ISSN No. 0976-5697

Volume 1, No. 4, Nov-Dec 2010

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

Protocal Usage Model & An Architecture Design for Bluetooth – A Short Range Transaction System

Kamani Krunalkumar C
Information Technology Center
Anand Agricultural University
Anand, India
kamanikrunal@aau.in

Kathiriya Dhaval R
MCA Department
L.D.R.P. Institute of Tech. and Research
Gandhinagar, India
drkathiriya@yahoo.com

Virparia Paresh V*
Department of Computer Science and Technology
Sardar Patel University
Vallabh Vidya Nagar, India
pvvirparia@yahoo.com

Ghodasara Yogesh R
College of Agricultural Information Technology
Anand Agricultural University
Anand, India
yrghodasara77@yahoo.co.uk

Abstract: Bluetooth is an open standard specification created to solve a simple problem, replace the cables used on mobile devices with radio frequency waves. The technology encompasses a simple low-cost, low-power, global radio system for integration into mobile devices. Such devices can form a quick ad-hoc secure "piconet" and communicate among the connected devices. This technology creates many useful mobile usage models because the connections can occur while mobile devices are being carried in pockets and briefcases (therefore, there are no line-of-sight restrictions). This technical paper provides a brief description of some of these usage models and explains how the Bluetooth architecture is optimized to enable them. The paper also focuses on one of the emerging technologies for constructing a Mobile ad hoc network: Bluetooth. Bluetooth can be exploited on small scales, to build ad hoc wireless Personal Area Networks (WPAN), i.e., networks that connect devices placed inside a circle with radius of 10 m. The Bluetooth technology is just starting to appear on the market and its architecture and protocols are not widely known

Keywords: Bluetooth, Scatter nets, Pico net, Routing, Protocol Stack, Connectivity

I. INTRODUCTION

From the beginning, Bluetooth technology was intended to hasten the convergence of voice and data to handheld devices, such as cellular telephones and portable computers. Through the efforts of its developers and the members of the Bluetooth Special Interest Group (SIG), it is now emerging with features and applications that not only remain true to its original intent, but also provide for broader uses of its technology.

The rest of this paper is organized as follows. First, in section 2, we take a look to Bluetooth as a technology, what possible uses it has and how it is implemented. In section 3, we look into the Protocols Stack in Bluetooth Architecture. Section 4, we look into the Protocols in Bluetooth Architecture Next, in section 5, we look into the Bluetooth Usage Models and Protocols and ad hoc networks in general. We discover if there are any problems and at the end, we sum up the discoveries and make the conclusions that can be made.

II. BLUETOOTH

In this section we view the concept and specifications of Bluetooth and discover what it is all about. First we look into the background and different ways to use Bluetooth.

A. Background

Bluetooth is an open wireless technology standard for exchanging data over short distances from fixed and mobile devices, creating Personal Area Network (PAN) with high levels of security. Created by telecoms vendor Ericsson in 1994, it was originally conceived as a wireless alternative to. RS-232 data cables. It can connect several devices, overcoming problems of synchronization. Today Bluetooth is managed by the Bluetooth Special Interest Group formed in May 1998[1]. The founding members were Ericsson, Nokia, Intel, IBM and Toshiba. Since then, almost all of the biggest companies in the telecommunications business (e.g. 3Com, Microsoft, Motorola) have joined the Bluetooth SIG and the number of the participating companies is now over 1,500.

Bluetooth can be used to connect almost any device to another device. The traditional example is to link a Personal Digital Assistant (PDA) or a laptop to a mobile phone. That way you can easily take remote connections with your PDA or laptop without getting your mobile phone from your pocket or messing around with cables. Bluetooth can also be used to form ad hoc networks of several (up to eight) devices, called piconets. This can be useful for example in a meeting, where all participants have their own Bluetooth compatible laptops, and want to share files with each other. [2]

B. Technical Specifications

Bluetooth devices are categorized into three different classes by the power they use. A class 3 device has a 1 mW transmission power and a range of 0.1-10 meters. A class 2 device has a transmission power of 1-2.5 mW and a 10-meter range. A class 1 device has a transmission power up to 100 mW and a range up to 100 meters. [3] The architecture of Bluetooth is formed by the radio, the base frequency part and the Link Manager. Bluetooth uses the radio range of 2.45 GHz. The theoretical maximum bandwidth is 1 Mb/s, which is slowed down a bit by Forward Error Correction (FEC).

