Bluetooth: A Brief Note on Technology

Surya Teja Paruchuri, *Student Member, IEEE*

***Abstract*—A short and brief review on my understanding of Bluetooth technology is presented in this paper. It includes the design goals of developing Bluetooth technology, PHY, MAC layer details, message transfer mechanisms. It also briefly visits the network topologies in Bluetooth specifically Piconet and Scatternet formation among Bluetooth devices.**

***Index Terms*— Bluetooth, PHY, MAC, protocol stack, Piconet, Scatternet.**

I. Introduction

THE omnipresence of computing devices has led to the development of local networks, many being ad-hoc in nature. IEEE 802.11 family of standards was developed to address the key issue of supporting local networks without requiring cables. The number of IEEE 802.11 devices grew in popularity owing much to availability of free ISM band of spectrum. While IEEE 802.11 standards did support data transfer over wireless medium, without need of purchased spectrum, these were still costly and heavy to be used as a cable replacement alternative for communication between devices within close proximity, such as keyboard, mouse, headsets of mobile etc. Bluetooth, developed by Ericsson amid controversial environment questioning the need for new standards, is currently being developed by Bluetooth Special Interest Group (SIG). Formation of SIG has accelerated development of Bluetooth, through explicitly specifying few protocols and adapting others.

Bluetooth is designed to operate in the ISM band. It uses Frequency Hopping Spread Spectrum (FHSS) technique, with 79 channels with a bandwidth of 1MHz each and 2.5MHz, 3.5MHz guard bands on lower and upper side of the spectrum. The maximum theoretical data rate over Bluetooth is 1Mbps, while recent developments are looking at increasing the data rates. It supports both synchronous and asynchronous traffic with Forward Error Correction (FEC) capability. While the core protocols such as PHY, MAC are developed explicitly for Bluetooth, various other protocols have been adapted from other technologies which include Object Exchange (OBEX) from Infrared (IR), TCP/UDP, IP, PPP. The asynchronous transfer method supports transfer of time insensitive data and synchronous traffic supports time-sensitive data like parsing audio data or call data to headsets.

Surya Teja Paruchuri, is a graduate student pursuing Master's in Telecommunications at Department of Electrical and Computer Engineering, University of Maryland, College Park. MD USA (e-mail: surya@umd.edu).

The next section discusses the three fold design objectives of developing Bluetooth and the later discusses the protocol stack and on Bluetooth profiles. The fifth section explains the PHY, MAC layer details. In Section six, topics related to network topologies are discussed in the view of Bluetooth. The last section gives certain practical examples and implementation details.

II. Design Goals

Bluetooth was primarily targeted as cheap cable replacement technology with equal importance placed on interoperability between Bluetooth devices in ad-hoc manner. The existing 802.11 wireless solutions were too heavy and expensive to be used as cable replacement alternatives. Thus Bluetooth's design goals mandated the solution to be a low cost and easily incorporation solution to support the existing cable based communication over short range wireless channels. Thus a relatively greater emphasis was placed on cable compatibility. On the other hand, compatibility within Bluetooth devices to communicate in an ad-hoc fashion was given equal significance.

III. Protocol Stack

Bluetooth has two specifications - Core Specifications and Profile specifications. The core specifications describe the protocol stack while the profile specification define hierarchical structure of core protocols needed to implement a service. Implementing core protocols independently and describing profiles necessary for an application led to interoperability between Bluetooth devices from different vendors- by simply listing all profiles a devices support. Thus by easy implementation of new profiles using a protocol on top of another the versatility of Bluetooth has been shown.

A. Core Specifications

Opting four different approaches for developing core specification protocol stack has led to fast development protocols and adoption of Bluetooth technology. The first approach resulted in development of Core protocols that were essential for the working of Bluetooth devices like PHY, MAC etc. Building from scratch enabled taking full advantage of Bluetooth in protocols like PHY, MAC layer. The second group of protocols supported the primary design goal of Bluetooth – Cable Replacement Protocols- which emulate RS232 serial port. The third set of protocols -Telephony Control Protocols- were meant to enable data/speech connections amongst Bluetooth devices. The last set of protocols are adopted protocols, which are adopted for Bluetooth technology. These include TCP/UDP, IP,PPP etc. The following table gives a brief description about few protocols [1].

