

Research Article

Novel MAC Protocol and Middleware Designs for Wearable Sensor-Based Systems for Health Monitoring

Kyeong Hur,¹ Won-Sung Sohn,¹ Jae-Kyung Kim,¹ and YangSun Lee²

¹ Department of Computer Education, Gyeongin National University of Education, Gyesan-Dong, San 59-12, 45 Gyodae-Gil, Gyeonggi-Gu, Incheon 407-753, Republic of Korea

² Department of Computer Engineering, Seokyeong University, 16-1 Jungneung-Dong, Sungbuk-Ku, Seoul 136-704, Republic of Korea

Correspondence should be addressed to Won-Sung Sohn; sohnws@ginue.ac.kr

Received 16 November 2012; Accepted 11 December 2012

Academic Editor: Sabah Mohammed

Copyright © 2013 Kyeong Hur et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

We propose a middleware platform built on wireless USB (WUSB) over wireless body area networks (WBAN) hierarchical protocol for wearable health-monitoring systems (WHMS). The proposed middleware platform is composed of time-synchronization and localization solutions. It is executed on the basis of WUSB over WBAN protocol at each wearable sensor node comprising the WHMS. In the platform, firstly, the time-synchronization middleware is executed. After that, a WHMS host calculates the location of a receiving sensor node by using the difference between the times at which the sensor node received different WBAN beacon frames sent from the WHMS host. The WHMS host interprets the status and motion of the wearable body-sensor objects.

1. Introduction

Wearable health-monitoring systems (WHMS) have drawn a lot of attention from the research community and the industry during the last decade as it is pointed out by the numerous and yearly increasing corresponding research and development efforts [1]. To address this demand, a variety of system prototypes and commercial products have been produced in the course of recent years, which aim at providing real-time feedback information about one's health condition, either to the user himself, to a medical center, or straight to a supervising professional physician, while being able to alert the individual in case of possible imminent health-threatening conditions. In addition to that, WHMS constitute a new means to address the issues of managing and monitoring chronic diseases, elderly people, postoperative rehabilitation patients, and persons with special abilities [1, 2].

Wearable systems for health monitoring may comprise various types of miniature sensors, wearable or even implantable [1]. These biosensors are capable of measuring significant physiological parameters like heart rate, blood pressure, body and skin temperature, oxygen saturation, respiration rate, electrocardiogram, and so forth. The obtained measurements are communicated either via a wireless or a wired link

to a central node, for example, a personal digital assistant (PDA) or a microcontroller board, which may then in turn display the according information on a user interface or transmit the aggregated vital signs to a medical center. The previous illustrates the fact that a wearable medical system may encompass a wide variety of components: sensors, wearable materials, smart textiles, actuators, power supplies, wireless communication modules and links, control and processing units, interface for the user, software, and advanced algorithms for data extracting and decision making.

The wireless body area network (WBAN) is a promising technology that can revolutionize next-generation healthcare applications. Developing a unifying WBAN standard that addresses the core set of technical requirements is the quintessential step to unleash the full potential of WBAN and is currently under discussion in the IEEE 802.15.6 Task Group [2]. The IEEE 802.15.6 WBAN standard is used in or around a body [2, 3]. It is designed to serve advanced medical and entertainment options enabled by this standard. It will allow medical equipment manufacturers and consumer electronics manufacturers to have small, power-efficient, and inexpensive solutions to be implemented for a wide range of devices. This standard considers effects on portable antennas due to the presence of a person (varying with male, female,

skinny, heavy, etc.), radiation pattern shaping to minimize specific absorption rate (SAR) into the body, and changes in characteristics as a result of the user motions.

WiMedia Alliance is developing the specifications of the PHY, MAC, and convergence layers for Ultrawideband (UWB) systems with participation from more than 170 companies. Also, it has been promoting the standardization and adaptation of UWB for high rate-wireless personal area network (HR-WPAN) that enables the multimedia and high-speed data communication [4, 5]. WiMedia Alliance has completed the specification of WiMedia distributed-MAC (D-MAC), and this enables various applications, such as wireless USB (WUSB) [4], Wireless 1394, and Wireless IP, to operate on WiMedia D-MAC. The WiMedia D-MAC supports a distributed-MAC approach. In contrast to IEEE 802.15.3, D-MAC makes all devices have the same functionality, and networks are self-organized and provide devices with functions such as access to the medium, channel allocation to devices, data transmission, quality of service, and synchronization in a distributed manner [5].

