contention slot.
- NODE_SLOT_POSITION: For slotted, a per node configuration in the range 1 to NUM_CONT_SLOTS. Each node **shall** have a different value configured.

K.2.3 Concept of Operations

A likely occurrence in multi-node HF subnetworks is that nodes need to communicate at the same time and may attempt to do so. Uncontrolled attempts to access and transmit on the same channel can lead to channel collisions when these transmissions occur at the same time. Mitigating the effects of collisions requires retransmissions and delays that lower channel throughput, but, when retransmissions remain uncontrolled, can further decrease throughput. Repeated collisions on an uncontrolled channel can degrade the network severely. Carrier-Sense Multiple Access (CSMA) concepts that include collision-avoidance (CA) can provide simple yet effective mechanisms for a (small) number of nodes to share a single- frequency HF subnetwork when they each have traffic to send.

The concept of operations for the CSMA protocol follows.

From an offline, non-operating state a node starts the protocol by entering a carrier-sensing state, where it listens on the channel for a radio-frequency carrier. The mechanisms a node uses for carrier-sense may vary but this Annex assumes that the carrier is sensed using either the Data-Carrier-Detect (DCD) signal available from the communications equipment~~, or from a Virtual Data-Carrier-Detect signal generated by~~

~~tracking the EOT-field of received DPDUs contained in previous transmissions~~.

A node with no data queued for transmission remains in the carrier-sensing state.

A node with queued data to send can sense an Idle channel by loss of DCD or predict end of transmission by looking at EOT information in received D_PDUs. In this situation the node will ~~that senses an idle channel enters the Listen- Before-Transmit state, and~~ sets a timer — the LBT_WAIT_TIMER — that identifies a time when it will be safe to transmit.

A second timer – CONT_WAIT_TIMER – is used to help avoid collisions. This can be set according to a Jitter or Slotted mechanism as described below.

Nodes will always listen for DCD and will not transmit when another node has taken the channel.

~~controls its exit from the state. The timer is restarted whenever the carrier is detected. If the LBT_WAIT_TIMER expires and the node still has queued data, the node enters a contention state. At any time in the Listen-Before-Transmit state, if the node no longer has any queued data, the node returns to the carrier-sensing state. Note that the LBT_WAIT_TIMER timer value specifies the maximum time a node will wait on an idle channel. The node will wait for a longer time — potentially a much longer time — on a busy channel because the LBT_WAIT_TIMER is restarted whenever carrier is sensed.~~

~~Nodes wait before contending for the channel and transmitting because of the time delays associated with the carrier-sense mechanism. A node waits to determine that the channel truly is idle. Detection of a busy channel — i.e., declaration of the DCD or VDCD signals — is not instantaneous. A node's transmissions will have a propagation delay between itself and any receiver. Then, DCD is declared by a receive modem on declaration of valid data and it takes the modem's decoder some number of interleaver frames to make this declaration, introducing additional delay. Additionally, for secure systems that use crypto between the HF modem and the STANAG 5066 system, the modem DCD either needs to use a crypto bypass or to be relayed through the cryptographic equipment, which may introduces additional delay. These delays influence the length of the listen- before-transmit interval that enables nodes to avoid — but not completely prevent — collisions on the channel.~~

~~On completion of the Listen-before-Transmit interval i.e., when the LBT_WAIT_TIMER expires, a node with queued data to transmit enters a contention state, where it uses one either a jitter or a slotted mechanisms set out below.~~

~~For both mechanisms, the contention-state's random waiting interval is quantized. It consists of a number of slots, each of a fixed width. The slot width is specified to insure that all nodes in the network can detect a node's transmission before the beginning of the next slot.~~

~~The slot width **shall** be chosen so that it is greater than the channel-sensing delays — i.e., the sum of the modem-, crypto-, propagation- and other delays.~~

K.2.4 Jitter Mechanism

With the jitter mechanism, a node selects a random interval to wait before it enters the transmit state. A node picks a random time to wait by picking a random slot to start its transmission.