TABLE I
CORE SPECIFICATIONS PROTOCOL STACK

Type	Protocol Name	Brief Description
Core	Bluetooth PHY layer	Describes and controls the radio characteristics of Bluetooth radio.
Core	Baseband	Physical connection establishment and clock synchronization
Core	Link Management Protocol (LMP)	Manages the link connection establishment and also existing links. (Similar to TCP layer in IP structure). Also responsible for security and authorization.
Core	Logical Link Control and Adaptation Protocol (L2CAP)	Responsible for segmentation and reassembly of packets. (Similar to Layer-3 in TCP/IP structure).
Core	Service Discovery Protocol (SDP)	This protocol is responsible for identifying Bluetooth devices in a given vicinity.
Cable Replace ment	Radio Frequency Communication Protocol (RFCOMM)	This protocol provides the services of a RS232 serial port to the protocols above it using the services of lower layer protocol- L2CAP.
Telepho ny Control	Telephony Control Specification Binary (TCS BIN)	Responsible for establishing speech connections.
Adopted	TCP/UDP	Connection oriented and connection less serices used in IP.
Adopted	IP	Transfer packets in a connectionless manner eg: Internet.
Adopted	PPP	Transfer data in a point-to-point manner over serial ports

B. Profile Specifications

Profile Specifications in Bluetooth are at the core of implementation of interoperability among Bluetooth devices. Two devices claiming to support same profile are required to have implemented the necessary protocols required to support applications of this profile. Thus a list of all the profiles that a device supports gives a complete picture of all the implemented protocols. The first version of Bluetooth consists of 13 profiles, all organized in tree type hierarchical structure with the Generic Access Profile acting as the root for the entire tree, i.e. every protocol uses the services of Generic Access Profile. As an example to support file transfer over Bluetooth (say pictures, audio files – to transfer and not to play over Bluetooth- or documents) file transfer protocol has to be implemented, which requires the services of General Object Exchange Profile to be implemented. The General Object Exchange Profile transfers the files in a serial communication way using services of RFCOMM which implicitly requires the existence of Generic Access Profile which includes the L2CAP, LMP, SDP, Baseband and PHY protocols.

IV. PHY AND MAC LAYER

A. PHY layer.

Though Bluetooth owes a good amount of its wide spread usage to the free ISM band, it has to deal with the interference from several devices operating in the same 2.4 GHz band. To minimize the collisions Bluetooth adopts Frequency Hopping Spread Spectrum (FHSS) by diving the band into 79 channels of 1MHz each and a fixed hopping rate of 1600 hops per second. This gives 1600 timeslots, each of 625µs each. The channels are labelled from 0 to 78 corresponding to 2.402GHz + n MHz, where n is the channel number. Gaussian Frequency Shift keying (GFSK) modulation is used in Bluetooth. In addition to have control over ranges, a power control mechanism is adopted in which 3 different power classes are made available. Class 1 has a maximum power output of 20dbm and class 3 has 0dbm as maximum power output.

B. MAC layer.

The MAC layer functionalities - primarily controlled by the baseband protocol- holds responsibility for synchronization, contenting for channel and other key functions. Bluetooth networks are based on Master-slave topology with a limit of 7 on the maximum number of active slaves a master can control in its piconet (This limitation comes from the 3-bit Active Member Address -AM_ADDR- given by master to each active slave. But a larger number of parking members can be present in a given piconet owing to large 8-bit Parking Member Address PM_ADDR). The apparent need for multiple access seen in master’s ability to control multiple slaves in a network is solved by using Time Division Duplex (TDD). The fact that a device is allowed to transmit on a particular frequency only for 625µs is taken advantage instead of having a varying timeslot length. Both on Uplink and Downlink use bandwidth of entire channel bandwidth- 1MHz - but are divided in time.

