IEEE 802.11
Wireless Access Method and Physical Layer Specifications

Title: **Fragmentation/Reassembly at the MAC layer.**

Motorola, Inc.
Wireless Data Group
50 E. Commerce Drive Suite M-1
Schaumburg, Illinois 60173

Presented By: Mark Demange
Tel: 1-708-576-7913
Fax: 1-708-538-5152
E-Mail: mark_demange@wes.mot.com

Abstract: The foundation MAC currently does not define a mechanism for fragmentation of an MSDU into multiple MAC frames. This submission presents a case for the addition of fragmentation to the foundation MAC. This submission also offers a method for integrating fragmentation into the current foundation MAC (DFWMAC)[1].

The authors would like to see a vote of YES to MAC issue 20.6 "Is there a need for fragmentation/re-assembly at the MAC layer?".

The authors would also like to see the fragmentation proposal given within this submission be adopted as a foundation for inclusion within the current foundation MAC.

It is the authors view that an 802.11 MAC standard without fragmentation can be considered broken since it seriously degrades the robustness of the MAC and its ability to operate in the presence of microwave oven interference.

1. Introduction:

The process of partitioning an MSDU into smaller MAC level frames is defined in this submission as fragmentation. The foundation MAC currently does define an implementation for fragmentation. It is the intent of this submission to address this issue (IEEE 802.11 MAC issue 20.6). Addition of fragmentation to the foundation MAC will provide the following benefits:

- Provide more efficient utilization of bandwidth by reducing the quantity of data required to be retransmitted due to interference.
- Provide more efficient utilization of bandwidth by reducing the quantity of data required to be retransmitted due to hidden stations.
- Allow optimal hopping of FH PHYs.

- Reduce or eliminate the variation in the start time of the time bounded portion of the MAC superframe.

This submission will first identify the aforementioned advantages of adding fragmentation to the current foundation MAC. A relative comparison of a system with and without fragmentation is then made for stations within a BSA. Finally, A proposal for implementation of fragmentation is made. This implementation is offered as a basis for inclusion in the MAC.

2. Advantages of fragmentation

The following discussion defines some of the advantages of including fragmentation within the MAC. It should be noted that this discussion does not address what the optimal fragmentation algorithm should be but rather that the fragmentation is necessary.

2.1 *Enhanced performance in the presence of microwave oven interference*

Wireless LAN systems operating within the ISM bands must operate in the presence of interference generated from other users of the band. One of the primary users of the 2.4 GHz ISM band is microwave ovens. Wireless LAN systems operating within this band should be designed to operate in the presence of this source of interference with minimal degradation in performance. If the characteristics of microwave oven interference and of typical data packets to be transmitted across the LAN system are considered, it is evident that a system wishing to share spectrum with the oven must fragment the end user data packets before transmitting to a station being interfered with by an oven.

Consider that a typical Ethernet packet of length 1100 bytes would result in a MAC frame of duration 8.8 ms. minimum (depending on frame overhead) at a 1 Mbps channel rate. Now consider that microwave ovens typically transmit using Pulse Amplitude Modulation with a 60 Hz frequency and 50% duty cycle. This corresponds to an 8.3 ms. ON time and an 8.3 ms. OFF time. The figures below show that the interference typically covers at least 10 MHz of the band at any instant in time.

The figures show the emissions of a microwave oven measured on two frequencies simultaneously. Three spectrum analyzers using a common trigger measured the ovens emissions at 2.450, 2,455, and 2.460 GHz simultaneously. Figure 1 shows the data at 2.450 GHz and Figure 2 shows the data at 2.460 GHz.

