RW-WLAN-nX MAC PHY Interface

Functional Specification

RW-WLAN-nX-MAC-PHY-IF-FS/4.06

Version 4.06

2019-01-16

Revision History

Version	Date	Revision Description	Author
1.00	2011-06-07	Initial Revision	Jerome Vanthournout
1.01	2012-04-17	Added unit for RSSI in RX Vector Changed *rxReq* behavior Added CCA false alarm case	Jerome Vanthournout
2.00	2012-08-09	Added 802.11ac support Reworked Tx and Rx vector for 802.11ac support	Jerome Vanthournout
2.01	2013-11-10	Updated for bandwidth Signaling (#3128)	Jerome Vanthournout
2.02	2014-01-13	Updated MAC abort description	Jerome Vanthournout
2.03	2015-03-12	Added support of anticipated txData request from the PHY through the MAC-PHY Interface. (#4637)	Jerome Vanthournout
2.04	2015-07-05	Added firstUser field on rxVector1 for MU-MIMO RX support.	Jerome Vanthournout
3.00	2015-07-06	Added MU-MIMO AP support (#5000)	Jerome Vanthournout
3.01	2016-03-22	Added UserPosition Field	Jerome Vanthournout
4.00	2017-06-12	Updated for 11ax support Reworked Tx & Rx vectors	Jerome Vanthournout
4.01	2017-12-05	Added macDataValid flag during TXVector transmission	Jerome Vanthournout
4.02	2018-01-31	Changed RU_ALLOCATION size	Jerome Vanthournout
4.03	2018-02-02	Moved STARTING_STS_NUM field in Txvector to a dedicated byte for HE_TRIG format	Jerome Vanthournout
4.04	2018-02-14	Added Rx Vector 2	Jerome Vanthournout
4.05	2018-11-22	Added RXVector format for HE_TB in STA mode	Jerome Vanthournout
4.06	2019-01-14	Added midamble field in Tx and Rx Vectors (#9546) Added RU_Size field in Rx Vector (#9547) Replaced UMRS by TRS	Jerome Vanthournout

Table of Contents

List of Figures

List of Tables

1 Overview

1.1 Document overview

This document describes the MAC-PHY interface architecture for the RW 802.11n/ac/ax MAC and Modem. The contents of this document are meant for MAC and PHY implementation teams.

1.2 Product overview

The main features of the MAC-PHY interface are:

1. CPU interface for access to PLME registers and PHY debug features based on AHB slave interface.
2. Separate interface for time critical information such as data, vector and control

1.3 List of abbreviations

Abbreviation	Meaning
AHB	AMBA High Performance Bus
CBW	Channel Band Width
CSD	Cyclic Shift Delay
CSI	Channel State Information
LVDS	Low Voltage Differential Signalling
MFMR	MIMO Feedback Matrices Report. In this document it corresponds to 3 types of Feedback Report Frames: CSI , Non-Compressed Beamforming Matrix and Compressed Beamforming Matrix.
MIMO	Multiple Input Multiple Output
NSS	Number of Spatial Steams
NSTS	Number of Space Time Streams
RCV	Reciprocity Correction Vector
RIFS	Reduced Inter-frame Spacing
SM	Steering Matrix
SMM	Spatial Map Matrix

2 Interface architecture

2.1 Introduction

The MAC-PHY interface comprises of two separate interfaces:

1. **Data and Control interface:** This interface is used for time critical data exchanges. It controls the states of the PHY and is used for data transfers between the MAC and the PHY. During a Transmission state, the MAC first passes the Tx Vector via this interface, and then passes the PSDU. During the Receive state, the PHY first passes the Rx Vector and then passes the received PSDU to the MAC. This interface is also used for passing some time-critical MIMO commands to the PHY. The interface works on the interface clock provided by the PHY (*mpIFClk*).
2. **PLME Interface:** This interface is used for non time-critical data exchanges between the CPU and the PHY, such as PLME register access, CSI information exchange, debugging etc. This interface is either an AHB Slave interface.

   The PLME interface on the PHY works on the AHB clock for single chip configuration. The PLME interface clock is asynchronous to internal PHY clocks.

2.2 Data and control interface

This interface defines the data and control signals between the MAC and PHY. The control signals control the various states of the PHY and the two data buses (*txData* and *rxData*) are used to transfer the vectors and PSDU between the MAC and the PHY.

The interface is assumed to be clocked at a clock frequency synchronous to the PHY. The clock frequency may be 120 MHz which achieves 960 Mbps over the interface.

This interface can optionally support MU-MIMO transmission thanks to the additional user interfaces (define in Table 2: Additional multi-user interface signal definition).

2.2.1 LSignal interface definition

Signal Name	Direction	Source clock	Description
mpIFClk		PHY clock	MAC PHY Interface Clock The data is generated and sampled by the PHY on this clock. The MAC-PHY IF block in the MAC synchronizes to this clock.
hardRstMPIFClk_n		PHY clock	Asynchronous Hardware Reset Active low hard reset signal synchronized to the *mpIFClk.* This reset line is used for the FFs in the *mpIFClk* domain.

Signal Name	Direction	Source clock	Description
txReq	MAC to PHY	mpIFClk	Transmit Request This signal is asserted by the MAC at the start of a transmission cycle, when it starts transmitting the TxVector to the PHY. It is de-asserted by the MAC one clock after the *txEnd_p* signal is generated by the PHY, thus ending the transmission cycle. If the *txReq* is de-asserted while the PHY is in a transmit cycle, it serves to abort the current transmission. The MAC waits for *txEnd_p* to be asserted before asserting *txReq* or *rxReq* again.
txData[7:0] / mimoCmd[7:0]	MAC to PHY	mpIFClk	Transmit Data / MIMO Command During transmission: This data bus carries the TxVector and the data to be transmitted. The TxVector is changed every *mpIFClk* clock by the MAC and *phyRdy* is not required. During data transmission, the MAC puts a new data byte on the bus if the *phyRdy* signal is asserted and it has data available to transmit. The MAC indicates that it has put valid data on the *txData* bus by asserting *macDataValid*. During idle (no transmission or reception cycle is ongoing): This data bus carries the MIMO Command. The MIMO Command is valid only if the *mimoCmdValid* signal is asserted. For more details on the MIMO commands refer to section *5.2*, *MIMO commands*.
macDataValid	MAC to PHY	mpIFClk	MAC Data Valid This signal is asserted when the MAC puts a data byte on the *txData*.
mimoCmdValid	MAC to PHY	mpIFClk	MIMO Command Valid This signal is used to indicate that *mimoCmd* bus (multiplexed TxData bus) contains a valid MIMO Command.

Signal Name	Direction	Source clock	Description
phyRdy	PHY to MAC	mpIFClk	PHY Ready During Transmission: This signal is asserted when the PHY is ready to receive data from the MAC on the *txData* lines during a transmission cycle. The MAC puts new data on the bus every *mpIFClk* tick, if *phyRdy* signal is asserted and it has data available to transmit. After the txEnd_p, it is asserted to indicate the validity of TxVector 2 on rxData port. During Reception: This signal is asserted when the PHY puts valid data on the *rxData* lines during a receive cycle. The MAC must sample this data when it detects *phyRdy* asserted. The *rxData* may change every clock. The number of *mpIFClks* the *phyRdy* is asserted is equal to the number of *rxData* bytes available at PHY.
txEnd_p	PHY to MAC	mpIFClk	Transmit End Pulse The modem asserts this signal to indicate end of transmit processing, and helps the MAC synchronize with the PHY and for transmit timing reference. It is the last signal generated by the PHY in a transmit cycle under all circumstances. In case of a normal transmission without error, this signal is generated while the modem outputs the last sample of the last data symbol. In case of an abnormal transmission (transmit abort from MAC, or transmit error detected by PHY), this signal is generated by the PHY when it has completed any transmit processing or internal state cleanup. The PHY detects an error in transmission when: ✓ If it finds an incorrect TxVector parameter ✓ If the *phyRdy* signal has been asserted but the MAC has not passed it adequate number of data bytes, thus leading to an underrun during the transmission.
rxReq	MAC to PHY	mpIFClk	Receive Request MAC asserts this signal to put the PHY in receive state. The MAC de-asserts this signal after the PHY signals *rxEnd*_p, indicating the last byte of the current packet on the *rxData* bus. If the MAC de-asserts *rxReq* while the PHY is actively receiving a PPDU, it is treated as Rx Abort. The MAC waits for the *rxEnd_p* signal from the PHY as an acknowledgement that the PHY has finished processing the abort condition.

Signal Name	Direction	Source clock	Description
rxData[7:0]	PHY to MAC	mpIFClk	Receive Data This data bus carries the RxVector and the PSDU received by the PHY. The PHY indicates that it has put valid data on the rxData bus by asserting *phyRdy*.
CCAPrimary20	PHY to MAC	mpIFClk	Clear Channel Assessment for the primary 20MHz This signal is asserted when the PHY senses that the medium is busy in the primary 20 MHz band. This signal is also asserted when the PHY senses that the medium is busy when operating in a single 20 MHz band.
CCASecondary20	PHY to MAC	mpIFClk	Clear Channel Assessment for the secondary 20MHz This signal is asserted when the PHY senses that the medium is busy in the secondary 20 MHz band. This signal is used in 40MHz, 80MHz, 80+80MHz and 160MHz mode. In 20MHz, this signal is forced low.
CCASecondary40	PHY to MAC	mpIFClk	Clear Channel Assessment for the secondary 40MHz This signal is asserted when the PHY senses that the medium is busy in the secondary 40 MHz band. This signal is used in 80MHz, 80+80MHz and 160MHz mode. In 20MHz and 40MHz, this signal is forced low.
per20MHzBitmap[7:0]	PHY to MAC	mpIFClk	Clear Channel Assessment for the secondary 40MHz Each bit of the per20MHzBitmap that is equal to 0 indicates an idle channel This signal is used in 80MHz, 80+80MHz and 160MHz mode. In 20MHz and 40MHz, this signal is not required.
rxEndForTiming_p	PHY to MAC	mpIFClk	Receive End for Timing Pulse This signal is asserted when the modem has received the last sample of the last data symbol from the RF. It is asserted for one *mpIFClk* period and is used by the MAC for receive timing reference.
rxErr_p	PHY to MAC	mpIFClk	Receive Error Pulse If an error (FormatViolation, UnsupportedRate, CarrierLost) occurs during reception, the PHY indicates this to MAC by asserting the *rxErr*_p signal. The MAC discards any data bytes given by the PHY until *rxEnd_p* is asserted. The MAC de-asserts *rxReq* after the PHY has signalled *rxEnd*_p.

Signal Name	Direction	Source clock	Description
rxEnd_p	PHY to MAC	mpIFClk	Receive End Pulse The PHY asserts this signal to indicate end of receive processing, and helps the MAC synchronize with the PHY. It is the last signal generated by the PHY in a receive cycle under all circumstances. In case of a normal reception without error, this signal is generated while putting the last data byte on the *rxData* bus. In case of an abnormal reception (receive abort from MAC, or error during receive detected by PHY), this signal is generated by the PHY when it has completed any receive processing or internal state cleanup. Note that when the MAC aborts the current reception this signal is asserted after the *rxReq* is de-asserted. Note that when receive error is detected by the PHY, this signal is asserted after *rxErr_p* has been asserted.
keepRFOn	MAC to PHY	mpIFClk	Keep RF On During TX: The MAC asserts this signal to signal to the PHY to not switch off the radio after the current TX cycle. This is to enable the PHY to transmit packets within RIFS. During RX: The MAC asserts this signal to indicate that the PHY should remain in RX state even after reception of the current packet. The PHY starts its search for packet next packet after the current packet is completely received on the air. This is to enable the PHY to receive packets arriving within RIFS.
phyErr_p	PHY to MAC	mpIFClk	PHY Error Pulse The PHY may assert this signal due to various reasons, like incorrect data passed through Tx Vector, transmit underrun detected by the PHY, incorrect command passed through the *mimoCmd* lines. This signal may also be asserted in scenarios where the PHY state machines are hung.
rifsRxDetected	PHY to MAC	mpIFClk	RIFS Reception Detected The PHY asserts this signal when it detects a RIFS separated PPDU from the last received PPDU. This signal is not directly used in the MAC HW, but is used for statistics and debugging purposes.

Table 1: Data and control interface signal definition

Signal Name	Direction	Source clock	Description

Signal Name	Direction	Source clock	Description
txDataUID[7:0]	MAC to PHY	mpIFClk	Transmit Data User ID During transmission: This data bus carries the ID of the targeted user during both TxVector and the data to be transmitted..
rxDataUID[7:0]	PHY to MAC	mpIFClk	Receive Data User ID During reception: This data bus carries the ID of the targeted user during both Rx Vectors and the data received

Table 2: Additional multi-user interface signal definition

2.2.2 Timing diagrams

2.2.2.1 Normal transmission

2.2.2.1.1 SIFS separation between PPDUs

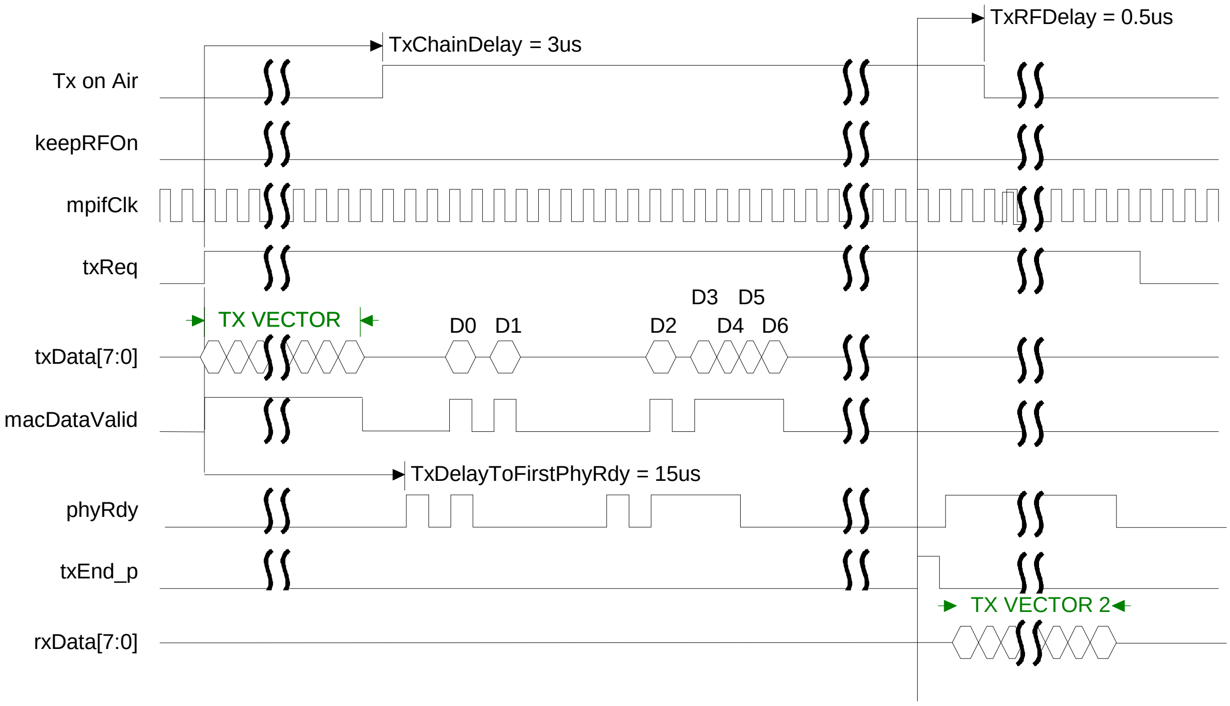

Figure 1 : Timing diagram for normal transmission with SIFS

1. The MAC asserts *txReq* TX-CHAIN-DELAY duration before the PPDU is required to start on air, and sends out the TxVector with the assertion of *macDataValid*. The PHY does not assert *phyRdy* during the TxVector. After receiving the TxVector, the PHY turns on the transmitter and starts the preamble transmission after the PA has ramped up. The TXVector is defined in section *3.1*, *Transmit Vector definition (TxVector)*. Note that the MAC may decide to de-assert the *macDataValid* in the middle of TxVector for some time and then complete the TXVector transmission.
2. Subsequently, data from the MAC is transferred to the PHY based on 2 signals: *macDataValid* and *phyRdy*. The *phyRdy* signal is registered and used in the MAC. If the MAC has data available with it, it transfers it on the *txData* lines, with the assertion of *macDataValid* signal. Then, two differents behavior are supported.
    a. The *phyRdy* signal is used as a strobe, and data is available on the interface earliest after 2 clocks from the assertion of *phyRdy*. The interaction of these two signals is explained in more detail in the *2.2.2.1.1.1*, *Signaling based on PHY strobe*.
    b. The *phyRdy* signal is used as a request, and *txData* and *macDataValid* are provided by the MAC when available. The interaction of these two signals is explained in more detail in the *2.2.2.1.1.2*, *Signaling based on PHY request*.
3. The PHY asserts *txEnd_p* signal to indicate to MAC that the PHY has passed the last sample of the last symbol to the RF. Then the PHY provides the txVector2 on the *rxData* lines with the assertion on *phyRdy* indicating the validity of data. Once the MAC has received the latest bytes of txVector2, it de-asserts the *txReq* signal. The PPDU ends on air after a delay of TX-RF-DELAY.