A Bluetooth device obtains its frequency hopping sequence from the master’s Bluetooth Address (BD_ADDR) and clock as seen in many wireless architecture. This minimizes the chances of collision from devices using the same frequency band for communications. Even in case of a collision the probability of successive collision occurrence is very slim owing to large differences in generating the frequency hopping sequence. This gives certain amount of robustness for interference from devices in same band. Additionally Bluetooth also supports Forward Error Correction (FEC).

Although Bluetooth doesn’t allow varying length of time slots, it permits either master or slave’s transmission to occupy an integral (1, 3, 5) multiples of the standard 625µs timeslots. In a situation where the neither the master and slave occupy multiple time slots, the frequency used for transmission either by master or slave should be the next frequency generated by the Frequency hopping sequence. Thus a master uses sequences such as g(0), g(2), g(4), and so on for transmission and the slave transmitting will be using g(1), g(3), g(5) and son on its transmission, where ‘g’ represents the hopping sequence generator function. In case if the slave uses multiple time slots to transmit data (say 3 timeslots), then it starts with the immediate next frequency- g(1) -generated by the hopping sequence after the master's transmission on say g(0). It then continues to transmit on g(2) and g(3) for next two 625 μs timeslots as decided. The master after slave's transmission begins it transmission only on g(4). In case of 5 timeslots transmission by slave the next frequency used by master will be g(6). The same is true if master chooses to transmit across multiple timeslots.

C. Connections and Packets in Bluetooth Networks

Bluetooth supports connection oriented - Synchronous Connection oriented (SCO)-service and connectionless – Asynchronous Connectionless (ACL)-service, based on the service being provided.

ACL supports asynchronous traffic such as file transfer. It is a point-to-multipoint type connections with the control vested in master, to poll members in a Bluetooth network (Piconet), thus allowing only slaves with the token-typically passed in a round robin fashion- to transmit. At any given time only a single ACL connection is allowed between a pair of devices, even if slave is allotted a guaranteed timeslot through SCO mechanism. SCO on the other hand is meant to support synchronous data such as voice, over point-to-point connections. An SCO provides a guaranteed bandwidth by reserving a timeslot for a transmission pair, enabling the slave to automatically transmit on the allotted timeslot. In addition to SCO Bluetooth also provides for an enhanced SCO (eSCO) type connection, very similar to SCO. eSCO differs from SCO in way that it supports multiple timeslot and asymmetric transmission.

There are three types of packets in a typical Bluetooth network- Control, ACL, SCO. Control Packets has information about Identity ID, NULL, POLL, and Frequency hopping Synchronization details. ACL packets are of two types DMx and DHx, with x representing the number of packets used in the burst (1,3,5). While both provide CRC based error detection capability, only DM provides a fixed 2/3 rate Forward Error Correction capability. The maximum payload and data rate depend on the way time slots are used- Symmetric or Asymmetric. Alternatively SCO and eSCO connections use HVx (with x 1, 2, 3-based on error rates 1/3, 2/3, or no FEC), EVy packets (with y=3, 4, 5). While HV type packets don't support for error detection, EV packets do through CRC mechanism.

V. BLUETOOTH NETWORKS

A Bluetooth device can work in either of the three topologies- point-to-point, single point-to-multipoint (piconet) and multipoint-to-multipoint (scatternet) topologies. While a piconet is a network of Bluetooth devices, scatternet is network of piconets. Two or more piconets connects to each other either through a Master-Slave bridge (Master of one piconet and slave of 2nd piconet) or Slave-Slave bridge.

