Proposed Text for Section 5. thru 5.2.13.4, Based on responses to Draft D1 Letter Ballot processed at March 1995 Meeting

Greg Ennis
Editor IEEE P802.11
16331 Englewood Av
Los Gatos CA 95032
Phone: +1 408 356 4775
Fax: +1 408 356 4775
E-Mail: gennis@netcom.com

Wim Diepstraten
AT&T WCND Utrecht
Zadelstede 1-10
3431 JZ Nieuwegein Netherlands
Phone +31 3402 97482
Fax: +31 3402 97555
E-Mail: wim.diepstraten@utrecht.attgis.com

Abstract: This paper presents the changes to section 5.1 thru 5.2 in the Draft Standard P802.11/D1 as a result of the Response to Draft D1 Letter Ballot processed at the March 1995 Meeting as shown in the companion Documents P802.11-95/67, P802.11-95/68 and P802.11-95/69.
Not all Letter Ballot comments were processed at the March 1995 Meeting.

Action: Adopt the changes in this paper to replace the relevant portions of Section 5 of P802.11/D1

5. MAC Sub-layer Functional Description

In the following sections, the MAC functional description is presented. Section 5.1 introduces the architecture of the MAC sublayer, including the distributed coordination function, the point coordination function and their coexistence in an 802.11 LAN. Sections 5.2 and 5.3 expand on this introduction and provide a complete functional description of each. Section 5.4 describes the security mechanisms within the MAC layer. Section 5.5 and 5.6 cover fragmentation and reassembly. Multirate support is addressed in Section 5.7. Section 5.8 reiterates the functional descriptions in the form of state machines.

5.1. MAC Architecture

The MAC is composed of several functional blocks: the MAC-LLC Service Interface, the MAC State Machines, the MAC Management State Machines and the MAC Management Information Base (MIB). The MAC-LLC Service Interface comprises the MAC Data Service and the MAC Management Service. These blocks perform the functions required to provide *contention* ~~asynchronous, time-bounded~~ and contention-free access control on a variety of physical layers. The functions are provided without reliance upon particular data rates or physical layer characteristics. The MAC provides both distributed and point coordination functions *and is able to support both asynchronous and time bounded service clases.* Figure 5-1 illustrates the MAC architecture.

The MAC-LLC Service Interface shall accept MAC service requests from higher layer entities and shall distribute those requests to either the MAC Data Service or the MAC Management Service as appropriate. The MAC Data and MAC Management Services shall interpret the service requests and shall cause the appropriate signals to be generated to initiate actions in the state machines. The MAC-LLC Service Interface shall also accept indications from the state machines and provide those indications to higher layer entities. The particular service requests and indications are described in Section 3.2.

The MAC State Machine shall provide the sequencing required to provide the distributed coordination function. The *MAC*~~Mac~~ State Machine shall provide the protocol sequencing necessary to provide asynchronous communication service. . *The MAC State Machine shall* ~~It may also~~ provide the sequencing required to provide the point coordination function and the associated time-bounded and contention-free communication services. *The implementation of the PCF portions of the MAC State Machine (and the associated Time-bounded and contention-free services) are optional.* -The MAC State Machine shall not interfere with time-bounded nor contention-free communications even if the optional point coordination function is not implemented.

The MAC Management State Machine shall provide the protocol sequencing required to provide the following services:

a) Association and re-association
b) Access to the MAC MIB
c) Timing synchronization
d) Power management
e) Authentication

~~[editor's note (Bob): what are the other management services provided?]~~

~~[editor's note (Bob): are there other services mgt needs to provide to support time-bounded and contention-free services?]~~

The MAC MIB shall provide storage of and access to all of the information required to properly manage the MAC.

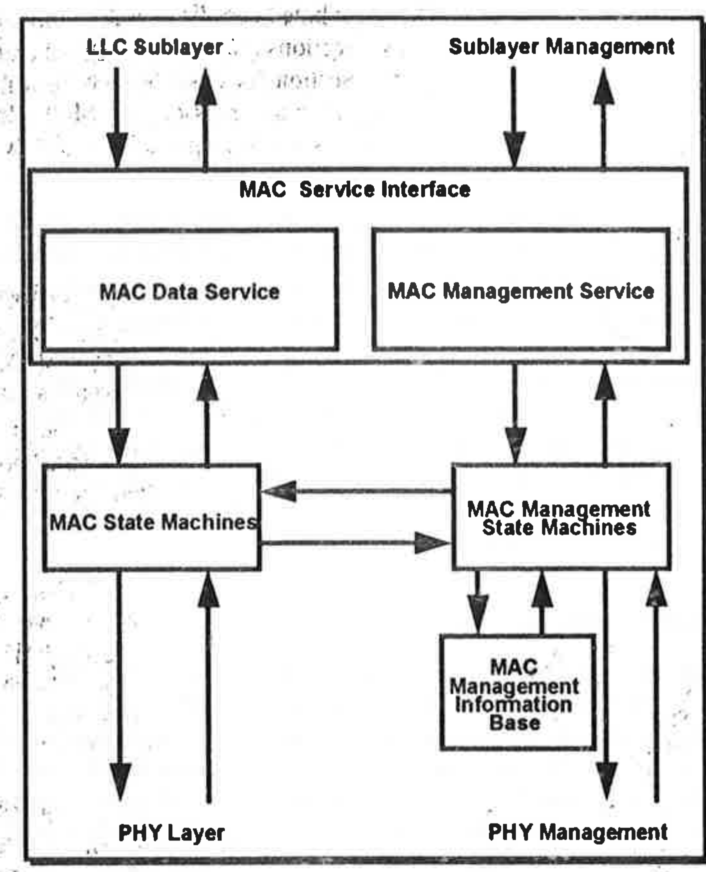


Figure 5-1: MAC Architecture Block Diagram

Viewed along a different axis, the MAC architecture can be described as shown in Figure 5-2 below as providing the point coordination function through the services of the distributed coordination function.

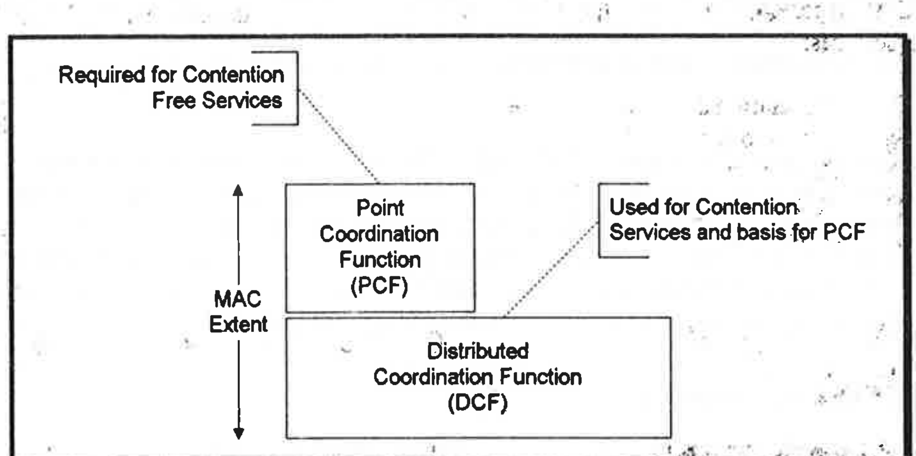


Figure 5-2: Alternative view of MAC architecture.

5.1.1. Distributed Coordination Function

The fundamental access method of the 802.11 MAC is a distributed coordination function known as carrier sense multiple access with collision avoidance, or CSMA/CA. The distributed coordination function shall be implemented in all stations ~~and access points~~. It is used within both ad hoc and infrastructure configurations.

A station wishing to transmit shall sense the medium to determine if another station is transmitting. If the medium is not busy, the transmission may proceed. The CSMA/CA distributed algorithm mandates that a gap of a minimum specified duration exist between contiguous frames ~~(a distributed interframe space, or DIFS).~~. A transmitting station shall ensure that the medium is idle for the required ~~DIFS~~ duration before attempting to transmit. If the medium is sensed busy ~~(a collision)~~ the station shall defer until the end of the current transmission. After deferral, the station shall select a random *backoff* interval and shall *decrement the interval counter while the medium is free.* ~~check that the medium remains idle for that interval.~~ A refinement of the method may be used under various circumstances to further minimize collisions - here the transmitting and receiving station exchange short control frames (**RTS** and **CTS** frames) prior to the data transmission. The details of CSMA/CA are described in Section 5.2. RTS/CTS exchanges are also presented in Section 5.2.

