Modeling and Simulation of Master-driven TDD Wireless Communication Systems

Tae-Jin Lee
School of Electrical and Computer Engineering
Sungkyunkwan University
300 Cheoncheon-Dong, Jangan-Gu
Suwon City, Kyunggi-Do, 440-746, Korea
tjlee@ece.skku.ac.kr

Abstract—

We model and simulate master-driven TDD wireless communication systems, e.g., Bluetooth systems. We model the Bluetooth system and use the BONeS simulation tool to conduct event-driven simulations. In order to support more than seven slave devices in a piconet, a park mode is considered and modeled. We evaluate the performance, i.e., throughput and delay, using simulations when multi-connections (both ACL and SCO connections) are present in a piconet. We show that the data rate of ACL connections may be less than 20 kbps when SCO connection(s) and more than six ACL connections are jointly supported in a piconet. In addition, if up to five ACL connections are supported, the average delay is shown to be maintained less than 20 msec. Our results can serve as a guideline to the design of master-driven TDD wireless communication systems with performance requirements.

I. INTRODUCTION

Communication systems are becoming more and more complex in an effort to enhance performance, to add new features, and to make them user-friendly. These trends are fueled by technological advances in various fields, e.g., telecommunications, semiconductor, computer, and software. Moreover, the requirement of shortening the development cycle of a system becomes tighter. Hence, it gets more difficult to build actual communication systems in a short duration. As a rememdy, modeling and siumlation of real systems before implementation is expected to play a more important role. It can also provide great flexibilty in simulating and testing performance under various operating conditions by slight modification of parameters and/or programs, which is not easy to be attained through making actual systems in a short period of time.

Bluetooth is a low power, low cost, short-range, master-driven TDD (Time Division Duplex) wireless communication system to make mobile and/or personal computer and communication devices communicate one another. It has been gaining much interest from many industries, drawing more than 2000 computer, communications, and semiconductor industry members, and as such it has set up an organization within IEEE, i.e., IEEE 802.15, to put vast effort in making the technology into a standard. As described in the Profiles of the Bluetooth Specification [4], Bluetooth is expected to provide numerous wireless services ranging from synchronization to wireless Internet between heterogeneous devices such as notebook PCs, PDAs, mobile phones, and headsets.

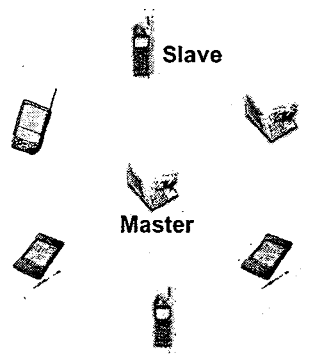


Fig. 1. A piconet with one master and six slaves.

In order to truly make various devices communicate in a piconet, a small network consisting of a master and slave devices, see Fig. 1, it is required to employ point-to-multipoint communication functionality in Bluetooth-enabled devices. To support point-to-multipoint communications, the master in a piconet can support up to seven active slave devices under a normal connection mode. In addition, it can support up to 255 slave devices by using a power saving mode, i.e., park mode. Although the Specification of the Bluetooth system [3] describes that Bluetooth has point-to-multipoint communication ability, Bluetooth-enabled products with one-to-many communication capability have not been pervasive so far.

Since the Bluetooth system is complex and it is hard to implement and evaluate performance of the actual system under the strict time requirement, it would be better to predict actual performance of Bluetooth systems by simulation so that it can be applied to the design of systems. Johansen et. al. [6] modeled the Bluetooth system for a few scenarios and evaluated performance. They focus on performance of a scatternet, i.e., a set of piconets, in which each piconet consists of a few Bluetooth-enabled devices. In [8], performance evaluation under interference between piconets is addressed, in which each piconet has a single master and slave. Das et. al. [2] tried to enhance performance of the Bluetooth system supporting up to seven active slave devices based on the simulation.