The random waiting period during the contention interval is intended to avoid collisions — the carrier-sense and listen-before-transmit states tend to synchronize the access attempts of multiple nodes and the random waiting time during the contention interval forces, with some probability of success, the attempts to occur at different times. With good choice of slot width, the probability of collision can be reduced to the probability that two or more contending nodes pick the same contention slot.

Quantizing the contention-state's waiting time and constraining nodes to attempt transmissions only at slot boundaries will reduce average wait time by reducing collisions. Unconstrained transmission- attempt times during the contention interval increases the possibility of collisions between contending nodes.

K.2.5 Slotted Mechanism

With the slotted mechanism, there is a slot assigned for each node on the network, and the total number of slots is equal to the number of nodes on the network.

Advantages of the slotted mechanism are:

1. Each node will transmit in its own slot, so there is no risk of nodes picking the same slot.
2. For small networks it is safe to use a small number of slots.
    a. Note that for a two-node network, slots are not needed at all, because of ~~a~~ the mechanism described ~~later~~in Section K.2.6~~K.2.6~~:

Disadvantages of the slotted mechanism are:

1. It requires more coordinated configuration than is needed by the jitter mechanism.
2. It is inefficient for larger networks, where Jitter is preferable.
3. It is not "fair" as priority is given to nodes with earlier slots. This may be an advantage in some deployments.
    a. The unfairness only becomes noticeable under high load. For a highly loaded channel, WTRP (Annex L) is likely to be preferable.

K.2.6 Optimizing for Two Active Nodes

Use of CSMA naturally introduces delays between transmissions in order to avoid collisions, while allowing multiple nodes to share a network. A common scenario on a lightly loaded network, for which Annex K is targeted, is that active communication is restricted to a pair of nodes. Annex K optimizes for this scenario, by allowing a node to transmit at the earliest safe point, when it knows that there is only a pair of nodes communicating. This gives useful performance optimization for this common scenario.

Note that either active node can introduce communication with a third node at any time, by establishing a CAS-1 link. However, this mechanism does block communication between other nodes until the pair of nodes has finished communicating. This is not expected to be a problem on target networks (lightly loaded).

K.3 SPECIFICATION AND PROTOCOL

The CSMA protocol is specified below in the following sections:

- K.3.1 Overall State Diagram , which presents the protocol as a state graph with directed transitions and transition events that may be conditional or unconditional;
- K.3.2 State Specifications, which presents detail of each state and outbound-transition tables that govern the actions that take place when the node is in the state, and the events that trigger transitions to the next state.

K.3.1 Overall State Diagram

Figure K-1 below shows the simplified CSMA state diagram in which only the states and transition events are shown. Each state is specified in detail in the following section.

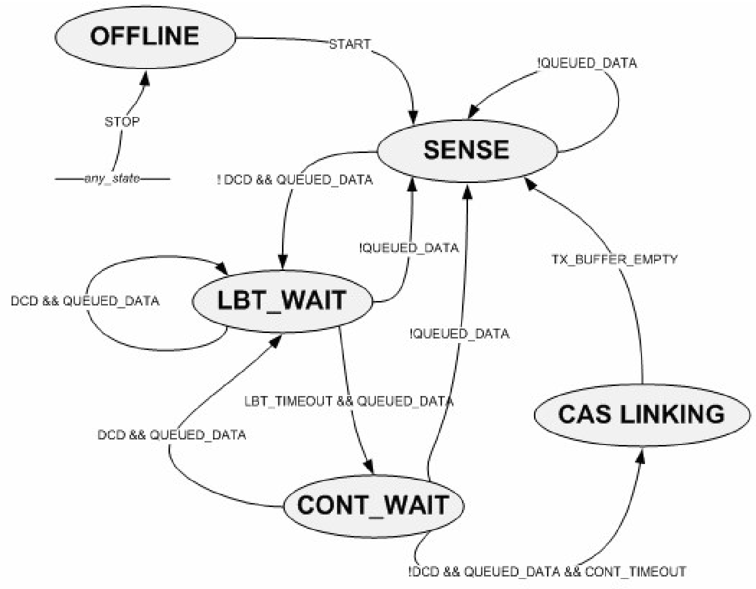

Figure K-1 — CSMA state diagram.

