IEEE 802.16t Direct Peer-to-Peer Communication

January 19th, 2023

# Introduction

This document presents Direct Peer to Peer communication between two DPP terminals, with no Base station infrastructure. DPP link is symmetrical in terms of Air interface protocol. Minimal a priori configuration is needed to establish the link.

# Terminology

## Direct Peer-to-Peer (DPP): Direct link between two terminals with no Base Station infrastructure in between.

## DPP Terminal: Each of the two terminals of the DPP link.

## CSMA/CA: Carrier Sense Multiple Access with Collision Avoidance

## CSMA/CA with RTS, CTS: CSMA/CA with Request to Send and Clear to Send.

# Requirements

## Symmetry means two DPP terminals are interchangeable.

## Support of Half duplex with the same frequency in both directions and distinct frequencies in each direction.

## Channel Access methods-CSMA/CA

## Security

## Priority

# Channel Access Methods

1. The following four schemes are proposed for DPP communication:
   1. Half Duplex with Carrier-Sense Multiple Access with Collision Avoidance (CSMA/CA). The same frequency is used in both directions.
   2. CSMA/CA with Request to Send (RTS), Clear to Send (CTS). The same frequency is used in both directions.
   3. Each of the two schemes above, can also use a distinct frequency to communicate in each direction. In this case the sensing is done against both transmit and receive frequency.
2. Based on the deployment/application scenario two parameters are configured.
   1. Maximum time of the burst.
   2. Maximum continuous time usage of the channel in one direction

The maximum time of the burst shall be used for doing the random back-off for re-transmissions. And maximum continuous time usage parameter shall be used to avoid excessive usage of the channel this considers the transmit and response time also.

# Detailed description of the four communication schemes

## CSMA/CA

In this scheme, the RX carrier is sensed first, and once confirmed it is idle, the transmission is started. In the case the channel is not idle, random backoff is used before carrier sense is tried again. The below figure shows the flow.

Diagram

Description automatically generated

Figure 1 CSMA/CA flowchart

This method has the known problem of hidden nodes hence is preferred in the scenario where bi-directional traffic flows with intervals e.g., voice call, this increases the possibility of hidden nodes sensing the channel is busy avoiding further collisions.

To reduce the probability of collisions, an Acknowledgement (ACK) message can be added as a configurable option. Sending of the ACK makes communication bi-directional and can help the hidden node to sense the channel is busy.

In case ACK has not been received the transmitter must do the random back off and initiate the channel access procedure again before doing the re-transmission.

## CSMA/CA with RTS, CTS

In addition to CSMA/CA, before transmitting the actual data, the DPP node sends a RTS message and waits for a CTS message. This helps resolve the hidden node problem. With the CTS response, automatic MCS selection converges faster due to the bidirectional traffic. With RTS/CTS, larger packets can be sent, RTS CTS can be used to inform other listening nodes about the transaction details to avoid further collisions. Below figure shows the flow:

Diagram

Description automatically generated

Figure 2 CSMA/CA RTS CTS flowchart

## Link Adaptation

1. DPP terminal shall begin the transmission with the Robust FEC, other DPP terminal shall send an unsolicited measurement report to the other DPP terminal during the association phase and whenever there is a significant change in measurements (when the MCS needs to be changed).
2. The report shall indicate the time frame of data for which the measurement was taken.
3. Use the valid report and update the MCS for the next transmission at the same time LA hold timer is started. LA hold timer is used to maintain the same MCS till timer expiry.

For measurement report structure refer 7.4.7. Figure 3 shows the flow of LA procedure.

Chart, diagram

Description automatically generated

Figure 3 LA procedure

# OTA protocol

There are no periodic transmissions, and DPP communication generates on-demand bursty traffic. The transmitter will be transmitting at the fixed configured power and the receiver will do the Automatic Gain Control. Figure 4 shows the OTA burst structure. The waveform used for the transmission is the same as mentioned in 802.16t PHY specification document with k=0, refer to the section “*3.4.2.2 SC-FDMA Baseband Signal Generation (Single subchannel)”.*

Refer to 802.16t PHY specification document, section “*3.6 Resolution of air interface resource allocation”*

Refer to 802.16t PHY specification document, section “*3.8* *Uplink transmitter” except the Ranging section.*

Waterfall chart

Description automatically generated

Figure 4 OTA burst structure.

## Gain Adjustment Period

The transmitter shall transmit the one slot worth alternate 1’s and 0’s BPSK modulated signal for a receiver to adjust the gain.

## Synchronization

Preamble is used as a synchronization signal carrying Gold sequence of length 63, details are mentioned in 802.16t PHY specification document, refer to the section “*3.7.2 Downlink Preamble Transmission”.*

## Control Message

The control message shall carry the information about the Data content such as the MCS, length of the allocation etc. It may carry the receiver and sender IDs or addresses. CTRL MSG shall be transmitted with the robust FEC. Refer 7.4.4 for message structure.