However, performance of a piconet with more than seven devices supporting both ACL (Asynchronous Connection-Less) and SCO (Synchronous Connection-Oriented) connections in a piconet has not been studied. In this paper, we model, simulate and evaluate a piconet consisting of more than seven slave devices with possible support of both ACL and SCO connections for point-to-multipoint communications in the Bluetooth system. We investigate the impact of SCO connections and a power saving mode, i.e., a park mode, on the performance (throughput and delay) and adopt the result to the design of a system. This paper is organized as follows. In §II, we describe modeling and simulation technigues of the Bluetooth system. We then evaluate performance of the system in §III. Conclusion is followed in §IV.

II. MODELING

Bluetooth is a low power and low cost wireless communication system covering a small personal area within 10m. It operates in 2.4 GHz unlicensed ISM (Industrial Science Medical) radio frequency band and uses frequency hopping transceiver, in which 79 frequencies are pseudo-randomly selected to carry signals of 1 MHz bandwidth resulting in fast 1600 hoppings per second. It modulates data bits to signals using a GFSK (Gaussian Frequency Shift Keying) modulation scheme and utilizes slotted TDD (Time Division Duplex), in which a master and slave devices can transmit packets at even-numbered and odd-numbered slots, respectively, with the slot interval of 625 μsec [3], [5].

In a piconet, two or more units sharing the same channel, slave devices with ACL (Asynchronous Connection-Less) connections are allowed to transmit data packets right after the TX slot of a master device, only when they are explicitly "polled" by the master. Thus ACL connections provide best-effort service. The maximum effective data rate one can achieve from an ACL connection between a single master and slave is 723 kbps. The number of slots for a packet can occupy varies among one, three, and five according to the amount of data to be transmitted. For SCO (Synchronous Connection Oriented) connections, periodic slots are strictly reserved for SCO packets between a master and slaves resulting in a guaranteed rate of 64 kbps.

Data packets can be protected by error correction schemes, e.g., FEC (Forward Error Correction) or ARQ (Automatic Repeat Request), depending on the status of wireless channel. Up to seven active slave devices can be supported in a piconet, and using a park mode, up to 255 devices can be supported. Piconets can form a scatternet to interconnect devices among piconets. Since Bluetooth is expected to be employed in small mobile devices like PDAs and mobile phones, battery power consumption needs to be minimized. In this sense, power saving modes, i.e., hold, sniff and park mode, are defined in the specification.

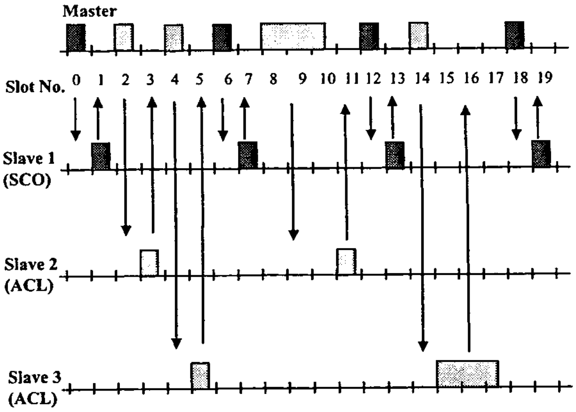


Fig. 2. ACL and SCO connections between a master and three slaves.

In this paper, we consider a piconet consisting of a single master and many slaves, see Fig. 1. Packet transmission and reception are synchronized by the system clock (625 μsec slot interval). We will focus on throughput and delay performance in point-to-multipoint communication environment. We try to simplify and abstract the system as much as possible while preserving the essence of the system which affects the actual performance of the system. We use BONeS simulation tool [1] to capture the behavior of the Bluetooth system. BONeS is the modeling and performance evaluation tool providing event-driven simulation environment and graphical user interface for the construction of functional blocks and simulation objects.

In order to poll multi-slaves, a polling list needs to be maintained in a master device. A master polls slaves in a round-robin fashion. In our model, multi-SCO connections (up to three SCO connections) can be supported. When there are seven active slaves in a piconet and a new device is about to enter the piconet, an existing slave may need to be parked since Bluetooth can support only seven active slaves.