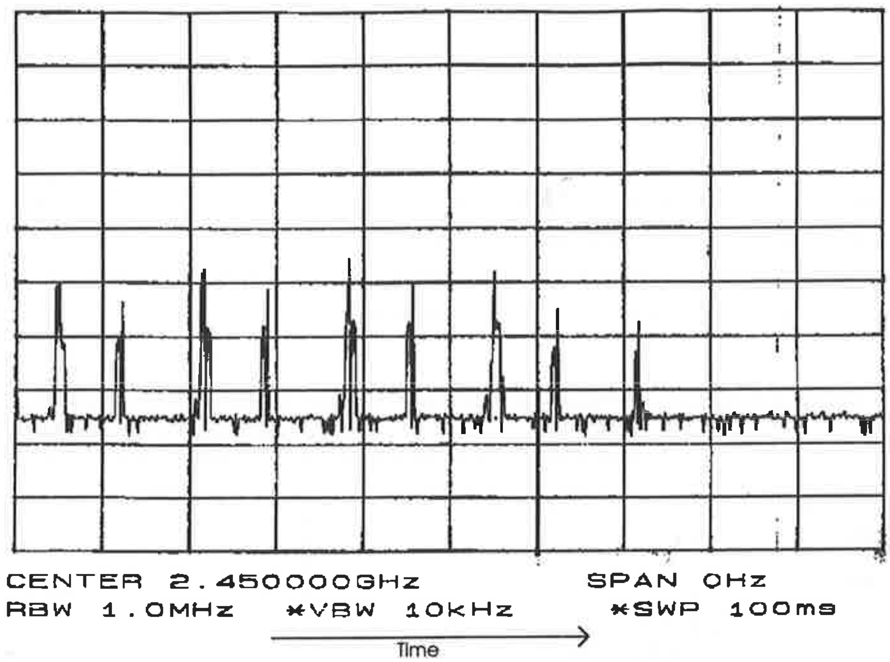


Figure 1: Microwave Oven Interference at 2.45 GHz – Time Domain

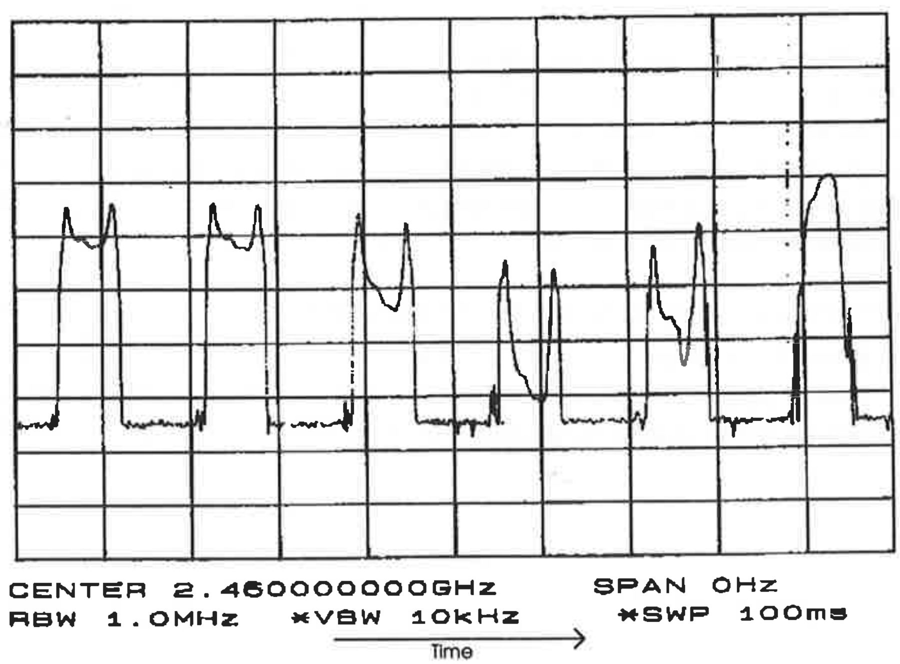


Figure 2: Microwave Oven Interference at 2.46 GHz – Time Domain

It is clear that a station under the influence of the microwave oven interference will be unable to successfully receive an 1100 byte MAC frame without errors. In effect, the receive throughput of the interfered station has dropped to zero for packets of duration 8.3 ms (1037 bytes) or greater so long as the station is operating on or near the ovens operating frequency. If the 1100 byte MAC frame is fragmented into two 550 byte MAC frames, then the receiving station now has an opportunity to receive at least half of the packet depending on the relative time between the data frames and the ON time of the oven. If the packet is delivered as three MAC frames then a minimum of one MAC frame and a maximum of two MAC frames of data can be successfully received by the receiving station.

Table 1 below shows the percentage of MAC frames from a single 1100 byte Ethernet packet that will be successfully received by a station subject to interference from an oven. Clearly, as the number of MAC frames per Ethernet packet is increased, the effective utilization of the available bandwidth between the oven's ON times is also increased.

Table 1: Fragmentation Performance With Microwave Oven Interference

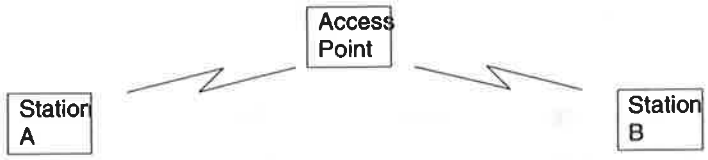
Frames per 1100 Byte Packet	% of packet received successfully during OFF time of oven 1 Mbps Data Rate	2 Mbps Data Rate
1 - no fragmentation	0%	0% to 100%
2	0 % - 50%	50% - 100%
3	33% - 66%	66% - 100%
4	50% - 75%	75% - 100%

Note that a direct sequence system will also be vulnerable to this type of interference. Depending on the actual interfered bandwidth and the signal strength of the interference relative to the desired signal, the direct sequence system will be unable to cope with the interference and a packet received during the ON time of the oven will be corrupted. Also note that the typical direct sequence system is using a 2 Mbps channel rate and will therefore need to fragment the packet into fewer MAC frames than a system operating at 1 Mbps in order to efficiently utilize the bandwidth available during the OFF times of the oven.

2.2 *Less retransmitted data in BSAs with hidden and sleeping stations*

A MAC/PHY without fragmentation is vulnerable to interference from hidden stations. The RTS/CTS mechanism employed in the foundation MAC decreases the probability of interference in many circumstances IF the other stations in the cell are awake to hear the RTS/CTS frame transaction.

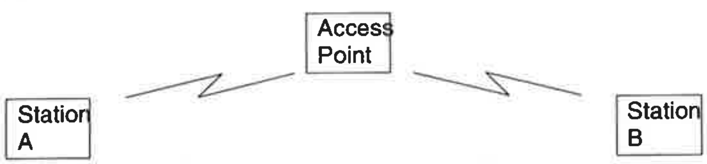
Consider the following situation without RTS/CTS employed:



Station A and B are hidden from each other and are both AWAKE
Station A starts to transmit data frame to Access Point
Station B senses channel as CLEAR (station A is hidden)
FH:
Station B transmits to AP and corrupts AP reception of data frame from station A
Station A's transmission corrupts AP reception of data frame from station B
Both stations required to retransmit
DS:
Station B transmits to AP and is not acknowledged by AP
Station B required to retransmit

Figure 3: Hidden Station Interference – Example 1

The next example shows a similar problem with RTS/CTS employed:



Station A and B are hidden from each other
Station B is sleeping
Station A transmits RTS to Access Point
Access Point transmits CTS to station A
Station A starts to transmit data frame
Station B wakes up and senses channel as CLEAR
FH:
Station B transmits to AP and corrupts AP reception of data frame from station A
Station A's transmission corrupts AP reception of data frame from station B
Both stations required to retransmit
DS:
Station B transmits to AP and is not acknowledged by AP
Station B required to retransmit

Figure 4: Hidden Station Interference – Example 2

Station B's transmission of a data frame in the above examples caused a collision at the Access Point. The duration of the collision is the time during which both Station A and Station B are simultaneously transmitting. If a collision occurs anywhere within a frame then the entire frame can be considered corrupt in a FH system. It is easy to see that if a packet is fragmented into multiple MAC frames then only the frames subject to the collision need to be retransmitted. This effectively improves the utilization of the channel.