5.1.2. Point Coordination Function

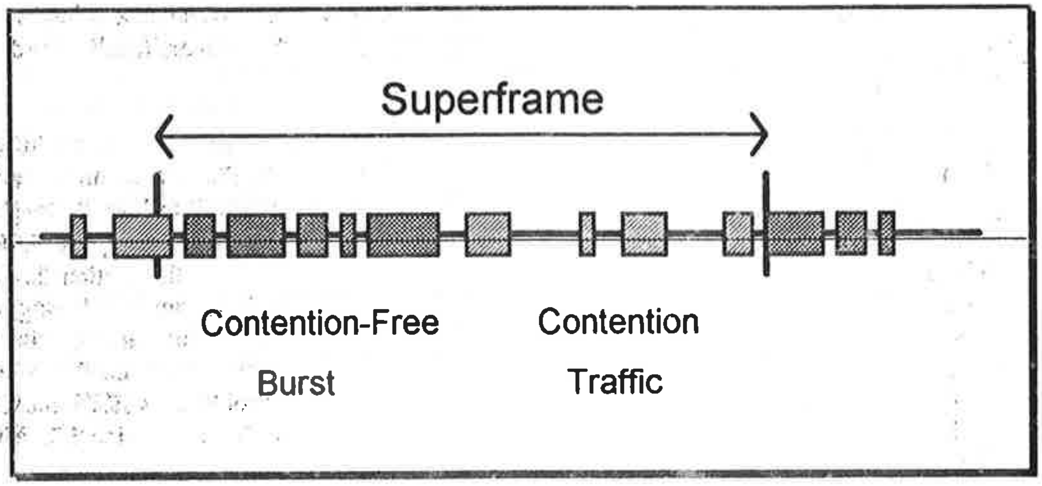
The 802.11 MAC may also incorporate an *optional*~~alternative~~ access method described as a point coordination function. This *optional*~~alternative~~ access method shall be implemented on top of the distributed coordination function. This access method uses a point coordinator to determine which station currently has the right to transmit. The operation is essentially that of polling with the point coordinator playing the role of the polling master. ~~The support of the point coordination function requires that the network configuration involves no overlapping point-coordinated BSS's on the same channel~~*The operation of the Point Coordination Function may require additional coordination, not sepcified in this standard, to permit effiecent operation in cases where multiple Point-Coordinated BSSs are operating on the same channel in overlapping physical space.*

The point-coordination function shall be built up from the distributed coordination function through the use of an access priority mechanism. Different classes of traffic can be defined through the use of different values for IFS, thereby creating prioritized access to the medium for those classes with a shorter IFS. The point coordination function shall use a ~~Point~~ IFS ~~(PIFS)~~ that is smaller than the ~~Distributed IFS (DIFS)~~*IFS* for data frames transmitted via the point coordination function. *The use of a smaller IFS implies that* ~~. Since PIFS is smaller than DIFS,~~ point-coordinated traffic shall have priority access to the medium.

The access priority provided by point-coordinated traffic may be utilized to create a **contention-free** access method. The priority access of the PIFS allows the point coordinator to "seize control" of the medium away from the other stations. The point coordinator can then control the frame transmissions of the stations so as to eliminate contention. ~~Also used in this regard is a mechanism in which the point coordinator shall signal to the other stations that such a burst is occurring, causing them to be silent during that period of the burst even if they don't directly hear the traffic.~~

5.1.3. Coexistence of DCF and PCF

~~Both t~~*T*he distributed coordination function and the point coordination function shall coexist without interference. The two access methods are integrated in a **superframe** in which a contention-free burst occurs at the beginning of the superframe, followed by a contention period. This is also described in Section 5.3. ~~The following figure depicts this mode of operation.~~



~~**Figure 5-3: Superframe Structure**~~

5.1.4. Fragmentation/Reassembly Overview

The process of partitioning a MAC Service Data Unit (MSDU) into smaller MAC level frames, MAC Protocol Data Units (MPDUs), is defined as fragmentation. *Fragmentation creates MPDUs smaller than the MSDU size to increase reliablity of successful transmission of the MSDU over a given PHY.* Fragmentation is ~~function of the source station~~ *accomplished at each immediate transmitter*. The process of recombining MPDUs into a single MSDU is defined as reassembly. Reassembly is ~~a function of the destination station~~*accomplished at each immediate recipient*.

~~The following are the Management Information Base (MIB) attributes used by fragmentation.~~

> ~~aMSDU_Size: This attribute specifies the maximum size of a MSDU, in octets, supported the 802.11 MAC. This is a fixed value.~~
>
> ~~aMax_Full_MPDU: This attribute specifies the maximum full size MPDU, in octets, that the attached PHY can transmit and is PHY dependent. This is a fixed value.~~
>
> ~~aMin_Full_MPDU: This attribute specifies the minimum full size MPDU, in octets, that the attached PHY can transmit and is PHY dependent. This is a fixed value.~~
>
> ~~aFragmentation_Threshold: This attribute specifies the current maximum size of a MPDU, in octets, that can be delivered to the PHY. An MSDU will be broken into fragments if its size exceeds the value of this attribute after adding MAC headers and trailers. The value of aFragmentation_Threshold must be less than or equal to aMax_Full_MPDU and greater than or equal to aMin_Full_MPDU. The default value for this attribute shall be equal to aMax_Full_MPDU.~~
>
> ~~aFragment_Payload: This attribute specifies current maximum size of a MPDU fragment, in octets. The value of this attribute equals aFragmentation_Threshold minus MAC headers and~~

~~trailers. The payload of a fragment shall never exceed this attribute. However, the size of the payload may be less than this attribute.~~

~~aMax_Transmit_MSDU_Lifetime: This attribute specifies the maximum amount of time allowed to transmit a MSDU.~~

~~aTransmit_MSDU_Timer: This attribute is replicated for each MSDU being transmitted. It is a timer that starts on the attempt to transmit the first fragment of the MSDU. If it exceeds aMax_Transmit_MSDU_Lifetime than all remaining fragments are discarded by the source station and no attempt is made to complete transmission of the MSDU.~~

~~aMax_Receive_MSDU_Lifetime: This attribute specifies the maximum amount of time allowed to receive a MSDU.~~

~~aReceive_MSDU_Timer: This attribute is replicted for each MSDU being received. It is a timer that starts on the reception of the first fragment of the MSDU. If it exceeds aMax_Receive_MSDU_Lifetime than all received fragments are discarded by the destination station.~~

~~The attributes are illustrated in Figure 5-4.~~

~~**Figure 5-4: MPDU and MSDU Definitions**~~

When a frame is received from the LLC with a MSDU size greater than aFragment_~~Payload~~*Threshold*, the frame must be fragmented. The MSDU is divided into MPDUs. Each MPDU is a fragment with a frame body no larger than aFragment_~~Payload~~*Threshold*. It is possible than any fragment may contain a frame body smaller than aFragment_*Threshold*~~Payload~~. An illustration of fragmentation is shown in Figure 5-5.

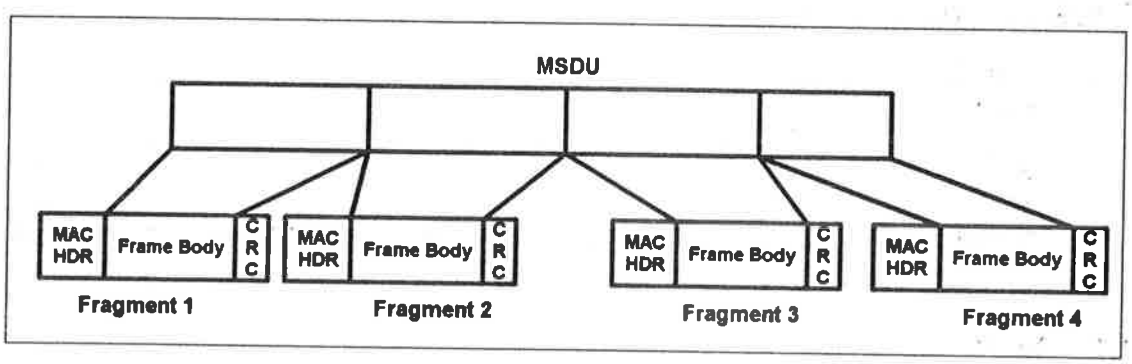


Figure 5-5: Fragmentation

5.1.5. MAC Data Service

~~*[The following sections need a close look to be sure that there is no ambiguity in translating service requests into state machine signals and vice versa - Bob]*~~

The MAC Data Service shall translate MAC service requests from LLC into *inputs*~~signals~~ utilized by the MAC State Machines. It shall also translate *outputs*~~signals~~ from the MAC State Machines into service indications ~~and confirmations~~ to LLC. The translations are given below.