Bluetooth specification designates the frequency hopping to be implemented with Gaussian Frequency Shift Keying (GFSK). The base frequency part of the Bluetooth architecture uses a combination of circuit and packet switching technologies. Bluetooth can support either one asynchronous data channel and up to three simultaneous synchronous speech channels, or one channel that transfers asynchronous data and synchronous speech simultaneously. The Link Manager is an essential part of the Bluetooth architecture. It uses Link Manager Protocol (LMP) to configure, authenticate and handle the connections between Bluetooth devices. It also operates the power management scheme, which is divided into three modes: sniff, hold and park. [4] As discussed earlier, several Bluetooth devices can form an ad hoc network. In these piconets, one of the Bluetooth devices will act as a master and the others are slaves. The master sets the frequency-hopping behavior of the piconet. It is also possible to connect up to 10 piconets to each other to form so-called scatternets.

III. BLUETOOTH PROTOCOL STACK

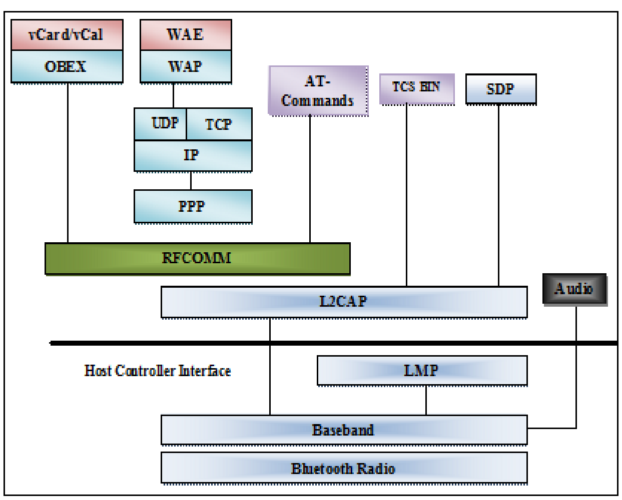

Figure 1 Bluetooth Protocol Stack

The ultimate objective of the Specification is to allow applications written in a manner that is conformant to the Specification to interoperate with each other. To achieve this interoperability, matching applications (e.g., corresponding client and server application) in remote devices must run over identical protocol stacks. The following protocol list is an example of a (top-to-bottom) protocol stack supporting a business card exchange application: vCard → OBEX → RFCOMM → L2CAP → Baseband. This protocol stack contains an internal object representation convention, vCard, and "over-the-air" transport protocols, the rest of the stack. Different applications may run over different protocol stacks. Nevertheless, each one of these different protocol stacks use a common Bluetooth data link and physical layer, see more details on the protocol layers in the next section. Figure 1[5] shows the complete Bluetooth protocol stack as identified in the Specification on top of which interoperable applications supporting the Bluetooth usage models are built. Not all applications make use of all the protocols shown in Figure 1. Instead, applications run over one or more vertical slices from this protocol stack. Typically, additional vertical slices are for services supportive of the main application, like TCS Binary (Telephony Control Specification), or SDP (Service Discovery Protocol). It is worth of mentioning that Figure 1 shows the relations how the protocols are using the services of other protocols when payload data needs to be transferred over air. However, the protocols may also have some other relations between the other protocols. E.g., some protocols (L2CAP, TCS Binary) may use LMP (Link Manager Protocol) when there is need to control the link manager.