The term middleware refers to the software layer located between the OS and the application layer. Middleware can be further classified based on “submiddleware” functionalities including time synchronization, location, battery power, and network functionalities. As in all distributed systems, time synchronization is very important in a sensor network since the design of many protocols and the implementation of applications require precise timing information. In forming an energy-efficient radio schedule, conducting in-network processing such as data fusion, performing RF ranging, and tracking a certain object; for example, time synchronization is certainly required.

Time-synchronization problem has been investigated thoroughly in Internet and LANs. Complex protocols such as NTP [6] have been developed to keep the Internet’s clocks ticking in phase. However, the peculiar constraints of sensors and adhoc network topologies of wireless sensor networks (WSNs) impose specific requirements on protocol design of time synchronization for WSN applications. First, since the amount of energy available to battery-powered sensors is quite limited, time synchronization must be implemented in an energy-efficient way. Second, the small size of a sensor node imposes restrictions on computational power and storage space. Therefore, traditional synchronization schemes such as NTP and GPS are not suitable for WSNs because of complexity and energy issues, cost efficiency, limited size, and so on [7].

In this paper, we propose a middleware platform built on wireless USB (WUSB) over wireless body area networks (WBANs) hierarchical protocol for wearable health-monitoring systems (WHMS). The proposed middleware platform is composed of time-synchronization and localization solutions. It is executed on the basis of WUSB over WBAN protocol at each sensor node comprising the WHMS. In the platform, firstly, the time-synchronization middleware is executed. After that, a WHMS host calculates the location of a receiving sensor node by using the difference between the times at which the sensor node received different WBAN beacon frames sent from the WHMS host. The WHMS host

interprets the status and motion of the attached body-sensor objects.

2. Features of IEEE 802.15.6 WBAN Protocol

To provide or support time-referenced allocations in its wireless body area network (WBAN), a hub shall establish a time base as specified in [3] which divides the time axis into beacon periods (superframes) regardless of whether it is to transmit beacons. In such cases, the hub shall transmit a beacon in each beacon period, except in inactive superframes, unless prohibited by regulations such as imposed in MICS band. The hub may shift (rotate) its beacon transmission time from one offset from the start of current beacon period to another offset from the start of the next beacon period, thereby shifting the time reference for all scheduled allocations, to prevent large-scale repeated transmission collisions between its WBAN and neighbor WBANs [3].

In cases where a hub is not to provide or support time-referenced allocations in its WBAN, it may operate with establishing neither a time base nor superframe boundaries and hence without transmitting beacons at all. Equivalently, a hub shall operate in beacon mode transmitting a beacon in every beacon period other than in inactive superframes to enable time-referenced allocations, unless regulations as applicable in the MICS band disallow beacon transmission. In the latter case, a hub shall operate in nonbeacon mode transmitting no beacons, with superframe and allocation slot boundaries established if access to the medium in its WBAN involves time referencing, or without superframe or allocation slot boundaries if access to the medium in its WBAN involves no time referencing. In the beacon mode, a hub shall divide each active beacon period into applicable access phases as illustrated in Figure 1. The hub may announce some superframes (beacon periods) as inactive superframes where it transmits no beacons and provides no access phases if there are no allocation intervals scheduled in those superframes [3].

The hub shall place the access phases—exclusive access phase 1 (EAP1), random access phase 1 (RAP1), type I/II access phase, exclusive access phase 2 (EAP2), random access phase 2 (RAP2), type I/II access phase, and contention access phase—in the order stated and shown above. The hub may set to zero the length of any of these access phases but shall not have RAP1 shorter than the guaranteed minimum length communicated in connection assignment frames sent to nodes that are still connected with it. To provide a nonzero length CAP, the hub shall transmit a preceding B2 frame [3].