2.2.2.1.1.1 Signaling based on PHY strobe

The PHY may not implement a FIFO at the interface. In this case, the *phyRdy* assertion has particular relevance to the number of bytes that need to be transferred on the interface. It must be asserted for the number of clocks equal to the number of bytes that need to be transferred between the MAC and the PHY. The MAC guarantees a byte on the *txData* bus 2 clocks after every *phyRdy* assertion. Since the *phyRdy* signal is used as a strobe by the MAC to supply data on the interface without checking its ability to deliver data, the PHY must satisfy two conditions:

- ✓ The first *phyRdy* should not be asserted before TX-DELAY-TO-FIRST-PHY-RDY delay from *txReq*
- ✓ Subsequent *phyRdy* assertions should not cause the MAC data pipeline to underrun

The truth table is as follows, where *phyRdyReg* is a registered version of *phyRdy*:

phyRdyReg	Behaviour at interface
0	Condition 1: PHY does not want data and no data is delivered on *txData* lines.
1	Condition 2: Data is delivered on *txData* lines on the next clock.

Table 3 : Truth table for signalling based on PHY strobe

In the timing diagram below, *popFIFO* is an internal MAC signal indicating that one data should be popped form the FIFO.

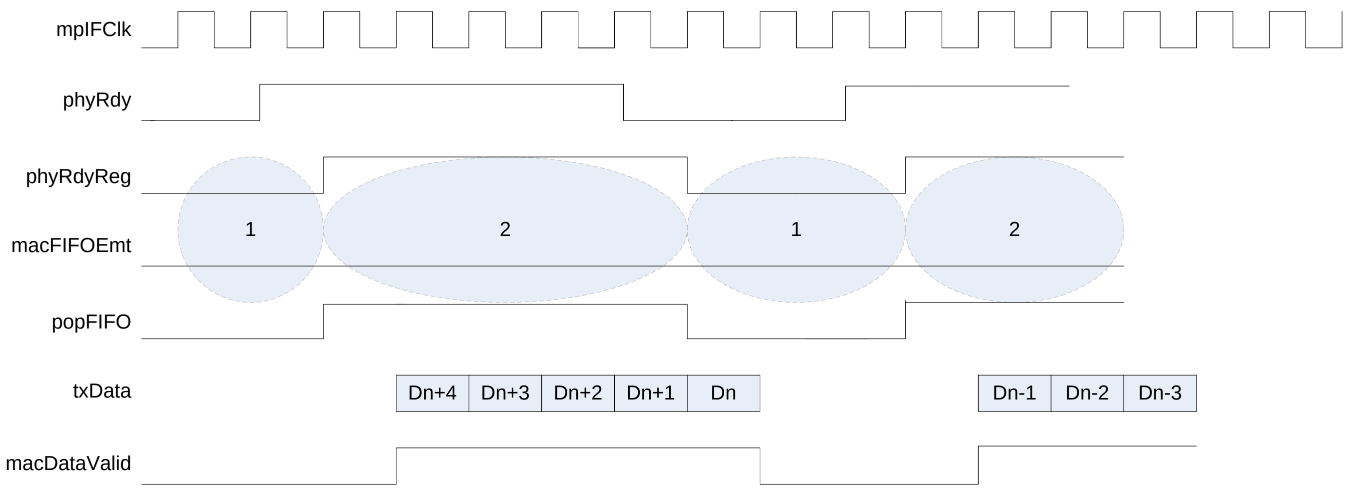

Figure 2 : Signaling based on PHY strobe

If the MAC does not have available *txData* when it receives a *phyRdy*, it is considered as an underrun situation.

In this configuration, the PHY may ignore the state of the *macDataValid* signal as the MAC guarantees the *macDataValid* two clock cycles after every *phyRdy*. However, it may be required for a specific implementation to pipeline the MAC-PHY interface and as a consequence; the additional delays might be introduced between the MAC and the PHY. For this reason, we recommend to always capture the *txData* based on *macDataValid* because the 2 clock cycle might not be guaranty anymore.

2.2.2.1.1.2 Signaling based on PHY request

The PHY may implement a FIFO at the interface. In this case, the *phyRdy* assertion has also particular relevance to the number of bytes that need to be transferred on the interface. It must be asserted for the number of clocks equal to the number of bytes that need to be transferred between the MAC and the PHY.

The main difference between 2.2.2.1.1.1, Signaling based on PHY strobe is that the MAC does not guaranty a *txData* bytes a certain number of clock cycle after every *phyRdy* assertion. This also allows the PHY to assert the *phyRdy* before TX-DELAY-TO-FIRST-PHY-RDY delay. The MAC counts the number of requested byte (number of clock cycles with *phyRdy* high) and delivers the requested data when available in its FIFO.

The consequence of this mechanism is the underrun detection. In this case, the underrun is detected by the PHY as the MAC is not obliged to deliver data 2 clock cycles after the *phyRdy*. If the MAC is too long to deliver the data, the PHY detects an underrun condition and assert the *phyErr_p*.

The PHY shall not have more than 16 pending requests (i.e *phyRdy* high for more than 16 clock cycles without *macDataValid*).

In the timing diagram below, *popFIFO* is an internal MAC signal indicating that one data should be popped form the FIFO.

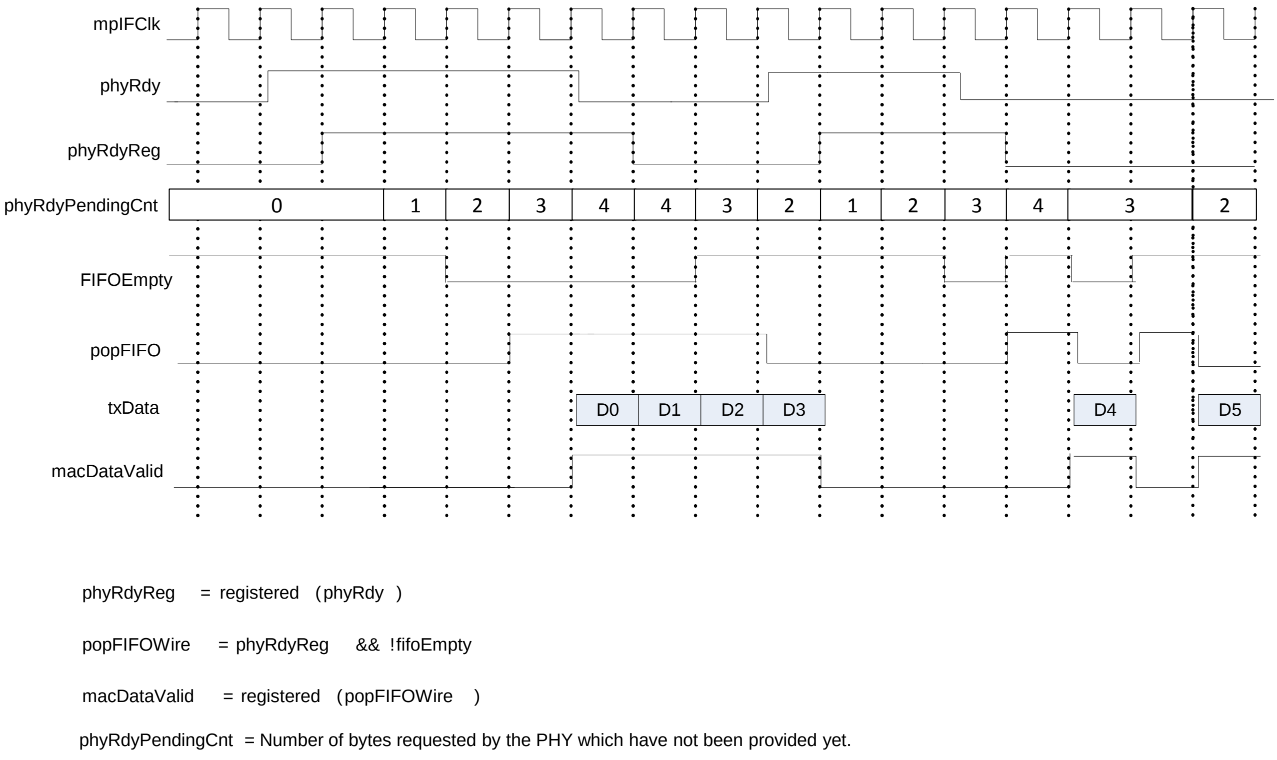

Figure 3 : Signaling based on PHY request

2.2.2.2 Abnormal transmission

2.2.2.2.1 Transmit abort from MAC

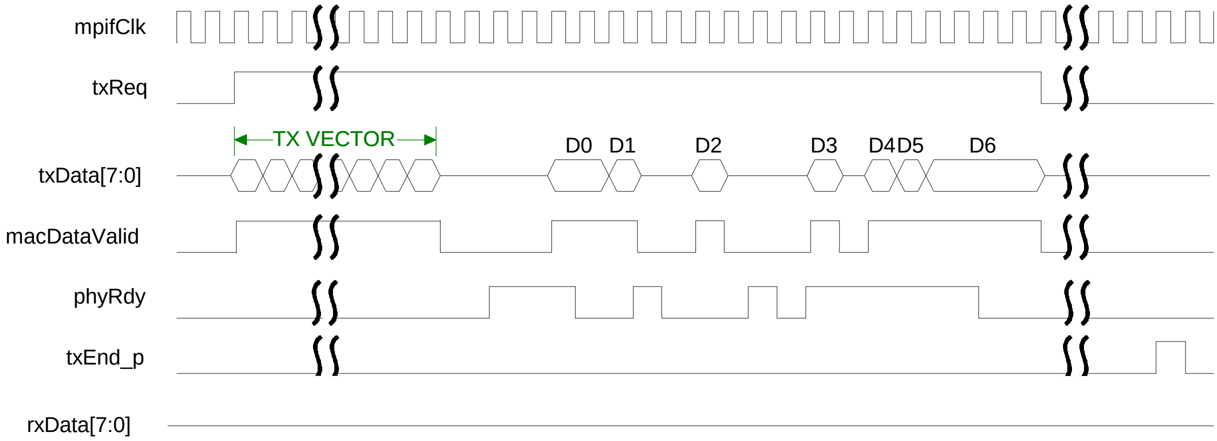

Figure 4 : Timing diagram for transmit abort from MAC

1. The MAC de-asserts the *txReq* before the required number of bytes (as indicated in the Tx Vector 1) have been transferred to the PHY. The PHY may abort the PPDU immediately, at a symbol boundary or it may ignore the error until it has transmitted the required bytes on air. If the PHY continues to sample the *txData* bus after the MAC de-asserts *txReq*, it will get garbage data.
2. The PHY asserts *txEnd_p* to signal the end of the transaction to the MAC when it has completed any transmit processing or internal state cleanup, and is used by the MAC to synchronize with the PHY. In this case, the Tx Vector 2 is not provided by the PHY.
3. The MAC can then assert *txReq* or *rxReq* as required.

Note that if a transmit underrun is detected by the MAC, a transmit abort condition will result on the interface.

2.2.2.2.2 Transmit error detected by PHY - incorrect transmit vector

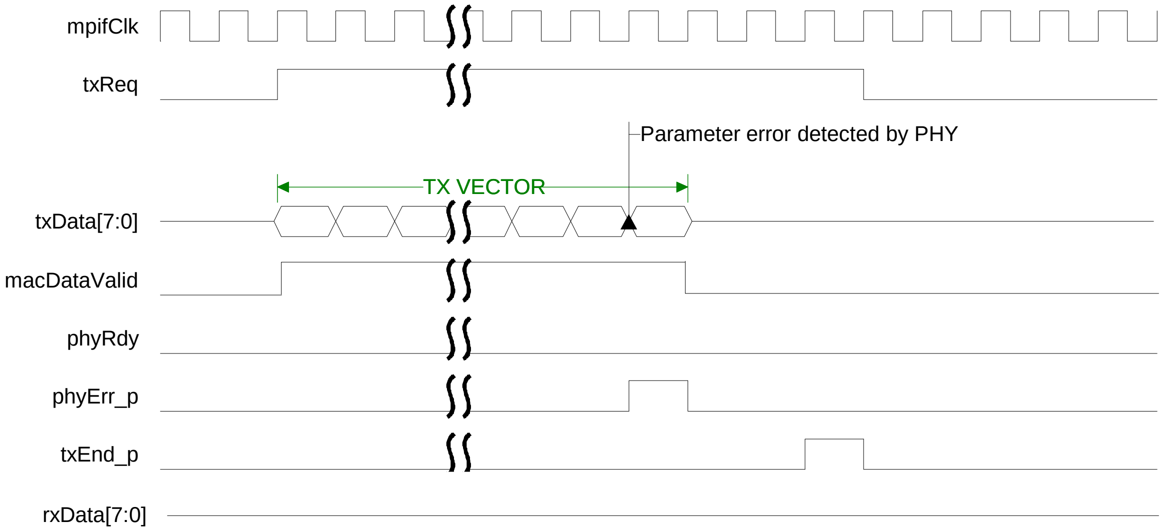

Figure 5 : Timing diagram for incorrect vector detected by PHY during transmission

1. The PHY may detect an error in the Tx Vector parameter from the MAC. It asserts *phyErr_p* to signal the error to the MAC. The MAC stops further transmit processing on detecting this signal and waits for the *txEnd_p* signal. The *txEnd_p* signal must be asserted at least 1 clock after the *phyErr_p*.
2. The PHY asserts *txEnd_p* to signal the end of the transaction to the MAC when it has completed any transmit processing or internal state cleanup, and is used by the MAC to synchronize with the PHY. In this case, the Tx Vector 2 is not provided by the PHY.
3. The MAC de-asserts the *txReq* and can then assert *txReq* or *rxReq* as required.

Note that this error is not expected to occur under normal operation. It is reported to the SW for debug purposes. The MAC HW retransmits the MPDU.

2.2.2.2.3 Transmit error detected by the PHY – under-run

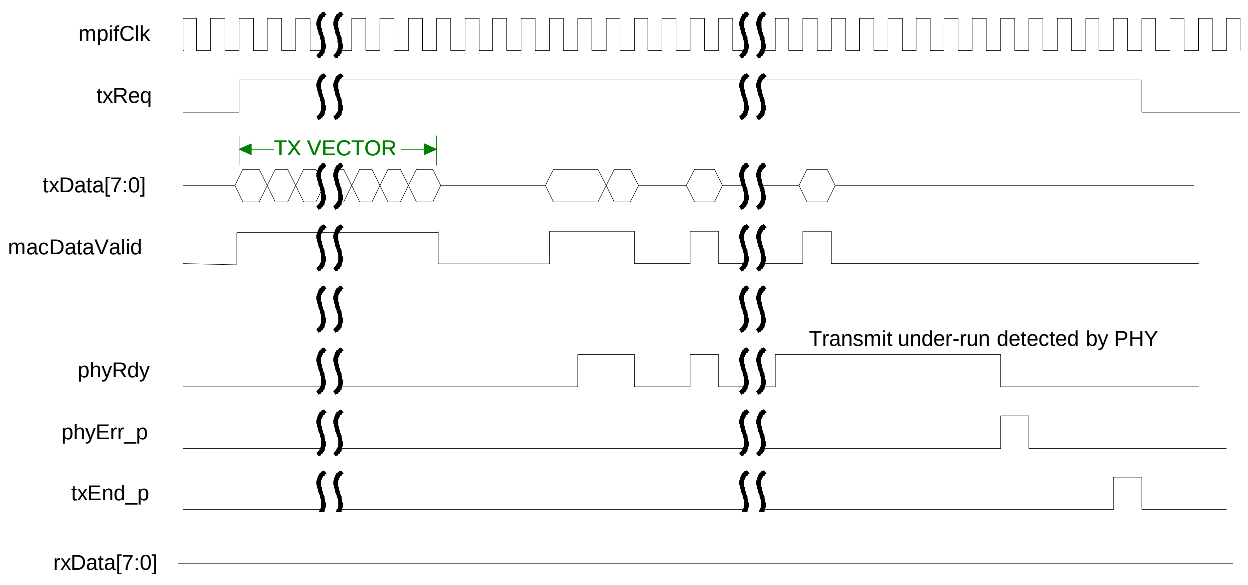

Figure 6 : Timing diagram for under-run detected by PHY during transmission

The PHY requires a certain number of Ndbps i.e. data bits per symbol (every 4 us /3.6 us for Long GI / Short GI respectively), depending on the data rate. If MAC is unable to provide the required number of data bits per symbol, then data under-run occurs in PHY.