The MA_DATA.request from LLC shall initiate one of the transmit cycles in the MAC State Machine. The psuedo-code below shall be used to translate this request into ~~particular~~ *inputs*~~signal indications~~ to the MAC State Machine.

```
Tx_data_req = { requested_service_class = async  &  length(MSDU) > RTS_threshold
                & destination_address <> (broadcast | multicast) }
~~Tx_broadcast_req = { requested_service_class = async  &  destination_address = broadcast }~~
Tx_multicast_req = { requested_service_class = async  &  destination_address = multicast }
Tx_unitdata_req = { requested_service_class = async  &  length(M-SDU) < RTS_threshold}
DA = { destination_address }
Length = { Rate_factor * ( length(MSDU) + Overhead ) }
~~Type = { ??? }~~
~~Control = { ??? }~~
Connection ID = integer. Note a value of zero is reserved for all asynchronous data requests
```

~~The MAC Data Service shall translate signals from the MAC State Machine to MA_DATA.confirmation as shown in the psuedo-code below.~~

~~transmission_status = { !Tx_failed }~~

The MAC Data Service shall translate *outputs*~~signals~~ from the MAC State Machine to MA_DATA.indication as shown in the following pseudo-code.

```
control = { type,control }
destination_address = { DA }
source_address = { SA }
M-SDU = { info_field }
reception_status = { !(CRC_error | Format_error) }
Connection ID = integer. Note a value of zero is used when there is no connection. ~~reserved for all
    asynchronous data requests~~
```

~~**5.1.6. Connection Control Service**~~

~~Start Connection Request with parameters of maximum length of MSDU size (= integer) and mean data interval (= integer in ms).~~

~~There are two possible corresponding indications:~~

~~Connection not granted indication.~~

~~Connection granted indication with parameter user connection ID (= integer), maximum length of MSDU size (= integer) and mean data interval (= integer in ms).~~

5.1.7. MAC Management Service

The MAC Management Service shall translate MAC service requests from external management entities into *inputs*signals utilized by the MAC Management State Machines. It shall also translate *outputs*signals from the MAC Management State Machines into service indications and confirmations to external management entities. The translations are given below.

The MAC Management Service shall translate a SM_MA_DATA.request from an external management entity as defined in the following psuedo-code.

```
Tx_data_req = { requested_service_class = async & length(M_SDU) < RTS_threshold
                & destination_address <> (broadcast | multicast) }
Tx_broadcast_req = { requested_service_class = async & destination_address = broadcast }
Tx_multicast_req = { requested_service_class = async & destination_address = multicast }
Tx_unitdata_req = { requested_service_class = async & length(M_SDU) > RTS_threshold}
DA = { destination_address }
Length = { Rate_factor * ( length(M_SDU) + Overhead ) }
Type = { ??? }
Control = { ??? }
```

The MAC Management Service shall translate signals from the MAC State Machine to SM_MA_DATA.confirmation as shown in the psuedo-code below.

```
transmission_status = { !Tx_failed }
```

[This should be qualified with the management frame type - Bob]

The MAC Management Service shall translate *outputs*signals from the MAC State Machine to SM_MA_DATA.indication as shown in the following psuedo-code.

```
control = { type,control }
destination_address = { DA }
source_address = { SA }
M_SDU = { info_field }
reception_status = { !(CRC_error | Format_error) }
```

[This also should be qualified with the management frame type - Bob]

5.2. Distributed Coordination Function

The basic medium access protocol is a Distributed Coordination Function (DCF) that allows for automatic medium sharing between *compatible*~~similar and dissimilar~~ PHYs through the use of CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance) and a random backoff time following a busy medium condition. In addition, all directed traffic uses immediate positive acknowledgements (ACK frame) where retransmission is scheduled by the sender if no ACK is received.

The CSMA/CA protocol is designed to reduce the collision probability between multiple stations accessing a medium, at the point where they would most likely occur. Just after the medium becomes free following a busy medium (as indicated by the CS function) is when the highest probability of a collision occurs. This is because multiple stations could have been waiting for the medium to become available again. This is the situation where a random backoff arrangement is needed to resolve medium contention conflicts.

Carrier Sense shall be performed both through physical and virtual mechanisms.

The virtual Carrier Sense mechanism is achieved by distributing medium busy reservation information through an exchange of special ~~small~~ RTS and CTS (medium reservation) frames prior to the actual data frame. *For stations & all AP's that do not initiate an RTS/CTS sequence, the duration information is also available in all data frames.* The RTS and CTS frames contain a duration field *that defines*~~for~~ the period of time that the medium is to be reserved to transmit the actual data frame *and the returning Ack.* This information is distributed to all stations within detection range of both the transmitter and the receiver, so also to stations that are possibly "hidden" from the transmitter but not from the receiver. This scheme can only be used for directed frames. When multiple destiniations are addressed by broadcast/multicast frames, then this mechanism is not used.

It can also be viewed as a Collision Detection mechanis*m.*~~m, b~~ *B*ecause the actual data frame is only transmitted when a proper CTS frame is received in response to the RTS frame, *this* result*s*~~ing~~ in a fast detection of a collision if it occurs on the RTS.

However the addition of these frames will result in extra overhead, which is especially *impacts*~~considerable for~~ short *data* frames. Also since all stations will likely be able to hear traffic from the AP but may not hear the traffic from all stations within a BSA~~, its use may be benificial for inbound traffic only~~.

However in situations where multiple BSS's utilizing the same channel do overlap, then the medium reservation mechanism will work accross the BSS boundaries, when RTS/CTS is also used for all traffic.

~~Therefore t~~*T*he use of the RTS/CTS mechanism is under control of RTS_Threshold attribute. *If the payload length of an MPDU is not less than this threshold, the MPDU will be send following an RTS/CTS exchange.*~~, which indicates the payload length under which the data frames should be send without any RTS/CTS prefix.~~

This parameter is a manageable object and can be set on a per station basis. This mechanism allows stations to be configured to use RTS/CTS either always, never or only on frames longer then a specified payload length.

Although a station can be configured not to initiate RTS/CTS to transmit its frames, every station shall respond to the duration information in the RTS/CTS frames to update its virtual Carrier Sense mechanism, and respond with a proper CTS frame in response to an addressed RTS frame.

The basic medium access protocol allows for stations supporting different set of rates to coexist, this is achieved by the fact that all stations are required to be able to receive any frame transmitted on a given

Basic Rate S~~set of rates~~., and must be able to transmit at (at least) one of these rates. All Multicast, Broadcast and Control frames (RTS, CTS and ACK) are always transmitted at one of this mandatory rates. This set of restrictions will assure that the Virtual Carrier Sense Mechanism described above will still work on multiple rate environments.

5.2.1. Physical Carrier Sense Mechanism

A physical carrier sense mechanism shall be provided by the PHY. See Section 8, Physical Service Specification for how this information is conveyed to the MAC. The details of carrier sense are provided in the individual PHY specification sections.

5.2.2. Virtual Carrier Sense Mechanism

A virtual carrier sense mechanism shall be provided by the MAC. This mechanism is referred to as the Net Allocation Vector(NAV). The NAV maintains a prediction of future traffic on the media based on duration information that is announced in RTS/CTS frames prior to the actual exchange of data. *The duration information is also available in all data and Ack frames* The mechanism for setting the NAV is described in 5.2.6.4

5.2.3. MAC-Level Acknowledgments

To allow detection of a lost or errored frame an ACK frame shall be returned to the source STA by the destination STA immediately following a successfully received frame. Success shall be determined by an identical CRC generated from the received frame and the FC field of the same frame. The gap between the received frame and the ACK frame shall be the SIFS. This technique is known as positive acknowledgement.

All directed frames of type Management and Asynchronous Data shall be acknowledged. The only exception is that under certain circumstances the PowerSavePOLL Control frame may be acknowledged instead of responding with the requested data.