## PDU

The PDUs will be transmitted over the air in the same manner as it is processed in the 802.16t. Refer to 802.16t PHY specification document, section “*3.8* *Uplink transmitter” except the Ranging section.*

# Higher layer

## DPP Terminal States

### Offline state

1. Each DPP terminal when turned ON will enter the Offline state by default.
2. Configuration happens during this state. Each DPP terminal will have a unique ID (MAC Address).
3. Each DPP terminal will have a priori information about the other DPP terminal part of the configuration.
4. When there are two distinct frequencies available the DPP terminal will compare its own MAC ID with the other DPP terminal and use the highest frequency for TX and lower for RX in case its MAC ID is higher than the other DPP terminal’s MAC ID, otherwise it selects lower frequency for TX and higher frequency for RX.

### Operational state

1. DPP terminals shall switch to an operational state based on any trigger like an external command.
2. During this state the DPP terminal starts to transmit periodic ASSOCIATE Request message indicating its own ID and the partner node ID with which it wants to communicate.
3. On receiving a response from the other DPP terminal, the DPP terminal enters the Association state.

### Association state

1. The DPP terminal in this state shall receive and transmit control messages (non-traffic) and will not transmit any traffic until it reaches Associated state.
2. The following activities shall happen during this state.

#### Identity verification

DPP terminals shall exchange their IDs (MAC address) using ASSOICATE Request/Response/Reject/Ack messages. DPP terminal receiving the ASSOCIATE request message shall match the received MAC ID of the other DPP terminal and send the ASSOCIATE Response if a match is found. If there is no match the node shall send ASSOCIATE Reject message.

Shown below in Figure 5 , DPP terminal 1 is the transmitter, and it is associating with DPP terminal 2 the receiver. Similarly, DPP terminal 2 shall associate with DPP terminal 1 for its transmission.

Timeline

Description automatically generated

Figure : Association message flow

#### Authentication

After identity verification, the nodes shall proceed to the authentication state where each node can independently authenticate the other as described in section 7.5.

### Associated State

1. After completing the activities of the association process state, the DPP terminal enters the Associated state.
2. In this state, DPP terminals shall transmit traffic data following the rules and specifications finalised during association process.

## OTA PDU structure

1. One transmission cycle constitutes to one burst.
2. A burst can have multiple PDUs.
3. There can be two types of PDU. Based on header type the receiver will identify the PDU type.
   1. Control PDU
      1. RTS, CTS,
      2. Security, SF, LA related
   2. Data PDU
      1. Traffic
4. PDU Header shall indicate Header type, encryption indication, PHS indication, PDU length, Data type ,PHS index and HCS.
5. A PDU can carry one or more control or data SDUs or packets. Concatenation is supported. For example, a node is waiting for transmission and packets get accumulated, then the packets can be concatenated to single PDU and transmitted, provided its under tolerable latency.

PDU header structure is described in 7.4.5

## Dynamic Packet Header Suppression

1. A repetitive portion of the data in the packet is suppressed by the sender and restored by the receiver depending on known rules called PHS rules. PHS rules help in reconstructing the packet correctly at the receiving end.
2. PHS rules are specific to service flows. Every service flow can have multiple rules.
3. PHS parameters include PHS depth, PHS field, PHS mask and PHS index.
4. PHS rules shall be dynamically created by any of the nodes and the rules specify the field values that can be suppressed.
5. Nodes shall learn the traffic and will trigger new rule when any repetitive field values in the traffic is observed.
6. PHS suppression will be applied after creation of rule, until then the traffic will go unsuppressed.
7. PHS index will be prefixed to PDU data and the PDU header will indicate when PHS is applied.
8. Each PHS rule will be identified using PHS index (PHSI).

Diagram

Description automatically generated

Figure PHS Creation Flow

Dynamic PHS related messages are described in section 7.4.8

## Message formats

### Association Messages

|  |  |  |
| --- | --- | --- |
| Syntax | Size(bit) | Notes |
| ASSOCIATE Request () { | --- | ---- |
| Message Type | 8 |  |
| Initiator Terminal ID | 48 | MAC address of initiating Terminal |
| Receptor Terminal ID | 48 | MAC address of partner terminal |
| } |  |  |

Table Associate Request

|  |  |  |
| --- | --- | --- |
| Syntax | Size(bit) | Notes |
| ASSOCIATE \_Response () { | --- | ---- |
| Message Type | 8 |  |
| Response | 8 | 0: Reject 1 : Accept |
| } |  |  |

Table Associate Response

### RTS

1. Message includes type, Sender ID, Receiver ID, sequence number and Length of data in bytes.
2. Each node shall maintain a sequence number which will be incremented after every transmission.