As seen in Figure 1, the complete protocol stack comprises of both Bluetooth specific protocols like LMP and L2CAP, and non-Bluetooth-specific protocols like OBEX (Object Exchange Protocol) and UDP (User Datagram Protocol). In designing the protocols and the whole protocol stack, the main principle has been to maximize the re-use of existing protocols for different purposes at the higher layers, instead of re-inventing the wheel once again. The protocol reuse also helps to adapt existing (legacy) applications to work with the Bluetooth technology and to ensure the smooth operation and interoperability of these applications. Thus, many applications already developed by vendors can take immediate advantage of hardware and software systems, which are compliant to the Specification. The Specification is also open, which makes it possible for vendors to freely implement their own (proprietary) or commonly used application protocols on the top of the Bluetooth-specific protocols. Thus, the open Specification permits the development of a large number of new applications that take full advantage of the capabilities of the Bluetooth technology [6].

IV. PROTOCOLS IN BLUETOOTH ARCHITECTURE

The Bluetooth protocol stack can be divided into four layers according to their purpose including the aspect whether Bluetooth SIG has been involved in specifying these protocols. The protocols belong into the layers in the following way.

Table I The Protocol and Layers in Bluetooth Protocol Stack

Protocol layer	*Protocols in the stack*
Bluetooth Core Protocols	Baseband, LMP, L2CAP, SDP
Cable Replacement Protocol	RFCOMM
Telephony Control Protocols	TCS Binary, AT-commands
Adopted Protocols	PPP, UDP/TCP/IP, OBEX, WAP, vCard, vCal, IrMC1, WAE

In addition to the above protocol layers, the Specification also defines a Host Controller Interface (HCI), which provides a command interface to the baseband controller, link manager, and access to hardware status and control registers. This interface is not discussed further in this paper, but more information can be obtained from the functional specification of Bluetooth host controller interface [7][27]. In Figure 1, HCI is positioned below L2CAP but this positioning is not mandatory but HCI can exist e.g., above L2CAP. The Bluetooth Core protocols comprise exclusively Bluetooth-specific protocols developed by the Bluetooth SIG. RFCOMM and the TCS binary protocol have also be developed by the Bluetooth SIG but they are based on the ETSI TS 07.10 [9] and the ITU-T Recommendation Q.931 [10], respectively. The Bluetooth Core protocols (plus the Bluetooth radio) are required by most of Bluetooth devices, while the rest of the protocols are used only as needed. Together, the Cable Replacement layer, the Telephony Control layer, and the Adopted protocol layer form application-oriented2 protocols enabling applications to run over the Bluetooth Core protocols. As mentioned earlier, the Bluetooth Specification is open and additional protocols (e.g., HTTP, FTP, etc.) can be accommodated in an interoperable fashion on top of the Bluetooth-specific transport protocols or on top of the application-oriented protocols shown in Figure 1.

A. Bluetooth Core Protocols

a. Baseband

The Baseband and Link Control layer enables the physical RF link between Bluetooth units forming a piconet [8]. As the Bluetooth RF system is a Frequency-Hopping-Spread-Spectrum system in which packets are transmitted in defined time slots on defined frequencies, this layer uses inquiry and paging procedures to synchronize the transmission hopping frequency and clock of different Bluetooth devices. It provides 2 different kind of physical links with their corresponding baseband packets, Synchronous Connection-Oriented (SCO) and Asynchronous Connectionless (ACL) which can be transmitted in a multiplexing manner on the same RF link. ACL packets are used for data only, while the SCO packet can contain audio only or a combination of audio and data. All audio and data packets can be provided with different levels of FEC or CRC error correction and can be encrypted. Furthermore, the different data types, including link management and control messages, are each allocated a special channel.

b. Audio

Audio data can be transferred between one or more Bluetooth devices, making various usage models possible and audio data in SCO packets is routed directly to and from Baseband and it does not go through L2CAP. Audio model is relatively simple within Bluetooth; any two Bluetooth devices can send and receive audio data between each other just by opening an audio link.

c. Link Manager Protocol (LMP)

The link manager protocol [11] is responsible for link set-up between Bluetooth devices. This includes security aspects like authentication and encryption by generating, exchanging and checking of link and encryption keys and the control and negotiation of baseband packet sizes. Furthermore it controls the power modes and duty cycles of the Bluetooth radio device, and the connection states of a Bluetooth unit in a piconet.

d. Logical Link Control and Adaptation Protocol (L2CAP)

The Bluetooth logical link control and adaptation protocol (L2CAP) [12] adapts upper layer protocols over the baseband. It can be thought to work in parallel with LMP in difference that L2CAP provides services to the upper layer when the payload data is never sent at LMP messages.