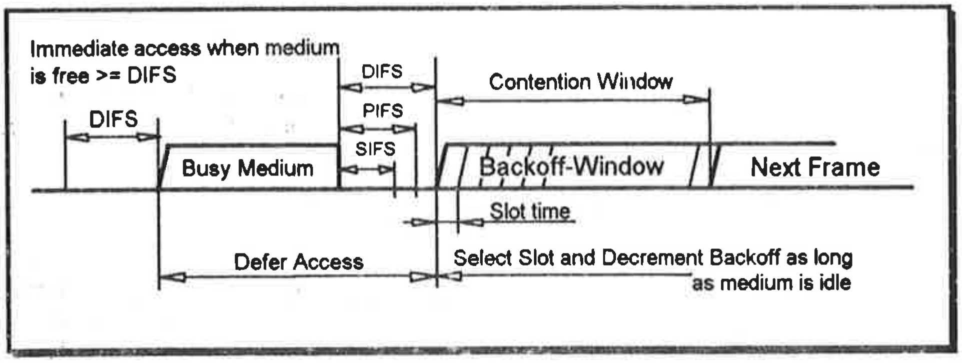
The lack of an ACK frame from a destination STA on any of the listed frame types shall indicate to the source STA that an error has occurred. Note however, that the destination STA may have received the frame correctly and the error has occurred in the ACK frame. This condition shall be indistinguishable from an error occurring in the initial frame.

5.2.4. Inter-Frame Space (IFS)

The time interval between frames is called the inter-frame space. A STA shall determine that the medium is free through the use of the carrier sense function for the interval specified. Three different IFS's are defined so as to provide a corresponding number of priority levels for access to the wireless media. *Three different IFSs are defined:*

- *SIFS* *Short Interframe Space*
- *PIFS* *PCF Interframe Space*
- *DIFS* *DCF Interframe Space*

It should be noticed that the different IFSs are independent of the station bitrate, and are fixed per each PHY (even in multi-rate capable PHYs), *The IFS timings are defined as time gaps on the medium. The standard shall specify the relation of the relative PHY MIB parameters to achieve the specified timegaps. The timing tolerances are specified in an explanatory section of the MIB.*



New Figure to illustrate IFS relations.

5.2.4.1. Short-IFS (SIFS)

This inter-frame space shall be used for an ACK frame, a CTS frame, *a Data frame of a fragmented MSDU, and,* by a STA responding to any polling as is used by the Point Coordination Function (PCF) (See Section 5.3, Point Coordination Function) ~~and between frames in the sequences described in section 4.3~~. Any STA intending to send only these frame types shall be allowed to transmit after the SIFS time has elapsed following a busy medium.

The SIFS has both a minimum and maximum specification. The maximum (SIFSmax) prevents another STA from claiming the medium and in physical terms is the maximum receive to transmit (R2T) turn around time allowed by the specific PHY. The transmit to receive (T2R) time need not be specified because it is only related to the stability of a specific implementation. Clearly T2R must be less than or equal to SIFSmax.

The minimum time (SIFSmin) prevents a STA from getting onto the medium too soon for another STA to process the transition. This minimum time may be very short. It is related to the need by a STA to see a minimum number of preamble bytes, so the length of the preamble needs to accommodate the difference between the minimum and maximum allowable SIFS timing. The assumption on the minimum is that the number of preamble bytes is fixed for a given PHY.

5.2.4.2. PCF-IFS (PIFS)

This PCF priority level shall be used only by the PCF to send any of the Contention Free Period (CFP) frames. The PCF shall be allowed to transmit after it detects the medium free for the period PIFS (PCF Interframe Space), at the start of and during a CF-Burst .

5.2.4.3. DCF-IFS (DIFS)

The DCF priority level shall be used by the DCF to transmit asynchronous MPDUs. A STA using the DCF shall be allowed to transmit after it detects the medium free for the period DIFS~~,~~ *and its backoff time has expired* ~~as long as it is not in a backoff period~~.

5.2.5. Random Backoff Time

STA desiring to initiate transfer of asynchronous MPDUs shall utilize the carrier sense function to determine the state of the media. If the media is busy, the STA shall defer until after a DIFS gap is detected, and then generate a random backoff period for an additional deferral time before transmitting. This process resolves contention between multiple STA that have been deferring to the same MPDU occupying the medium.

Backoff Time = *INT(*CW * Random()*)* * Slot time

where:

CW = An integer between CW_{min} and CW_{max}

Random() = *Pseudo random number between 0 and 1*

~~**[Need definition for Random() function. JES.]**~~

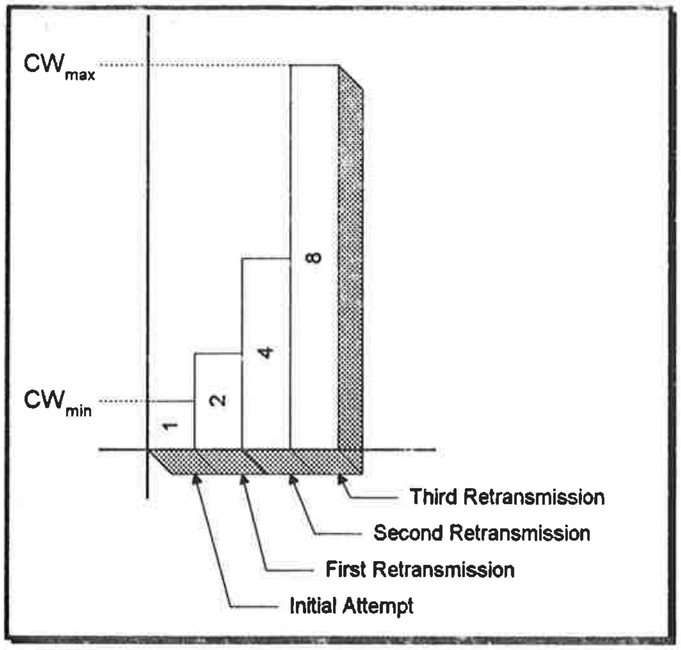
Slot Time = Transmitter turn-on delay + medium propagation delay + medium busy detect response time, *and is PHY dependent*~~.~~

~~**[Note to the PHY Group: We need numbers to help define the Slot Time. JES]**~~

The Contention Window (CW) parameter shall contain an initial value of CW_{min} for every MPDU queued for transmission. The CW shall *double at every retry until it reaches Cwmax* ~~increase exponentially after every retransmission attempt., up to a maximum value CW_{max}~~. *The CW will remain at CWmax for the remaining of the retries.* This is done to improve the stability of the access protocol under high load conditions. See Figure 5-6.

Suggested values are for: CWmin=31, Cwmax = 255.

~~**[Note: OK, I'm taking a stab at this here. Consider it a place holder for "correct" values of CW. JES.]**~~



The numbers in this figure need to be changed into 31,63,127,255 respectively, which are given as an example.

Figure 5-6: Exponential Increase of CW

CWmin and CWmax are MAC constants that should be fixed for all MAC implementations, because they effect the access fairness between stations.

~~[What happens when the number of retransmission attempts reaches the CWmax limit? Can a STA attempt transmission forever, or should we have a failure mechanism defined?. See RE-TRANSMIT_Limit in RTS/CTS Recovery Procedure section. JES.]~~

5.2.6. DCF Access Procedure

The CSMA/CA access method is the foundation of the Distributed Coordination Function. *The operational rules vary sligthly between Distributed Coordination Function and Point Coordination function.*~~This access method shall be used when there is no PCF detected and when in the Contention Period of a Superframe when using a PCF.~~

~~Initial transmission or access to the media using the DCF is divided into two cases; when the media has been free for greater than or equal to a DIFS and when it has not. Initial transmission is defined as an attempt to transfer an RTS, Data, Poll, Request or Response MPDU for the first time.~~

5.2.6.1. Basic Access

Basic access refers to the core mechanism a STA uses to determine whether it has permission to transmit.

Both the Physical and Virtual Carrier Sense functions are used to determine the busy state of the medium. When either of them indicate a busy medium, the medium shall be considered busy. The opposite of a busy medium shall be known as a free medium.

A STA with a pending MPDU may transmit when it detects a free medium for greater than or equal to a DIFS time. This rule applies both when using the DCF access method exclusively and when using the PCF access method in the Contention Area.

If the medium is busy when a STA desires to initiate a *RTS,* Data, Poll, *and Management,* ~~Request or Response~~ MPDU transfer, and only a DCF is being used to control access, the Random Backoff Time algorithm shall be followed.

Likewise, if the medium is busy when a STA desires to initiate a *RTS,* Data, Poll, *and Management*~~Request or Response~~ MPDU transfer, and a Contention Period portion of a Superframe is active (See 5.3 PCF), the Random Backoff Time algorithm shall be followed.