1. In the diagram above, the PHY asserts *phyRdy*, and expects data on the *txData* lines. The MAC runs out of data and hence does not assert the *macDataValid*. The PHY is unable to provide required number of data bits for the transmission of the next OFDM symbol. The PHY may abort the PPDU immediately, at a symbol boundary or it may ignore the error until it has transmitted the required bytes on air. If the PHY continues to sample the *txData* bus in this condition, it will get garbage data.
2. The PHY asserts *phyErr_p* to signal the error to the MAC. The MAC stops further transmit processing on detecting this signal and waits for the *txEnd_p* signal.
3. The PHY asserts *txEnd_p* to signal the end of the transaction to the MAC when it has completed any transmit processing or internal state cleanup, and is used by the MAC to synchronize with the PHY. In this case, the Tx Vector 2 is not provided by the PHY.
4. The MAC de-asserts the *txReq* and can then assert *txReq* or *rxReq* as required.

Note that this error is not expected to occur under normal operation and is a system design issue. It is reported to the SW for debug purposes. The MAC HW retransmits the MPDU.

2.2.2.3 Multi User transmission

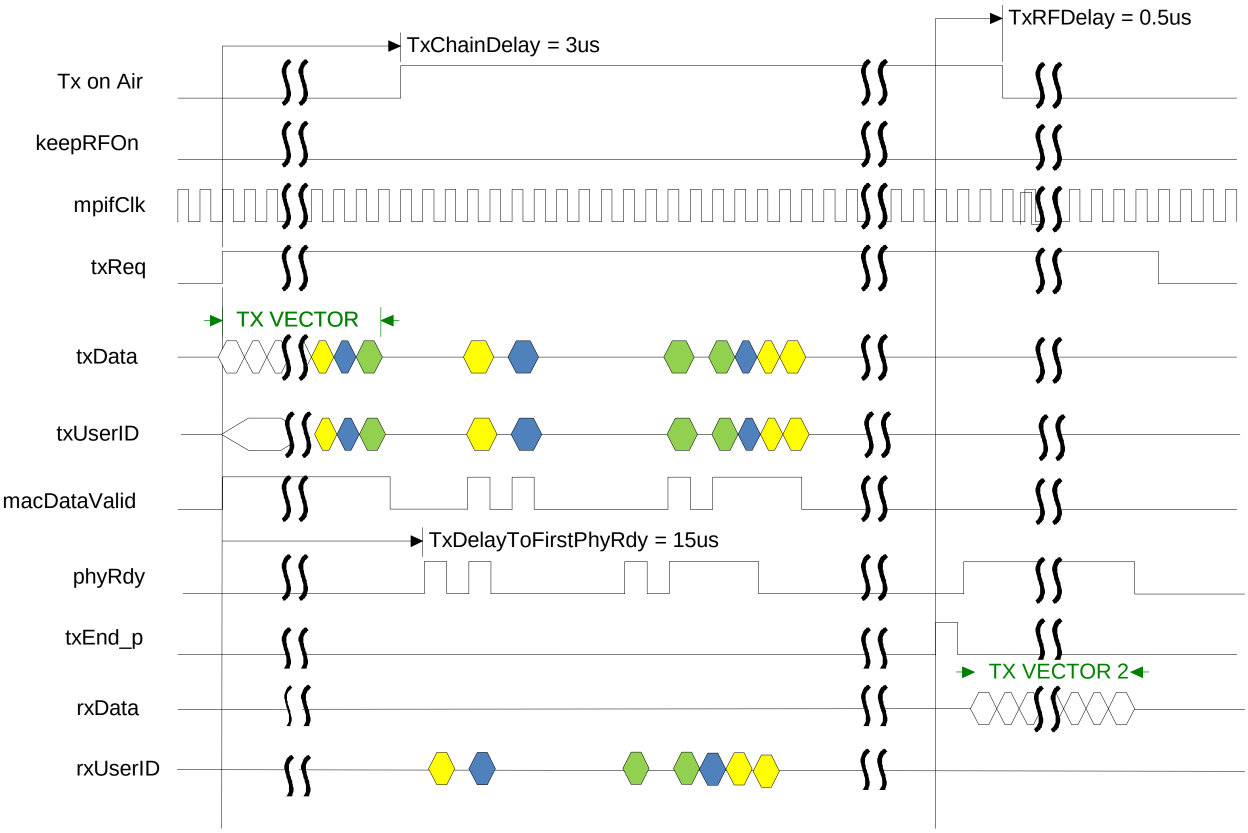

Figure 7 : Timing diagram for Multi User transmission

1. The MAC asserts *txReq* TX-CHAIN-DELAY duration before the PPDU is required to start on air, and sends out the TxVector on *txData* using *macDataValid* to validate each byte of the txVector. In parallel, it also sends out the User Specific part of the TxVectors and indicates the targeted user using *txUserID*. On the common part of the TxVectors, the txUserID is undefined and shall not be considered by the PHY.
2. The PHY does not assert *phyRdy* during the TxVector. After receiving the TxVector, the PHY turns on the transmitter and starts the preamble transmission after the PA has ramped up. The Tx Vector is defined in section *3.1*, *Transmit Vector definition (TxVector)*.
3. As for a standard transmission, data from the MAC is transferred to the PHY based on *macDataValid* and *phyRdy* signals. The *phyRdy* signal is registered and used in the MAC. If the MAC has data available with it, it transfers it on the *txData* lines, with the assertion of *macDataValid* signal. In case of Multi User transmission, alone to the *phyRdy*, the PHY indicates the user for which it needs a new data using rxUserID.
4. When the PHY has got all the data for all the users (defined in the user specific *Length*), it does not assert the user specific *phyRdy* anymore and waits for the completion of the other user transmissions. In such case, the PHY handles the Multi User padding.
5. The PHY asserts *txEnd_p* signal to indicate to MAC that the PHY has passed the last sample of the last symbol to the RF.
6. Following the *txEnd_p*, the PHY provides on *rxData* lines the TxVector2.
7. Once completely received, the MAC de-asserts the *txReq* signal. The PPDU ends on air after a delay of TX-RF-DELAY. The MAC does not assert *keepRFOn* signal since it does not intend to transmit after RIFS.

2.2.2.4 Normal reception

2.2.2.4.1 SIFS separation between PPDUs

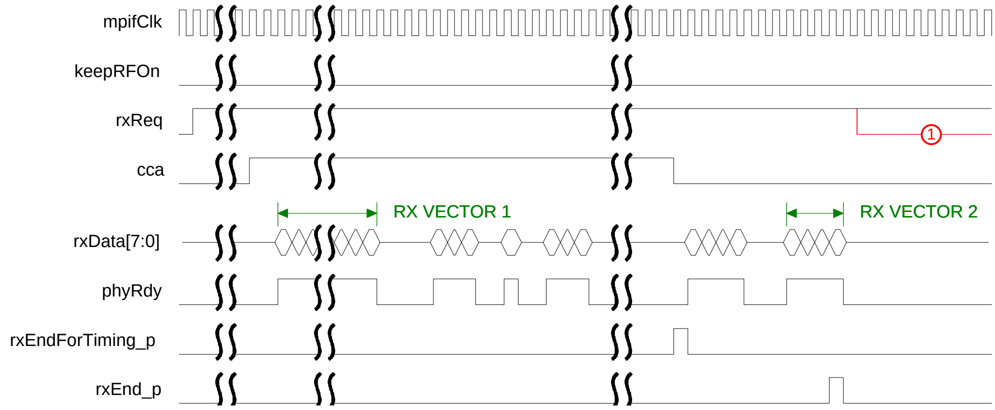

Figure 8 : Timing diagram for normal reception with SIFS

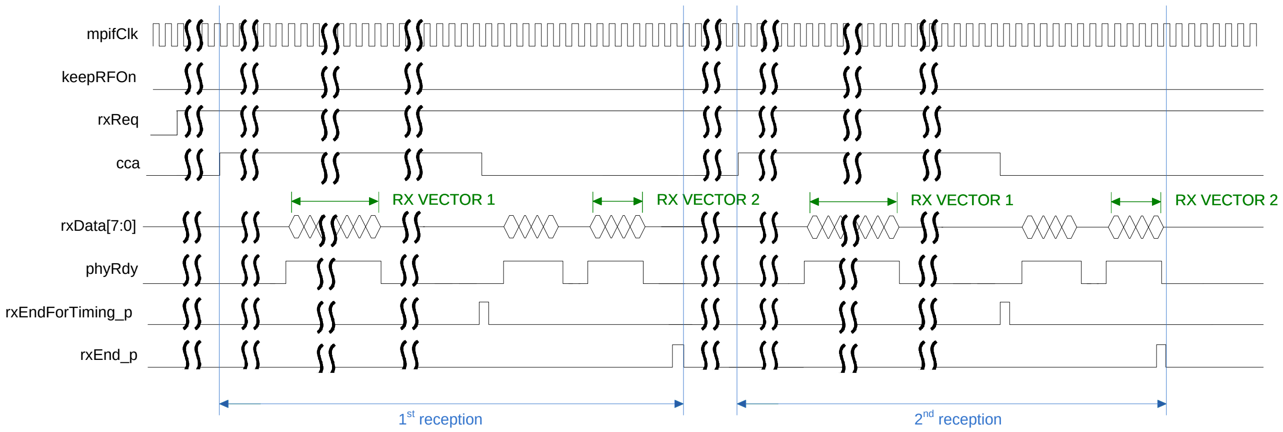

Figure 9 : Timing diagram for two consecutive receptions

1. The MAC puts the PHY in receive state by asserting the *rxReq*. The PHY turns on the RF RX chain and starts the packet detection algorithm after the RX RF has ramped up.
2. Once PHY detects the valid preambles, it asserts CCAPrimary20 and depending of the bandwidth reception, the CCASecondary20 and CCASecondary40.
3. Once the PHY receives all the header symbols (SIGNAL), after decoding them, PHY passes the first part of the Rx Vector to the MAC. The PHY asserts *phyRdy* indicating that the *rxData* contains valid data. The Rx Vector is defined in section *3.2*, *Receive Vector definition (RxVector)*. It is allowed to insert wait-states (forcing *phyRdy* low) during the RxVector. The MAC is informed that a reception is on-going when the first byte of the RxVector is received.
4. The PHY processes the received data and passes the data bytes to the MAC by putting data on the *rxData[7:0]* bus and asserting the *phyRdy*.
5. Once the PHY receives all the symbols on air as per the length decoded from the header, it asserts *rxEndForTiming*_p to provide a timing reference to the MAC.

6. Depending on its CCA logic, the PHY de-asserts the *CCA* signals, indicating that the medium is free.
7. After all the payload bytes have been transferred to the MAC, the PHY passes the second part of the Rx Vector to the MAC.
8. The PHY asserts the *rxEnd_p* along with the last data byte put on *rxData* bus. Then, the MAC leaves the *rxReq* asserted for the next reception. However, the MAC can also decide to *de*-asserts *rxReq* one clock cycle after the *rxEnd_p* if it wants a full reset of the PHY (case highlighted in red). The MAC does not assert *keepRFOn* signal since it does not intend to receive after RIFS.

Note 1: There is no dependency on the *CCA* signals for the MAC during reception. The *CCA* signals are used only by the backoff algorithm to decide if the medium is busy.

Note 2: After the reception is over, if an immediate response needs to be transmitted, the MAC asserts *txReq* even if the CCA indication is asserted. The assertion of *txReq* cancels the CCA indication and state machine in the PHY and immediately puts the PHY into a transmit state. In this case, the *rxReq* is also de-asserted.

2.2.2.4.2 RIFS separation between PPDUs

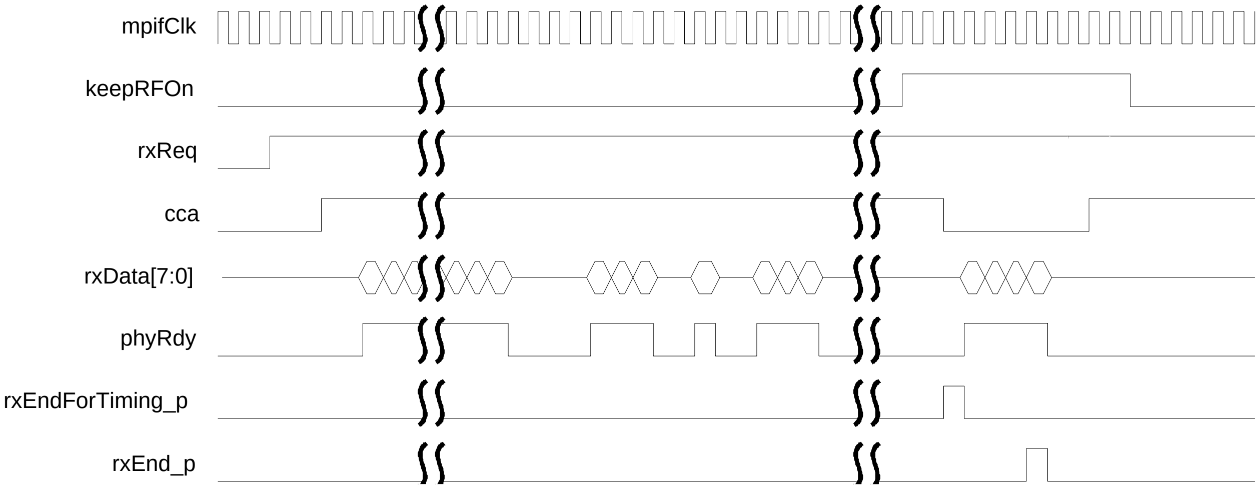

Figure 10 : Timing diagram for normal reception with RIFS

1. During reception if the MAC expects to receive further frames within RIFS, the MAC should indicate this to PHY by asserting the *keepRFOn* signal some time before the end of the first reception on air. Note that the timing of the *keepRFOn* signal from the MAC is undefined; the only condition that needs to be satisfied is that the *keepRFOn* should be asserted at least 1 clock before *rxEndForTiming_p* is asserted by the PHY. When the *keepRFOn* signal is asserted, the PHY does not turn off the RF RX chain, thereby avoiding the RX ramp on delay. The MAC should not assert *keepRFOn* signal if it does not expect to receive any successive frames or when it wants to immediately perform a transmission.
2. The MAC keeps asserted the *rxReq* signal after the assertion of *rxEnd_p* signal.
3. The MAC de-asserts the *keepRFOn* signal once the *rxEnd_p* has been received. Note that the timing of the *keepRFOn* signal from the MAC is undefined; the only condition that needs to be satisfied is that the *keepRFOn* should be de-asserted at least 1 clock after *rxEnd_p* is asserted by the Modem.
4. The PHY re-starts the packet detection and the AGC procedure once the current packet's last symbol has been received on air. While receiving preambles of the next burst frame, the PHY may be processing the last symbol of the previous frame.
5. If the *keepRFOn* signal is asserted, but the MAC does not subsequently assert *rxReq* and a packet is detected, the PHY does not pass this packet to the MAC. The PHY's behavior in this case is undefined w.r.t. the MAC-PHY interface, subject to the above condition.

The *keepRFOn* signal is asserted to indicate to the PHY that a RIFS spaced reception **may** follow. This may not always be true. If the MAC decides that a transmission should follow, the receive RIFS procedure is cancelled by de-asserting *keepRFOn*. Subsequently, the MAC will start a new transmission cycle by asserting *txReq*.

