IEEE P802.15  
Wireless Specialty Networks

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| IEEE 802.15.13  Text input for non-beacon-enabled MAC | | | | |
| Date: 2018-10-25 | | | | |
| Author: | | | | |
| Name | Affiliation | Address | Phone | Email |
| Nikola Serafimovski | pureLiFi |  |  | nikola.serafimovski@purelifi.com |
| Chong Han | pureLiFi |  |  | Chong.han@purelifi.com |

Abstract

# This document contains a text proposal for the MAC algorithm for non-beacon-enabled OWPAN.

1. **Overview**
2. **Normative references**
3. **Definitions, acronyms, and abbreviations**
4. **General description**
5. **MAC protocol specification**
   1. **MAC functional description**
      1. **Superframe structure**
      2. **Channel access**
      3. **Non-beacon-enabled OWPAN**

The MAC for non-beacon-enabled OWPAN starts with the transmission of a control frame (RA) from the coordinator, which carries the information required for new devices to initiate an association procedure. Following the control frame, a contention-based random access period starts. During the contention-based random access period, devices compete on the UL channel sending Poll Request frames to the coordinator to enable the coordinator to poll the device in the following polling period. In the contention-free period, the coordinator can poll one or multiple devices within one frame transmission. Devices are effectively polled by the coordinator and, as soon as they recognize the beginning of the polling frame (polling information is contained in the frame header at the beginning of the frame), they are allowed to transmit on UL channels. The flow chart in Figure 1 illustrates the logic for the non-beacon-enabled OWPAN MAC.

A close up of a sign

Description automatically generated

Figure Flow chart of the non-beacon-enabled OWPAN MAC

The polling mechanism enables very low-complexity synchronization of the transmission slots and also enables the coordinator to dynamically adapt and assign the slots for transmissions. The specific polling algorithm and order is outside the scope of this specification and is left as a system designer's choice. The concept is presented in the example in Figure 2.

A screenshot of a cell phone

Description automatically generated

Figure 2 non-beacon-enabled structure example

* + 1. **Contention-based Polling Period (CAPP)**

Figure 3 presents a flow chart for the contention-based random access period process. The coordinator sends a control frame (RA) periodically at the beginning of a polling cycle. The first valid downlink frame (i.e., DATA, MAC Management or MAC Control frames) after the control frame (RA) marks the beginning of the contention period, i.e., the first valid frame after the control frame is interpreted as a poll to all stations that are not connected to transmit their Poll Request. All stations that have already been associated with the coordinator may ignore any polling during the contention period. If no stations have been connected to the coordinator, the first valid frame would be the subsequent control frame. After receiving the control frame, the device can send a Poll Request. The coordinator starts polling a station in the next polling round after receiving the Poll Request.

A close up of a map

Description automatically generated

Figure Flow chart for contention-based random access period

In case several devices want to connect at the same time, a collision may occur. If it is not polled during the next polling round, a device assumes a collision has occurred and proceeds accordingly with a back-off and an attempt to reconnect. Then, the random backoff is introduced. For example, one device retries to establish a connection after 2 control frames (RA), another device retries after 5 control frames (RA). A device transmits a poll request frame immediately after it decodes a valid MAC frame header. Hence, as depicted in Figure 2, it will not wait for the entire frame on the downlink from the coordinator to be transmitted. Upon a successful polling request (the coordinator has successfully received the station's request to be polled and has indicated this by polling the station within the next transmission round), a station exchanges the necessary association and authentication information with the coordinator via association and authentication control frames. Upon successful authentication and association, the device is assigned a short address which will identify the device among all other devices connected to the same coordinator.

* + 1. **Contention-Free Polling Period (CFPP)**

During the contention-free polling period each device is required to be able to respond to a Poll frame received from a coordinator. A device should also be able to request to be polled by an active coordinator. When polled by the coordinator, a device may transmit only one MPDU. The acknowledgments for any data frame can "piggyback" on the transmission of any management or data frame (including Null frames). If a frame is not acknowledged, retransmission of the frame will be rescheduled by the coordinator. If a polled device does not have any data or acknowledgement to transmit, it simply ignores the polling request. The coordinator will stop polling a device provided it does not receive any response from the device for 5 seconds. The concept is given in the flow chart in Figure 4.