The basic access mechanism is illustrated in the following diagram.

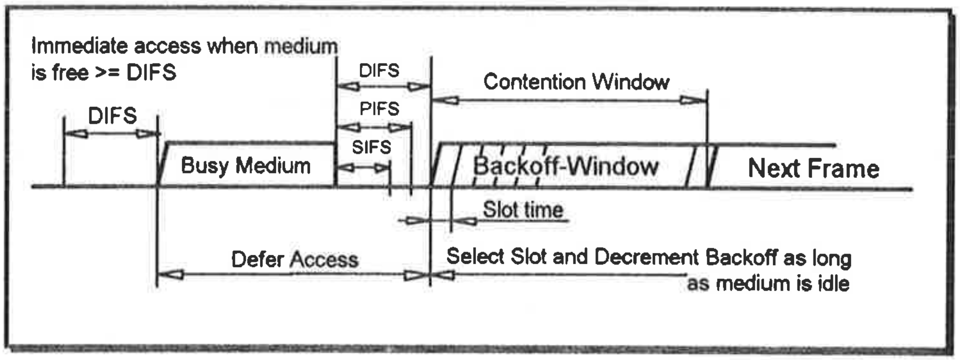


Figure 5-7: Basic Access Method

5.2.6.2. Backoff Procedure

The backoff procedure shall be followed whenever a STA desires to transfer an MPDU and finds the medium busy.

The backoff procedure consists of selecting a backoff time from the equation in Section 5.2.5 Random Backoff Time. The Backoff Timer shall decrement *by slottime amount after every slottime, while*~~only when~~ the medium is free. The Backoff Timer shall be frozen while the medium is sensed busy. Decrementing the Backoff Timer shall resume whenever a medium free period longer than DIFS is detected. Transmission shall commence whenever the Backoff Timer reaches zero.

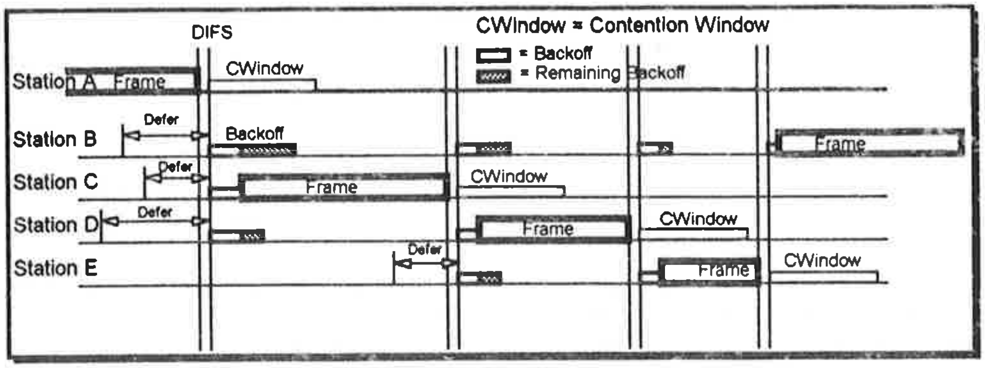


Figure 5-8: Backoff Procedure

A station that has just transmitted a *MSDU*~~frame~~ and has another *MSDU*~~frame~~ ready to transmit (queued), shall perform the backoff procedure. This requirement is intended to produce a level of fairness of access amongst STA to the medium.

The effect of this procedure is that when multiple stations are deferring and go into random backoff, then the station selecting the lowest delay through the random function will win the contention. The advantage of this approach is that stations that lost contention will defer again until after the next DIFS period, and will then likely have a shorter backoff delay than new stations entering the backoff procedure for the first time. This method tends toward fair access on a first come, first served basis.

5.2.6.3. RTS/CTS Recovery Procedure and Retransmit Limits

Many circumstances may cause an error to occur in a RTS/CTS exchange.

For instance, CTS may not be returned after the RTS transmission. This can happen due to a collision with another RTS or a DATA frame, or due to interference during the RTS or CTS frame. It can however also be that CTS fails to be returned because the remote station has an active carrier sense condition, indicating a busy medium time period.

If after an RTS is transmitted, the CTS fails in any manner within a predetermined CTS_Timeout (T1), then a new RTS shall be generated while following the basic access rules for backoff. Since this pending transmission is a retransmission attempt, the CW shall be doubled as per the backoff rules. ~~This process shall continue until the RTS_RE-TRANSMIT_Counter reaches an RTS_RE-TRANSMIT_Limit~~. *This process shall continue until the aRTS_Retry_Counter reaches an aRTS_Retry_Max limit.*

~~***[Need definition of CTS_Timeout (T1), RTS_RE-TRANSMIT_Limit, ACK_RE-TRANSMIT_Limit and ACK_Window (T3). JES]***~~

The same backoff mechanism shall be used when no ACK frame is received within a predetermined ACK_Window (T3) after a directed DATA frame has been transmitted. Since this pending transmission is a retransmission attempt the CW will be greater than one as per the backoff rules. This process shall continue until the *aData_Retry_Counter*~~ACK_RE-TRANSMIT_Counter~~ reaches an *aData_Retry_Max limit* ~~ACK_RE-TRANSMIT_Limit~~.

~~***[Note that the RTS_ RE-TRANSMIT_LIMIT is expected to be set to 8-16, while the no-Ack_Retransmit_Limit is expected to be set to 2-4.: from 93/190, JES]***~~

[Is there any requirement for interaction between RTS and ACK retransmission? Does one effect the other in any way? JES]

5.2.6.4. Setting the NAV Through Use of RTS/CTS Frames

In the absence of a PCF, reception of RTS and CTS, *Data and Ack* frames are the only events that shall set the NAV to a non-zero duration. Various conditions may reset the NAV.

RTS and CTS frames contain a Duration field based on the medium occupancy time of the MPDU from the end of the RTS or CTS frame until the end of the ACK frame. (See Section 4: RTS and CTS Frame Structure.) All STA receiving a these frame types with a valid FCS field *but with the exception of the station that is addressed* shall interpret the duration field in these frames, and maintain the Net Allocation Vector (NAV). *Stations receiving a valid frame should update their NAV with the information received in the Duration field, but only when the new NAV value is greater then the current NAV value.*

Maintenance of the NAV shall consist of an internal state accurate to *1 microsecond*X nS of the busy/free condition of the medium. Figure 5-9 indicates the NAV for stations that can hear the RTS frame, while other stations may only receive the CTS frame, resulting in the lower NAV bar as shown. Although the NAV effectively will "count-down" from a non-zero value, only the fact of whether the NAV is non-zero or not is necessary for correct protocol operation.

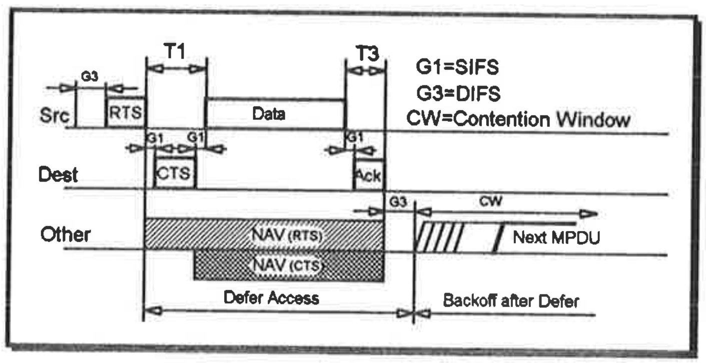


Figure 5-9: RTS/CTS/DATA/ACK MPDU

5.2.6.5. Control of the Channel

The Short Interframe Space (IFS) is used to provide an efficient MSDU delivery mechanism. Once a station has contended for the channel, it will maintain control of the channel until it has sent all of the fragments of a MSDU, *and received their corresponding Acks, or until it failed to receive an Ack for a specific fragment*. After all fragments have been transmitted, the station will relinquish control of the channel.

Once the station has contended for the channel, it will continue to send fragments until either all fragments of a MSDU have been sent, an acknowledgment is not received, or the station can not send any additional fragments due to a dwell time boundary.

Figure 5-10 illustrates the transmission of a multiple fragment MSDU using the IFS.