In the case of a DS system there is a processing gain advantage which allows the AP to reject the transmission from station B. However, station B is still required to retransmit the entire data frame. Again, fragmentation will allow only the data subject to the collision to be retransmitted.

2.3 *Removes constraints on practical dwell times for FH systems*

A MAC with no provisions for fragmentation requires that all data packets are sent at full length across the channel. A full size Ethernet packet (1518 bytes) is approximately 12 ms long on a 1 Mbps channel. In order for a FH MAC/PHY to support these packet sizes a FH station must send the entire packet on one frequency prior to hopping to the next frequency. The MAC must be designed in a manner which ensures maximum utilization of the available bandwidth in each hop interval. (Note that the PHY committee passed a vote in the January 1993 meeting stating that the MAC maximize the use of bandwidth in each hop interval.) Three obvious options exist for allowing the MAC to accomplish the task of maximizing the utilization of each hop interval.

1. Use a fixed dwell time significantly longer than 12 ms.
2. Allow the dwell time to stretch when needed to allow transmission of a frame near the dwell boundary.
3. Use a fixed dwell time and dynamically fragment the packet into a frame size that maximizes use of each hop interval.

2.3.1 Fixed dwell time

The first option is to keep the dwell lengths fixed and just defer the transmission until the next dwell when there is sufficient time in the dwell to accommodate the entire packet. This causes a waste of bandwidth since the end of the dwells tend to be vacant. Additionally, in a heavily loaded BSA, the beginning of dwells become prone to collisions since many stations may have deferred their transmissions until the new dwell. The solution to the wasted bandwidth problem is to lengthen the dwell time so that the lost bandwidth is small relative to the total bandwidth available in the dwell. Given 12 ms. Ethernet packets, a dwell length of 120 ms. would still leave up to 10% of the bandwidth unutilized and still leave a tendency for collisions at the beginning of the dwell. Unfortunately, the longer dwells tend to negate the advantages of frequency hopping by requiring a station to remain on a bad channel for longer than desired. The bad channel could be due to a frequency selective fade and or interference from another user of the band. In the latter case the problem is compounded if the receiving station is subject to interference from a microwave oven. Not only is the station unable to successfully receive a frame due to the interference but the station must also remain on the bad frequency for an extended period of time.

2.3.2 Stretchable dwell time

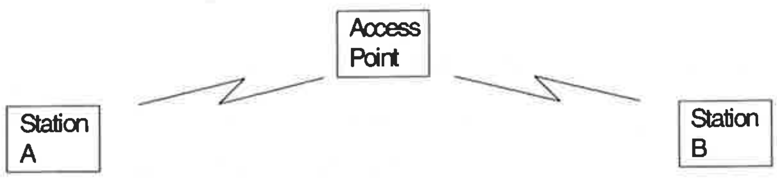
The second option is to stretch the current dwell sufficiently to allow transmission of the complete packet. This option has a three negative side effects associated with it:

1. Increased number of back-offs for stations exercising power management.
2. Degraded power management.
3. Time variation in the start of the time bounded services superframe.

It should also be noted that the January 93 PHY committee PASSED 20-1-1 the following motion: "The hop rate shall be configurable in the MAC but fixed within a given BSA. It does not have to adapt." This motion indicates that a stretchable dwell time is unacceptable from the PHY committee perspective. Nonetheless, a description of the disadvantages of such an implementation follows.

2.3.2.1 ***Increased number of back-offs for stations exercising power management.***

Stretching is undesirable since all stations sleeping during the beginning of the stretch will hop ahead to the next frequency since they were unaware of the stretch. Now some significant portion of the cell is on the new frequency while the remainder is on the old frequency. Now any of the stations electing to transmit on the new frequency are subject to backing off since it is very possible that their destination is on the old frequency. This destination in many cases will be the Access Point (client-server applications) which is still on the old frequency because it can hear the transmission causing the dwell to stretch. A likely case is that the AP is the destination of the packet which is causing the dwell to be stretched in the first place.



Station B is sleeping
Station A transmits data frame to AP on frequency 1 - stretching dwell
Station B unaware of stretch wakes up on frequency 2 and senses channel - clear
Station B transmits data frame to AP
Station B invokes backoff algorithm since AP doesn't acknowledge data frame

Figure 5: Dwell Stretch Induced Back-off

Note that if the sleeping station (station B) woke up before the hop then it could sense the channel before transmitting and avoid the back-off. However, if station B were hidden from station A, then it would not accurately sense the state of the channel and the problem would still exist.

2.3.2.2 ***Time variation in the start of the time bounded services superframe.***

Dwell stretching could also causes a variation in the start time of the time bounded portion of the MAC superframe. The foundation MAC indicates the use of a TX-Blackout COULD be used. If the blackout is used then the MAC is wasting available bandwidth by prohibiting transmissions with no information present in the frame.