2.2.2.5 Multi User reception

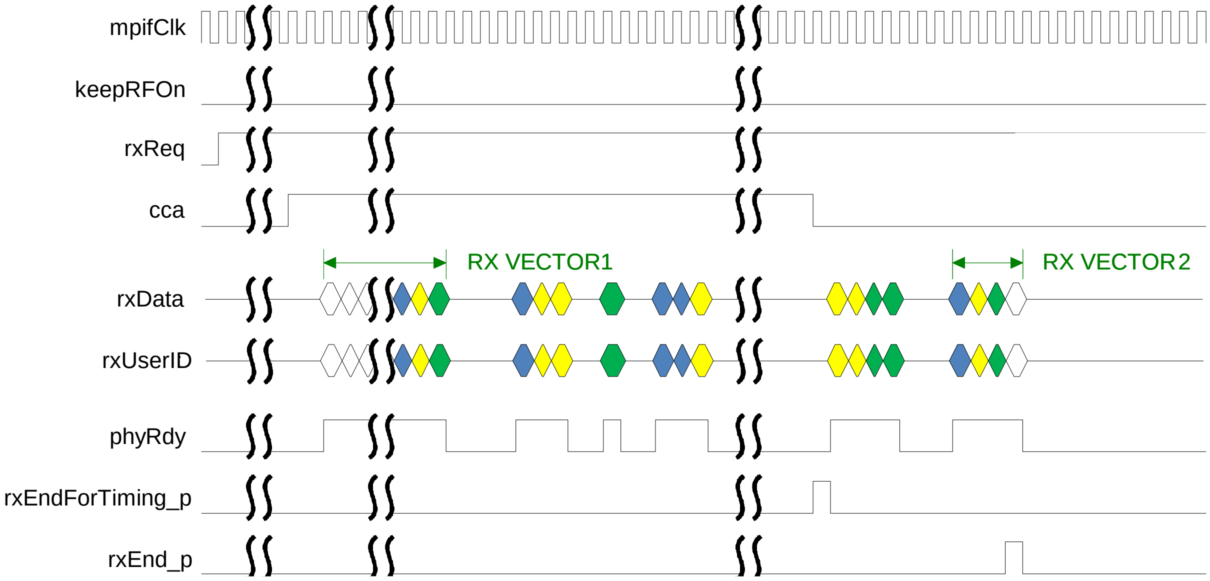

Figure 11 : Timing diagram for Multi User reception

A mutli user reception is similar to a standard reception expecting that the PHY provides along with the rxData the rxUserID. The rxUserID indicates which user the rxData is associated with. On the common part of the RxVectors, the rxUserID is undefined and shall not be considered by the MAC.

1. During a Multi-User reception if the MAC expects to receive further frames within RIFS, the MAC should indicate this to PHY by asserting the *keepRFOn* signal some time before the end of the first reception on air. Note that the timing of the *keepRFOn* signal from the MAC is undefined; the only condition that needs to be satisfied is that the *keepRFOn* should be asserted at least 1 clock before *rxEndForTiming_p* is asserted by the PHY. When the *keepRFOn* signal is asserted, the PHY does not turn off the RF RX chain, thereby avoiding the RX ramp on delay. The MAC should not assert *keepRFOn* signal if it does not expect to receive any successive frames or when it wants to immediately perform a transmission.
2. The MAC keeps asserted the *rxReq* signal after the assertion of *rxEnd_p* signal.
3. The MAC de-asserts the *keepRFOn* signal once the *rxEnd_p* has been received. Note that the timing of the *keepRFOn* signal from the MAC is undefined; the only condition that needs to be satisfied is that the *keepRFOn* should be de-asserted at least 1 clock after *rxEnd_p* is asserted by the Modem.
4. The PHY re-starts the packet detection and the AGC procedure once the current packet's last symbol has been received on air. While receiving preambles of the next burst frame, the PHY may be processing the last symbol of the previous frame.
5. If the *keepRFOn* signal is asserted, but the MAC does not subsequently assert *rxReq* and a packet is detected, the PHY does not pass this packet to the MAC. The PHY's behavior in this case is undefined w.r.t. the MAC-PHY interface, subject to the above condition.

2.2.2.6 Abnormal receptions

2.2.2.6.1 Receive abort from the MAC

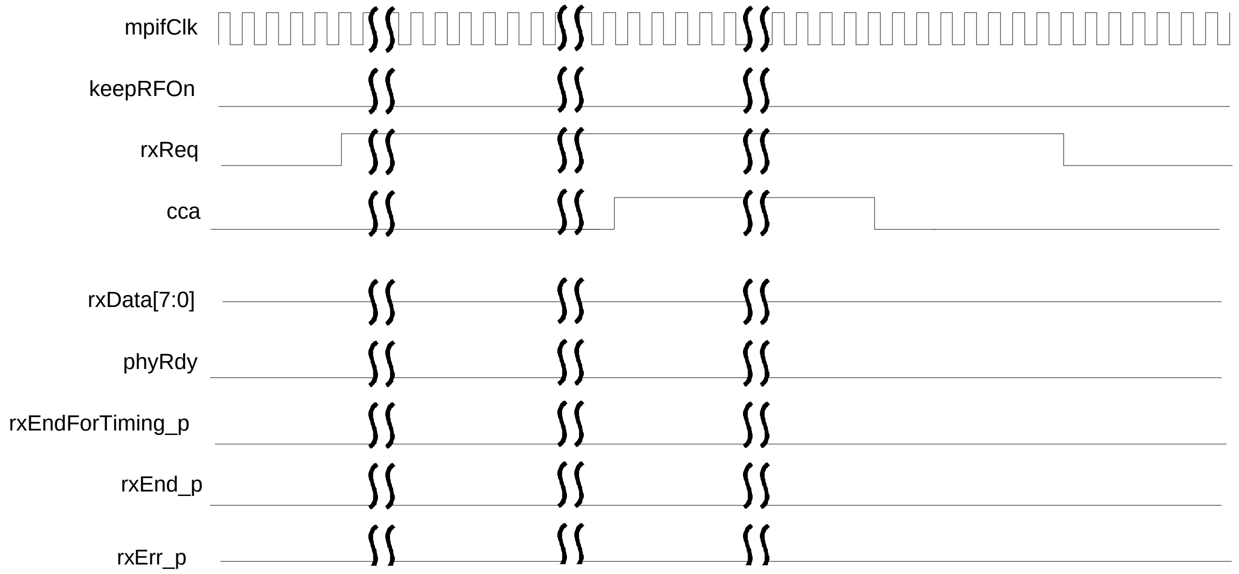

Figure 12 : Timing diagram for receive abort 1 from MAC

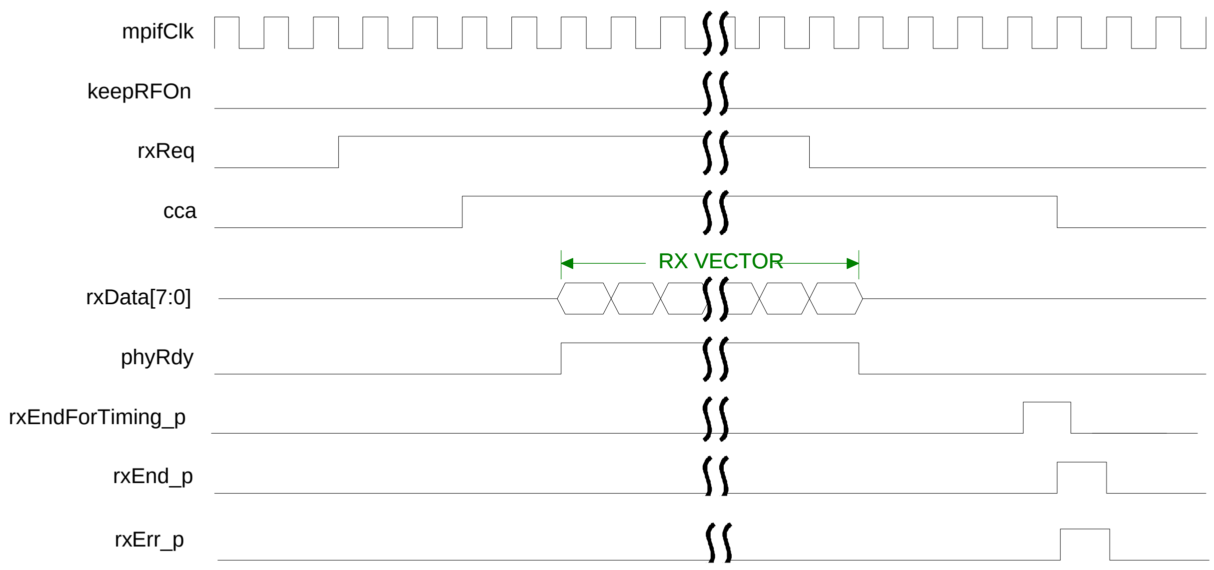

Figure 13 : Timing diagram for receive abort 2 from MAC

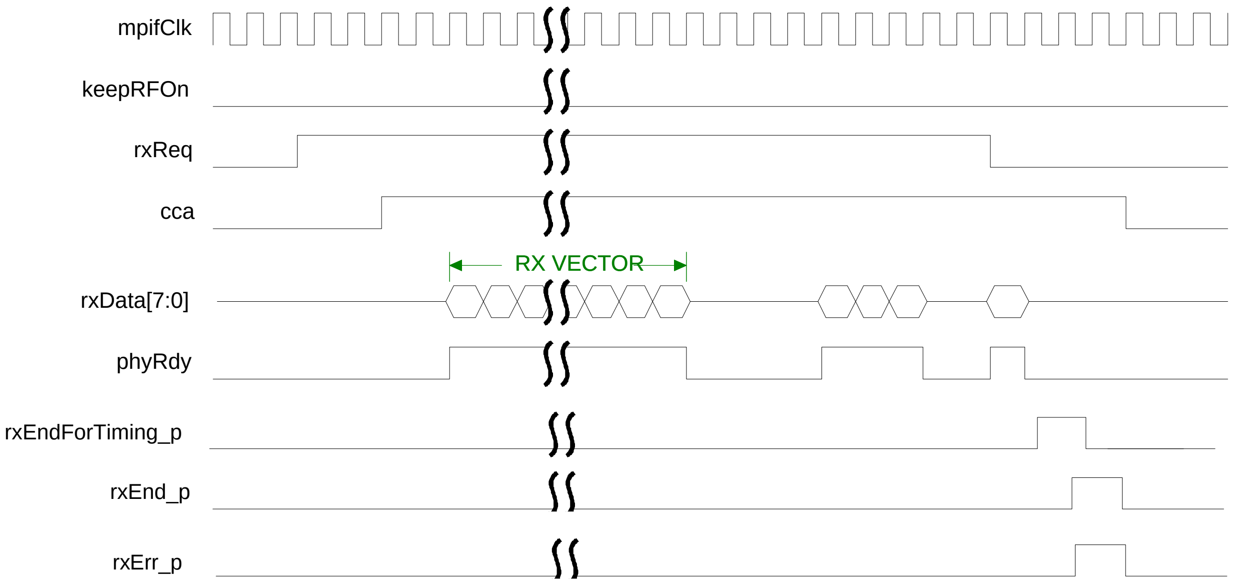

Figure 14 : Timing diagram for receive abort 3 from MAC

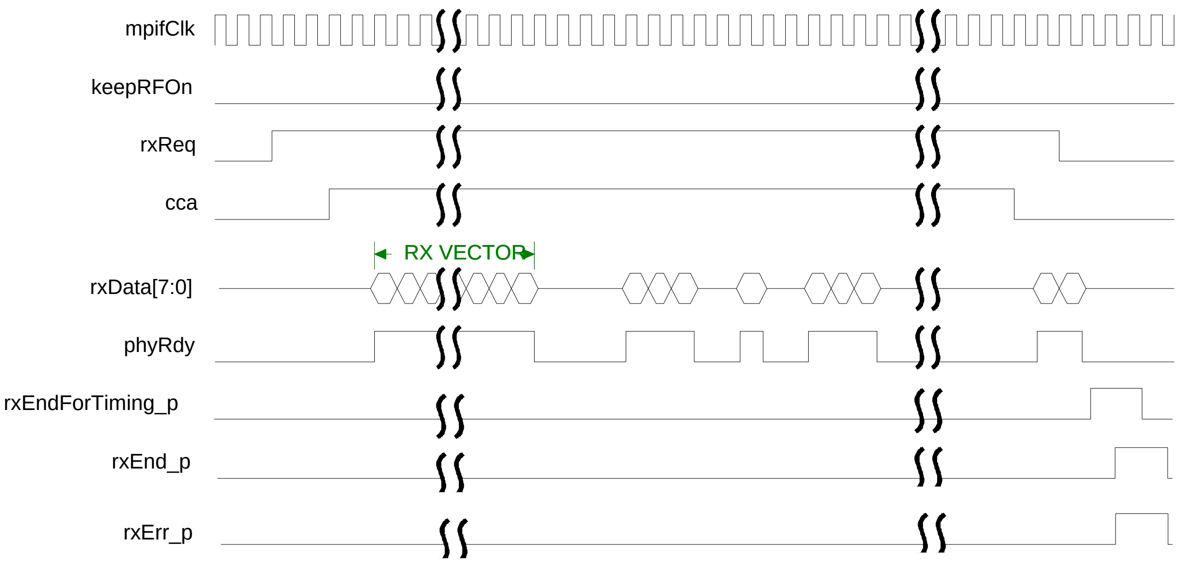

Figure 15 : Timing diagram for receive abort 4 from MAC

1. The MAC de-asserts the *rxReq* before the required numbers of bytes (as indicated in the Rx Vector) have been transferred to the PHY. The PHY may abort the PPDU immediately or at a symbol boundary.
2. The four timing diagrams above show four different conditions for receive abort from MAC.
   a. In *Figure 12 : Timing diagram for receive abort 1 from MAC,* the MAC aborts the reception before the Rx Vector reception.
   b. In *Figure 13 : Timing diagram for receive abort 2 from MAC,* the MAC aborts the reception during or after the Rx Vector reception.
   c. In *Figure 14 : Timing diagram for receive abort 3 from MAC,* the MAC aborts the reception before the PPDU has been completely received on air.

d. In *Figure 15 : Timing diagram for receive abort 4 from MAC*, the MAC aborts the reception after the PPDU has been completely received on air, but before the PPDU has been completely transferred to the MAC.

3. The PHY asserts *rxEnd_p* to signal the end of the transaction to the MAC when it has completed any receive processing or internal state cleanup, and is used by the MAC to synchronize with the PHY.
4. The MAC can then assert *txReq* or *rxReq* as required.

Note 1: The *rxEndForTiming*_p and *rxErr_p* are also asserted from the PHY in this condition.

Note 2: An abort of an active reception from the MAC is not expected under ordinary circumstances. It may occur (for instance) when the MAC makes a decision that the PLCP parameters (e.g. Length) is incorrect and such a packet is surely in error. In can be also generated in case of coexistence.

2.2.2.6.2 Receive error detected by the PHY – Packet detection false alarm

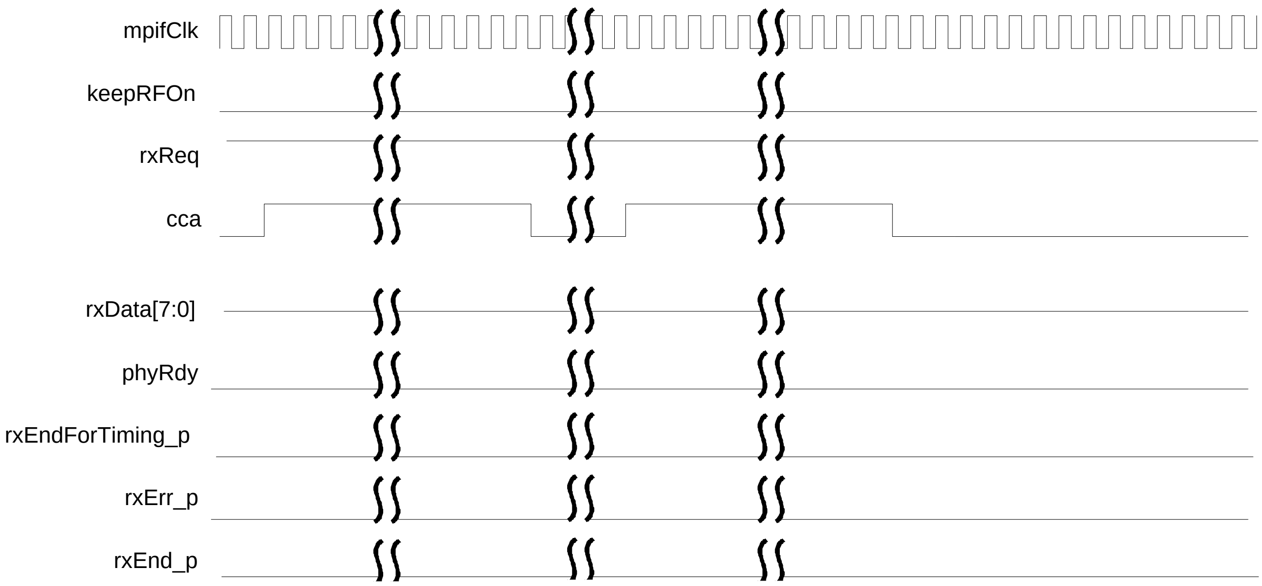

Figure 16 : Timing diagram for packet detection false alarm

1. In case of false alarm, it may happen that the CCA status goes back to IDLE before starting the PLCP reception. In this case, except the CCA indication, no signaling is provided to the MAC by the PHY.

