collision is assumed to have occurred and the data packet is transmitted again after waiting another random amount of time. CSMA/CA thus provides a way of sharing access over the air. This explicit ACK mechanism also handles interference and other radio-related problems very effectively.

[**0015**] The MAC layer is designed to interface with many different types of devices used in a WLAN, including access points, mobile units and client bridges. Such WLAN devices have differing characteristics. There is therefore a need in the art for a mechanism for an improved apparatus and method for the MAC layer to operate based on the characteristics of the device that is connected to the MAC layer.

SUMMARY

[**0016**] The present invention is based on the division of the 802.11 MAC operation into two entities: a lower MAC and an upper MAC. The lower MAC functionality will be provided by a combination of software commands and an RF driver. The RF driver is intended to be appliance independent. In other words, the RF driver will not change based on what type of 802.11 node is being implemented: Mobile Unit (MU), Access Point (AP), Wireless AP (WLAP), phone, pager, etc. The RF driver will not be hardware platform independent, due to variations in DMA capability and allotted timers. The upper MAC is intended to be hardware platform independent, but appliance dependent (i.e. it will change based on the product being implemented such as PCMCIA, USB, PCI, etc.).

BRIEF DESCRIPTION OF THE DRAWINGS

[**0017**] **FIG. 1** illustrates a basic block diagram of a radio frequency transceiver.

[**0018**] **FIG. 2** illustrates a basic block diagram of an improved radio frequency transceiver with the lower MAC and upper MAC split over two chipsets.

DETAILED DESCRIPTION

[**0019**] Turning to **FIG. 1**, shown is a basic conceptual overview of a transceiver that may be used in a wireless local area network (WLAN) device. Shown is an antenna **10** that is capable of receiving and transmitting radio frequency signals. Once signals are received by antenna **10**, these signals are passed on to the radio **20** for converting from the higher radio frequency to the analog baseband signal. The analog baseband signal is passed on the baseband processor **30**, where the baseband analog signal is converted to a digital signal and processed to extract the data transmitted over the radio frequency signal. This signal is passed on to the device interface **40**, where the information contained within the signal is processed to determine the actions necessary to pass on to the device **50**. Such devices may be, for example, an access point, mobile terminal, network interface card or client bridge. The process may be reversed to send radio frequency signals through the antenna **10**. In one embodiment of **FIG. 1**, each of the radio **20**, baseband processor **30** and device interface **40** are fabricated on a separate ASIC.

[**0020**] Applying the 802.11 protocol, the radio **20** and baseband processor **30** performed much of the PHY functionality and the device interface **40** performed much of the MAC functionality. While the nature of the MAC functions coordinated by device interface **40** differ based on the nature of the device **50** that is connected to the transceiver, many MAC functions that have traditionally been performed by device interface **40** are independent of the nature of the device **50**. When all the MAC functions are on the device interface **40**, however, this results in needless duplication of the hardware and software to perform MAC functions that are common to each device **50**.

[**0021**] This process can be made more efficient by division of the 802.11 MAC operation into two entities: a lower MAC and an upper MAC. The lower MAC functionality may be provided by a combination of software commands and an RF driver. The RF driver is intended to be appliance independent. In other words, the RF driver will not change based on what type of 802.11 node is being implemented: Mobile Unit (MU), Access Point (AP), Wireless AP (WLAP), phone, pager, etc. In contrast, the upper MAC is intended to be hardware platform independent, but appliance dependent (i.e. it will change based on the product being implemented).

[**0022**] This design presents the advantage that for each product used on a WLAN, only the hardware and software necessary to run the upper MAC must be designed and configured for that product. The lower MAC will remain the same and will not require new design to implement the necessary hardware and software.

[**0023**] **FIG. 2** illustrates an embodiment of the present invention incorporating the concept described above. Shown is a baseband processor **100** chipset along with a device interface chipset **200**. For receiving data, the baseband processor **100** includes a mechanism **110** to receive signals converted from the higher radio frequency radio signal to the analog baseband signal. These signals are then processed by analog to digital converters **120**, a digital modem **130** and the PLCP (physical layer convergence procedure) processor **140**. The PLCP processor **140** is a PHY sublayer designed to work with the MAC by minimizing the variations in the PHY layer necessitated by the variations of the physical media. The baseband processor **100** also includes the lower MAC **150**.

[**0024**] The device interface chipset **200** includes the upper MAC **210** and the I/O interface **220** which is designed to communicate **230** with a host device **240** based on various protocols such as PCMCIA, USB and PCI.

[**0025**] The hardware and software necessary to implement the upper MAC **210** will reside on the device interface chipset **200** while the hardware and software necessary to implement the lower MAC **150** will reside on the baseband processor **100**. Although the upper MAC **210** implementation will vary based on the product it interfaces with, the general features of the device interface chipset may include one or more of the following:

PC Card Host Interface
CardBus Host Interface
CIS Logic and RAM
Arbitration plus Interleaver
Internal RAM
Local Bus Interface
Reset logic