A type I/II access phase is either a type I or type II access phase as described below but not both. The two type I/II access phases may be both of type I, both of type II, or one of type I and the other of type II. If one is of type I and the other is of type II, either one may appear before the other. The hub may schedule uplink allocation intervals, downlink allocation intervals, and bilink allocation intervals anywhere in a type I access phase. It may improvise type I and type II polled allocation intervals as well as posted allocation intervals anywhere outside the scheduled allocation intervals

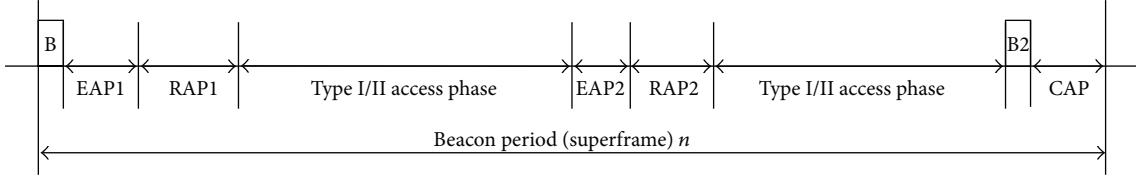


FIGURE 1: Layout of access phases in a beacon period (superframe) for beacon mode.

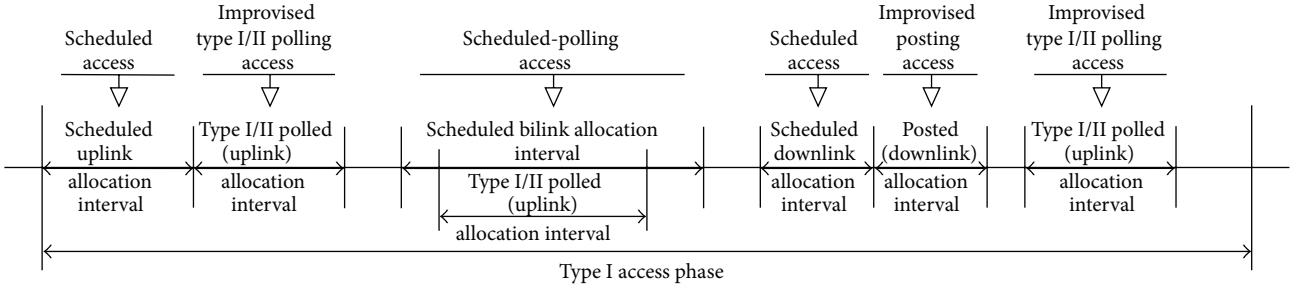


FIGURE 2: Allocation intervals and access methods permitted in a type I access phase.

in the type I access phase. It may also provide type I and type II polled allocation intervals within scheduled bilink allocation intervals in a type I access phase.

A type I polled allocation is conveyed in terms of its time duration (the maximum time the polled node may use for its frame transactions in the allocation). And a type II polled allocation is conveyed in terms of a frame count (the maximum number of frames the polled node may transmit in the allocation). These allocation intervals along with the corresponding access methods by which they are obtained are illustrated in Figure 2.

The hub may schedule bilink allocation intervals and delayed bilink allocation intervals anywhere in a type II access phase, except that it shall not schedule any bilink allocation intervals after delayed bilink allocation intervals in the same type II access phase. It may improvise type II, but not type I, polled allocation intervals as well as posted allocation intervals anywhere outside the scheduled bilink allocation intervals and delayed bilink allocation intervals in the type II access phase. It may also provide type II, but not type I, polled allocation intervals within scheduled bilink allocation intervals and delayed bilink allocation intervals. These allocation intervals along with the corresponding access methods by which they are obtained are illustrated in Figure 3 [3].

In exclusive access phase 1 (EAP1), random access phase 1 (RAP1), exclusive access phase 2 (EAP2), random access phase 2 (RAP2), and contention access phase (CAP), allocations may only be contended allocations, which are nonreoccurring time intervals valid per instance of access. The access method for obtaining the contended allocations shall be CSMA/CA if pRandomAccess is set to CSMA/CA, or slotted Aloha access if pRandomAccess is set to slotted Aloha.

A hub or nodes may obtain contended allocations in EAP1 and EAP1 if it needs to send data type frames of the highest user priority (i.e., containing an emergency or medical event

report). The hub may obtain such a contended allocation pSIFS after the start of EAP1 or EAP2 without actually performing the CSMA/CA or slotted Aloha access procedure. Only nodes may obtain contended allocations in RAP1, RAP2, and CAP to send management or data type frames [3].