2.2.2.6.3 Receive error detected by the PHY – CarrierLost during PLCP or FormatViolation

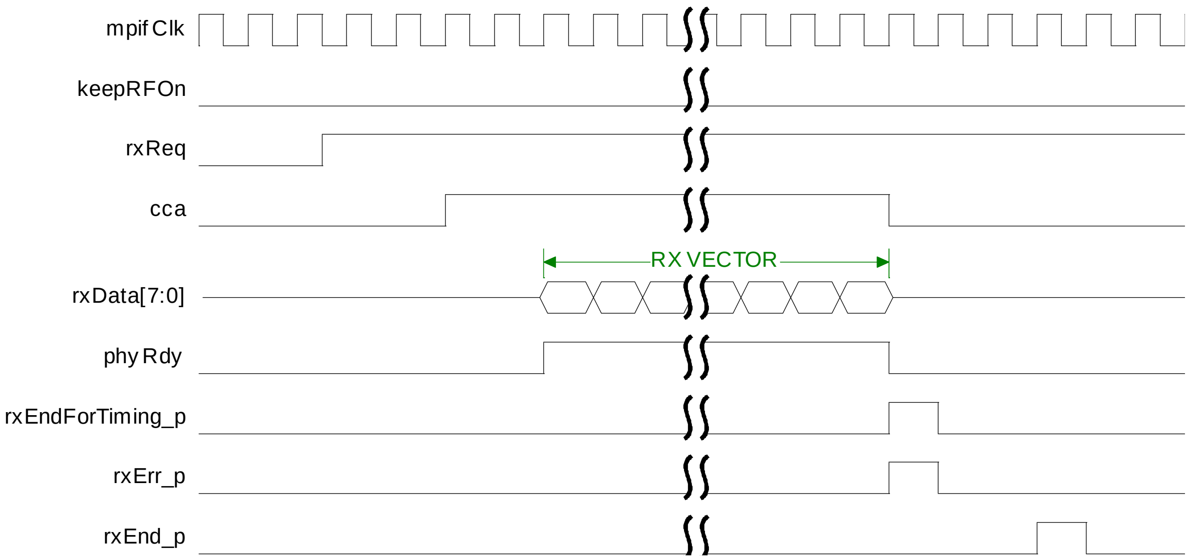

Figure 17 : Timing diagram for CarrierLost during PLCP or FormatViolation detected by PHY during reception

1. During the PLCP header reception, two error conditions can occur:
   a. Change in RSSI causes the CCA status to be IDLE before complete reception of the PLCP. This is CarrierLost error.
   b. The parity check or CRC check of the PLCP fails. This is *FormatViolation* error.
2. The PHY asserts *rxErr_p* signal to indicate to the MAC that PHY has encountered an error during reception.
3. The PHY asserts *rxEndForTiming*_p to provide a timing reference to the MAC. It is understood that this condition may lead to uncertainty in determining slot boundary due to uncertainty in the perceived packet end time.
4. The PHY asserts *rxEnd_p* to signal the end of the transaction to the MAC when it has completed any transmit processing or internal state cleanup, and is used by the MAC to synchronize with the PHY.
5. Note that the MAC does not de-assert the *rxReq* in this case, unless it wants to start a transmission cycle.

2.2.2.6.4 Receive error detected by the PHY – CarrierLost after PLCP or UnsupportedRate

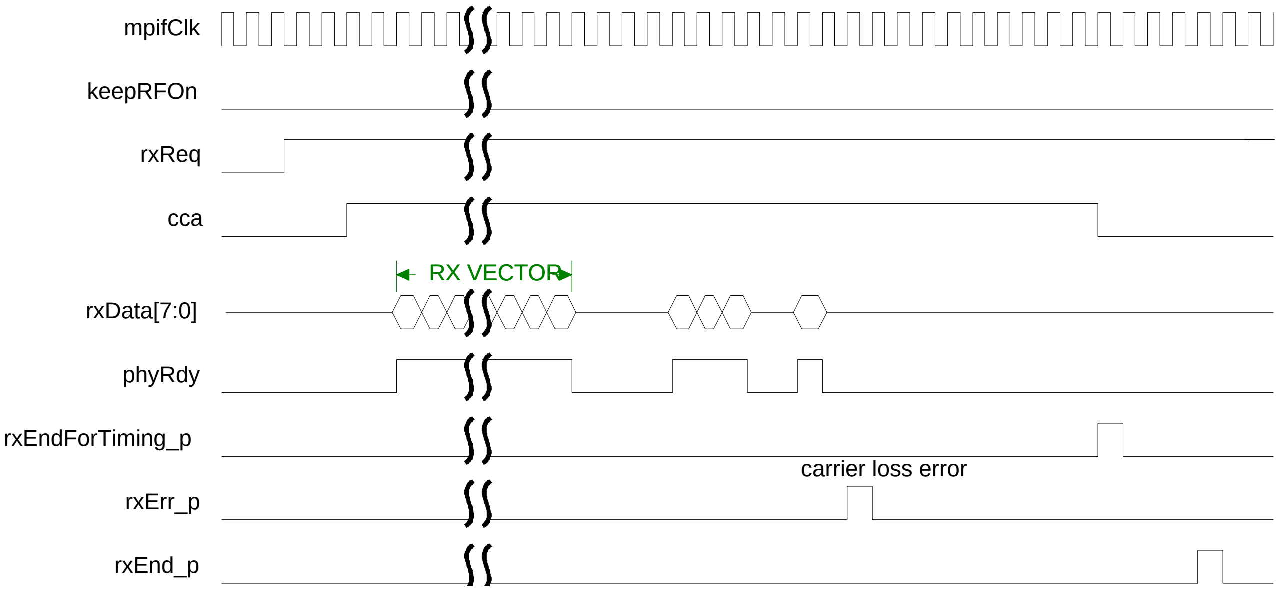

Figure 18 : Timing diagram for CarrierLost after PLCP or UnsupportedRate detected by PHY during reception

1. After the PLCP header reception is successful, two error conditions can occur:
    a. Change in RSSI causes the CCA status to be IDLE before complete reception of the PSDU. This is CarrierLost error.
    b. An indicated rate in the SIGNAL field is not receivable. This is an UnsupportedRate error.
2. The PHY asserts *rxErr_p* signal to indicate to the MAC that PHY has encountered an error during reception.
3. The PHY is subsequently capable of two behaviors:
    a. CCA behavior as defined in the specification, i.e. CCA indicates a busy medium for the intended duration of the PPDU. The *rxEndForTiming_p* is asserted to provide a timing reference to the MAC at the end of the intended reception.
    b. Proprietary behavior, i.e. abort the current frame reception and try to demodulate another frame. The PHY asserts *rxEndForTiming*_p to provide a timing reference to the MAC. It is understood that this condition may lead to uncertainty in determining slot boundary due to uncertainty in the perceived packet end time.
4. The PHY asserts *rxEnd_p* to signal the end of the transaction to the MAC when it has completed any transmit processing or internal state cleanup, and is used by the MAC to synchronize with the PHY.
5. Note that the MAC does not de-assert the *rxReq* in this case, unless it wants to start a transmission cycle.

2.2.3 MAC and PHY latencies

2.2.3.1 Transmit chain latencies

Parameter	DSSS/ CCK	OFDM & MIMO-OFDM	Description
TX-RF-DELAY	0.5µs	0.5µs	Transmit RF delay Delay after modem output to air during transmission. Accounts for the transmit analog/RF delay.
TX-MDM-DELAY	TBDµs	2.5µs	Transmit Modem Delay Delay from the assertion of the *txReq* by MAC to the start of the preamble transmission by modem to analog/RF components. Accounts for the modem transmit data path processing latency.
TX-CHAIN-DELAY	TBDµs	3µs	Transmit Chain Delay Delay from the assertion of the *txReq* to the start of the preamble on air. Accounts for the complete PHY transmit data path processing. It is a sum of the TX-RF-DELAY and the TX-MDM-DELAY.
MAC_PROC-DELAY	2µs	2µs	MAC Processing Delay. This is the delay allowed for the MAC processing.
TX-DELAY-TO-FIRST-PHY-RDY-DSSS/CCK	TBDµs	NA	The minimum delay from the *txReq* to first *phyRdy* from the PHY. This delay indicates the time available for MAC from assertion of *txReq* to keep the first data byte/word ready for a NON-HT-DSSS/CCK transmission with short preamble. This delay takes the following delays into account: The max of (TxRamp On time, preamble processing time), the preamble transmission time, the data path processing time (txPLCPdelay).
TX-DELAY-TO-FIRST-PHY-RDY-OFDM	NA	18.15µs	The minimum delay from the *txReq* to first *phyRdy* from the PHY. This delay indicates the time available for MAC from assertion of *txReq* to keep the first data byte/word ready for a NON-HT-OFDM transmission. This delay takes the following delays into account: The max of (TxRamp On time, preamble processing time), the preamble transmission time, the data path processing time (txPLCPdelay).

Parameter	DSSS/ CCK	OFDM & MIMO-OFDM	Description
TX-DELAY-TO-FIRST-PHY-RDY-MF-1-SS	NA	33.6µs	The minimum delay from the *txReq* to first *phyRdy* from the PHY. This delay indicates the time available for MAC from assertion of *txReq* to keep the first data byte/word ready for a HT-MF with 1 SS transmission. This delay takes the following delays into account: The max of (TxRamp On time, preamble processing time), the preamble transmission time, the data path processing time (txPLCPdelay).
TX-DELAY-TO-FIRST-PHY-RDY-MF-2-SS	NA	37.6µs	The minimum delay from the *txReq* to first *phyRdy* from the PHY. This delay indicates the time available for MAC from assertion of *txReq* to keep the first data byte/word ready for a HT-MF with 2 SS transmission. This delay takes the following delays into account: The max of (TxRamp On time, preamble processing time), the preamble transmission time, the data path processing time (txPLCPdelay).
TX-DELAY-WITH-RF-ON	0.5µs	0.5µs	Transmit Chain Delay with RF On Delay from the assertion of the *txReq* to the start of the preamble on air when the *keepRFOn* signal is asserted before the previous transmission ended i.e. when the TX RF Chain is already ramped up. It is equal to TX-RF-DELAY.
MIN-TXREQ-FALL-TO-RISE	TBD	TBD	Minimum TxReq Fall To Rise Minimum duration between the falling edge of *txReq* and next rising edge. This duration is used when a transmission is immediately followed by another transmission without a reception in between.
MIN-RXREQ-FALL-TO-TXREQ-RISE	TBD	TBD	Minimum RxReq Fall to TxReq Rise Minimum duration between the falling edge of *rxReq* and rising edge of *txReq*. This duration is used when a reception is immediately followed by a transmission.

Table 4: Transmit latencies

2.2.3.2 Receive chain latencies

Parameter	DSSS/ CCK	OFDM & MIMO-OFDM	Description
CCA-MDM-DELAY	*TBD*	3.5µs	CCA Modem Delay Delay of the modem in providing the CCA indication to the MAC. Accounts for the modem CCA procedure.
RX-RF-DELAY	0.5µs	0.5µs	Receive RF delay Delay from air to modem input during reception. Accounts for the receive analog/RF delay.

Parameter	DSSS/ CCK	OFDM & MIMO-OFDM	Description
CCA-DELAY	TBD	4.0µs	CCA Delay Delay of the PHY in providing the CCA indication to the MAC. It is a sum of the RX-RF-DELAY and the CCA-MDM-DELAY.
RX-MDM-DELAY	TBD	9.6µs	Receive Modem Delay Delay from end of the last symbol at the input to the modem to the input to the MAC during reception. Accounts for the modem receive data path processing latency.
RX-CHAIN-DELAY	TBD	10.1µs	Receive Chain Delay Delay from end of the last symbol on the air to the input to the MAC during reception. Accounts for the PHY receive data path processing latency. It is a sum of the RX-RF-DELAY and the RX-MDM-DELAY.
MIN-RXREQ-FALL-TO-RISE	TBD	TBD	Minimum RxReq Fall to Rise Minimum duration between the falling edge of *rxReq* and next rising edge. This duration is to be used when a reception is immediately followed by another reception without a transmission in between.
MIN-TXREQ-FALL-TO-RXREQ-RISE	TBD	TBD	Minimum TxReq Fall to RxReq Rise Minimum duration between the falling edge of *txReq* and rising edge of *rxReq*. This duration is to be used when a transmission is immediately followed by a reception.

Table 5 : Receive chain latencies

2.2.3.3 Slot duration calculations

The IEEE slot duration for an OFDM PHY is 9µs.

Parameter	Value
CCA-DELAY	4.0µs
MAC-PROC-DELAY	2µs
TX-CHAIN-DELAY	3µs
Total	9µs

Table 6 : Slot duration calculations

Thus, the 9µs allowed by the standard *[1]* is satisfied.

2.2.3.4 SIFS duration calculations

The IEEE SIFS duration for an OFDM PHY is 16µs.

Parameter	Value
RX-CHAIN-DELAY	11µs
MAC-PROC-DELAY	2µs
TX-CHAIN-DELAY	3µs
Total	16µs

Table 7 : SIFS duration calculations

Thus, the 16µs allowed by the standard *[1]* is satisfied.

2.3 PLME interface

This PLME interface is used for non-time critical data exchanges such as:

1. PLME Register Programming
2. CSI data collection for Calibration and Explicit Beamforming
3. MCS calculation
4. Programming the Filter Coefficients
5. PHY Debug

This interface is either an AHB Slave interface.

2.3.1 PLME AHB slave interface

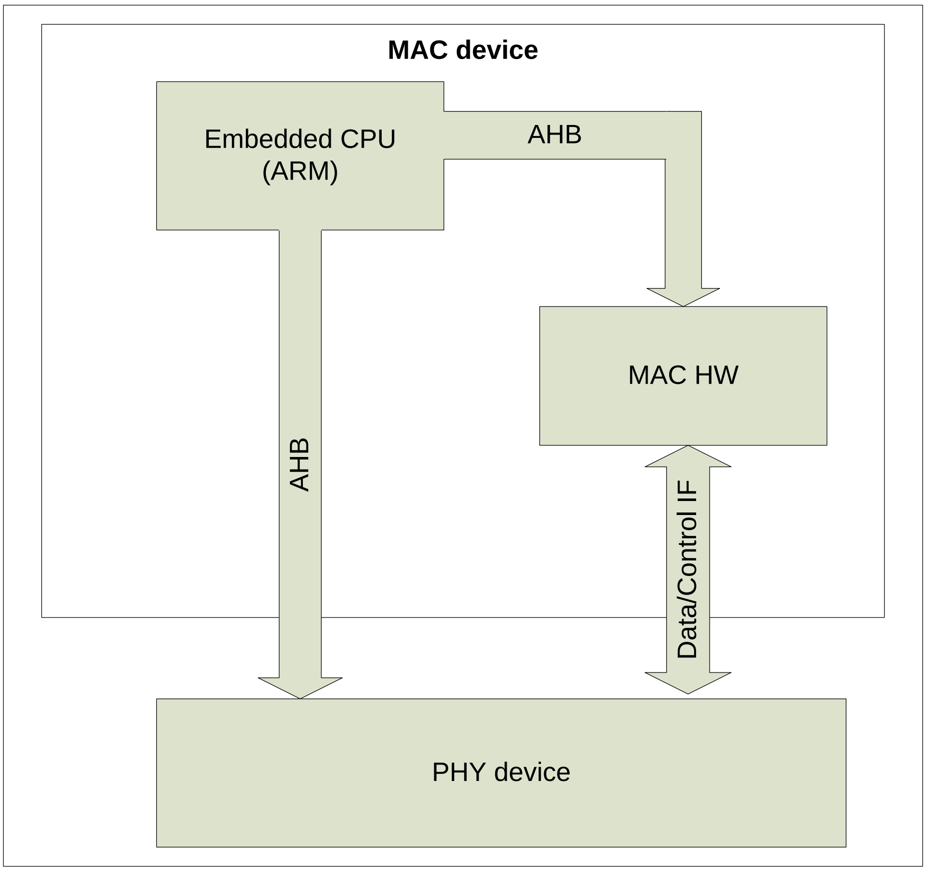

Figure 19 : AHB Slave interface to the Modem in a dual-device environment

The AHB slave supports the following transactions:

1. Read
2. Write
3. Burst Read

The data bus width is 32 bits.

For signal interface definitions and the protocol details refer *[2]*.

3 Transmit and Receive Vector definitions

The Tx Vector definition starts from the LSB. The same is applicable to Rx Vector also.

3.1 Transmit Vector definition (TxVector)

The Tx Vector is transmitted on the *txData* lines to the PHY at the start of the transmit cycle with the assertion of *txReq*. For more details see section *2.2.2.1.1*, *SIFS separation between PPDUs*. Its length and content depend on the frame format and number of users.