We propose a First-Active-First-Park scheduling mechanism for the parking/unparking of slaves. In the mechanism, the oldest active slave in the piconet is forced to enter a park mode first when parking of a slave is required due to a new connection (Fig. 3). In addition after some pre-specified parking time ($Tpark$) exceeds for a parked slave, the oldest one among active slaves is parked, and then the parked slave with expired parking time is unparked and enters the active slaves (Fig. 3). This scheduling mechanism will provide fair access to channel bandwidth to all the slaves in a piconet while admitting new connections. Parameters for a park mode are tabularized in Table I. The simulation environment on the BONeS simulation tool is illustrated in Fig. 4.

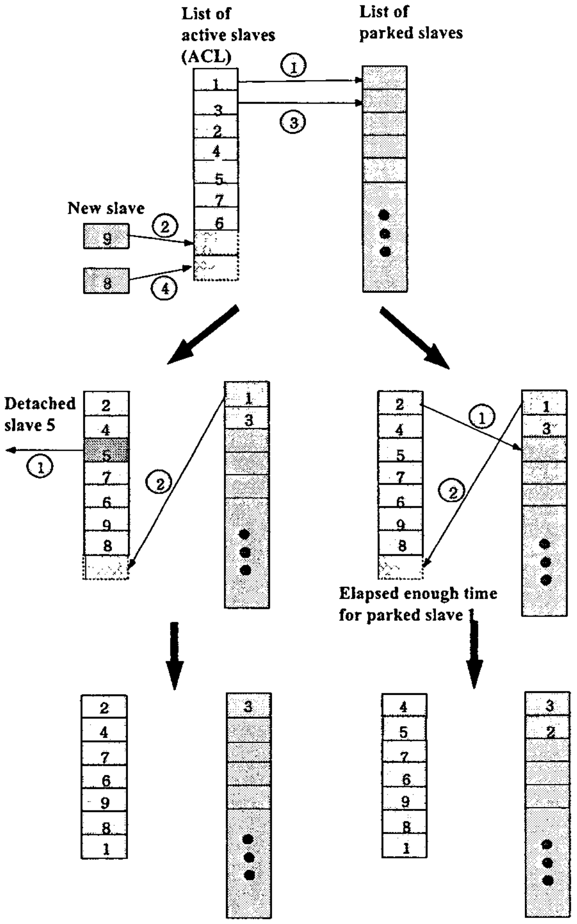
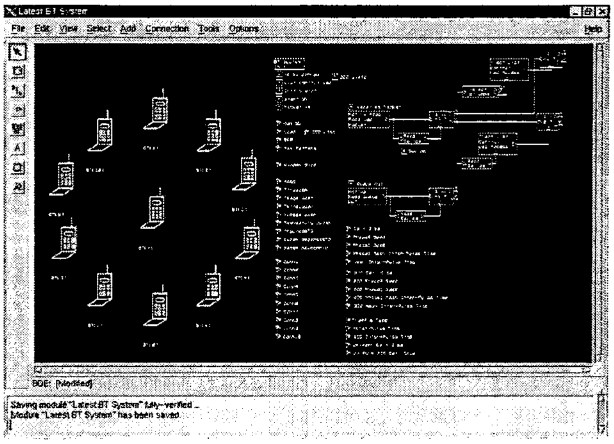


Fig. 3. Control of parking and unparking slave devices.



Saving module "Latest BT System" fully-verified ...
Module "Latest BT System" has been saved.

Fig. 4. Simulation environment using BONeS tool.

Park parameter	Value	Unit
T_B	2.56	sec
N_B	24	slots
N_{B_sleep}	1	
D_{access}	80	slots
T_{access}	4	slots
M_{access}	4	
T_{park}	5	sec

TABLE I
PARK PARAMETERS USED IN THE SIMULATION.

III. PERFORMANCE EVALUATION

A. Throughput performance

Throughput performance is evaluated when there are one to ten slave devices, for which an ACL or SCO connection is established each. We assume that packets are always transmitted and received in the corresponding TX/RX slots. In other words, a master polls a slave with an ACL connection in a round-robin fashion and it transmits a packet to the polled slave. When a slave is polled by a master, the slave sends a packet to the master in the consecutive TX slot of the reception of the packet from the master. Various packet types can be used for ACL connections when there are no SCO connections in a piconet. Only DH1 or DM1 packets are assumed to be transmitted if at least one SCO connection is present in a piconet. In order to see the maximum possible performance, asymmetric TX/RX is considered, where different packet types are used for down (master to slave) and up (slave to master) stream traffic. In our simulation, each master and slave is assumed to use a specific packet type, e.g., DH5 for a master and DH1 for slaves.