To send data type frames of the highest user priority based on CSMA/CA, a hub or a node may treat the combined EAP1 and RAP1 as a single EAP1, and the combined EAP2 and RAP2 as a single EAP2, so as to allow continual invocation of CSMA/CA and improve channel utilization. To send data type frames of the highest user priority based on slotted Aloha access, a hub or a node may treat RAP1 as another EAP1 but not a continuation of EAP1, and RAP2 as another EAP2 but not a continuation of EAP2, due to the time-slotted attribute of slotted Aloha access.

3. Data Flow in WUSB Protocol

WUSB is the technology-merged USB with UWB based on success of wired USB, and it can apply to WPAN applications as well as PAN applications like wired USB. Because WUSB specification has defined high-speed connection between a host and a device for the compatibility with USB 2.0 specification, it can be adapted easily for wired USB applications. WUSB connects WUSB devices with the WUSB host using a “hub and spoke” model [4]. The WUSB host is the “hub” at the center, and each device sits at the end of a “spoke.” Each spoke is a point-to-point connection between the host and the device. Like this, the network formed by one host and several devices is referred to as the WUSB cluster.

WUSB hosts can support up to 127 devices and because WUSB does not have physical ports there is no need, nor any definition provided, for hub devices to provide port expansion. There is only one host in any WUSB cluster. And that host performs to transmit/receive a data with devices in the WUSB cluster. Also, it schedules the exchange of data

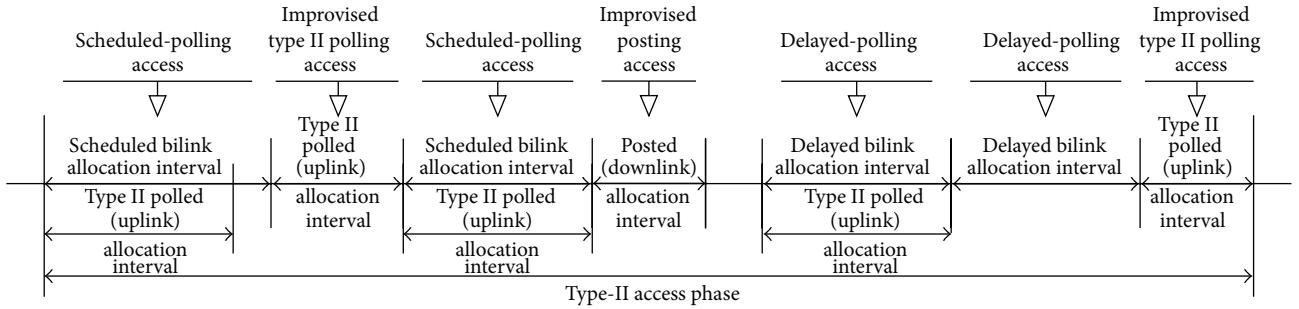


FIGURE 3: Allocation intervals and access methods permitted in a type II access phase.