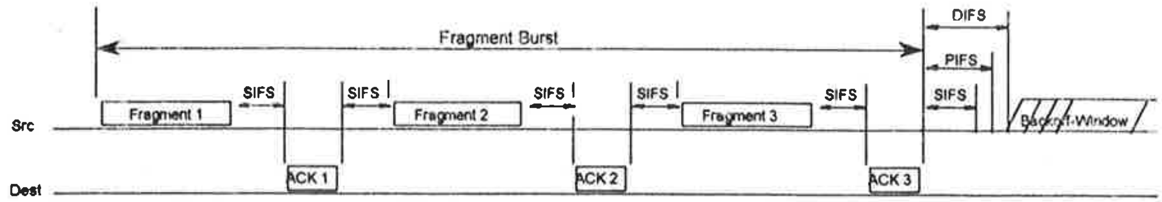


Figure 5-10: Transmission of a Multiple Fragment MSDU using IFS

The source station transmits a fragment then releases the channel and waits for an acknowledgment. When the source station releases the channel following its fragment, it will immediately monitor the channel for an acknowledgment frame from the destination station.

When the destination station has finished sending the acknowledgment, the SIFS following the acknowledgment is then reserved for the source station to continue (if necessary) with another fragment. The station sending the acknowledgment does not have permission to transmit on the channel immediately following the acknowledgment.

The process of sending multiple fragments after contending for the channel is defined as a fragment burst.

If the source station receives an acknowledgment but there is not enough time to transmit the next fragment and receive an acknowledgment due to an impending dwell boundary, it will contend for the channel at the beginning of the next dwell time.

If the source station does not receive an acknowledgment frame, it will attempt to retransmit the ~~fragment at a later time (~~according to the backoff algorithm~~)~~. When the time arrives to retransmit the fragment, the source station will contend for access in the contention window.

After a station contends for the channel to retransmit a fragment of a MSDU, it will start with the last fragment that was not acknowledged. The destination station will receive the fragments in order since the source sends them one at a time, in order. It is possible however, that the destination station may receive duplicate fragments. This will occur if the destination station sends an acknowledgment and the source does not receive it. The source will resend the same fragment after executing the backoff algorithm and contending for the channel.

A station will transmit after the SIFS only under the following conditions during a fragment burst:

> The station has just received a fragment that requires acknowledging.
>
> The source station has received an acknowledgment to a previous fragment, has more fragment(s) for the same MSDU to transmit, and there is enough time left in the dwell time to send the next fragment & receive an acknowledgment.

The following *rules*~~guidelines~~ also apply.

> When a station has transmitted a frame other than a fragment, it *may not* ~~does not have priority to~~ transmit on the channel following the acknowledgment for that frame, *without going through a backoff.*.
>
> ~~When a MSDU has been successfully delivered, the station does not have priority to transmit on the channel following the last acknowledgment of the last fragment.~~

When a MSDU has been succesfully delivered, and want to transmit a subsequent MSDU, then it should go through a backoff.

Only unacknowledged fragments are retransmitted.

If a multiple fragment MSDU does not require an acknowledgment (for example, a broadcast/multicast packet transmitted by the Access Point), the source station will transmit all fragments of the MSDU without releasing the channel as long as there is enough time left in the dwell time. If there is not, the station will transmit as many fragments as possible and recontend for the channel during the next dwell time.*The spacing between fragments of a broadcast/multicast frame shall be equal to the SIFS period.*

5.2.6.6. RTS/CTS Usage with Fragmentation

The following is a description of using RTS/CTS for *the first fragment of* a fragmented MSDU. *RTS/CTS will also be used for retransmitted fragments if their size warrents it.* The RTS/CTS frames define the duration of the first frame and acknowledgment. The duration field in the data and acknowledgment frames specifies the total duration of the next fragment and acknowledgment. This is illustrated in Figure 5-11.

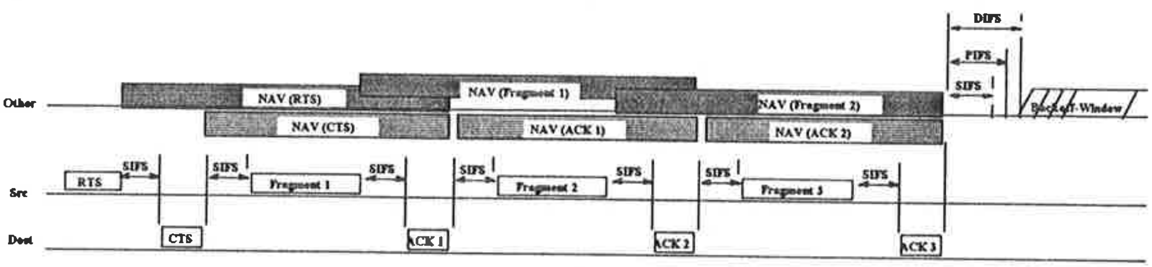


Figure 5-11: RTS/CTS with Fragmented MSDU

Each frame contains information that defines the duration of the next transmission. The RTS will update the NAV to indicate busy until the end of ACK 1. The CTS will also update the NAV to indicate busy until the end of ACK 1. Both Fragment 1 and ACK 1 will update the NAV to indicate busy until the end of ACK 2. This is done by using the duration field in the DATA and ACK frames. This will continue until the last Fragment and ACK which will have the duration set to zero. Each Fragment and ACK acts as a virtual RTS and CTS, *therefore no RTS/CTS frame needs to be generated even though subsequent fragments are larger the aRTS_Threshold.*

In the case where an acknowledgment is not received by the source station, the NAV will be marked busy for next frame exchange. This is the worst case situation. This is shown in Figure 5-12. If the acknowledgment is not sent by the destination station, stations that can only hear the destination station will not update their NAV and be free to access the channel. All stations that hear the source will be free to access the channel after the NAV from Frame 1 has expired.

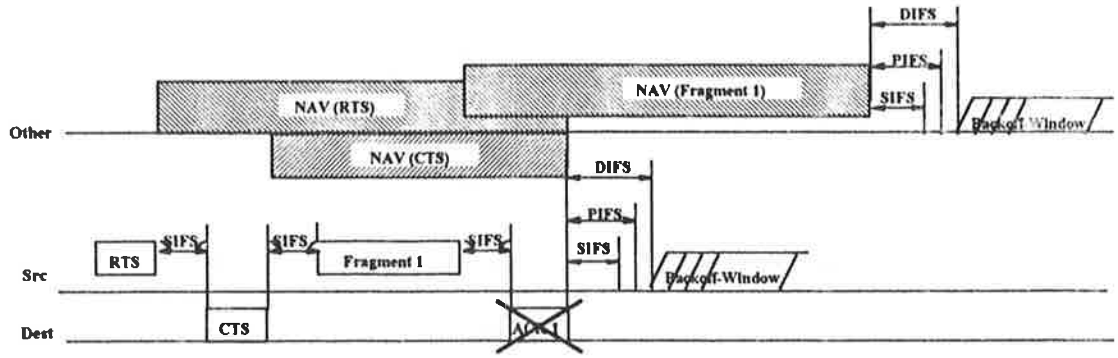


Figure 5-12: RTS / CTS with Transmitter Priority with Missed Acknowledgment

The source station must wait until the *Ack timeout*~~NAV (Fragment 1) expires~~ before attempting to contend for the channel after not receiving the acknowledgment.

5.2.7. Directed MPDU Transfer Procedure Using RTS/CTS

Figure 5-9 shows the Directed MPDU transfer procedure with the use of RTS/CTS. In certain circumstances the DATA frames will be preceded with an RTS and CTS frame exchange that include duration information.

STA shall use an RTS/CTS exchange for directed frames only when the length of the MPDU is greater than the length threshold indicated by the RTS_Threshold attribute. The RTS_Threshold attribute shall be set to a MPDU length threshold in each STA.

The RTS_Threshold attribute shall be a managed object within the MAC MIB, and its value can be set and retrieved by the MAC LME. The RTS_Threshold attribute shall be constrained to range (0 ... Maximum MPDU Length). The value 0 shall be used to indicate that no MPDU shall be delivered without the use of *RTS/CTS. Values of RTS_Threshold ≥ MDPU_Maximum shall indicate that all MPDU shall be delivered with RTS/CTS.*

~~This parameter provides control over both direction and length since it is maintained on a per STA basis. So as an example, only inbound long frames may use the RTS/CTS option.~~

The asynchronous payload frame (e.g. DATA) shall be transmitted after the end of the CTS frame and an SIFS gap period. No regard shall be give to the busy or free status of the medium.

5.2.7.1. Directed MPDU Transfer Procedure without RTS/CTS

Following the basic access mechanism, the source STA shall transmit the asynchronous payload frame (e.g. DATA). The destination STA shall follow the ACK Procedure.