If the blackout is not used then the time bounded data may need to contain additional information to account for the worst case variation in the start time of the contention free area of the superframe. For example, transmission of voice might require 60 ms. of voice be transmitted every 50 ms. The receiving station would then buffer the additional 10 ms. of voice samples to allow for a stretch of up to 10 ms. If there was a stretch then the receiving station would use the buffered data until the next frame was received. If there was no stretch then the buffered data would be thrown out and the process would repeat. Again, this is inefficient use of the bandwidth since 10 ms. of voice samples would be sent twice.

2.3.2.3 ***Degraded power management.***

The second option of stretching the dwell also negatively impacts the achievable power management obtainable from the stations since they wake up to listen for a TIM or DTIM only to find that the dwell was stretched. This increases the achievable ratio of Awake time to Asleep time. This problem is most noticeable in heavily loaded systems and impacts all power managed stations – even those with little data activity.

2.3.3 Fixed dwell time with fragmentation

The third option to ensure maximum utilization of bandwidth within a hop interval is to fragment the MSDU into multiple MAC frames. This option does not allow the hop interval to stretch. The sourcing station , in this case, only transmits the number of sub-packet frames which can be fit into the current hop interval. The table below shows the maximum wasted bandwidth for a 1 Mbps FH system utilizing fixed hop intervals and fixed frame lengths.

Table 2: Impact of Fragmentation On Unutilized Bandwidth

	Maximum unutilized bandwidth in each hop interval		
Frame size	20 ms. hop interval	50 ms. hop interval	100 ms. hop interval
1518 bytes - no fragmentation	60.7%	24.3 %	12.1%
759 bytes	30.4%	12.1%	6.1%
506 bytes	20.2%	8.1%	4.0%
380 bytes	15.2%	6.1%	3.0%
Dynamic	Approximately 0%	Approximately 0%	Approximately 0%

Note that as the number of fragments per MSDU is increased the utilization of the bandwidth is improved. It needs to be reiterated that the long dwell times decrease the effectiveness of a FH system. Simulation of system performance as a function of dwell duration will prove useful in determining the best balance be hopping effectiveness and utilized bandwidth.

An obvious enhancement to this option is to allow a station to dynamically adjust the size of the transmitted frame to fully utilize the bandwidth available in the remainder of the current hop interval. This allows the wasted bandwidth to be reduced even more depending on the actual implementation. This submission proposes an implementation for dynamic fragmentation.

The option of using fragmentation allows efficient use of the channel without requiring long hop intervals. The addition of fragmentation to the MAC eliminates the requirement for stretching of the hop interval and its associated drawbacks with hidden/sleeping stations.

3. The cost of adding fragmentation to the system

The following discussion addresses the impact of adding fragmentation to the foundation MAC. The discussion assumes that frame windowing has been included with the fragmentation. Two cases are considered - stations in fringe areas of the BSA and stations not operating within the fringe areas.

3.1 *Impact on stations in fringe areas.*

Due to the cost of infrastructure and the natural tendency for people to expect products to work in environments which the manufacturers engineers would like to avoid, it is safe to assume that a large number of users will attempt to extend the coverage of the system to its extremes. In an environment where the users are evenly distributed throughout the coverage area, approximately 20% of the users will be in the outer 10% of the coverage area and 10% of the users will be in the outer 5% of the coverage area.

The users in these fringe areas will be operating near the sensitivity of the PHY. There is a Frame Error Rate (FER) floor defined by the Bit Error Rate (BER) of the PHY. The FER near the sensitivity of the PHY can be estimated from the PHY's BER using the following equation:

$$FER = 1 - (1 - BER)^{number\_of\_bits}$$

The average number of attempts to transmit the frame without error is equal to the following:

$$Number\_of\_attempts = (1 - FER)^{-1}$$

The average number of bytes required to transmit an Ethernet packet:

$$Average\_number\_of\_bits\_TXd = Number\_of\_bits * (1 - FER)^{-1}$$

The table below shows the expected average number of bytes required to be transmitted to ensure valid reception by the receiving station for various frame lengths based on a specified BER of $1 * 10^{-5}$ for the PHY.

Table 3: Fragmentation Performance In Fringe Areas

Frames per 1100 byte MSDU	FER per frame (30 bytes overhead per frame)	Average Bytes TX'd per frame	Total Bytes TX'd per packet
1 – no fragmentation	8.6%	1237	1237
2 – (550 + 30 OH) bytes	4.5%	607	1215
3 – (367 + 30 OH) bytes	3.1%	409	1228
4 – (275 + 30 OH) bytes	2.4%	313	1250

The above table shows little or no penalty associated with fragmentation for typical 1100 byte packets. This indicates that the benefits of using fragmentation as presented in the discussions above can be obtained with little or no additional overhead introduced into the system.

3.2 *Impact on stations not in a fringe area.*

The BER performance of the PHY for stations operating in areas other than the fringe area will typically be $1 * 10^{-6}$ or better. In this case the FER for a 1100 byte frame is less than 1%. For stations operating in this region of the BSA the overhead due to fragmentation should be insignificant from a users perspective. The example below shows the effective impact of adding fragmentation for users operating in the strong signal areas of the BSA.