K.3.2 State Specifications

Specifications for CSMA operation for each state are provided below. For specification-purposes, transitions from each state to the next state shall be controlled by the accompanying outbound-transition tables, which specify:

- The current state,
- The event that triggers the transition,
- The action that shall be taken as a result of the transition,
- The next state to which the protocol transitions,
- The timer (if any) that is started.

K.3.2.1 Offline State (OFFLINE)

A node in the OFFLINE state **shall** neither send nor receive data.

Outbound transitions from the OFFLINE state **shall** conform to the table below.

Table 1 — OFFLINE-State Outbound-Transition Table

state	event	condition	action	next state	start timer
OFFLINE	START	start event received from the subnetwork	Neither send nor receive	SENSE	*none*

A node **shall** transition to the SENSE state when it receives a START signal from the subnetwork management function (i.e.., when the nodes starts operating).

K.3.2.2 Carrier-Sensing State (SENSE)

~~A node in the SENSE state **shall** listen for an idle channel, which is declared whenever the DCD signal is false (i.e, a !DCD signal).~~The SENSE state is always used when there is no queued data. When there is queued data in SENSE state, the channel is monitored so that the node can progress to transmission. When an idle channel is detected (DCD), predicted (EOT) or implied (SELF) then LBT_WAIT_TIMER is set and transition to LBT_WAIT.

Outbound transitions from the SENSE state **shall** conform to the table below.

Table 2 — SENSE-State Outbound-Transition Table

state	event	condition	action	next state	start timer
SENSE	State Entry~~!DCD (idle channel)~~	~~data QUEUED for transmission~~QUEUED_DATA AND		LBT_WAIT	LBT_WAIT_TIMER set to LBT_WAIT_TIMER_SELF~~LBT_WAIT_TIMER~~
SENSE	DATA_QUEUE_NOT_EMPTY	NOT DCD~~no data for transmission~~		~~SENSE~~LBT_WAIT	IF LBT_WAIT_TIMER not active then LBT_WAIT_TIMER set to 0~~*none*~~
SENSE	DCD_FALL	NOT EOT_ACTIVE AND QUEUED_DATA		LBT_WAIT	LBT_WAIT_TIMER set to LBT_WAIT_TIMER_DCD
SENSE	DCD_FALL	NOT EOT_ACTIVE AND NOT QUEUED_DATA		SENSE	LBT_WAIT_TIMER set to LBT_WAIT_TIMER_DCD
SENSE	EOT_RECEIVED	QUEUED_DATA		LBT_WAIT	LBT_WAIT_TIMER set to EOT + LBT_WAIT_TIMER_EOT
SENSE	EOT_RECEIVED	NOT QUEUED_DATA		SENSE	LBT_WAIT_TIMER set to EOT + LBT_WAIT_TIMER_EOT
SENSE	STOP			OFFLINE	

A node with no data queued for transmission **shall** remain in the SENSE state regardless of channel activity. The following transitions are noted:

1. On state entry, if there is queued data and no DCD, LBT_WAIT_TIMER set to LBT_WAIT_TIMER_SELF. This is because SENSE will be invariably entered after transmission with DCD false. This transition deals with the scenario where a node transmits twice. When another node transmits, SENSE state will be re-entered with DCD true.
2. If data arrives (DATA_QUEUE_NOT_EMPTY) and DCD is not set, transition to LBT_WAIT. DCD transition to false and EOT has been received while queue

Formatted Table

Formatted: Indent: Left: 0 cm

Formatted: Font: Not Italic

Formatted: Font: 12 pt, Character scale: 100%

is empty will have set LBT_WAIT_TIMER and this may still be running. If LBT_WAIT_TIMER is not running, LBT_WAIT_TIMER is set to 0, which will cause LBT_WAIT to transition immediately to CONT_WAIT.