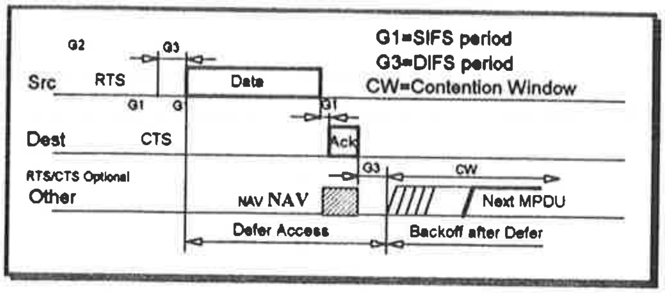


Figure 5-13: Directed Data/ACK MPDU

~~**[Note to PHY group contained in 93/190: Note: It is desirable that the PHY has an "End Delimiter" detection means, so that the end of the "Medium Busy" can be detected bit-synchronously, minimizing detection delay. ED: This would effect the SIFS, etc. gap times.]**~~

5.2.8. Broadcast and Multicast MPDU Transfer Procedure

In the absense of a PCF, when Broadcast or Multicast MPDUs are transferred from an AP to a STA, or from one STA to other STA's, only the basic access mechanism shall be used. Regardless of the length of the frame, no RTS/CTS exchange shall be used. In addition, no ACK shall be transmitted by any of the receipients of the frame.

Any Broadcast or Multicast MPDUs transferred from a STA ~~to an AP~~ *with a To_DS bit set* shall, in addition to conforming to the basic access mechanism of CSMA/CA, obey the rules for RTS/CTS exchange. In addition, the AP shall transmit an ACK frame in response just as if the MPDU were directed traffic.

Multicast MSDUs shall be propagated throughout the ESS.~~The scope of a multicast is an ESS~~.

There is no MAC level recovery on Broadcast or Multicast frames except for those frames sent *with an To_DS bit set*~~via an AP.~~ As a result the reliability of this traffic is reduced due to the increased probability of lost frames from interference or collisions.

5.2.9. ACK Procedure

Upon successful reception of a data or management frame with the ToAP bit set, an AP shall always generate an ACK frame. An ACK frame shall be transmitted by the destination STA which is not an AP whenever it successfully receives a unicast data frame or management frame, but not if it receives a broadcast or multicast data frame. The transmission of the ACK frame shall commence after an SIFS period without regard to the busy/free state of the medium.

The Source STA shall wait an Ack_timeout amount of time *without receiving an Ack frame* before concluding that the MPDU failed.

~~**[Should there be a specification of how long the Source STA waits for the ACK? Something like SIFS + err. tollerance.? (YES. There is Ack_timeout in the state machine diagrams. Bob) JES]**~~

This policy induces some probability that a pending frame in a neighboring BSA (using the same channel) could be corrupted by the generated ACK. However if no ACK is returned because a busy medium was detected, then it is guaranteed that the frame would be interpreted as in error due to the ACK timeout, resulting in a retransmission.

6.2.10. DCF Pseudo Code Example

[These have not been edited for accuracy. Please check. JES]

The following rules need to be applied when transmitters use the DCF Asynchronous Services.

[I suppose these timers, T1 and T3 should be specified? JES]

[Timers are specified:

T1: CTS_timeout

T3: ACK_timeout

Bob.]

When transmitting a unicast MPDU using RTS/CTS exchange:
If medium is free longer then DIFS, then transmit RTS.
Else defer until DIFS gap is detected, and go into backoff.
If CTS is received within T1 after RTS, then transmit the DATA after SIFS.
Else go into Retransmit_Backoff.
If Ack not received within T3 then go into Retransmit_Backoff.

When transmitting a unicast MPDU without the RTS/CTS exchange:
If medium is free longer then DIFS, then transmit DATA.
Else defer until DIFS gap is detected, and go into backoff.
If Ack not received within T3 then go into Retransmit_Backoff.

When transmitting a Broadcast/Multicast MPDU:
If medium free (No NAV and no CS) longer then DIFS, then transmit DATA.
Else defer until DIFS gap is detected, and go into backoff.

The following rules need to be applied by receiving stations:

If RTS frame is detected but station is not the destination, Then:
Update the NAV with the Duration information and start a T1 timer.
Else
Return a CTS frame when medium free (no NAV and no CS) after SIFS.

If T1 timer expires, and CS is not active at that time, then clear the NAV.

If CTS frame is detected Then: Update the NAV with the Duration information.

If station is the destination of a unicast DATA frame, Then:

~~Transmit Ack after SIFS when CRC was correct.~~

5.2.11. Duplicate Detection and Recovery

Since MAC-level acknowledgments and retransmissions are incorporated into the protocol, there is the possibility that a frame may be received more than once. Such duplicate frames shall be filtered out within the destination MAC.

Duplicate frame filtering is facilitated through the inclusion of a dialog token (consisting of a sequence number and fragment number) field within DATA and MANAGEMENT frames. MPDUs which are part of the same MSDU shall have the same sequence number, and different MSDUs will (with a very high probability) have a different sequence number.

The sequence number is generated by the transmitting station as an incrementing sequence of numbers.

The receiving station shall keep a cache of recently-received <source-address, sequence-number, fragment-number> tuples.

A destination STA shall reject a frame which has the RETRY bit set in the CONTROL field as a duplicate if it receives one which matches both source-address, sequence-number and fragment-number in the cache .

There is the small possibility that a frame will be improperly rejected due to such a match; however, this occurrence would be rare and would simply result in a lost frame similar to an FCS error in ethernet.

The Destination STA shall perform the ACK procedure even if the frame is subsequently rejected due to duplicate filtering.

~~Since MAC-level acknowledgments and retransmissions are incorporated into the protocol, there is the possibility that a frame may be received more than once. Such duplicate frames shall be filtered out within the destination MAC.~~

~~Duplicate frame filtering is facilitated through the inclusion of an **MPDU ID** field within the individual frames of an MPDU, including the DATA and ACK frames. Frames which are part of the same MPDU shall have the same ID, and different MPDUs will (with a very high probability) have a different ID.~~

~~The MPDU ID is a 16 bit hash of the 2 octet Network ID field, 6 octet source address and a 1 octet sequence number maintained by the source STA. The hashing of this information into a smaller field reduces overhead, particularly within ACK frames.~~

~~***[Hash algorithm should be defined. JES]***~~

~~A destination STA shall reject a frame which has the RETRY bit set in the CONTROL field as a duplicate if it receives one which matches a value of recent MPDU IDs kept in the MPDU_ID_CACHE. The MPDU_ID_CACHE shall keep the last X MPUD IDs on a FIFO basis for the purpose of comparison with the most recent MPDU ID.~~

~~***[Do we need to specify the depth of the MPDU_ID_CACHE to achieve interoperability??? Implication: If not, don't overspecify. JES]***~~

~~There is the small possibility that a frame will be improperly rejected due to such a match; however, this occurrence would be rare and would simply result in a lost frame similar to an FCS error.~~

~~Destination STA shall perform the ACK procedure even if the frame is subsequently rejected due to duplicate filtering.~~