At the end of the transmission, the PHY provides to the MAC the TxVector response containing information about the transmission (see *3.1.9 Transmit Vector Response*).

3.1.1 Transmit Vector Common Part for all frames

TxVector	txData	Field	Description
0	txData[3:0]	FORMAT	Format and Modulation This field indicates the format and the modulation of the PPDU, and is encoded as follows: 4'b0000: NON-HT 4'b0001: NON-HT-DUP-OFDM 4'b0010: HT-MM 4'b0011: HT-GF 4'b0100: VHT 4'b0101: HE-SU 4'b0110: HE-MU 4'b0111: HE-EXT-SU 4'b1000: HE-TB

TxVector	txData	Field	Description
0	txData[6:4]	CH_BANDWIDTH	Channel Bandwidth of the transmitted frame This field is encoded as follows: 3'b000: 20MHz 3'b001: 40MHz 3'b010: 80MHz 3'b011: 160MHz or 80+80MHz 3'b100: HE-CBW-PUNC80-PRI for preamble puncturing in 80 MHz, where in the preamble only the secondary 20 MHz is punctured 3'b101: HE-CBW-PUNC80-SEC for preamble puncturing in 80 MHz, where in the preamble only one of the two 20 MHz sub-channels in secondary 40 MHz is punctured 3'b110: HE-CBW-PUNC160-PRI20 for preamble puncturing in 160 MHz or 80+80 MHz, where in the primary 80 MHz of the preamble only the secondary 20 MHz is punctured 3'b111: HE-CBW-PUNC160-SEC40 for preamble puncturing in 160 MHz or 80+80 MHz, where in the primary 80 MHz of the preamble the primary 40 MHz is present In case of HE-EXT_SU, 3'b000 : 242-tone RU 3'b001 : Higher frequency 106-tone RU within the primary 20 MHz Other : Reserved
0	txData[7]	PREAMBLE_TYPE	Preamble Type For *formatMod* = NON-HT and the transmission modulation is DSSS/CCK, this bit is encoded as follows: 1'b0: Short Preamble 1'b1: Long Preamble It is reserved for other *formatMod* types.
1	txData[7:0]	ANTENNA_SET	Antenna Set Indicates which antennas to be used for transmission. Each bit corresponds to one antenna & at a time maximum 4 bits can be set out of 8 bits.

TxVector	txData	Field	Description
2	txData[7:0]	TXPWR_LEVEL	Transmit Power Level The transmit power level for the current PPDU is provided on this field as follow : 2's Complement 8'h80 : -128 dBm 8'hFF : -1 dBm 8'h00 : 0 dBm 8'h01 : 1 dBm 8'h3F : 127dBm
3	txData[2:0]	N_TX	Number of Transmit Chains. The number of transmit chains is indicated by nTx + 1, i.e. nTx = 0 indicates 1 Transmit Chain, nTx = 1 indicates 2 Transmit Chains, etc.
3	txData[6]	TIME_OF_DEPARTURE_REQUESTED	Time of Departure Requested When set, it indicates that the MAC entity requests that the PHY entity measures and reports time of departure parameters corresponding to the time when the first PPDU energy is sent by the transmitting port. When reset, it indicates that the MAC entity requests that the PHY entity neither measures nor reports time of departure parameters.
3	txData[7]	CONTINUOUS_TX	Enable continuous transmit mode. 1'b1: The modem starts a transmission based on the TX Vector information with an infinite frame length. This is only used for RF testing. 1'b0: Normal operation.
4	txData[7:0]	L_LENGTH[7:0]	Legacy Length of the PPDU This field takes a value from 0 to 4095
5	txData[3:0]	L_LENGTH[11:8]	

TxVector	txData	Field	Description
5	txData[7:4]	L_RATE[3:0]	Legacy Rate of the PPDU. 1 Mbps: 4'b0000 2 Mbps: 4'b0001 5.5 Mbps: 4'b0010 11 Mbps: 4'b0011 6 Mbps: 4'b1011 9 Mbps: 4'b1111 12 Mbps: 4'b1010 18 Mbps: 4'b1110 24 Mbps: 4'b1001 36 Mbps: 4'b1101 48 Mbps: 4'b1000 54 Mbps: 4'b1100 When transmitting a HT, VHT or HE PPDU, this field always indicates 6 Mbps.
6	txData[7:0]	SERVICE[7:0]	Service Scrambler initialization, 7 null bits + 9 reserved null bits except in case of CH_BANDWIDTH_IN_NON_HT. For Dynamic Bandwidth management, the first 7 bits are not null and contain the Dynamic Bandwidth selection information) and are used as is by the PHY to initialize the scrambler. However, when the service field will be shifted through the scrambler, the first 7 bits should be masked to 0. If the Dynamic Bandwidth management is not used, the first 7 bits are null. In this case, the PHY uses its internal mechanism for scramble initialization.
7	txData[7:0]	SERVICE[15:8]	

Table 8 : Transmit Vector Common Part for all frames

3.1.2 Transmit Vector Specific Part for NON-HT and NON-HT-DUP frames

TxVector	txData	Field	Description
8	txData[0]	TRIGGER_RESPONDING	
9	txData[7:0]	SMMINDEX	Spatial Map Matrix Index Indicates which spatial map matrix to be used. In case of beamforming, the smmindex indicates the pointers in the index of the beamforming report memory. Refer Section *5.1*, *Spatial map matrix table* for details

Table 9 : Transmit Vector Specific Part for NON-HT and NON-HT-DUP frames

3.1.3 Transmit Vector Specific Part for HT-MM and HT-GF frames

TxVector	txData	Field	Description
8	txData[0]	SOUNDING	Indicates whether this PPDU is Sounding 1'b0: NOT_SOUNDING 1'b1: SOUNDING
8	txData[1]	SMOOTHING	Smoothing recommended 1'b1: Smoothing recommended as part of channel estimation 1'b0: Smoothing not recommended as part of channel estimation
8	txData[2]	GI_TYPE	Guard Interval Type 1'b0: LONG_GI (800 ns) used during the transmission 1'b1: SHORT_GI (400 ns) used during the transmission
8	txData[3]	AGGREGATION	MPDU Aggregate 1'b0: PPDU is not an A-MPDU 1'b1: PPDU is an A-MPDU
8	txData[4]	STBC	Space Time Block Coding It indicates if STBC is used
8	txData[6:5]	NUM_EXT_SS	Number of Extension Spatial Streams Number of extension spatial streams to be transmitted (0 – 3).
9	txData[7:0]	SMMINDEX	Spatial Map Matrix Index Indicates which spatial map matrix to be used. In case of beamforming, the smmindex indicates the pointers in the index of the beamforming report memory. Refer Section *5.1*, *Spatial map matrix table* for details
10	txData[6:0]	MCS	Modulation Coding Scheme This field represents the MCS.
10	txData[7]	FEC_CODING	FEC Coding 1'b0: Use BCC 1'b1: Use LDPC
11	txData[7:0]	LENGTH[7:0]	Length of the HT PPDU This field takes a value from 0 – 65535
12	txData[7:0]	LENGTH[15:8]	

Table 10 : Transmit Vector Specific Part for HT-MM and HT-GF frames

3.1.4 Transmit Vector Specific Part for VHT frames

TxVector	txData	Field	Description

TxVector	txData	Field	Description
8	txData[0]	SOUNDING	Indicates whether this PPDU is Sounding 1'b0: NOT_SOUNDING 1'b1: SOUNDING
8	txData[1]	BEAMFORMED	BeamFormed Indicate that the transmission is beamformed.
8	txData[2]	GI_TYPE	Guard Interval Type 1'b0: LONG_GI (800 ns) used during the transmission 1'b1: SHORT_GI (400 ns) used during the transmission
8	txData[3]	*Reserved*	
8	txData[4]	STBC	Space Time Block Coding It indicates if STBC is used
8	txData[5]	DOZE_NOT_ALLOWED	TXOP PS Not Allowed 1'b0: Doze mode during a TXOP allowed. 1'b1: Doze mode during a TXOP not allowed.
9	txData[7:0]	PARTIAL_AID[7:0]	Partial AID Provides an abbreviated indication of the intended recipient(s) of the PSDU. Integer: range 0-511.
10	txData[0]	PARTIAL_AID[8]	
10	txData[6:1]	GROUP_ID[5:0]	Group ID The PHY uses the *grouID* to select between SU-MIMO or MU-MIMO transmission. If *groupID* is different than 0 or 63 then the transmission will be MU-MIMO. Otherwise, the transmission will be SU-MIMO.
11	txData[2:0]	N_USER	Number of User Indicates the number of users with nonzero space-time streams. 3'b000 : *Reserved* 3'b001 : 1 user 3'b010 : 2 users 3'b011 : 3 users 3'b100 : 4 users *Other are Reserved*
12+5*UserIndex	txData[7:0]	SMMINDEX	Spatial Map Matrix Index Indicates which spatial map matrix to be used. In case of beamforming, the smmindex indicates the pointers in the index of the beamforming report memory. Refer Section *5.1*, *Spatial map matrix table* for details

TxVector	txData	Field	Description
13+5*UserIndex	txData[3:0]	MCS	Modulation Coding Scheme This field represents the MCS.
13+5*UserIndex	txData[6:4]	NSS	Number of Spatial Stream This field indicates the number of Spatial Stream and is coded as follow: 3'b000: 1SS 3'b001: 2SS … 3'b111: 8SS
13+5*UserIndex	txData[7]	FEC_CODING	FEC Coding 1'b0: Use BCC 1'b1: Use LDPC
14+5*UserIndex	txData[7:0]	LENGTH[7:0]	Length of the PPDU This field takes a value from 0 – 1048575
15+5*UserIndex	txData[7:0]	LENGTH[15:8]	
16+5*UserIndex	txData[3:0]	LENGTH[19:16]	
16+5*UserIndex	txData[7:6]	USER_POSITION	User Position In case of MU-MIMO transmission (GROUP_ID other than 0 or 63), it indicates the user position inside the MU-MIMO frame. Note that the USER_POSITION defined in this field is independent of the *txUserID* of the MAC-PHY interface.

Table 11 : Transmit Vector Specific Part for VHT frames

3.1.5 Transmit Vector Common Part for all HE frames

TxVector	txData	Field	Description
8	txData[0]	SOUNDING	Indicates whether this PPDU is Sounding 1'b0: NOT_SOUNDING 1'b1: SOUNDING
8	txData[1]	BEAMFORMED	BeamFormed Indicate that the transmission is beamformed.
8	txData[3:2]	GI_TYPE	Guard Interval Type 2'b00: 0.8 us 2'b01: 1.6 us 2'b10: 3.2 us
8	txData[4]	STBC	Space Time Block Coding It indicates if STBC is used

TxVector	txData	Field	Description
9	txData[0]	UPLINK_FLAG	UP Link Flag Set to 1 if the HE PPDU is addressed to an AP. Set to 0 otherwise
9	txData[1]	BEAM_CHANGE	Beam Change Set to 1 to indicate that the pre-HE-STF portion of the PPDU is spatially mapped differently from HE-LTF1. Set to 0 to indicate that the pre-HE-STF portion of the PPDU is spatially mapped the same way as HE-LTF1 on each tone.
9	txData[2]	DCM	Dual Carrier Modulation Set to 1 to indicate that dual carrier modulation is used for the HE-Data field. Set to 0 to indicate that dual carrier modulation is not used for the HE-Data field.
9	txData[4:3]	HE_LTF_TYPE	Type of HE-LTF Indicates the type of HE-LTF. 2’b00: 1x HE-LTF for 3.2 µs 2’b01: 2x HE-LTF for 6.4 µs 2’b10: 4x HE-LTF for 12.8 µs 2’b11: *Reserved*
9	txData[5]	DOPPLER	Doppler Indicates whether the doppler effect should be considered for the PPDU or not.
9	txData[6]	MIDAMBLE	Midamble Periodicity Indicates the midamble periodicity in number of OFDM symbols in the Data field. 1’b0: Periodicity of 10 symbols 1’b1: Periodicity of 20 symbols
10	txData[5:0]	BSS_COLOR	BSS Color Indicate the BSS color of the AP.
11	txData[6:0]	TXOP_DURATIO N	Duration of TXOP Indicates a duration that is used to update the NAV for this TXOP. If TXOP_DURATION[0] = 0, TXOP duration is 8* TXOP_DURATION[6:1] us If TXOP_DURATION[0] = 1, TXOP duration is 512+128* TXOP_DURATION[6:1] us If TXOP_DURATION = 127, it indicates UNSPECIFIED
12	txData[3:0]	SPATIAL_REUSE	Spatial Reuse Indicates the spatial reuse parameter value

Table 12 : Transmit Vector Common Part for all HE frames

3.1.6 Transmit Vector Specific Part for HE-SU and HE-EXT-SU frames

TxVector	txData	Field	Description
13	txData[7:0]	SMMINDEX	Spatial Map Matrix Index Indicates which spatial map matrix to be used. In case of beamforming, the smmindex indicates the pointers in the index of the beamforming report memory. Refer Section *5.1*, *Spatial map matrix table* for details
14	txData[3:0]	MCS	Modulation Coding Scheme This field represents the MCS.
14	txData[6:4]	NSS	Number of Spatial Stream This field indicates the number of Spatial Stream and is coded as follow: 3'b000: 1SS 3'b001: 2SS … 3'b111: 8SS
14	txData[7]	FEC_CODING	FEC Coding 1'b0: Use BCC 1'b1: Use LDPC
15	txData[7:0]	LENGTH[7:0]	Length of the PPDU This field takes a value from 0 – 1048575
16	txData[7:0]	LENGTH[15:8]	
17	txData[3:0]	LENGTH[19:16]	
17	txData[6:4]	PACKET_EXTENSION	Nominal Packet Padding The duration of the Nominal Packet Padding field in 4us unit. Possible values are 0 µs, 8 µs and 16 µs. 3'b000: PE0 for 0 µs 3'b010: PE2 for 8 µs 3'b100: PE4 for 16 µs

Table 13 : Transmit Vector Specific Part for HE-SU and HE-EXT-SU frames

3.1.7 Transmit Vector Specific Part for HE-MU frames

TxVector	txData	Field	Description
13	txData[0]	SIG_B_COMPRESSION_MODE	SIG B Compression Mode Used to differentiate full bandwidth MU-MIMO from OFDMA MU PPDU. In case of full bandwidth MU-MIMO set to 1, otherwise set to 0.

TxVector	txData	Field	Description
13	txData[1]	DCM_SIG_B	Dual Carrier Modulation on SIG B Set to 1 to indicate that dual carrier modulation is used for the HE-SIG-B field. Set to 0 to indicate that dual carrier modulation is not used for the HE-SIG-B field.
13	txData[4:2]	MCS_SIG_B	MCS of SIG B Indicates the modulation and coding scheme used for HE-SIGB field. Integer in the range 0 to 5
14	txData[7:0]	RU_ALLOCATION	RU Allocation Indicates the RU assignment in the frequency domain. It also indicates the number of users in each RU. For RUs of size greater than or equal to 106-tones that support MUMIMO, It indicates the number of users multiplexed using MU-MIMO.
15	txData[7:0]	RU_ALLOCATION	RU Allocation Indicates the RU assignment in the frequency domain. It also indicates the number of users in each RU. For RUs of size greater than or equal to 106-tones that support MUMIMO, It indicates the number of users multiplexed using MU-MIMO.
16	txData[7:0]	RU_ALLOCATION	RU Allocation Indicates the RU assignment in the frequency domain. It also indicates the number of users in each RU. For RUs of size greater than or equal to 106-tones that support MUMIMO, It indicates the number of users multiplexed using MU-MIMO.
17	txData[7:0]	RU_ALLOCATION	RU Allocation Indicates the RU assignment in the frequency domain. It also indicates the number of users in each RU. For RUs of size greater than or equal to 106-tones that support MUMIMO, It indicates the number of users multiplexed using MU-MIMO.
18	txData[7:0]	N_USER	Number of User Indicates the number of users. 7’d0 : *Reserved* 7’d1 : 1 user 7’d2 : 2 users …
19+7*UserIndex	txData[7:0]	SMMINDEX	Spatial Map Matrix Index Indicates which spatial map matrix to be used. In case of beamforming, the smmindex indicates the pointers in the index of the beamforming report memory. Refer Section *5.1*, *Spatial map matrix table* for details

TxVector	txData	Field	Description
20+7*UserIndex	txData[3:0]	MCS	Modulation Coding Scheme This field represents the MCS.
20+7*UserIndex	txData[6:4]	NSS	Number of Spatial Stream This field indicates the number of Spatial Stream and is coded as follow: 3'b000: 1SS 3'b001: 2SS … 3'b111: 8SS
20+7*UserIndex	txData[7]	FEC_CODING	FEC Coding 1'b0: Use BCC 1'b1: Use LDPC
21+7*UserIndex	txData[7:0]	LENGTH[7:0]	Length of the PPDU This field takes a value from 0 – 1048575
22+7*UserIndex	txData[7:0]	LENGTH[15:8]	
23+7*UserIndex	txData[3:0]	LENGTH[19:16]	
23+7*UserIndex	txData[6:4]	PACKET_EXTENSION	Nominal Packet Padding The duration of the Nominal Packet Paddin field in 4us unit. Possible values are 0 µs, 8 µs and 16 µs. 3'b000: PE0 for 0 µs 3'b010: PE2 for 8 µs 3'b100: PE4 for 16 µs
24+7*UserIndex	txData[7:0]	STAID[7:0]	STA ID Indicates the STA ID
25+7*UserIndex	txData[2:0]	STAID[10:8]	
25+7*UserIndex	txData[7:3]	USER_POSITION	User Position In case of MU-MIMO transmission, it indicates the user position inside the MU-MIMO frame. Note that the USER_POSITION defined in this field is independent of the *txUserID* of the MAC-PHY interface.