EXAMPLE:

Assume a client-server type network operating system using a packet-acknowledgment type protocol. In this type of system the maximum PPS rate that the station can expect to achieve is equal to the reciprocal of the round trip delay from the transmission of the packet to the transmission of the next packet. The data packets in this system are 1100 bytes and the network acknowledgments are 100 bytes. In this system it is assumed that the server and the client each have a 3 ms. response time. The server takes 3 ms. to generate an acknowledgment in response to a packet and the sourcing station requires 3 ms. to generate the next packet in response to the last received acknowledgment. Overhead of 30 bytes per MAC frame is assumed. The table below shows the round trip delay, the maximum Packet Per Second (PPS) rate, and the effective throughput of a station operating in this environment. This example is based on a single packet followed by a single network level acknowledgment. Delays due to transmission on the wire (if applicable) are neglected. The FER is assumed to be 0.0%. Frame windowing is assumed in this example.

Table 4: Fragmentation Performance

Frames per 1100 Byte Packet	Round trip delay at 1 Mbps	Maximum PPS and Throughput achievable at 1 Mbps	Round trip delay at 2 Mbps	Maximum PPS and Throughput achievable at 2 Mbps
1 - no fragmentation	16.1 ms. = 9.04 + 3 + 1.04 + 3	62.1 PPS = 547 Kbps	11.0 ms. = 4.52 + 3 + 0.52 + 3	90.9 PPS = 800 Kbps
2 – (550 + 30 OH) bytes	16.3 ms. = 9.28 + 3 + 1.04 + 3	61.3 PPS = 540 Kbps	11.2 ms. = 4.64 + 3 + 0.52 + 3	89.3 PPS = 786 Kbps
3 – (367 + 30 OH) bytes	16.5 ms. = 9.52 + 3 + 1.04 + 3	60.6 PPS = 533 Kbps	11.3 ms. = 4.76 + 3 + 0.52 + 3	88.5 PPS = 779 Kbps
4 – (275 + 30 OH) bytes	16.8 ms. = 9.76 + 3 + 1.04 + 3	59.5 PPS = 524 Kbps	11.4 ms. = 4.88 + 3 + 0.52 + 3	87.7 PPS = 772 Kbps

The above table shows that partitioning a packet into 4 MAC frames will decrease the maximum available throughput of a 1 Mbps station from 547 Kbps to 524 Kbps - a decrease of less than 5 %. To the end user, this is an imperceptible decrease in performance.

The above examples show that little if any performance is given up by adding fragmentation. On the other hand, all of the benefits outlined in the previous section are gained.

4. An implementation proposal for fragmentation

Although the channel will support large MAC frames, fragmentation of large data packets into smaller fragments will allow a more efficient channel usage given the likelihood of microwave oven interference, collisions, and hidden terminals as described in the previous section. The wireless MAC must be able to fragment and reassemble user data packets into acceptably small fragments. The fragmentation and reassembly mechanisms must also allow for fragments to be retransmitted. For the purposes of this description a 'dwell' will refer to the duration of time spent on a single frequency in a FH system. Therefore in a FH PHY the PHY will hop to the next frequency in the hop sequence at the end of the current dwell. For other systems a 'dwell' will refer to the period of time spanning from the start of transmission of a beacon (TIM) until just before the start of transmission of the next beacon. Optionally, other systems could elect to allow stretching of the duration between TIMs (as currently defined in the foundation MAC) in which case the 'dwell' would correspond to the time from the start of the time bounded services superframe until the just before the start of the next time bounded services superframe.

The following discussion assumes that AP relay between devices is supported by the MAC. If this is not the case then the proposed methodology still holds true with the appropriate references to AP relay removed from the discussion.

4.1 *Control of the Channel*

Any channel access procedure with MAC level acknowledgments requires a mechanism to pass control of the channel between the sourcing station and the destination station. The current foundation MAC provides such a mechanism and can be used as is with the addition of fragmentation.

The following example illustrates the mechanism for maintaining control of the channel if windowing is implemented. The window size in the example is 2.

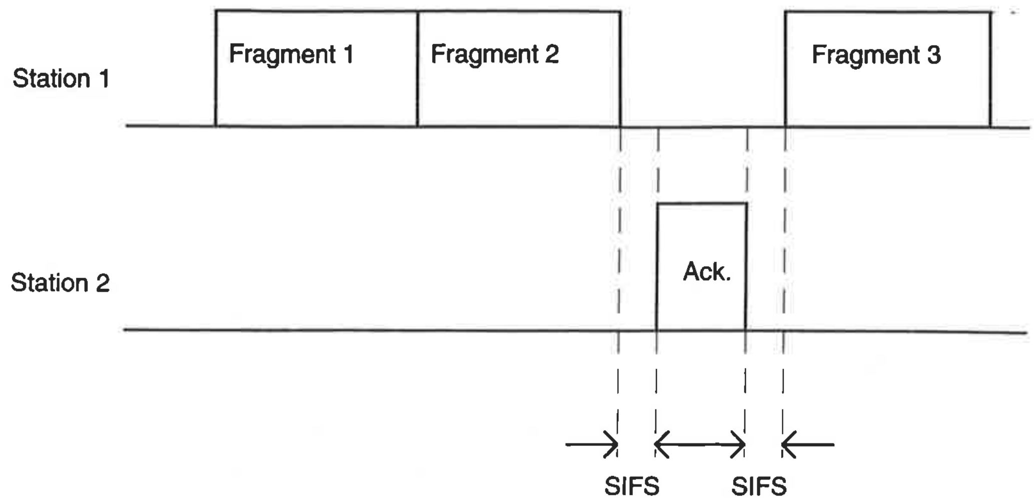


Figure 6: Channel Control With Windowing

If windowing is not implemented then the control of the channel is maintained in the same manner as shown in the figure below.