3. If DCD transitions to false (DCD_FALL event) then LBT_WAIT_TIMER is set to LBT_WAIT_TIMER_DCD. If there is queued data, state transitions to LBT_WAIT.
4. If an EOT is received (EOT_RECEIVED event) then LBT_WAIT_TIMER is set to EOT + LBT_WAIT_TIMER_EOT. The timer is set to wait until the predicted end of transmission plus the fixed LBT_WAIT_TIMER_EOT. If there is queued data, state transitions to LBT_WAIT.
5. If STOP is received, the state transitions to OFFLINE.

Formatted: Numbered + Level: 1 + Numbering Style: 1, 2, 3, … + Start at: 1 + Alignment: Left + Aligned at: 0.63 cm + Indent at: 1.27 cm

~~A node with data to send **shall** wait in the SENSE state until it detects an idle channel, and then **shall** transition to the LBT_WAIT state. The value that LBT_WAIT_TIMER is to set depends on a number of conditions.~~

~~1. LBT_WAIT_TIMER **shall** be set to zero if the node determines that the only active CAS-1 link is between the node and one other node. This is determined if one the following conditions are true:~~

~~a. If the received transmission contains D_PDUs directly addressed to the local node and there are no D_PDUs directly addressed to other nodes. Note that the received transmission **may** contain non-ARQ D_PDUs addressed to broadcast and/or multicast destinations; or~~

~~b. A transmission is received, but the local node is not able to parse any D_PDUs in the transmission, and the local node is known to be a peer in all known CAS-1 transmissions. Under these conditions it is able to transmit safely, as no other node is expected to transmit at this point. In order to determine this, a node needs to monitor the status of all CAS-1 links on the channel (not just the CAS-1 links that the node is involved in).~~

~~The first condition **shall** be checked. The second condition **may** be checked.~~

~~2. LBT_WAIT_TIMER **shall** be set to LBT_WAIT_TIMER_VALUE_DCD if the !DCD event was determined by DCD and not by VDCD.~~

~~3. LBT_WAIT_TIMER **shall** be set to LBT_WATI_TIMER_VALUE_VDCD if !DCD event was determined by VDCD (EOT).~~

K.3.2.3 Listen-Before-Transmit-Wait State (LBT_WAIT)

The LBT_WAIT state is entered with LBT_WAIT timer set. Where EOT has been detected, DCD is predicted to fall before end of timer. When EOT has not been detected, DCD will be down on state entry. Data queues will not be empty on state entry.

~~A node in the LBT_WAIT state **shall** wait while continuing to sense channel activity, to ensure that the channel is truly idle.~~

Outbound transitions from the LBT_WAIT state **shall** conform to the table below.

Table 3 — LBT_WAIT-State Outbound-Transition Table

state	event	condition	action	next state	start timer
LBT_WAIT	LBT_WAIT_TIMER ~~DCD~~ ~~(busy channel)~~	~~data QUEUED for transmitDC~~		~~LBT_WAIT~~SENSE	~~LBT_WAIT_TIMER~~ ~~(restarts timer)~~
LBT_WAIT	LBT_WAIT_TIMER ~~timeout event~~	~~data QUEUED for transmission~~ NOT DCD	compute CONT_WAIT_TIMER_VALUE	CONT_WAIT	CONT_WAIT_TIMER
LBT_WAIT	DCD_RAISE	~~no data for transmission~~		SENSE	~~*none*~~
LBT_WAIT	DATA_QUEUE_EMPTY			SENSE	
LBT_WAIT	EOT_RECEIVE		Cancel LBT_WAIT_TIMER	LBT_WAIT	LBT_WAIT_TIMER set to EOT + LBT_WAIT_TIMER_EOT
LBT_WAIT	STOP			OFFLINE	

Formatted Table

Notes on the state table:

1. A successful transition from this state is driven by expiry of the LBT_WAIT_TIMER. If DCD is true when this timer expires, the link is busy and state reverts to SENSE. Otherwise, CONT_WAIT_TIMER_VALUE is calculated as specified below. This is used to sent CONT_WAIT_TIMER and transition to CONT_WAIT state.
2. If DCD transitions to true (DCD_RAISE event) then another node is using the link and state transitions to SENSE.
3. If the data queue is emptied, for example by operator action (DATA_QUEUE_EMPTY event) then state transitions to SENSE.
4. If an EOT is received (EOT_RECEIVE event) the LBT_WAIT_TIMER is started with value EOT plus LBT_WAIT_TIMER_EOT.