~~*[The following paragraph should be discussed by the group. JES]*~~

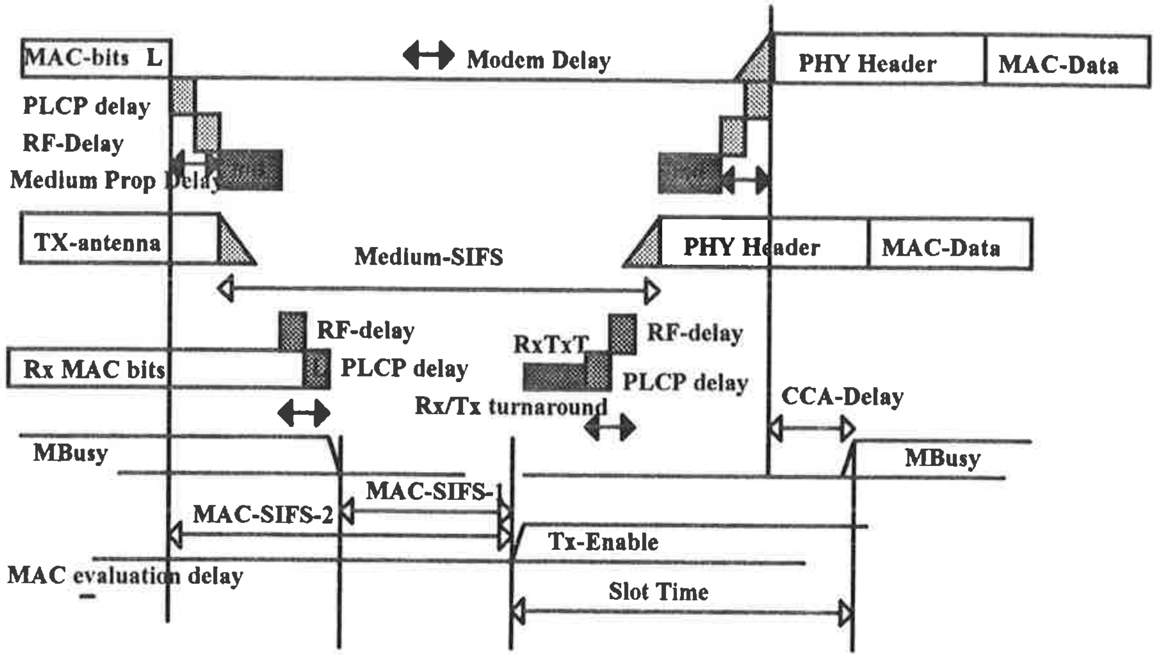
~~5.2.12. Fast Response Possibility~~

~~Note that instead of an Ack frame, it is also possible to directly transmit the response frame back to the transmitter of the received frame. This would allow a class of fast implementations, which could for instance directly respond to a Poll frame with the requested Data frame itself, which in turn should be acknowledged by an Ack frame.~~

~~Another example is in the Contention Free (CF) period, where stations respond to a Poll bit in frames coming from the AP.~~

5.2.13. *DCF Timing relations*

This section formulates the relation between the IFS specifications as have been defined as time gaps on the medium, and the associated MIB variables that are provided per PHY.



~~Distributed Time Bounded Service (DTBS)~~

~~An optional Distributed Time Bounded Service (DTBS) may be based on the connectionless-mode MAC Service provided by the DCF. DTBS can be characterized as a "best effort" service providing bounded transit delay and delay variance.~~

~~DTBS requires a mechanism to map requested Quality of Service (QoS) onto channel access priority. QoS parameters include transit delay, delay variance, and user priority. If the MAC Service user does not explicitly state QoS parameters, the MAC Service provider shall use default values. MAC Service requests that cannot be satisfied are rejected by the MAC Service provider, thus avoiding overload conditions.~~

~~DTBS assumes that the MAC Service provides multiple hierarchically independent levels of channel access priority. Hierarchical independence means that increasing load from lower priority classes does not degrade the performance of higher priority classes.~~

~~[We may wish to promote time bounded service to a second level heading in order to provide sufficient levels of description. - KEL]~~

~~5.2.13.1. Quality of Service~~

~~Associated with each MAC connectionless-mode transmission, certain measures of QoS are requested by the sending MAC Service user when the primitive action is initiated. The requested measures (or parameter values and options) are based on a priori knowledge by the MAC Service user of the service(s) made available to it by the MAC Service provider. Knowledge of the characteristics and type of service provided (i.e., the parameters, formats, and options that affect the transfer of data) is made available to the MAC Service user through some layer management interaction prior to (any) invocation of the MAC connectionless-mode service. Thus the MAC Service user not only has knowledge of the characteristics of~~

~~the parties with which it can communicate, it also has knowledge of the statistical characteristics of the service it can expect to be provided with for each MAC Service request.~~

~~5.2.13.1.1. Transit Delay~~

~~Transit delay is the elapsed time between MA-UNITDATA.request primitives and the corresponding MA-UNITDATA.indication primitives. Elapsed time values are calculatetd only on MSDUs that are transferred successfully.~~

~~Successful transfer of a MSDU is defined to occur when the MSDU is transferred from the sending MAC Service user to the intended receiving MAC Service user without error.~~

~~For connectionless-mode transfer, transit delay is specified independently for each MAC connectionless-mode transmission. In general, satisfaction of the transit delay bound is managed by the sender.~~

~~5.2.13.1.2. Delay Variance~~

~~Delay variance is the jitter associated with transit delay. In general, satisfaction of the delay variance bound is managed by the receiver and may be used to regenerate the regular periodic interval of related sequences of MSDUs.~~

~~5.2.13.1.3. User Priority~~

~~The MAC Service user may transfer to the MAC Service provider a priori knowledge about the characteristics of the parties with which it can communicate via the user priority QoS parameter.~~

~~5.2.13.2. Mapping QoS onto Channel Priority~~

~~There is a standardized mapping of QoS Transit Delay and Delay Variance parameters to initial Time to Live (TTL). The initital transmit queue position is determined by TTL, possibly qualified by the QoS User Priority parameter. All MSDUs in the transmit queue count down their associated TTL while waiting to reach the head of the queue and be dequeued for transmission.~~

~~The channel access priority is determined, in a standardized way, from remaining TTL at dequeue time. At transmission time, the measured queue delay must be subtracted from the TTL to give the Residual Time to Live (RTL) i.e. the time left before the MSDU becomes out of date. RTL may be used in subsequent handling of the MSDU. If RTL should become less than or equal to zero, the MSDU should in all cases be discarded.~~

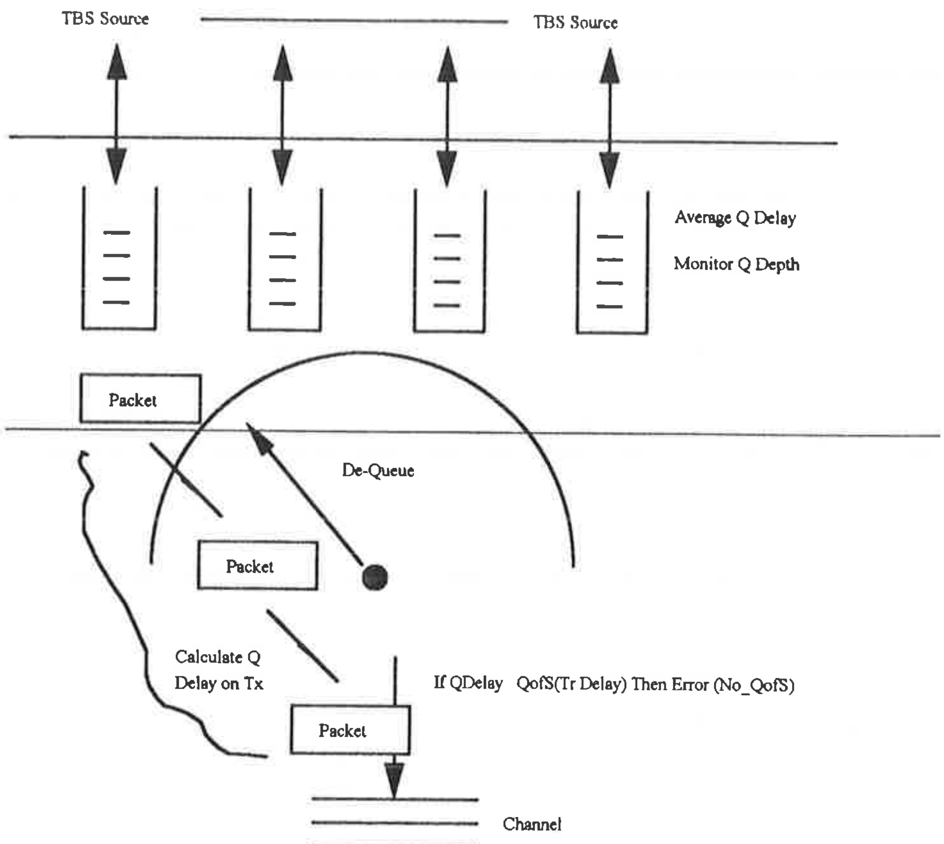


Figure 5-14: Mapping QoS onto Channel Access Priority

[The queueing behavior described could potentially be placed above the MAC pending decision of the 802.11 Plenary . - KEL]

5.2.13.3. Partitioning of Channel Capacity

Partitioning of channel capacity amongst conceptual user classes (e.g. low priority async requests and higher priority time bounded requests) is a natural side effect of the mapping of TTL to channel access priority at dequeue time. Since *all* queued MSDUs progress towards the head of the queue as a function of their decreasing TTL, the relationship between channel access priority and conceptual user class is a function of channel load.

5.2.13.4. Channel Access Priority Mechanism