Table 14 : Transmit Vector Specific Part for HE-MU frames

The structure from RxVector byte 19 to 25 is repeated for each user.

3.1.8 Transmit Vector Specific Part for HE-TB frames

TxVector	txData	Field	Description
12	txData[7:4]	SPATIAL_REUSE2	
13	txData[3:0]	SPATIAL_REUSE3	
13	txData[7:4]	SPATIAL_REUSE4	
14	txData[7:0]	HE_SIGA_RESERVED[7:0]	HE SIGA Reserved bits Indicates the Reserved field setting for HE-SIG-A2.
15	txData[0]	HE_SIGA_RESERVED[8]	
15	txData[3:1]	NUM_HE_LTF	Number of HE LTF Indicate the number of HE-LTF symbols - 1.
15	txData[4]	HE_LTF_MODE	HE LTF Mode Indicates whether the UL MU MIMO transmission uses single stream pilots or a mask on each spatial stream of the LTF sequence by a distinct orthogonal code. It is only present for full bandwidth MU-MIMO.
15	txData[5]	LDPC_EXTRA_SYMBOL	LDPC Extra Symbol Indicates the presence of the extra OFDM symbol for LDPC in an HE TB PPDU. Set to 1 if an extra OFDM symbol for LDPC is present. Set to 0 if an extra OFDM symbol for LDPC is not present.
16	txData[2:0]	STARTING_STS_NUM	Starting STS Number Indicates the starting STS number in the global space-time streams for the UL MU MIMO.
17	txData[7:0]	RU_ALLOCATION	RU Allocation RU allocation in the whole bandwidth.
18	txData[7:0]	SMMINDEX	Spatial Map Matrix Index Indicates which spatial map matrix to be used. In case of beamforming, the smmindex indicates the pointers in the index of the beamforming report memory. Refer Section *5.1*, *Spatial map matrix table* for details
19	txData[3:0]	MCS	Modulation Coding Scheme This field represents the MCS.

TxVector	txData	Field	Description
19	txData[6:4]	NSS	Number of Spatial Stream This field indicates the number of Spatial Stream and is coded as follow: 3'b000: 1SS 3'b001: 2SS … 3'b111: 8SS
19	txData[7]	FEC_CODING	FEC Coding 1'b0: Use BCC 1'b1: Use LDPC
20	txData[7:0]	LENGTH[7:0]	Length of the PPDU This field takes a value from 0 – 1048575
21	txData[7:0]	LENGTH[15:8]	
22	txData[3:0]	LENGTH[19:16]	
22	txData[6:4]	PACKET_EXTENSION	Packet Extension In case of Trigger Frame, PACKET_EXTENSION[1:0] : Pre-FEC Padding Factor PACKET_EXTENSION[2] : PE Disambiguity In case of TRS, PACKET_EXTENSION[2:0] : Default PE Duration (in unit of 4us). 3'b000 : 0 us 3'b001 : 4 us 3'b010 : 8 us 3'b011 : 12 us 3'b100 : 16 us
22	txData[7]	TRIGGER_METHOD	Trigger Method Indicates the method used to trigger this HE TB PPDU transmission. 1'b0: Trigger Frame 1'b1: TRS Control field
23	txData[6:0]	RU_TONE_SET_INDEX	RU Tone Set Index Indicates the RU tone set used for an NDP feedback report PPDU. This field is valid only in case of HE-TRIG NDP frame (LENGTH = 0)

TxVector	txData	Field	Description
23	txData[7]	FEEDBACK_STATUS	Feedback Status Indicates the value of the FEEDBACK_STATUS bit used to encode the feedback. This field is valid only in case of HE-TRIG NDP frame (LENGHT = 0)

Table 15 : Transmit Vector Specific Part for HE-TRIG frames

It may occur that the MAC insert a pause in the TxVector transmission by de-asserting the macDataValid in order to leave more time for HE_TB A-MPDU preparation. This pause shall be inserted between the byte 1 and 20 of the txVector, just before the first byte of LENGTH.

3.1.9 Transmit Vector Response

TxVector	rxData	Field	Description
0	rxData[7:0]	TIME_OF_DEPARTURE[7:0]	Time Of Departure The locally measured time when the first frame energy is sent by the transmitting port, in units equal to 1/TIME_OF_DEPARTURE_ClockRate.
1	rxData[7:0]	TIME_OF_DEPARTURE[15:8]	
2	rxData[7:0]	TIME_OF_DEPARTURE[23:16]	
3	rxData[7:0]	TIME_OF_DEPARTURE[31:24]	
4	rxData[7:0]	TIME_OF_DEPARTURE_CLOCKRATE[7:0]	Time Of Departure Clock Rate The clock rate, in units of MHz, is used to generate the TIME_OF_DEPARTURE value.
5	rxData[7:0]	TIME_OF_DEPARTURE_CLOCKRATE[15:8]	

Table 16 : Transmit Vector Response Part for all frames

3.2 Receive Vector definition (RxVector)

The Rx Vector is transmitted on the *rxData* lines by the PHY at the start and end of the receive cycle. For more details see section *2.2.2.4.1*, *SIFS separation between PPDUs*. Its length and content depend on the frame format and number of users.

3.2.1 Receive Vector Common Part for all frames

RxVector1	rxData	Field	Description

RxVector1	rxData	Field	Description
0	rxData[3:0]	FORMAT	Format and Modulation This field indicates the format and the modulation of the PPDU, and is encoded as follows: 4'b0000: NON-HT 4'b0001: NON-HT-DUP-OFDM 4'b0010: HT-MF 4'b0011: HT-GF 4'b0100: VHT 4'b0101: HE-SU 4'b0110: HE-MU 4'b0111: HE-EXT-SU 4'b1000: HE-TB
0	rxData[6:4]	CH_BANDWIDTH	Channel Bandwidth of the transmitted frame This field is encoded as follows: 3'b000: 20MHz 3'b001: 40MHz 3'b010: 80MHz 3'b011: 160MHz or 80+80MHz 3'b100: HE-CBW-PUNC80-PRI for preamble puncturing in 80 MHz, where in the preamble only the secondary 20 MHz is punctured 3'b101: HE-CBW-PUNC80-SEC for preamble puncturing in 80 MHz, where in the preamble only one of the two 20 MHz sub-channels in secondary 40 MHz is punctured 3'b110: HE-CBW-PUNC160-PRI20 for preamble puncturing in 160 MHz or 80+80 MHz, where in the primary 80 MHz of the preamble only the secondary 20 MHz is punctured 3'b111: HE-CBW-PUNC160-SEC40 for preamble puncturing in 160 MHz or 80+80 MHz, where in the primary 80 MHz of the preamble the primary 40 MHz is present
0	rxData[7]	PREAMBLE_TYPE	Preamble Type For *formatMod* = NON-HT and the transmission modulation is DSSS/CCK, this bit is encoded as follows: 1'b0: Short Preamble 1'b1: Long Preamble It is reserved for other *formatMod* types.

RxVector1	rxData	Field	Description
1	rxData[7:0]	ANTENNA_SET	Antenna Set Indicates which antennas to be used for transmission. Each bit corresponds to one antenna & at a time maximum 4 bits can be set out of 8 bits.
2	rxData[7:0]	RSSI_LEGACY	Received Signal Strength in dBm measured during Legacy Preamble (from -128 to 0) 2's Complement 8'h80 : -128 dBm 8'hFF : -1 dBm 8'h00 : 0 dBm 8'h01-8'h7F : *Reserved*
3	rxData[7:0]	L_LENGTH[7:0]	Legacy Length of the PPDU This field takes a value from 0 to 4095
4	rxData[3:0]	L_LENGTH[11:8]	
4	rxData[7:4]	L_RATE[3:0]	Legacy Rate of the PPDU. 1 Mbps: 4'b0000 2 Mbps: 4'b0001 5.5 Mbps: 4'b0010 11 Mbps: 4'b0011 6 Mbps: 4'b1011 9 Mbps: 4'b1111 12 Mbps: 4'b1010 18 Mbps: 4'b1110 24 Mbps: 4'b1001 36 Mbps: 4'b1101 48 Mbps: 4'b1000 54 Mbps: 4'b1100 When transmitting a HT, VHT or HE PPDU, this field always indicates 6 Mbps.
5	rxData[7:0]	RSSI	Received Signal Strength in dBm measured during Data Same coding than RSSI_LEGACY

Table 17 : Receive Vector Common Part for all frames

3.2.2 Receive Vector Specific Part for NON-HT and NON-HT-DUP-OFDM frames

RxVector1	rxData	Field	Description
6	rxData[0]	DYN_BANDWIDTH_IN_NON_HT	Dynamic Bandwidth 1'b0: Static bandwidth operation 1'b1: Dynamic bandwidth operation

RxVector1	rxData	Field	Description
6	rxData[2:1]	CH_BANDWIDTH_IN_NON_HT	Channel Bandwidth In Non HT Channel Bandwidth In Non HT used in case of Bandwidth Signalling. This field is encoded as follows: 2'b00: 20MHz 2'b01: 40MHz 2'b10: 80MHz 2'b11: 160MHz or 80+80MHz
6	rxData[7]	L_SIG_VALID	L-SIG Valid This bit is set by the modem when the L-SIG is valid.

Table 18 : Receive Vector Specific Part for NON-HT and NON-HT-DUP frames

3.2.3 Receive Vector Specific Part for HT-MM and HT-GF frames

RxVector1	rxData	Field	Description
6	rxData[0]	SOUNDING	Indicates whether this PPDU is Sounding 1'b0: NOT_SOUNDING 1'b1: SOUNDING
6	rxData[1]	SMOOTHING	Smoothing recommended 1'b1: Smoothing recommended as part of channel estimation 1'b0: Smoothing not recommended as part of channel estimation
6	rxData[2]	GI_TYPE	Guard Interval Type 1'b0: LONG_GI (800 ns) used during the transmission 1'b1: SHORT_GI (400 ns) used during the transmission
6	rxData[3]	AGGREGATION	MPDU Aggregate 1'b0: PPDU is not an A-MPDU 1'b1: PPDU is an A-MPDU
6	rxData[4]	STBC	Space Time Block Coding It indicates if STBC is used
6	rxData[6:5]	NUM_EXT_SS	Number of Extension Spatial Streams Number of extension spatial streams to be transmitted (0 – 3).
6	rxData[7]	L_SIG_VALID	L-SIG Valid This bit is set by the modem when the L-SIG is valid.
7	rxData[6:0]	MCS	Modulation Coding Scheme This field represents the MCS.

RxVector1	rxData	Field	Description
7	rxData[7]	FEC	FEC Coding 1'b0: Use BCC 1'b1: Use LDPC
8	rxData[7:0]	LENGTH[7:0]	Length of the HT PPDU This field takes a value from 0 – 65535
9	rxData[7:0]	LENGTH[15:8]	

Table 19 : Receive Vector Specific Part for HT-MM and HT-GF frames

3.2.4 Receive Vector Specific Part for VHT frames

RxVector1	rxData	Field	Description
6	rxData[0]	SOUNDING	Indicates whether this PPDU is Sounding 1'b0: NOT_SOUNDING 1'b1: SOUNDING
6	rxData[1]	BEAMFORMED	BeamFormed Indicate that the frame was beamformed.
6	rxData[2]	GI_TYPE	Guard Interval Type 1'b0: LONG_GI (800 ns) used during the transmission 1'b1: SHORT_GI (400 ns) used during the transmission
6	rxData[4]	STBC	Space Time Block Coding It indicates if STBC is used
6	rxData[5]	DOZE_NOT_ALLOWED	TXOP PS Not Allowed 1'b0: Doze mode during a TXOP allowed. 1'b1: Doze mode during a TXOP not allowed.
6	rxData[6]	FIRST_USER	First User In case of MU, indicates if the user position inside the received MU-MIMO frame is the first one or not. 1'b0: User position is not the first one. 1'b1: User position is the first one. This bit is reserved in case of SU reception.
7	txData[7:0]	PARTIAL_AID[7:0]	Partial AID Provides an abbreviated indication of the intended recipient(s) of the PSDU. Integer: range 0-511.
8	txData[0]	PARTIAL_AID[8]	

RxVector1	rxData	Field	Description
8	txData[6:1]	GROUP_ID[5:0]	Group ID The PHY uses the *grouID* to select between SU-MIMO or MU-MIMO transmission. If *groupID* is different than 0 or 63 then the transmission will be MU-MIMO. Otherwise, the transmission will be SU-MIMO.
9	rxData[3:0]	MCS	Modulation Coding Scheme This field represents the MCS.
9	rxData[6:4]	NSS	Number of Spatial Stream This field indicates the number of Spatial Stream and is coded as follow: 3'b000: 1SS 3'b001: 2SS … 3'b111: 8SS
9	rxData[7]	FEC	FEC Coding 1'b0: Use BCC 1'b1: Use LDPC
10	rxData[7:0]	LENGTH[7:0]	Length of the VHT PPDU This field takes a value from 0 – 1048575
11	rxData[7:0]	LENGTH[15:8]	
12	rxData[3:0]	LENGTH[19:16]	

Table 20 : Receive Vector Specific Part for VHT frames

3.2.5 Receive Vector Specific Part for all HE frames

RxVector1	rxData	Field	Description
6	rxData[0]	SOUNDING	Indicates whether this PPDU is Sounding 1'b0: NOT_SOUNDING 1'b1: SOUNDING
6	rxData[1]	BEAMFORMED	BeamFormed Indicate that the transmission is beamformed.
6	rxData[3:2]	GI_TYPE	Guard Interval Type 2'b00: 0.8 us 2'b01: 1.6 us 2'b10: 3.2 us
6	rxData[4]	STBC	Space Time Block Coding It indicates if STBC is used

RxVector1	rxData	Field	Description
7	rxData[0]	UPLINK_FLAG	UP Link Flag Set to 1 if the HE PPDU is addressed to an AP. Set to 0 otherwise
7	rxData[1]	BEAM_CHANGE	Beam Change Set to 1 to indicate that the pre-HE-STF portion of the PPDU is spatially mapped differently from HE-LTF1. Set to 0 to indicate that the pre-HE-STF portion of the PPDU is spatially mapped the same way as HE-LTF1 on each tone.
7	rxData[2]	DCM	Dual Carrier Modulation Set to 1 to indicate that dual carrier modulation is used for the HE-Data field. Set to 0 to indicate that dual carrier modulation is not used for the HE-Data field.
7	rxData[4:3]	HE_LTF_TYPE	Type of HE-LTF Indicates the type of HE-LTF. 2'b00: 1x HE-LTF for 3.2 µs 2'b01: 2x HE-LTF for 6.4 µs 2'b10: 4x HE-LTF for 12.8 µs 2'b11: *Reserved*
7	rxData[5]	DOPPLER	Doppler Indicates whether the doppler effect should be considered for the PPDU or not.
8	rxData[5:0]	BSS_COLOR	BSS Color Indicate the BSS color of the AP.
9	rxData[6:0]	TXOP_DURATIO N	Duration of TXOP Indicates a duration that is used to update the NAV for this TXOP. If TXOP_DURATION[0] = 0, TXOP duration is 8* TXOP_DURATION[6:1] us If TXOP_DURATION[0] = 1, TXOP duration is 512+128* TXOP_DURATION[6:1] us