|  |  |  |
| --- | --- | --- |
| Syntax | Size(bit) | Notes |
| Ready To Send () { | --- | ---- |
| Message Type | 8 |  |
| Sender ID | 8 |  |
| Receiver ID | 8 |  |
| Sequence number | 8 | Transmission sequence number |
| Length of Data | 32 | Length of data waiting to be transmitted in bytes |
| } |  |  |

Table 3 : RTS message

### CTS

Message includes type, Sender ID, Receiver ID, sequence number, MCS, expected duration of transmission in slots based on MCS, Received CINR and RSSI

|  |  |  |
| --- | --- | --- |
| Syntax | Size(bit) | Notes |
| Clear To Send () { | --- | ---- |
| Message Type | 8 |  |
| Sender ID | 8 |  |
| Receiver ID | 8 |  |
| Sequence number | 8 | Transmission sequence number |
| MCS | 8 |  |
| Repetition | 8 |  |
| Duration | 16 | Transmission duration in slots |
| CINR | 8 | Measured on received data |
| RSSI | 16 | Measured on received data |
| } |  |  |

Table 4 : CTS message

### CTRL MSG

|  |  |  |
| --- | --- | --- |
| Syntax | Size(bit) | Notes |
| Control message () { | --- | ---- |
| Message Type | 8 |  |
| Sender ID | 8 |  |
| Receiver ID | 8 |  |
| MCS | 8 |  |
| Repetition | 8 |  |
| Length of Data | 32 |  |
| CINR | 8 | Measured on received data |
| RSSI | 16 | Measured on received data |
| } |  |  |

Table 5 : CTRL Message

### PDU Header

|  |  |  |
| --- | --- | --- |
| Syntax | Size(bit) | Notes |
| PDU\_header () { | --- | ---- |
| Header Type | 1 | 0 : Control 1:Data |
| Encryption indication | 1 | 0: Off 1:On |
| PHS indication | 1 | 0: Off 1:On |
| Length | 16 | Length in bytes of the PDU including the header and the CRC if |
| Payload data ID or type | 8 |  |
| If (PHS indication == 1) { PHS index } | 8 |  |
| Reserved | 5 |  |
| HCS | 8 | CRC for the above 5 bytes |
| } |  |  |

Table 6 : PDU header

### ACK

|  |  |  |
| --- | --- | --- |
| Syntax | Size(bit) | Notes |
| Acknowledgement message () { | --- | ---- |
| Message Type | 8 |  |
| Acknowledgement number | 8 | Sequence number received for transmision which is ACKed. |
| } |  |  |

Table 7 : ACK message

### Measurement Report

|  |  |  |
| --- | --- | --- |
| Syntax | Size(bit) | Notes |
| Measurement\_report () { | --- | ---- |
| Message Type |  |  |
| Sender ID |  |  |
| Receiver ID |  | MAC address |
| Sequence number | 8 | Measurement report corresponding to the sequence number of the received data burst. |
| CINR | 8 | Averaged CINR measurement report |
| RSSI | 16 | Averaged RSSI measurement report |
| MCS | 8 |  |
| Repetition | 8 |  |
| } |  |  |

Table 8 : Measurement report

### Dynamic PHS messages

|  |  |  |
| --- | --- | --- |
| Syntax | Size(bit) | Notes |
| PHS Request () { | --- | ---- |
| Message Type | 8 |  |
| PHSI | 8 | Identifies the PHS rule |
| PHS size | 8 | Size of the PHS Field |
| PHS Mask | 48 | Bitmask that determines which bytes of the PHSF that needs to be suppressed |
| PHS Field | variable | Field values |
| } |  |  |

Table PHS Request

|  |  |  |
| --- | --- | --- |
| Syntax | Size(bit) | Notes |
| PHS Response () { | --- | ---- |
| Message Type | 8 |  |
| Response | 8 | 0: Reject 1 : Accept |
| } |  |  |

Table PHS Response

|  |  |  |
| --- | --- | --- |
| Syntax | Size(bit) | Notes |
| PHS\_Ack () { | --- | ---- |
| Message Type | 8 |  |
| } |  |  |

Table PHS ACK

## Security related Authentication and Key management

1. Each node in the network will include a unique private / public RSA key pair.
2. Each node will be pre-configured with its peer MAC address and the peer’s public RSA key.
3. Each node will have a X.509 certificate which shall include, at a minimum the following fields:

|  |  |
| --- | --- |
| Field | Content |
| MAC address | The node’s owned MAC address |
| Manufacturer | A string indicating the node’s manufacture (e.g. “Ondas Inc.) |
| Owner | A string indicating the node’s owner (e.g. “railroad A”) |