A close up of a sign

Description automatically generated

Figure Flow chart for Contention-free polling period

Any frames transmitted from the coordinator to the device are treated as polling frames, except for the control frame (RA). This can be assumed because every packet transmission contains a polled device number, which enables the stations to keep track of the polling order. A newly connected device is made aware of its queue number with the first valid packet it receives from the coordinator, which contains both its queue number and its MAC address. Furthermore, the coordinator will not start another polling request before the current polled device finishes it transmission on uplink. Hence, no collisions in the uplink are possible and each downlink packet transmission can function as a polling frame. In case the response to a poll request is not detected by the coordinator, the coordinator will attempt to poll the next device in the queue. Upon detection of a subsequent downlink packet by the device whose response was not detected, the device will stop transmission immediately in order to avoid any collisions in the uplink as illustrated in Figure 5. This functionality is necessary because the coordinator does not expect to always receive a response to its poll. If a device has no acknowledgment or information to transmit, it will simply ignore its possibility to transmit. However, a device will need to send a response even though it does not have acknowledgment or information to send after a while. The purpose of this is to keep the connection alive so that the coordinator will not consider the link is invalid. If there is no information to be transmitted to a device, which is next in the queue, a Data Null frame is used as a polling frame.

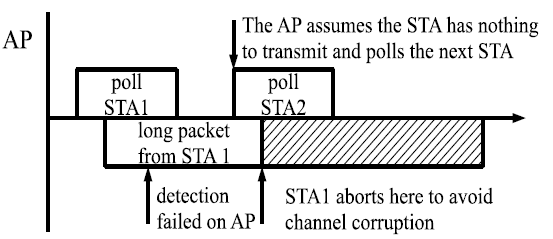


Figure 5 Consequences of the coordinator failing to recognize a poll response

* + 1. **Acknowledgement and retransmission**

The reception of every packet at the MAC layer level has to be acknowledged by the receiving side (coordinator or device). The flow chart in Figure 6 shows the logic of acknowledgement and retransmission processes.

A close up of text on a white background

Description automatically generated

Figure Flow chart for acknowledgement and retransmission processes

If a packet is not acknowledged by the receiver (either the device or the coordinator) at the next polling round (i.e., when the receiver is polled next time), it is deemed that a retransmission is needed for the packet. Packets may arrive out of order. Hence, even if a device or a coordinator does not receive the expected acknowledgment during a given polling round, it can proceed with the transmission of the next packet. If packets are not acknowledged after 4 retransmissions, they are considered lost and are dropped. Figure 7 illustrates the acknowledgment procedure.