A. Piconets

A piconet formation is initialized by potential master by sending out the Inquiry Access Code (IAC) ID packet on all 32 wake-up channels on the available 79 channels. Upon receiving IAC ID packet during periodic Inquiry Scan state the potential slave responds to the potential master with a Frequency Hop Synchronization (FHS) packet which contains the information about device's Bluetooth Address (BD_ADDR), clock information and other details, after a random amount of back-off time to avoid any collisions. Then the slave enters a page scan state where it waits for the paging message from the slave. But the Potential master on receiving the FHS packet may choose to wait for time to collect and all the potential slave's information and synchronization. Later on the master calculates the page hopping sequence based on received slave's BD_ADDR and clock information. Then master device sends out a page message with Device Address Code (DAC) over the paging sequence. The slave then echoes back with the same DAC packet to acknowledge the receipt, which allows the master to communicate it's BD_ADDR and clock information in FHS packet. The slave responds to FHS again with DAC, thus finishing the initial handshake and enters connection state. The Master enters connection state only when it has completed all the synchronization and paging activities with every potential slave.

All the slaves in a piconet are given either an active member address (AM_ADDR) or a Parking Member Address (PM_ADDR). AM_ADDR is given to slaves in Active or Hold or Sniff State. A piconet can have a maximum of 7 actively participating members owing to the limitations of 3-bit AM_ADDR. A slave in Active state actively participates using it's AM_ADDR, while a slave in Sniff state is only interested in a specific timeslot which I should have already informed to the master. On the other hand in Hold mode, a slave holds all it's activities in a piconet, as it can stay active simultaneously in two different piconets. Thus this state/mode is generally used by bridge connecting two or more piconets. A slave in a park state is given a PM_ADDR, but cannot participate actively in the piconet. Owing to 8-bit PM_ADDR a piconet can have more slaves in parking state. Slaves in sniff, hold and park save energy by entering reduced power mode between sleep cycles.

B. Scatternets

As coverage area and the number of devices (max. of 7) is limited in a piconet, Bluetooth tend to form scatternet to increase the reach and devices limit in the network. As mentioned earlier scatternets are formed by bridging several piconets together, using one of the two methods- Master-Slave bridge or Slave-Slave bridge. A scatternet can be formed either in a centralized way or distributed fashion. In a centralized approach a leader is elected and then leader decides the network topology. In distributed approach, localized clusters are formed and a local leader of the cluster are elected. The elected cluster leaders then choose one among themselves as the leader based parameters such as but not limited to number of devices in the localized cluster. In addition there are also semi-centralized approaches.

VI. QOS AND SCHEDULING IN BLUETOOTH

Although Bluetooth supports asymmetric Time Division Duplexing (TDD) – ie, can allot more timeslots (3, 5) in a traffic intensive direction, QoS can still be improved by using better and efficient scheduling algorithms and packet Segmentation And Reassembly (SAR) within frame structure. Such techniques include classifying devices/links based on utilization and allocating resources accordingly, better SAR at upper layers either to fit maximum data into a frame – to maximize the efficiency – or transmit the packet as soon as to achieve the least delay.

VII. Conclusion

Though operating in same ISM frequency band, through Frequency hopping Bluetooth performs better than FDMA systems [1]. As the number devices operating in the band proliferates with time, the performance of Bluetooth would depend more on other factors such as distance, transmitted power and amount of inference seen by receiver from the surroundings etc. In addition to these, recent developments like Bluetooth Low Energy (BLE), message dissipating ad-hoc LANs and pervasive sensing including Wireless Sensor Network (WSN)/ Internet of Things (IoT), Bluetooth seems to be promising technology due to ease of adding new profiles to support developing services with a lot of scope for development in areas such as scheduling and SAR.

References

[1] *Kwok, Y.; Lau, V., "Bluetooth WPAN," in Wireless Internet and Mobile Computing: Interoperability and Performance , 1, Wiley-IEEE Press, 2007, pp.285-307 doi: 10.1002/9780470167960.ch10 Available: http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=5236934&queryText=bluetooth&refinements=4291944823&searchField=Search_All*