L2CAP provides connection-oriented and connectionless data services to the upper layer protocols with protocol multiplexing capability, segmentation and reassembly operation, and group abstractions. L2CAP permits higher level protocols and applications to transmit and receive L2CAP data packets up to 64 kilobytes in length. Although the Baseband protocol provides the SCO and ACL link types, L2CAP is defined only for ACL links and no support for SCO links is specified in Bluetooth Specification 1.0.

e. Service Discovery Protocol (SDP)

Discovery services are crucial part of the Bluetooth framework. These services provide the basis for all the usage models. Using SDP, device information, services and the characteristics of the services can be queried and after that, a connection between two or more Bluetooth devices can be established. SDP is defined in the Service Discovery Protocol specification [13].

C. Cable Replacement Protocol

a. RFCOMM

RFCOMM is a serial line emulation protocol and is based on ETSI 07.10 specification. This “cable replacement” protocol emulates RS-232 control and data signals over Bluetooth baseband, providing both transport capabilities for upper level services (e.g. OBEX) that use serial line as transport mechanism. RFCOMM is specified in [14].

D. Telephony Control Protocol

a. Telephony Control – Binary (TCS BIN)

Telephony Control protocol - Binary (TCS Binary or TCS BIN), a bitoriented protocol, defines the call control signaling for the establishment of speech and data calls between Bluetooth devices. In addition, it defines mobility management procedures for handling groups of Bluetooth TCS devices.

b. Telephony Control – AT Commands

Bluetooth SIG has defined the set of AT-commands by which a mobile phone and modem can be controlled in the multiple usage models. In Bluetooth, AT-commands, which are utilized, are based on ITU-T Recommendation V.250 and ETS 300 916 (GSM 07.07).

E. Adopted Protocols

a. PPP

In the Bluetooth technology, PPP is designed to run over RFCOMM to accomplish point-to-point connections. PPP is the IETF Point-to-Point Protocol and PPP-

Networking is the means of taking IP packets to/from the PPP layer and placing them onto the LAN. Usage of PPP over Bluetooth is described in.

b. TCP/UDP/IP

These protocol standards are defined by the Internet Engineering Task Force and used for communication across the Internet. Now considered as the most widely used protocol family in the world, TCP/IP stacks have appeared on numerous devices including printers, handheld computers, and mobile handsets. Access to these protocols is operating system independent, although traditionally realized using a socket programming interface model. The implementation of these standards in Bluetooth devices allows for communication with any other device connected to the Internet: The Bluetooth device, should it be a Bluetooth cellular handset or a data access point for example is then used as a bridge to the Internet.

TCP/IP/PPP is used for the all Internet Bridge usage scenarios in Bluetooth 1.0 and for OBEX in future versions [15]. UDP/IP/PPP is also available as transport for WAP [16].

c. OBEX Protocol

IrOBEX (shortly OBEX) is a session protocol developed by the Infrared Data Association (IrDA) to exchange objects in a simple and spontaneous manner. OBEX, which provides the same basic functionality as HTTP but in a much lighter fashion, uses a client-server model and is independent of the transport mechanism and transport API, provided it realizes a reliable transport base. Along with the protocol itself, the "grammar" for OBEX conversations between devices, OBEX also provides a model for representing objects and operations. In addition, the OBEX protocol defines a folder-listing object, which is used to browse the contents of folders on remote device.
In the first phase, RFCOMM is used as sole transport layer for OBEX. Future implementations are likely to support also TCP/IP as a transport.