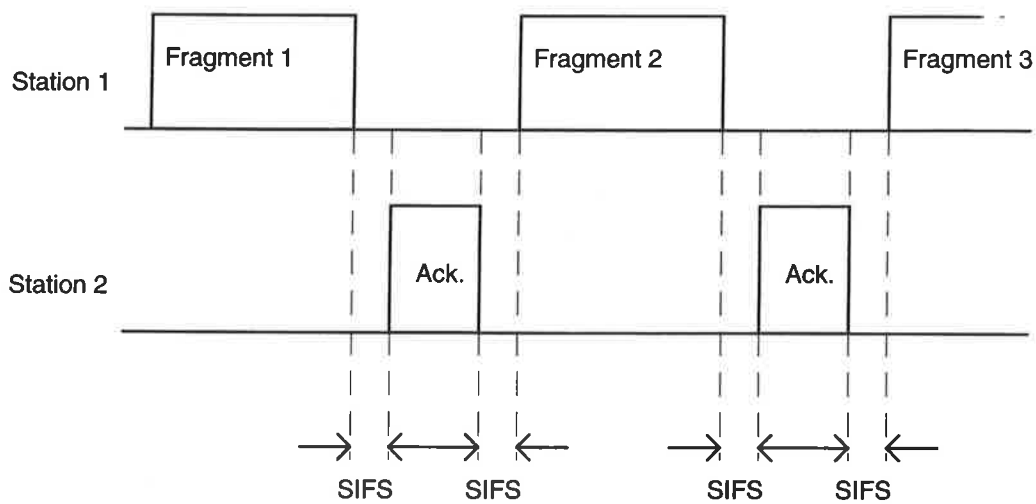


Figure 7: Channel Control Without Windowing

4.2 *Fragmentation*

Whenever possible, the size of the data portion (or *payload*) of a fragment shall be some fixed (configurable) number of bytes. Let this number of bytes be denoted by *max_payload*. The payload of a fragment shall never be larger than *max_payload*. However, in certain cases, the size of the payload may be less than *max_payload*.

When data needs to be transmitted, the number of bytes in the payload of the fragment shall be determined based on the time at which the fragment is to be transmitted for the first time. Once the fragment is transmitted for the first time, its contents shall be fixed until it is successfully delivered to the receiving station.

The number of data bytes in the payload of a fragment shall depend on the values of the following two variables at the instant the fragment is to be transmitted for the first time:

1. The time remaining in the current dwell.
2. The number of bytes in the packet that have not yet been transmitted for the first time.

Since the control of the channel may be lost at a dwell boundary, it is desirable for the acknowledgment of a fragment to be transmitted before the devices cross the dwell boundary. Hence, if there is not enough time remaining in the dwell to transmit a fragment with an M byte payload, the number of bytes in the payload shall be reduced to the maximum number of bytes that will allow the fragment plus the MAC acknowledgment to fit within the time remaining in the dwell. This strategy will (on the first transmission attempt) prevent the time near the end of the dwell from being wasted.

However, the number of bytes in the payload of a fragment shall be greater than some (configurable) minimum quantity, denoted *min_payload*. If there is not enough time remaining in the dwell to fit a fragment with a *min_payload* byte payload plus acknowledgment, the device shall not create another fragment for transmission in the current dwell. The device shall wait until the next dwell time to create and transmit a fragment with a *max_payload* byte payload (provided there are at least *max_payload* more bytes remaining in the packet). Devices can violate this minimum payload size rule only if the remaining number of bytes in the packet is less than *min_payload*. If the number of bytes that have not yet been transmitted is less than *min_payload*, the last fragment of the packet will have to contain less than *min_payload* bytes. (I.e., the payload of the fragment will not be padded to bring its length up to *min_payload*.)

If a fragment requires retransmission, its contents and length shall remain fixed for the lifetime of the data packet at that device. In other words, after a fragment is transmitted once, we will not allow the contents or length of that fragment to fluctuate to accommodate the dwell boundaries. By fixing a fragments size for the lifetime of the packet the overhead associated with the fragmentation process is minimized and the fragment reassembly process is simplified. Let the *fragmentation set* refer to the contents and length of each of the fragments that make up the data packet. The fragmentation set is created at a device as soon as the fragments are attempted for the first time. The fragmentation set remains fixed for the lifetime of the packet at the transmitting device. If an Access Point must relay a packet between two wireless stations, the Access Point will create its own fragmentation set when it comes time to transmit that packet. That is, the Access Point will not necessarily use the same fragmentation set used by the source wireless station.

Each fragment will contain a fragment ID number. When a device is transmitting a packet, each window of fragments will contain the fragments in order of lowest ID to highest ID except in the case of fragment retransmissions. Retransmitted fragments are transmitted after all other fragments have been transmitted. This is not required but simplifies the implementation.

If, when retransmitting a fragment, there is not enough time remaining in the dwell to allow transmission of the fragment plus the acknowledgment, the device shall wait until the start of the next dwell before retransmitting that fragment.

The following examples show how a data packet is fragmented and transmitted according to the above strategy.

Example 1:

This example assumes the maximum and minimum number of bytes in a fragment is M = 200 bytes and N = 25 bytes, respectively. Station 1 (STA1) is transmitting a 385 byte packet to Station 2 (STA2) with a fragment window size of two.

