IEEE 802.11
Wireless Access Method and Physical Layer Specifications

A Parametric MAC-PHY Interface Model

Date: August 14 1992

Author: Jonathon Y.C. Cheah
Hughes Network Systems
10450 Pacific Center Court
San Diego
CA 92121
Phone(619) 452-4847/ fax(619) 546-1953/ jcheah@oscar.hns.com

Introduction:

The need to satisfy diverse physical layers' implementation strategy has presented a challenge to the concept of a single unique MAC model. Soon it becomes obvious that there is no one exhaustive list of parameters that can cover all possible PHY implementation techniques and variations. Indeed, if such a list were to exist, the multitude of redundant parametric machine states will render the MAC model hopelessly complicated and inordinately expensive. This realization points to two obvious solutions. One is to produce a MAC model that is PHY dependent. This is a highly undesirable solution in the commercial arena but an extremely simple way to fulfill an official mandate in the standard making process. The other, is to create a MAC model that can satisfy all PHY structures by using PHY definable parameters.

Definable Parametric structures:

It is conceivable that there will be m numbers of parameters from the upper OSI layers, and n numbers of parameters from the PHY layer. The fundamental function of the MAC layer is to provide output functions so that the input parameters it receives from the PHY and the upper OSI layers can be processed appropriately. The problems that confronted IEEE802.11 MAC designers are the input parameters emanating from the upper layers may be the same but the expected response from the MAC layer is entirely different dependent on the different PHY layers. Similarly, PHY layer output parameters have different meanings dependent on the nature of the PHY. Thus, under this environment, a PHY dependent MAC is the easiest solution. However, given the large variety of PHY structures, the PHY dependent MAC will be small in market share, and the anticipated economy of scale cannot be realized. There is another solution.

The common vision of MAC being a fixed logic state machine should be modified. If a set of load-able "states" can be introduced, then the PHY dependency can be rendered transparent to the MAC hardware structure. In other words, MAC wakes up by a minimum "boot-state," then it loads the PHY dependent states into its universal state machine. In this it changes its "personality" dependent on the load-able "states." The "soft states" can be a PHY specific ROM or it can be a "configuration" file from the upper layer systems.

Using this concept, a major MAC design task is to define a universal MAC structure, or "load-able" state machine topology. The initial means to conceptualize this MAC structure is by defining a set of PHY independent primitives at the MAC-PHY interface. The following is a first cut at these definitions:

- **MAC to PHY primitives:**

  MAC_DRIVEN_PHY_PASSIVE_n:

These are a group of primitives that originated from MAC to PHY for specific tasks to be performed by the PHY layer. The population of these primitives has to be determined. The nature of these primitives needs be defined. For example, the universal MAC may be determined to support 10 load-able primitives and 5 of which are carried by 2 output pins each and the balance by 1 output pin each. Each primitive has a resident memory allocation of n octets, or n registers equivalent of GALs (Generic Array Logics).

Examples:

Define MAC_DRIVEN_PHY_PASSIVE_1 ≡Channel change

In FHSS this may indicate to PHY to switch hopping sequences. In DSSS, it is to change correlation sequence. In FDMA, PHY changes frequency. Each PHY decides it interpretation. In IR, for instance, PHY designers may save this definable primitive for something else, because there may not be a need to change channels.

The activation of this primitive may be caused by the upper layer complaining about long delays or MAC internal complaint about too many bad data. So a typical logic state equation can be loaded as:

$\text{Delay} \cap \text{Bad\_data} \Rightarrow \text{\_Passive\_1}$

MAC_DRIVEN_PHY_AUTO_n.

These are a group of primitives that originated from MAC to PHY for a certain request to be carried out by the PHY layer. Requests are issued to PHY layers where specific steps to satisfy the requests are not controlled by the MAC layer. PHY can perform such tasks autonomously. The examples are:

- Optimize channel diversity.
- Adaptive power control.

A typical state equation can be:

$\text{Acquisition} \cup \text{Sync\_loss} \Rightarrow \text{\_AUTO\_5}$

- **PHY to MAC Primitives:**

These are the primitives originated from the PHY to MAC. There are only 3 of such categories:

MAC_DRIVEN_PHY_RESPONSE_2n:

These are the matching set of primitives to both the MAC_DRIVEN_PHY_PASSIVE_n and MAC_DRIVEN_PHY_AUTO_n: There are 2n of such primitives. They acknowledge the tasks given by MAC have been carried out.

PHY_DRIVEN_FLAGS_n:

These are the primitives that are transferred to MAC without MAC's solicitation. Most of these primitives should be health and status related. For example: The PHY has lost signal synchronization

while transferring a data packet. The manner in which each PHY would report this event will be different. If PHY_DRIVEN_FLAGS_6 is defined as oscillator loses lock in DSSS, then the load-able state in MAC may take this primitive as a high priority interrupts input and a signal re acquisition in the form of MAC_DRIVEN_PHY_AUTO_x primitive will be passed to PHY. In IR, the same _FLAGS_6 may be assigned to IR PIN Diode failure, and a different state machine will be activated.

SIGNAL_QUALITY_n:

This set of primitives is logically be categorized under PHY_DRIVEN_FLAGS_n, but it deserves a category of its own because these primitives are those passed accompanying the data in a bit by bit basis. This is a class of "soft-decision" bits intended for FEC uses. Undoubtedly, other uses can be defined under the parametric MAC structures.

- Parameter Independent Primitives:

This is a set of primitives that is totally PHY implementation independent. There is only one such primitive that can be identified. No doubt, more of the same category of primitives will be generated.

PHY_KILL:

For all communications systems, there is an emergency "kill" command that shuts down all PHY activities. This is unequivocally universal in system implementation.

The Definable PHY-Independent Structure:

The discussions above are referring to a MAC sub structure at the MAC-PHY interface. The normal MAC related state machine resides above the "PHY-Independent Structure." By using parametric primitive passing concept, then most of the common MAC state machine can be shared. Functions such as OSI MAC management, and upper layer interfaces are very well suited in this grouping.

It is now obvious that what the PHY-Independent structure is based on a programmable gate array concept where the standard dictates how big this gate array can be by allowing the number of "input pins" (i.e., the number of definable MAC-PHY primitives) and how many "output pins" (i.e., PHY-MAC primitives).

It is imperative to guarded against the temptation to regress to the more familiar PHY dependent primitives as was observed in the PHY and MAC task groups by those are familiar only with one type of PHY implementation technique or the other. The important to challenge the introduction of each primitive category is whether it is perfectly independent of all PHY implementations. If not, then it should remain as a "definable " parameter. It is important to bear in mind that any non-definable primitive added is a cost carried by the MAC hardware, ultimately the cost of IEEE802.11 products.

The figure below shows the conceptual "Parametric MAC-Phy Interface."

Conclusions:

In this paper, a solution is put forward to overcome the dependency of MAC in PHY implementation techniques. In this way, the idea of a universal MAC can be accomplished. The solution also frees the constraints placed on the systems implementors to adopt a particular FY structure so as to utilize the common MAC hardware. Thus

diverse Phy media such as IR, Sonics and radio can benefit from the scale of economy in the MAC hardware production.