d. Content Formats

vCard and vCalendar are open specifications developed by the versit consortium and now controlled by the Internet Mail Consortium. These specifications define the format of an electronic business card and personal calendar entries and scheduling information, respectively. vCard and vCalendar do not define any transport mechanism but only the format under which data is transported. By adopting the vCard and vCalendar, the SIG will help further promote the exchange of personal information under these well defined and supported formats. The vCard and vCalendar specifications are available from the Internet Mail Consortium and are being further developed by the Internet Engineering Task Force (IETF).
Other content formats, which are transferred by OBEX in Bluetooth, are vMessage and vNote . These content formats are also open standards and are used to exchange messages and notes. They are defined in the IrMC specification, which also defines a format for the log files that are needed when synchronizing data between devices. [17]

B. WAP

Hidden computing usage models can be implemented using the WAP features. The Wireless Application Protocol (WAP) Forum is building a wireless protocol specification that works across a variety of wide-area wireless network technologies. The goal is to bring Internet content and telephony services to digital cellular phones and other wireless terminals. In Figure 2, the protocol stack of the WAP framework is depicted.

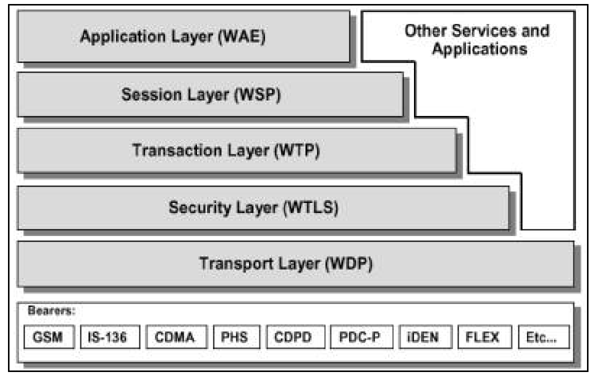

Figure 2 WAP Protocol Framework

The idea behind the choice of WAP is to reuse the upper software applications developed for the WAP Application Environment (WAE). These include WML and WTA browsers that can interact with applications on the PC. Building application gateways which mediate between WAP servers and some other application on the PC makes it possible to implement various hidden computing functionality, like remote control, data fetching from PC to handset etc. WAP servers also allow for both content push and pull between PC and handset, bringing to life concepts like information kiosks.

WAP framework also opens up the possibility of custom applications for handsets that use WML and WML Script as "universal" Software Development Kit [16][28].

f. Content Formats

Supported content formats for WAP over Bluetooth are WML, WMLScript, WTA event, WBMP, and vCard/vCal. These are all part of WAE.

V. BLUETOOTH USAGE MODELS AND PROTOCOLS

Each usage model is accompanied by a Profile. Profiles define the protocols and protocol features supporting a particular usage model. Bluetooth SIG has specified the profiles for these usage models. In addition to these profiles, there are four general profiles that are widely utilized by these usage model oriented profiles. These are the generic access profile (GAP) [18], the serial port profile [19], the service discovery application profile (SDAP) [20], and the generic object exchange profile (GOEP) [15].

A. File Transfer

The file transfer usage model offers the ability to transfer data objects from one device (e.g., PC, smart-phone, or PDA) to another. Object types include, but are not limited to, .xls, .ppt, .wav, .jpg, and .doc files, entire folders or directories or streaming media formats. Also, this usage model offers a possibility to browse the contents of the folders on a remote device. In addition, simple push and exchange operations, e.g., business card exchange are covered in the object push profile, with vCard specified as the format for pushed business card content. In Figure 3, the required protocol stack presented for this usage model is presented. The figure does not show the LMP, Baseband, and Radio layers although those are used underneath (See Figure 1)

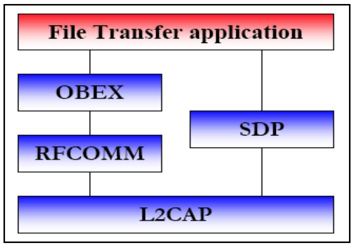

Figure 3 Protocol Stack for File Transfer Protocol

B. Internet Bridge

In this usage model, mobile phone or cordless modem acts as modem to the PC, providing dial-up networking and fax capabilities without need for physical connection to the PC [19]. The dial-up networking scenario of this usage model needs a two-piece protocol stack (in addition to the SDP branch), which is shown in Figure 4. The AT-commands are needed to control the mobile phone or modem and another stack (E.g., PPP over RFCOMM) to transfer payload data. The fax scenario has a similar protocol stack but PPP and the networking protocols above PPP are not used and the application software sends a facsimile directly over RFCOMM [14].