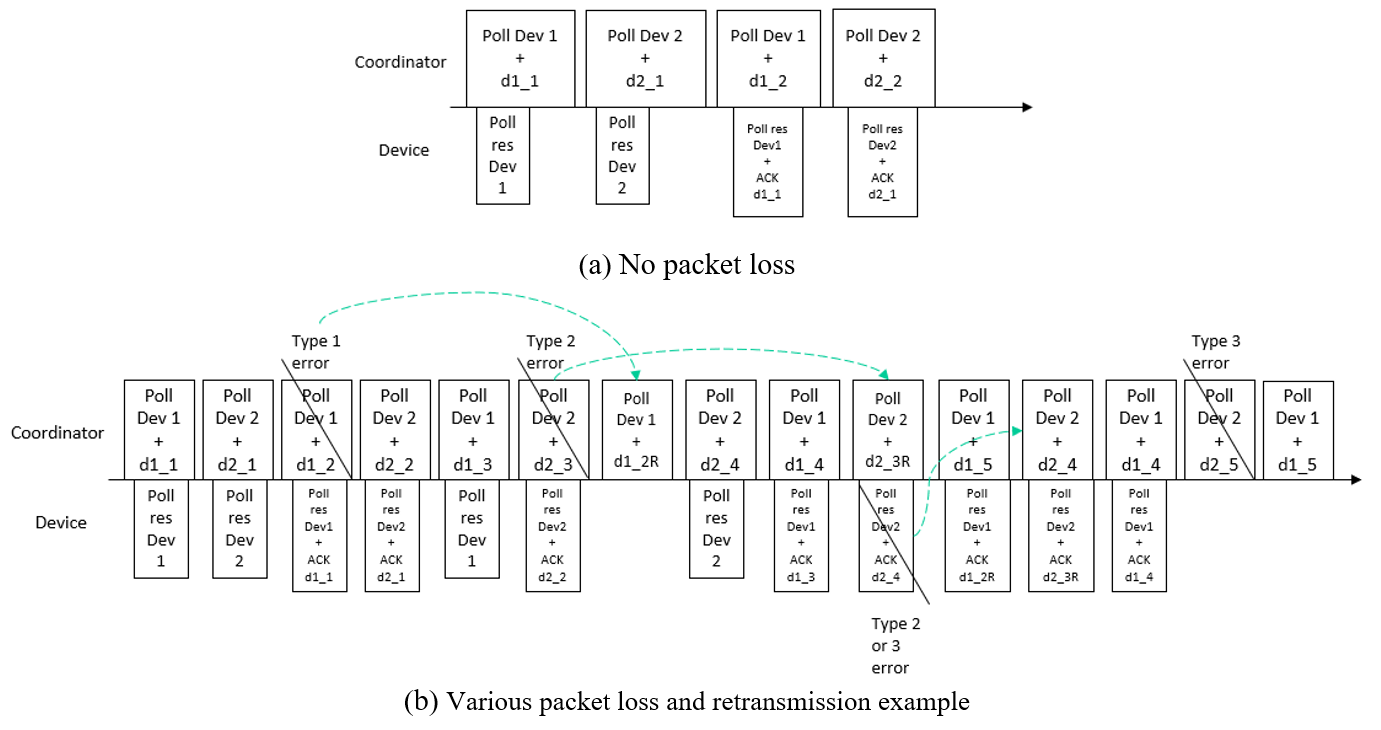


Figure 7 Downlink transmission example scenarios

Figure 7 shows an example for acknowledgment and retransmission. In this example, two devices are connected to the coordinator. The coordinator has a data frame (d1\_1) to transmit to the device 1, hence it polls the device and sends the data. Device 1 responds to the poll with a payload frame or a management frame as soon as it receives the PN sequence of the received frame. Note that if the high-reliability control header at the PHY is enabled, the device can respond as soon as this information is decoded at the PHY layer and the MLME is notified even before the entire DATA portion of the PHY packet is received and decoded. After both the downlink and the uplink packet transmissions are complete, the coordinator sends a data frame to Device 2. Device 2 replies to the poll with a payload frame as soon as the PN sequence is received, since the information of the current polled device was available from the high-reliability control header of the previous Poll. After completion of the transmission in both directions, the coordinator attempts to send a second data frame (d1\_2) to the device 1. Device 1 is polled again, hence it takes the opportunity to send the ACK for the first data frame d1\_1 to the coordinator. Device 2 acknowledges his reception once polled in another round. The Figure 7 (a) shows the above process when no packet loss exists.

Figure 7 (b) demonstrates a more complex example when three types of errors are considered.

1. Payload CRC error:
   * The error does not affect the decoding of the high-reliability control header.
   * The device which is supposed to be polled responds to the coordinator.
   * The device cannot acknowledge the coordinator for this frame due to the failure of decoding the DATA frame.
   * The lack of ACK results in a retransmission from the coordinator.
2. Header error:
   * The high-reliability control header cannot be decoded.
   * The device can still respond to the coordinator although the header is undecodable since each high-reliability header contains the information for the current polled device and the next device to poll.
   * Leads to no response for the next Poll frame as well as no reception of the frame d2\_3.
3. The packet is not detected successfully:
   * It has the lowest probability of occurrence.
   * It also affects the next Poll frame and receives no Poll response from the current polled device.