First, throughput performance is evaluated when one SCO connection is present (DH-1HV3, DH-1HV2), see Fig. 5. Since we have a SCO connection, 33 % of total available slots must be reserved for the voice (SCO) traffic [3]. So the number of slots available for ACL connections becomes smaller resulting in lowered performance for ACL connections. When one ACL and one SCO connection are established in a piconet, the rate of the ACL connection is 114 kbps (DH-1HV3), which can be compared with the rate of 370 kbps in previous simulation with two ACL connections (DH5). This 69.2 % decrease in throughput results from reduced available slots and packet size. Throughput performance becomes worse as the number of ACL connections increases. When six ACL connections and one SCO connection are supported, the rate shows 19 for DH-1HV3. If nine ACL connections and one SCO connection are supported using a park mode, the rate drops to 12 kbps for DH-1HV3, respectively.

Next, when two SCO connections are supported (DH-2HV3), performance is considered, see also Fig. 5. In this case, two SCO connections occupy more slots (67 % of total available slots) than one SCO connection, and thus lower data rates are expected. When three ACL connections with DH packets and two SCO connections with HV3 packets are in a piconet, performance already starts to become less than 20 kbps, i.e., 19 kbps. It can even degrade

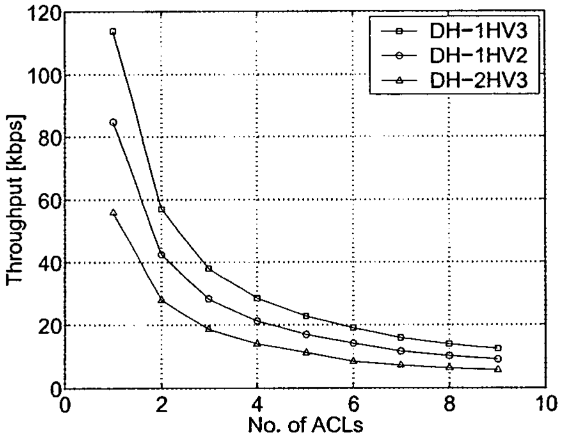


Fig. 5. Throughput of ACL connections with DH packets and SCO connections.

to 7 kbps if more than eight ACL and two SCO connections communicate using a park mode in the piconet (DH-2HV3). Thus, when many Bluetooth-enabled devices are contending channel bandwidth for both SCO and ACL connections in a piconet, these low data rates may not be suitable for some applications requiring high throughput.

B. Delay performance

Next, we evaluate delay performance. In the previous subsection, we have shown that SCO connections may significantly degrade system performance. Thus, if high throughput is required for the purpose of efficient data transmission like file transfers, one is likely to refrain from simultaneous support of SCO and ACL links. That is, use of only ACL connections would be a typical scenario for throughput-intensive applications. Thus, we investigate delay performance in such cases.

We consider one to seven slave devices with ACL connections in a piconet. In this scenario, packet types are adaptively selected among DM1, DM3 and DM5 according to the amount of data waiting for transmission in buffers. We consider only DM packets assuming that data packets are vulnerable to bit errors due to error-prone wireless channel characteristics. To ensure reliable data transmission, FEC (Forward Error Correction) and ARQ (Automatic Repeat Request) [3] are employed. We simulated simple wireless channel model, where BER (Bit Error Rate) is constant in a smulation. Although it needs to be modeled more accurately, we believe that the BER model captures the performance variation under changing BER. The BER of wireless channel is assumed to be 0.001.