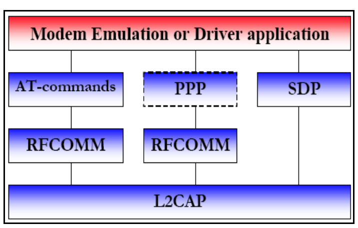

Figure 4 Dial up Networking Protocol Stack

C. LAN Access

In this usage model, multiple data terminals (DTs) use a LAN access point (LAP) as a wireless connection to a Local Area Network (LAN). Once connected, the DTs operate as if it they were connected to the LAN via dialup networking. The DT can access all of the services provided by the LAN. The protocol stack is nearly identical to the protocol stack in the Internet bridge usage model except that the AT commands are not used. [21]

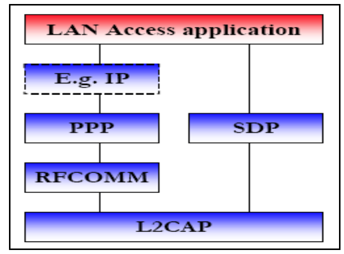

Figure 5 Protocol Stack for LAN (PPP) Usage Model

D. Synchronization

The synchronization usage model provides a device-to-device (phone, PDA, computer, etc.) synchronization of the PIM (personal information management) information, typically phonebook, calendar, message, and note information. Synchronization requires business card, calendar and task information to be transferred and processed by computers, cellular phones and PDAs utilizing a common protocol and format. The protocol stack for this usage model is presented in Figure 6. In the figure, the synchronization application block represents either an IrMC client or an IrMC server software.

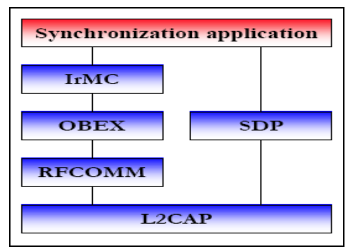

Figure 6 Protocol Stack for Synchronization

E. Three-in-One Phone

Telephone handsets built to this profile may connect to three different service providers. First, telephones may act as cordless phones connecting to the public switched telephone network (PSTN) at home or the office and incurring a fixed line charge. This scenario includes making calls via a voice basestation, making direct calls between two terminals via the base station and accessing supplementary services provided by an external network. Second, telephones can connect directly to other telephones for the purpose of acting as a “walkie-talkie” or handset extension. Referred to as the intercom scenario, the connection incurs no additional charge. Third, the telephone may act as a cellular phone connecting to the cellular infrastructure and incurring cellular charges. The cordless and intercom scenarios use the same protocol stack, which is shown in Figure 7. The audio stream is directly connected to the Baseband protocol indicated by the L2CAP bypassing audio arrow[22][23].

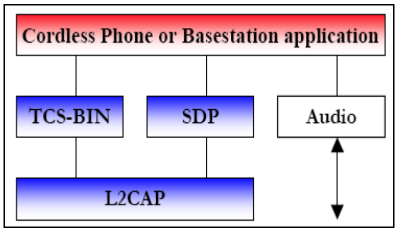

Figure 7 Protocol Stack for Cordless Phone and Intercom Scenario

F. Ultimate Headset

The headset can be wirelessly connected for the purpose of acting as a remote device's audio input and output interface. The headset increases the user's freedom of movement while maintaining call privacy. A common example is a scenario where a headset is used with a cellular handset, cordless handset, or personal computer for audio input and output. The protocol stack for this usage model is depicted in Figure 8 . The audio stream is directly connected to the Baseband protocol indicated by the L2CAP bypassing audio arrow. The headset must be able to send AT-commands and receive result codes. This ability allows the headset to answer incoming calls and then terminate them without physically manipulating the telephone handset [24][25].