Table 21 : Receive Vector Specific Part for all HE frames

3.2.6 Receive Vector Specific Part for HE-SU and HE-EXT-SU frames

RxVector	rxData	Field	Description

RxVector	rxData	Field	Description
10	rxData[3:0]	PE_DURATION	Packet Extension Duration The duration of the PE field. 3'b000: PE0 for 0 µs 3'b001: PE1 for 4 µs 3'b010: PE2 for 8 µs 3'b011: PE3 for 12 µs 3'b100: PE4 for 16 µs
10	rxData[7:4]	SPATIAL_REUSE	Spatial Reuse Indicates the spatial reuse parameter value
11	rxData[7:0]	*Reserved*	
12	rxData[3:0]	MCS	Modulation Coding Scheme This field represents the MCS.
12	rxData[6:4]	NSS	Number of Spatial Stream This field indicates the number of Spatial Stream and is coded as follow: 3'b000: 1SS 3'b001: 2SS … 3'b111: 8SS
12	rxData[7]	FEC_CODING	FEC Coding 1'b0: Use BCC 1'b1: Use LDPC
13	rxData[7:0]	LENGTH[7:0]	Length of the PPDU This field takes a value from 0 – 1048575
14	rxData[7:0]	LENGTH[15:8]	
15	rxData[3:0]	LENGTH[19:16]	

Table 22 : Receive Vector Specific Part for HE-SU and HE-EXT-SU frames

3.2.7 Receive Vector Specific Part for HE-MU frames

RxVector	rxData	Field	Description
10	rxData[3:0]	PE_DURATION	Packet Extension Duration The duration of the PE field. 3'b000: PE0 for 0 µs 3'b001: PE1 for 4 µs 3'b010: PE2 for 8 µs 3'b011: PE3 for 12 µs 3'b100: PE4 for 16 µs

RxVector	rxData	Field	Description
10	rxData[7:4]	SPATIAL_REUSE	Spatial Reuse Indicates the spatial reuse parameter value
11	rxData[0]	SIG_B_COMPRESSION_MODE	SIG B Compression Mode Used to differentiate full bandwidth MU-MIMO from OFDMA MU PPDU. In case of full bandwidth MU-MIMO set to 1, otherwise set to 0.
11	rxData[1]	DCM_SIG_B	Dual Carrier Modulation on SIG B Set to 1 to indicate that dual carrier modulation is used for the HE-SIG-B field. Set to 0 to indicate that dual carrier modulation is not used for the HE-SIG-B field.
11	rxData[4:2]	MCS_SIG_B	MCS of SIG B Indicates the modulation and coding scheme used for HE-SIGB field. Integer in the range 0 to 5
11	rxData[7:5]	RU_SIZE	RU Size Indicates the size of the RU allocated to the user encoded as follow : 3'd0: RU-26 3'd1: RU-52 3'd2: RU-106 3'd3: RU-242 3'd4: RU-484 3'd5: RU-996 Others: Reserved
12	rxData[3:0]	MCS	Modulation Coding Scheme This field represents the MCS.
12	rxData[6:4]	NSS	Number of Spatial Stream This field indicates the number of Spatial Stream and is coded as follow: 3'b000: 1SS 3'b001: 2SS … 3'b111: 8SS
12	rxData[7]	FEC_CODING	FEC Coding 1'b0: Use BCC 1'b1: Use LDPC
13	rxData[7:0]	LENGTH[7:0]	Length of the PPDU

RxVector	rxData	Field	Description
14	rxData[7:0]	LENGTH[15:8]	This field takes a value from 0 – 1048575
15	rxData[3:0]	LENGTH[19:16]	

Table 23 : Receive Vector Specific Part for HE-MU frames

3.2.8 Receive Vector Specific Part for HE-TB frames (as STA)

RxVector	rxData	Field	Description
10	rxData[7:0]	SPATIAL_REUSE	Spatial Reuse Indicates the spatial reuse parameter value
11	rxData[7:0]	SPATIAL_REUSE	Spatial Reuse Indicates the spatial reuse parameter value

3.2.9 Receive Vector Specific Part for HE-TB frames (as AP)

RxVector	rxData	Field	Description
10	rxData[7:0]	Reserved	Reserved
11	rxData[7:0]	N_USER	Number of User Indicates the number of users. 7'd0 : *0 user Indicate a NDP report.* 7'd1 : 1 user 7'd2 : 2 users …
12+7*UserIndex	rxData[3:0]	MCS	Modulation Coding Scheme This field represents the MCS.
12+7*UserIndex	rxData[6:4]	NSS	Number of Spatial Stream This field indicates the number of Spatial Stream and is coded as follow: 3'b000: 1SS 3'b001: 2SS … 3'b111: 8SS
12+7*UserIndex	rxData[7]	FEC_CODING	FEC Coding 1'b0: Use BCC 1'b1: Use LDPC
13+7*UserIndex	rxData[7:0]	LENGTH[7:0]	Length of the PPDU This field takes a value from 0 – 1048575
14+7*UserIndex	rxData[7:0]	LENGTH[15:8]	

RxVector	rxData	Field	Description
15+7*UserIndex	rxData[3:0]	LENGTH[19:16]	
16+7*UserIndex	rxData[7:0]	STAID[7:0]	STA ID Indicates the STA ID
17+7*UserIndex	rxData[2:0]	STAID[10:8]	

Table 24 : Receive Vector Specific Part for HE- TRIG frames

The structure from RxVector byte 12 to 17 is repeated for each user.

If N_USER is 0, it indicates a HE NDP triggered frame (see *3.2.10 Receive Vector Specific Part for HE-NDP-TRIG frames*)

3.2.10 Receive Vector Specific Part for HE-NDP-TRIG frames

RxVector	rxData	Field	Description
11	rxData[7:0]	N_USER	Number of User Indicates the number of users. 7'd0 : 0 user Indicate a NDP report.
12	rxData[7:0]	NDP_REPORT[7:0]	NDP Report Provides the vector of the detected FEEDBACK_STATUS for each P-matrix code and each RU_TONE_SET_INDEX for the first 20MHz channel
13	rxData[7:0]	NDP_REPORT[15:8]	
14	rxData[7:0]	NDP_REPORT[23:16]	
15	rxData[7:0]	NDP_REPORT[31:24]	NDP Report Provides the vector of the detected FEEDBACK_STATUS for each P-matrix code and each RU_TONE_SET_INDEX for the second 20MHz channel This part is provided by the PHY only if the Channel BW of the HE-NDP-TRIGGER frame is 40MHz
16	rxData[7:0]	NDP_REPORT[39:32]	
17	rxData[7:0]	NDP_REPORT[47:40]	NDP Report Provides the vector of the detected FEEDBACK_STATUS for each P-matrix code and each RU_TONE_SET_INDEX for the second 40MHz channel This part is provided by the PHY only if the Channel BW of the HE-NDP-TRIGGER frame is 80MHz
18	rxData[7:0]	NDP_REPORT[55:48]	
19	rxData[7:0]	NDP_REPORT[63:56]	
20	rxData[7:0]	NDP_REPORT[71:64]	
21	rxData[7:0]	NDP_REPORT[79:72]	NDP Report

RxVector	rxData	Field	Description
22	rxData[7:0]	NDP_REPORT[87:80]	Provides the vector of the detected FEEDBACK_STATUS for each P-matrix code and each RU_TONE_SET_INDEX for the second 80MHz channel This part is provided by the PHY only if the Channel BW of the HE-NDP-TRIGGER frame is 160MHz or 80+80MHz
23	rxData[7:0]	NDP_REPORT[95:88]	
24	rxData[7:0]	NDP_REPORT[103:96]	
25	rxData[7:0]	NDP_REPORT[111:104]	
26	rxData[7:0]	NDP_REPORT[119:112]	
27	rxData[7:0]	NDP_REPORT[127:120]	
28	rxData[7:0]	NDP_REPORT[135:128]	
29	rxData[7:0]	NDP_REPORT[143:136]	

Table 25 : Receive Vector Specific Part for HE-NDP-TRIG frames

3.2.11 Receive Vector 2

Fields transmitted by the PHY at the end of a reception:

mpIFClk number	Parameter	Description
0	rcpi1[7:0]	Receive Channel Power Indicator. It is the RF power measured over the data portion of the received frame, on the 1st RX chains. The valid values are 0 to 220. The range of power which can be indicated by this field is from -110 dBm (RCPI = 0) to 0 dBm (RCPI = 220). RCPI is a monotonically increasing function of the power in dBm. The values 221 to 254 are reserved. The value 255 indicates no measurement is available.
1	rcpi2[7:0]	Receive Channel Power Indicator. It is the RF power measured over the data portion of the received frame, on the 2nd RX chains. The valid values are 0 to 220. The range of power which can be indicated by this field is from -110 dBm (RCPI = 0) to 0 dBm (RCPI = 220). RCPI is a monotonically increasing function of the power in dBm. The values 221 to 254 are reserved. The value 255 indicates no measurement is available.

mpIFClk number	Parameter	Description
2	rcpi3[7:0]	Receive Channel Power Indicator. It is the RF power measured over the data portion of the received frame, on the 3rd RX chains. The valid values are 0 to 220. The range of power which can be indicated by this field is from -110 dBm (RCPI = 0) to 0 dBm (RCPI = 220). RCPI is a monotonically increasing function of the power in dBm. The values 221 to 254 are reserved. The value 255 indicates no measurement is available.
3	rcpi4[7:0]	Receive Channel Power Indicator. It is the RF power measured over the data portion of the received frame, on the 4th RX chains. The valid values are 0 to 220. The range of power which can be indicated by this field is from -110 dBm (RCPI = 0) to 0 dBm (RCPI = 220). RCPI is a monotonically increasing function of the power in dBm. The values 221 to 254 are reserved. The value 255 indicates no measurement is available.
4	evm1[7:0]	Error Vector Magnitude for space time stream 1. (Not supported yet by PHY)
5	evm2[7:0]	Error Vector Magnitude for space time stream 2. (Not supported yet by PHY)
6	evm3[7:0]	Error Vector Magnitude for space time stream 3. (Not supported yet by PHY)
7	evm4[7:0]	Error Vector Magnitude for space time stream 4. (Not supported yet by PHY)

Table 26 : Receive Vector 2

3.3 Recommendations & restrictions for RW-WLAN-HE TxVector parameters

In general, the following rules apply:

1. Nsts >= Nss
2. Ntx >= Nsts

Once the MCS has been determined (from LAA or otherwise), the Nss is known.

- ✓ Assuming no STBC, Nsts = Nss. Nss can be 1 or 2.
- ✓ With STBC = 1, Nss can only be 1. This should automatically come from the MCS calculation. Nsts = 2.

Field	Description

Field	Description
txPwrLevel	Transmit Power Level Restriction: None Recommendation: None
chBW	Channel Bandwidth Restriction: Shall be aligned with the PHY capabilities Recommendation: None
chOffset	Channel offset Restriction: None Recommendation: None
smoothing	Smoothing recommended Restriction: None Recommendation: Sounding frames are not smoothed. Transmit beam-formed frames are not smoothed.
antennaSet	Antenna Set Restriction: Can only be set to the following values: 2'b00, 2'b01, 2'b10 and 2'b11 since only 2 antennas are supported. Recommendation: Set equal to value 2'b11.
smmIndex	Spatial Map Matrix Index Restriction: Recommendation: TBD
mcs	Modulation Coding Scheme Restriction: Depend on FormatMod Recommendation: None
preType	Preamble Type Restriction: None Recommendation: None
formatMod	Format and Modulation Restriction: None Recommendation: None
numExtnSS	Number of Extension Spatial Streams Restriction: TBD Recommendation: TBD
stbc	Space Time Block Coding Restriction: Cannot be set if Ntx = 1. Recommendation: None

Field	Description
fecCoding	FEC Coding Restriction: None Recommendation: None
sounding	Indicates whether this PPDU is Sounding Restriction: None Recommendation: used only for FLA and I-BF.
legLength	Legacy Length of the PPDU Restriction: Cannot exceed 4095. Recommendation: None
legRate	Legacy Rate of the PPDU. Restriction: 6 Mbps for HT-MF, VHT, HE PPDU Recommendation: None
service	Service Restriction: None Recommendation: None
Length	Length of the HT/VHT/HE PPDU Restriction: Cannot exceed *aPPDUMaxTime* on air. Recommendation: None
nTx	Number of Transmit Chains Restriction: Ntx must be >= Nss of the PPDU, e.g. a PPDU transmitted with MCS 15 cannot have Ntx = 1. Recommendation: Set to maximum number of transmit chains available in the solution - 1.
shortGI	Short Guard Interval Restriction: TBD Recommendation: TBD
aggregation	MPDU Aggregate Restriction: None Recommendation: None

Table 27 : Recommendations & restrictions for RW-WLAN-nX TxVector parameters

4 PHY address space

As mentioned in Section *2.3*, *PLME interface*, the MAC uses the PLME interface to access various register and memory spaces within the PHY. These are:

1. PHY Registers

The address space of the PHY is described in *[3]*.

5 Functional description

5.1 Spatial map matrix table

The Spatial Map Matrix (SMM) table stored in the PHY maps the SMM Index to the type of Qk matrix to be used by the PHY for the mapping of space time streams on to different RF chains during transmission. This index is passed through the TxVector. Some of the SMMs are hard-coded in HW and some of them are stored in PHY memory.

The SMM can be classified into following categories:

1. Hard coded in the logic: These matrix elements take values 1 or 0 and do not require much storage space. e.g. Direct Mapping.
2. Stored in RAM: These matrix values are updated at run time by the PHY during normal operation. e.g. the Steering Matrix calculated during beamforming.

5.1.1 Mapping of SMM indices to type of matrices

The following table shows the mapping from SMM Index to the different Matrices:

SMM Index	Matrix Definition
8’h00	Direct Mapping – Identity Matrix (Qk = I)
8’h01	Indirect Mapping using Hadamard matrix
8’h02	Indirect Mapping using Hadamard matrix
8’h03-8’h7F	*Reserved*
8’h80-8’hFF	Beamforming Steering Matrix stored in the RAM

Table 28 : SMM Index types

5.2 MIMO commands

MIMO commands are passed from the MAC to the PHY to enable Calibration and Implicit Beamforming.

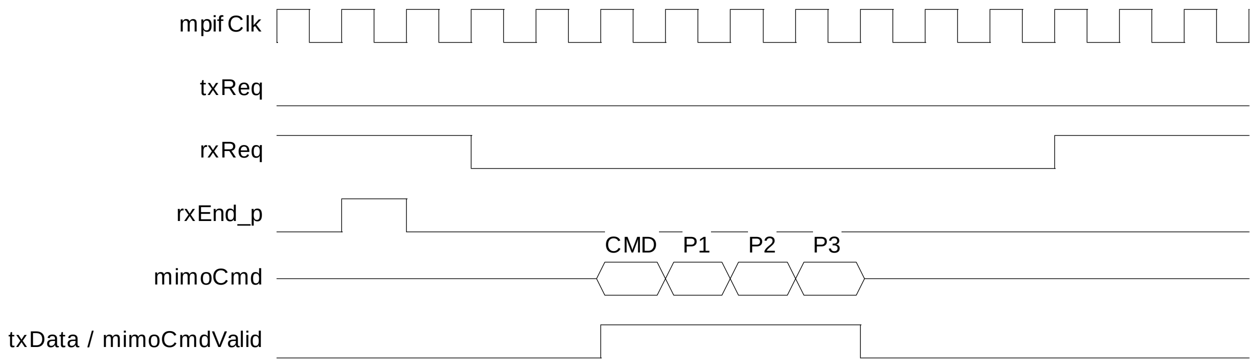

Figure 20 : Timing diagram for MIMO command procedure

When the PHY is not in TX state, the *txData* lines are idle, and hence are multiplexed with *mimoCmd* bus. The *mimoCmd* is validated by the *mimoCmdValid*. This command interface is synchronous to *mpIFClk*. When a PPDU is passed to the MAC on the interface, the MAC determines if it is a valid PPDU for which a MIMO command can be issued. The MIMO commands are issued within a few clocks after a received PPDU has been completely passed to the MAC, before the start of a new transmission or reception cycle, i.e. when the *txReq* and *rxReq* signals are de-asserted.

The MIMO command cycle lasts for 4 clocks. The first clock is the Command phase. The next 3 clocks are used for passing various parameters related to that command.

5.2.1 MIMO command details

The following commands are currently defined:

Command	Command Code	Description
IDLE	4’d0	Invalid or no command.
Reserved	8’d1 – 8’d255	Reserved for future use.

Table 29 : MIMO commands

The following table gives the list of parameters passed during each phase of MIMO command transfer. The fields are mentioned with the bit width and should be mapped from LSB to MSB, in the order mentioned in the table.

References

[1] Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications, IEEE Std 802.11™-2016

[2] AMBA Specification Rev 2.0

[3] RW-WLAN-nX PHY Functional Specification (*RW-WLAN-nX-Modem-FS*)

[4] RW-WLAN-nX MAC Functional Specification (*RW-WLAN-nX-MAC-HW-FS*)

[5] Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Enhancements for High Efficiency, IEEE P802.11ax™-Draft2.1