As traffic sources, MMPP (Markov-Modulated Poisson Process) is used, which is shown to be well matched with multimedia traffic [7]. Each source is modeled by a two-state Markov chain, where on- and off-periods are exponentially distributed with mean of α^{-1} and β^{-1} sec, respectively, see Fig. 6. The rate transition matrix between on- and off-period is characterized by

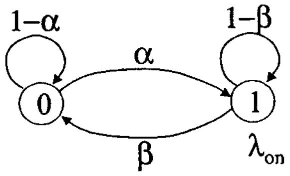


Fig. 6. A two-state model of MMPP traffic source.

$$Q = \begin{bmatrix} -\beta & \beta \\ \alpha & -\alpha \end{bmatrix}.$$

Thus the steady state probability of being in on-period $\pi_{on} = \beta/(\alpha + \beta)$. During on-period, packets are generated with Poisson arrival rate of λ_{on} packets/sec, and no packets are generated during off-period. Thus the average packet generation rate

$$\bar{\lambda} = \pi_{on}\lambda_{on}, \qquad \text{packets/sec.}$$

Let the average service rate of Bluetooth

$$\bar{\mu} = \lambda_{on}, \qquad \text{packets/sec}$$

and the traffic load $\rho = \bar{\lambda}/\bar{\mu}$. The average message size, average number of packets generated during on-period, is given by

$$\bar{m} = \alpha^{-1}\lambda_{on}, \qquad \text{packets.}$$

It is assumed that $\bar{m} = 250$ packets with the packet size of 100 bytes. If we assume symmetric TX/RX, the maximum rate for a single slave would achieve is 286.7 kbps. And if seven slaves share the bandwidth, 40.96 kbps is allocated to each slave. So we let average service rate of each connection $\bar{\mu} = 40.96$ kbps. One can then design $\lambda_{on}, \alpha^{-1}$, and β^{-1} to generate desired traffic. We have generated three types of traffic, i.e., $\rho = 0.2, 0.5$ and 0.8, whose effective rates are 8, 22, and 33 kbps, respectively. For each direction, i.e., master-to-slave and slave-to-master, there exists a traffic source and an associated data buffer.

Average packet delay, time between arrival of a new packet (data) into a buffer for transmission and reception of the packet at a receiver, is measured for each of traffic load as the number of ACL connections changes. Fig. 7 shows the average packet delays measured at slaves and a master, respectively. Up to four ACL connections, the average delay is shown to be less than 10 msec, which is suitable for delay-sensitive applications, and it increases as the number increases. When the number of ACLs is five, the average delay still remains within 20 msec. When the traffic load is heavy, i.e., $\rho = 0.8$, and the number of connections is seven, the delay would exceed 100 msec. Thus, one can estimate

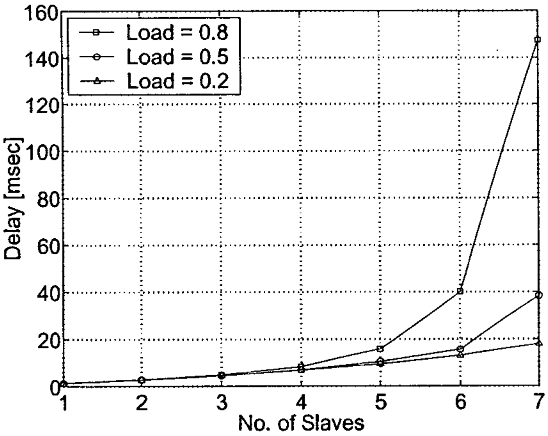


Fig. 7. Average delay of packets from slaves to a master with no SCO connections.

the average delay given traffic load and a number of slave devices, which can be used for the design of a system

We have also simulated the varying channel environment to observe the impact of error-prone channel environment, see Fig. 8 and 9. In this simulation, the number of ACL connections is two. In order to protect bit errors that may be caused during transmission, FEC is employed. The packet type used in the simulation is DM5. The throughput and delay may deteriorate if BER exceeds 0.002, which can be observed in real fading channel environment. If BER is greater than 0.005, the performance becomes worse more than 50 %.