5. If STOP event is received, state transitions to OFFLINE.

When transitioning to CONT_WAIT state, ~~At any time while in the LBT_WAIT state, detection of a busy channel (i.e., receipt of a DCD or VDCD signal) **shall** restart the LBT_WAIT_TIMER with a value of LBT_WAIT_TIMER_VALUE_DCD; the node **shall** remain in the LBT_WAIT state in this case.~~

~~At any time while in the LBT_WAIT state, detection of an empty data queue (e.g., it may have been emptied by a management function) **shall** force a transition to the SENSE state~~CONT_WAIT_TIMER_VALUE is set based on a number of considerations.~~.~~

The optimization for two active nodes described in Section K.2.6~~K.2.6~~ **may** be used. If this procedure is followed and the conditions described in this section for immediate transfer are met, then CONT_WAIT_TIMER_VALUE is set to zero.

Otherwise, either the Jitter or Slotted approach **shall** be used.

If the Jitter approach is taken, CONT_WAIT_TIMER_VALUE shall be set to multiple of CONT_SLOT_WIDTH, multiplied by a random number in the range 0 to NUM_CONT_SLOTS-1.

If the slotted approach is taken, the value depends on which node made the last transmission. When sending ARQ data, it is expected that another node will transmit next. When sending Non-ARQ, this may not be the case.

If the most recent transmission was made by the local node, CONT_WAIT_TIMER shall be set to:

NODE_SLOT_POSITION * CONT_SLOT_WIDTH

If a different node made the most recent transmission, CONT_WAIT_TIMER shall be set to:

~~If the LBT_WAIT_TIMER expires, CONT_WAIT_TIMER is set and calculated in the following manner.~~

1. ~~If the Jitter approach is taken, CONT_WAIT_TIMER **shall** be set to multiple of CONT_SLOT_WIDTH, multiplied by a random number in the range 0 to NUM_CONT_SLOTS-1.~~
2. ~~If the slotted approach is taken and the last transmission was not made by the local node, CONT_WAIT_TIMER shall be set to:~~

(NODE_SLOT_POSITION -1) * CONT_SLOT_WIDTH

3. ~~If the last transmission was made by the local node, CONT_WAIT_TIMER shall be set to:~~

~~NODE_SLOT_POSITION * CONT_SLOT_WIDTH~~

~~The third condition is to address sending Non-ARQ data. When sending ARQ data, it~~

Formatted: Numbered + Level: 1 + Numbering Style: 1, 2, 3, … + Start at: 1 + Alignment: Left + Aligned at: 0.63 cm + Indent at: 1.27 cm

Formatted: Font: Bold

Formatted: Font: Bold

Formatted: English (United Kingdom)

Formatted: No bullets or numbering

Formatted: Indent: First line: 1.27 cm

~~is expected that another node will transmit next. When sending Non-ARQ, this may not be the case.~~

K.3.2.4 Contention-Wait State (CONT_WAIT)

On entry to the CONT_WAIT state, data will be queued, DCD will be false and CONT_WAIT_TIMER will be set.

Outbound transitions from the CONT_WAIT state **shall** conform to the table below.

Table 4 — CONT_WAIT-State Outbound-Transition Table

state	event	condition	action	next state	start timer
CONT_WAIT	~~DCD (busy channel)~~DCD_RAISE	~~data QUEUED for transmission~~		~~LBT_WAIT~~SENSE	~~LBT_WAIT_TIMER (restarts timer)~~
CONT_WAIT	~~!DCD &&~~ CONT_WAIT_TI~~I~~MER	~~data QUEUED for transmission~~		LINKING	~~*none*~~
CONT_WAIT	DATA_QUEUE_EMPTY	~~no data for transmission~~		SENSE	~~*none*~~
CONT_WAIT	STOP			OFFLINE	

Formatted Table

Detection of DCD indicated by DCD_RAISE event (i.e., detection of a busy channel) while in the CONT_WAIT state is taken as an indication that the node has 'lost' the contention round to another node~~, and thus, if the node still has data to transmit, it **shall** restart the LBT_WAIT_TIMER with value of LBT_WAIT_TIMER_DCD and transition to the LBT_WAIT state to wait once again for an idle channel~~. It will transition to SENSE state to try again.