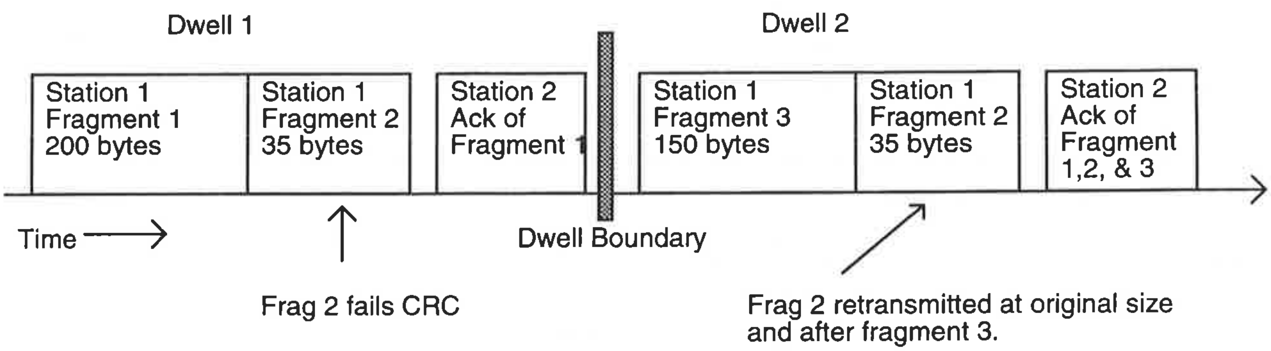


Figure 8: Fragmentation Near A Dwell Boundary – Example 1

STA1 transmits the first window of fragments near a dwell boundary. Fragment 1 is a full size fragment of 200 bytes, but there is not enough time for fragment 2 to be 200 bytes. Therefore, fragment 2 is only 35 bytes. Suppose, for some reason, STA2 received fragment 1 successfully, but failed CRC on fragment 2. Hence, STA2's acknowledgment indicates that only fragment 1 was successfully received. STA1's second fragment window follows at the start of the next dwell and consists of the remaining 150 bytes of the packet in fragment 3 and a retransmitted fragment 2 (35 bytes). STA2 transmits an acknowledgment indicating that fragments 1,2, and 3 were received correctly.

Example 2:

This example also assumes the maximum and minimum number of bytes in a fragment is M = 200 bytes and N = 25 bytes, respectively. Station 1 (STA1) is transmitting a 585 byte packet to Station 2 (STA2) with a fragment window size of two.

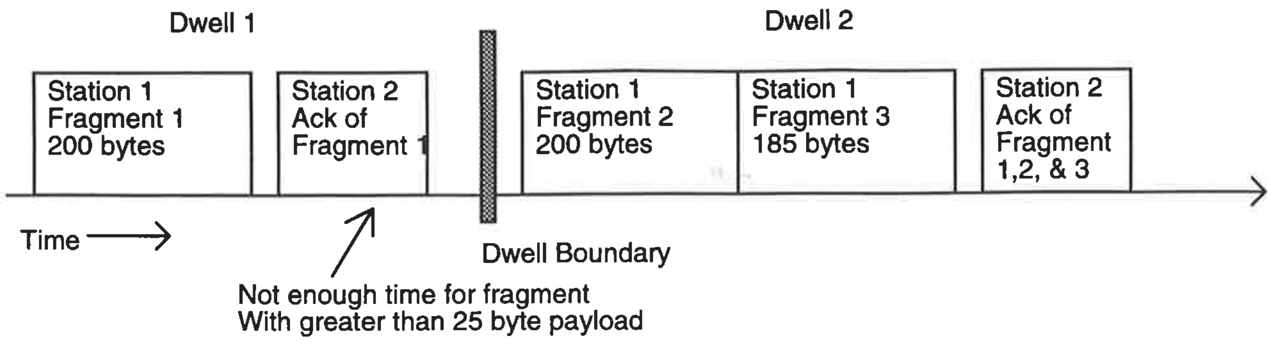


Figure 9: Fragmentation Near A Dwell Boundary – Example 2

Within the initial fragment window, the first fragment is transmitted containing 200 bytes, but there is not enough time left in the dwell for a fragment containing 25 bytes (the minimum payload size) in the payload plus the associated acknowledgment. Hence, fragment 2 is deferred until the start of the next dwell time and has a payload of 200 bytes. STA2 sees the SIFS space expire and sends an acknowledgment of fragment 1. The second fragment window (fragments 2 and 3) is transmitted. STA2 then acknowledges the fragments 1,2, and 3.

Example 3:

A situation not shown in the above examples concerns the case when STA1 does not receive an ACK of an entire fragment window and must back off and retransmit the window later. STA1 may end up retransmitting the window near the end of a dwell as shown in the figure below. In this figure, fragment 1 will fit in the current dwell with room for an acknowledgment, but fragment 2 must be deferred until the next dwell. Since STA1 gave up the channel to defer fragment 2, STA2 must send an acknowledgment before the end of the dwell. STA1 continues with fragment 2 in the next dwell.