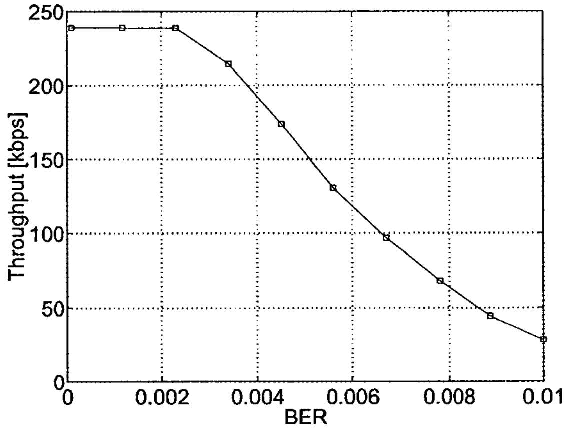


Fig. 8. Throughput performance as BER changes (in master).

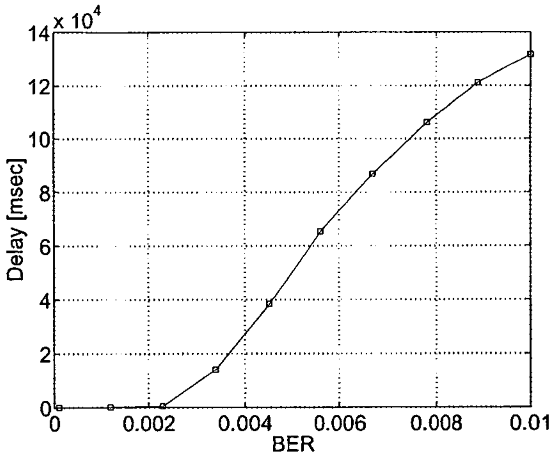


Fig. 9. Delay performance as BER changes (in master).

IV. CONCLUSION

We have modeled, simulated and evaluated the performance of a master-driven TDD wireless communication system, the Bluetooth system, supporting point-to-multipoint communications. It has been shown that the performance of ACL connections may degrade significantly, i.e., less than 20 kbps, when SCO connection(s) and more than six ACL connections are jointly supported. Thus, when many Bluetooth-enabled devices are contending channel bandwidth for both SCO and ACL connections in a piconet, resulting data rates may not be suitable for throughput-intensive applications. We have shown that one can control more than seven slave devices using a park mode with graceful degradation of performance. We have also shown that average delay experienced by a master and slaves is less than 20 msec if up to five slaves with ACL connections are supported. In addition, it has been shown that it indicates good throughput and delay performance under the BER of 0.002. In order to guarantee low delay requirements, the number of slaves may need to be limited. Our simulation provides a performance metric when multi-slaves (both ACL and SCO connections) are supported using a park mode. These results can serve as a guideline to the design of Bluetooth systems with throughput and delay requirements before implementation.

REFERENCES

[1] Cadence. *BONeS Designer ver. 4.0.* 1999.

[2] A. Das, A. Ghose, A. Razdan, H. Saran, and R. Shorey. Enhancing performance of asynchronous data traffic over the Bluetooth wireless ad-hoc network. In *Proc. of IEEE INFOCOM*, pages 591–600, 2001.

[3] Bluetooth Special Interest Group. *Core, Specification of the Bluetooth System ver. 1.1.* Nov. 2000.

[4] Bluetooth Special Interest Group. *Profiles, Specification of the Bluetooth System ver. 1.1.* Nov. 2000.

[5] J. C. Haartsen and S. Mattisson. Bluetooth - a new low-power radio interface providing short-range connectivity. *Proceedings of the IEEE*, 88(10):1651–1661, Oct. 2000.

[6] P. Johansson, N. Johansson, U. Körner, J. Elg, and G. Svennarp. Short range radio based ad-hoc networking: Performance and properties. *Proc. of IEEE ICC 99*, pages 1414–1420, 1999.

[7] S. Q. Li and C. L. Hwang. On the convergence of traffic measurement and queueing anlysis: A statistical-match queueing (SMAQ) tool. *IEEE/ACM Trans. Networking*, pages 95–110, Feb. 1997.

[8] S. Zürbes, W. Stahl, K. Matheus, and J. Haartsen. Radio network performance of Bluetooth. *Proc. of IEEE VTC 2000*, pages 1563–1567, 2000.