Expiration of the CONT_WAIT_TIMER ~~while the channel remains unoccupied (i.e. without any detection of DCD)~~ is taken as an indication that the node has 'won' the contention round and thus ~~, in this case and if the node still has queued data for transmission,~~ the node **shall** transition to the LINKING state.

~~At any time while in the CONT_WAIT state, detection of~~DATA_QUEUE empty event indicates an empty data queue (e.g., it may have been emptied by a management function) and **shall** force a transition to the SENSE state.

STOP event causes transition to OFFLINE state.

K.3.2.5 ~~CAS Linking~~Transmission State (LINKING)

In transmission state, the node can transmit data.

Outbound transitions from the LINKING state shall conform to the table below.

Table 5 — LINKING-State Outbound-Transition Table

state	event	condition	action	next state	start timer
LINKING	~~(TX_BUFFER_EMPTY)~~ ~~or (TX_TIME >~~END OF TRANSMISSION			SENSE	~~none~~
LINKING	STOP		Complete transmission before going offline	OFFLINE	

Formatted Table

In transmission state a node **shall** transmit some or all of the queued data. The choice of which data to send **shall** follow the procedures specified in STANAG 5066. When data transmission is complete, the nodes **shall** transition to SENSE state.

Formatted: Font: Bold

Formatted: Font: Bold

Formatted: Font: Bold

If STOP event is received, complete transmission and then transition to OFFLINE state.

~~A node in the LINKING state transmit until its transmit-buffer is empty (i.e., TX_BUFFER_EMPTY == TRUE) or until the time it has transmitted exceeds the maximum transmit interval allowed to it (N.B.: a node's maximum transmit interval can be limited by various considerations, whether it is the 127.5 second limitation imposed by the EOT field size, or the time it takes to transmit the maximum number of unacknowledged DPDUs allowed by the ARQ protocol, or some other lesser time imposed by the subnetwork management function.~~

K.4 SETTING SCALAR CONTROL PARAMETERS

Operation of the CSMA protocol is controlled by the scalar parameters listed in the table below.

Table 6~~7~~ —Default values for CSMA Scalar Parameters

Parameter Name	Default Value	Units	Default-Value Name; Comments
CONT_SLOT_WIDTH	3	seconds	CONT_SLOT_WIDTH; optimization of this value requires that it be a function of the modem preamble duration, data-rate, interleaver duration.
NUM_CONT_SLOTS	16	integer	For a slotted configuration, NUM_CONT_SLOTS should be set to the number of nodes on the network. For Jitter; the value selected is a balance between probability of one and only one node selecting the winning slot and acceptable delay (eg. 3 nodes, 90%probability, 16 slots).

LBT_WAIT_TIMER_VALUE_DCD	30	seconds	This needs to be set to a long value as the measurement may be premature due to fades
LBT_WAIT_TIMER_VALUE_EOT~~VDCD~~	3	seconds	A short time is recommended to allow for potential errors in EOT calculation. This also needs to be long enough to give nodes waiting for this sufficient time to detect a node that responds immediately.
LBT_WAIT_TIMER_VALUE_SELF	3	Seconds	This value will need to be tuned so that repeat transmissions are correctly timed.

K.5 Changes since Edition 34

Edition 4 introduces a number of new timers and scalars, to meet the goals set out in Section K.2.1~~K.2.1K.2.1~~, which extend the capabilities of Edition 3.

Because Annex K requires that all nodes are configured consistently, this does not introduce interoperability issues. A deployment with a mix of Edition 3 and Edition 4 systems will need to be configured with Edition 3 settings only.

Status is changed from Informative to Optional. As Annex K controls timers it impacts interoperability and is not merely Informational. This change also brings it in line with Annex J.

The state machine in Edition 3 was poorly specified and this is corrected in Edition 4.

Formatted Table

Formatted: Font: 10 pt