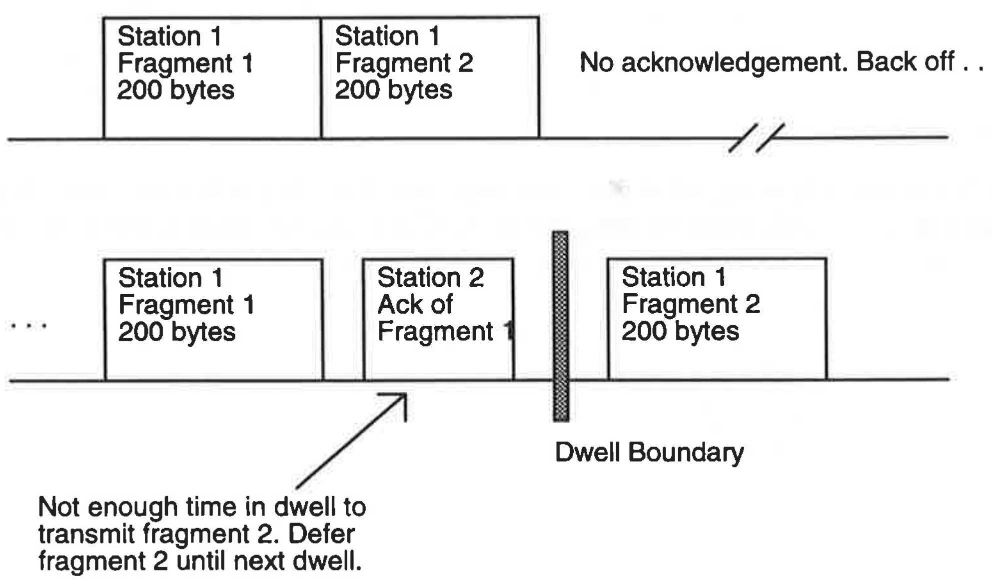


Figure 10: Retransmission Of A Fragment Window

4.3 *Data Packet Reassembly*

Each data fragment must contain enough information to allow the complete data packet to be reassembled from its constituent fragments. The header of each data fragment contains the following information that is used by the receiving device to correctly reassemble the complete data packet:

- Frame type (data, acknowledgment, etc.).
- Source address
- Destination address
- The packet sequence number. This field allows the receiver to check that all incoming fragments belong to the same data packet.
- A fragment ID number. Fragments of an MSDU are numbered sequentially, 1, 2, 3, etc.
- An end-of-packet indicator to inform the receiver that the fragment ID of the fragment corresponds to the total number of fragments that make up the complete data packet. Only the last fragment of the data packet will have this bit set to one. All other fragments of the data packet will have this bit set to zero.

The receiving device can reconstruct the packet by piecing together fragments in order of increasing fragment ID number. If the fragment with the end-of-packet bit set to one has not yet been received, then the receiving device knows that the data packet is not yet complete. As soon as the device receives the fragment with the end-of-packet bit set to one, the device knows that no more fragments with a fragment ID higher than that fragment's ID will be sent for the current data packet.

To properly reassemble packets, a receiving device must discard any duplicated fragments received over the RF. If a device receives a fragment with the same source, destination, sequence number, AP bit, relay bit, and fragment ID number as a previous fragment, then the device must discard the duplicate fragment.

If a device has previously accepted at least one fragment of a data packet, and more fragments of the same packet arrive with different relay bit or Access Point bit settings, then the device must proceed as follows. First, obviously, addressing rules must be followed to determine whether the device is allowed to accept the new fragments. If the device is allowed to accept the fragment with the new relay bit and Access Point bit setting, then the device should keep the fragments with the new settings and throw out all fragments with the old settings. This situation can occur when the device's acknowledgment failed to reach the sending device thereby causing the sending device to request relay from the Access Point. (Relay is discussed in more detail in a separate submission.)

4.4 *Frame Formats*

4.4.1 Data Frames

The following element needs to be added to the data frames to support the addition of fragmentation to the MAC.

1 element with 1 information octet specific to the element. This octet is defined as shown below:

1 bit	7 bits
Last Fragment	Fragment ID #

Figure 11: Information Octet For Data Frames

The Last Fragment bit indicates if the current fragment is the last fragment of the packet. The Fragment Number field is a binary representation of the fragment number of the data frame (fragment 1 - 0000001, fragment 2 - 0000010 . . .)

The entire fragmentation element is depicted below:

Code	
More	Link = number of subsequent octets
Last Fragment	Fragment ID #

Figure 12: Fragmentation Element For Data Frames

An additional octet corresponding to an updated NAV value may be added to the above element if the RTS/CTS mechanism is employed. This might be of value since the fragments of the packet could cross a dwell boundary and therefore make the originally calculated NAV value of questionable accuracy.

4.4.2 Acknowledgment Frames

The acknowledgment frame needs to have an additional variable length field to indicate which fragments have been received correctly. The acknowledgment frame is shown below:

1 - N octets

Fixed Header	Received Fragments	CRC8

Figure 13: Acknowledgment Frame Format

The received fragments field is a bit map representation of the fragments of the packet that have been successfully received. Normally only 1 octet would be required for the received fragments field. If however, transmission of the packet required more than 8 fragments then the Received Fragments field would contain an additional octet. This is shown below:

1	2	3	4	5	6	7	8	9	10						16

Figure 14: Received Fragments Field Of Acknowledgment Frame

An octet corresponding to an updated NAV value may be added if the RTS/CTS mechanism is employed. This might be of value since the fragments of the packet could cross a dwell boundary and therefore make the originally calculated NAV value of questionable accuracy. If this additional octet were added then it would be placed before the Received Fragments field as shown below:

1 - N octets

Fixed Header	Duration (NAV)	Received Fragments	CRC8

Figure 15: Acknowledgment Frame With RTS/CTS

5. Conclusion

This submission has proposed a strategy for fragmenting packets which is compatible with the current foundation MAC.

The authors would like to see fragmentation added to the current MAC. (Issue 20.6 – YES). The authors would also like to see the strategy for fragmentation proposed within this submission serve as the basis for inclusion in the MAC.

6. References

[1] Distributed Foundation Wireless Medium Access Control; DOC: IEEE P802.11-93/190; Wim Diepstraten, Greg Ennis, and Phil Belanger

[2] A Packet Delivery and Relay Strategy for the DFWMAC; DOC: IEEE P802.11-94/39; Mark Demange et al.

[3] Frame Windowing at the Mac Layer; DOC: IEEE P802.11-94/38; Rick White et al.