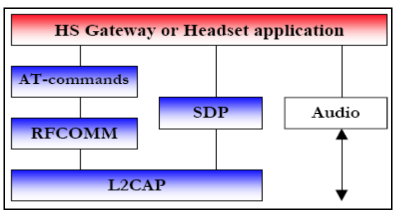

Figure 8 Ultimate HeadSet Protocol Stack

VI. SUMMARY

The Bluetooth protocols are intended for rapidly developing applications using the Bluetooth technology. The lower layers of the Bluetooth protocol stack are designed to provide a flexible base for further protocol development. Other protocols, such as RFCOMM, are adopted from existing protocols and these protocols are only modified slightly for the purposes of Bluetooth. The upper layer protocols are used without modifications. In this way, existing applications may be reused to work with the Bluetooth technology and the interoperability is ensured more easily. The purpose of the Specification is to promote the development of interoperable applications targeted at the highest priority usage models identified by the SIG's marketing team. However, the Specification also services as a framework for further development. Authors are developing architecture in which mobile handset works as a router using Bluetooth technology within this framework. Using the Bluetooth technology with the capabilities of current computers and communications devices, the possibilities for new future wireless applications are unlimited.

VII. REFERENCES

[1] http://en.wikipedia.org/wiki/Bluetooth
[2] Kamani K.C., Kathiriya D. R.,Virparia P. V., Ghodasara Y. R. & Kamani G. J. , Study-How NFC helpful to speed Bluetooth Transactions, Statistics Computer and Applications, Saurashtra University, Rajkot. 13-March-2010.
[3] Kamani K.C., Madurai Kamraj University, M. Phil. Thesis, "Wireless Snesor Network and Near Filed Communication", 2007.
[4] http://www.wirelessdevnet.com/channels/bluetooth/features/bluetooth.html
[5] http://www.cs.utk.edu/~dasgupta/bluetooth/blueprotocols.htm
[6] Bluetooth Protocol Architecture, Version 1.0
[7] I Stojmenovic & N. Zaguia, Bluetooth scatternet formation in ad hoc wireless networks.
[8] B.A. Forouzan, Data Communications and Networking, 4th Edition, Chapter 14.2.1 - Bluetooth Architecture.
[9] ETSI, TS 07.10 version 6.3.0
[10] International Telecommunication Union, "ITU-T Recommen dation Q.931"
[11] Bluetooth Special Interest Group, LMP Specification
[12] Bluetooth Special Interest Group, L2CAP Specification, August,1999
[13] Bluetooth Special Interest Group, SDP Specification
[14] Bluetooth Special Interest Group, RFCOMM with TS 07.10
[15] Bluetooth Special Interest Group, Generic Object Exchange Profile.
[16] http://www.javvin.com/protocolWAP.htm
[17] The Internet Mail Consortium, vCard - The Electronic Business Card Exchange Format, Version 2.1
[18] Bluetooth Special Interest Group, Generic Access Profile.
[19] Bluetooth Special Interest Group, Serial Port Profile.
[20] Bluetooth Special Interest Group, Service Discovery Application Profile.
[21] Amoroso E., Fundamentals of Computer Security Technology, Prentice Hall.
[22] Bluetooth, The Bluetooth Specification, v.1.0B.
[23] http://www.bluetooth.com/developer/specification/specification.asp
[24] Hermelin M. & Nyberg K., Correlation Properties of the Bluetooth Combiner
[25] Bradbury D., Disable the Cable, Personal Computer World, March 2000
[26] http://www.blueradios.com/bluetooth_architecture.pdf
[27] Joronen J., Bluetooth Tunkee Piireihin, Prosessori.
[28] http://www.docstoc.com/docs/25404940/Bluetooth-Protoc ol -Architecture
[29] http://www.tutorialspoint.com/wap/wap_architecture.htm