1. Each node shall authenticate the other node independently.
2. Key management shall employ two MAC message types: PKM-REQ and PKM-RSP, as per section 6.3.2.3.9 in 802.16-2017 and with the format listed in table 6-66 and table 6-67 there. Attributes of the different messages shall be TLV encoded as per these tables.
3. The keying materials (AK, KEK, TEK) shall be directional. The sending node of each direction shall be the generator of the AK from which a KEK and message authentication keys are derived. Pre-PAK generation is per section 7.2.2.2.1 of 802.16-2017. AK derivation is per section 7.2.2.2.3 of 802.16-2017. KEK derivation is per section 7.2.2.2.4 of 802.16-2017 and TEK is per section 7.2.2.2.6 of 802.16-2017.
4. A node (node 1), after identifying the node to communicate with (node 2), will send a PKMv2 RSA-Request to the other node to request RSA authentication. The PKMv2 RSA-Request message shall include the following attributes:

|  |  |
| --- | --- |
| Attribute | Content |
| Random1 | A 64-bit random number generated in the sending node (node 1) |
| Certificate | Contains the sending node’s X.509 certificate |
| Sig | An RSA signature over all the other attributes in the message using node1 private RSA key |

1. Upon receiving a PKMv2 RSA-Request, the receiving node (node 2) will authenticate the sending node (node 1) by using node 1 public RSA key. If authenticated, the receiving node will reply to the sending node with a PKMv2 RSA-Reply message. The PKMv2 RSA-Reply will include the following attributes:

|  |  |
| --- | --- |
| Attribute | Content |
| Random1 | The 64-bit random number received in the PKMv2 RSA-Request message |
| Random2 | A 64-bit random number generated by the sending node |
| Encrypted pre-Pak | RSA-OAEP-Encrypt (PubKey(node1), pre-PAK | node1 MAC Address) |
| Key lifetime | PAK aging timer |
| Key sequence number | PAK sequence number |

1. The message includes a pre-primary authorization key (Encrypted pre-Pak). The pre-Pak shall be encrypted with node 1 public key.
2. The pre-PAK shall be encrypted with node 1 public key. The received Random number is returned from the PKMv2 RSA-Request message, along with a random number supplied by the sending node, thus enabling assurance of key liveness.
3. The primary authorization key will be used to generate keys to encrypt all traffic from node 2 to node 1.
4. Authentication process includes the following:
   1. Verifying RSA Sig.
   2. Comparing the MAC address in the X.509 with the sender MAC address
   3. Validating the Manufacture and the Owner strings in the X.509 certificate
5. Upon receiving a PKMv2 RSA-Request, if the receiving node (node 2) fails to authenticate the sending node (node 1) as described in 4, the receiving node (node 2) will reply to the sending node with a PKMv2 RSA-Reject message. The PKMv2 RSA-Reject will include the following attributes:

|  |  |
| --- | --- |
| Attribute | Content |
| Random1 | The 64-bit random number received in the PKMv2 RSA-Request message |
| Random2 | A 64-bit random number generated by the sending node |
| Error Code | Error code identifying the reason for rejection of the authentication request |
| Display String (optional) | Display string providing reason for rejection of authorization request |
| Sig | An RSA signature over all other attributes in the message |

* 1. The Error-Code and Display-String attributes describe to the requesting node the reason for the RSA-based authorization failure.
  2. The Sig attribute indicates a RSA signature over all the other attributes in this message, and the node 2 private key is used to make a RSA signature

1. A node (node 1) sends a PKMv2 RSA-Acknowledgment message in response to a PKMv2 RSA-Reply message. Only if the value of the Auth Result Code attribute is failure, then the Error-Code and Display-String attributes can be included in this message.

The attributes of the PKMv2 RSA-Acknowledgment message are:

|  |  |
| --- | --- |
| Attribute | Content |
| Random2 | The 64-bit random number received in the PKMv2 RSA-Reply message in Random2 attribute |
| Auth Result Code | Indicating result (success or failure) |
| Error Code | Error code identifying the reason for rejection of the authentication request |
| Display String (optional) | Display string providing reason for rejection of authorization request |
| Sig | An RSA signature over all other attributes in the message |

1. The Sig attribute indicates a RSA signature over all the other attributes in this message, and node 1 private key is used to make a RSA signature.
2. After achieving initial authorization, each node periodically seeks reauthorization with its peer node; reauthorization is initiated for each direction by the receiving node. The directional authorization and reauthorization is managed by the node’s PKMv2 Authorization state machine. A node shall maintain its authorization status with the other node in order to be able to refresh aging TEKs.

Reauthentication period shall be a configurable parameter (typically, 24 hours).

1. TEK state machines manage the refreshing of TEKs as per section 7.2.2.9 of 802.16-2017.