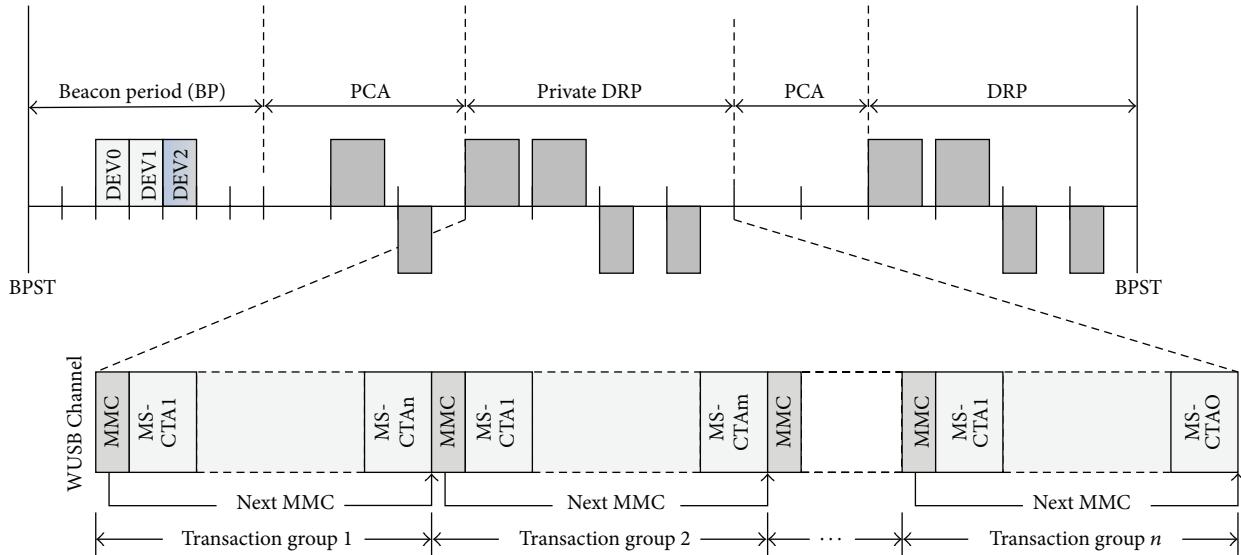


FIGURE 4: The example of the data exchange between WUSB devices through WiMedia D-MAC.

between WUSB host and WUSB devices and allocates time slots and channel bandwidths to WUSB devices in its own cluster. Because each WUSB cluster can overlap with the other at minimum interference, it can coexist with several WUSB clusters within the same communication environment. The distributed nature of D-MAC protocol can provide full mobility support and achieves scalable, fault-tolerant medium access method [4]. Thus, WUSB protocol based on WiMedia D-MAC can provide full mobility support.

WUSB defines a WUSB Channel which is encapsulated within a WiMedia MAC superframe via private DRP reservation blocks. The WUSB Channel is a continuous sequence of linked application-specific control packets, called micro-scheduled management commands (MMCs), which are transmitted by the host within the private DRP reservation blocks. Figure 4 shows the relationship between WiMedia MAC and WUSB.

The microscheduled management command (MMC) is the fundamental element of the Wireless USB protocol. MCs are used to help devices discover information about a WUSB cluster, notify their intentions, manage power, and schedule data transmissions efficiently to attain very high

throughputs. The general structure of an MMC Control packet is showed in Figure 5 and detailed in Table 1.

A WUSB Channel consists of a continuous sequence of MMC transmissions from the host. The linked stream of MCs is used primarily to dynamically schedule channel time for data communications between host applications and WUSB endpoints. An MMC specifies the sequence of microscheduled channel time allocations (MS-CTAs) up to the next MMC within a reservation instance or to the end of a reservation instance. It may be followed by another MMC without the existence of MS-CTAs between the two MCs. In this case, the MMC is only used to convey command and control information. The channel time between two MCs may also be idle time, where no MS-CTAs are scheduled. The MS-CTAs within a reservation instance can only be used by the devices that are members of the associated WUSB cluster. The direction of transmission and the use of each MS-CTA are fully declared in each MMC instance. An MMC can declare an MS-CTA during any channel time following the MMC.

An MMC contains the information elements necessary to identify the WUSB Channel or declare any MS-CTAs or

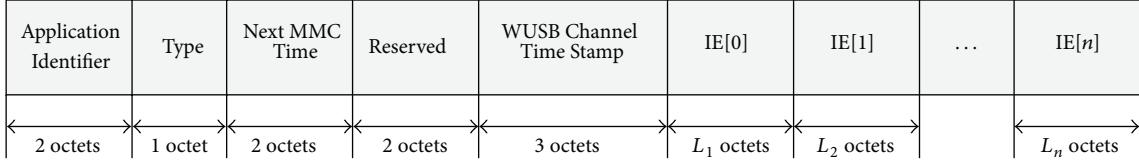


FIGURE 5: The general structure of an MMC Control packet.

TABLE 1: Detailed field description of MMC Control packet.

Field	WUSB equivalents
Application Identifier	Wireless USB: 0100H
Type	MMC command type: 01H
Next MMC Time	The number of microseconds from the beginning of this field indicates the value of the next MMC packet
Reserved	This field is reserved and should be set to “zero”
WUSB Channel Time Stamp	Timestamp provided by the host based on a free running timer in the host. The value in this field indicates the value of the host free running clock when MMC transmission starts. This timestamp is formatted into two fields as follows: (i) 0~6 bits: microsecond count. The microsecond count rolls over after 125 microseconds. Each time it rolls over the 1/8th millisecond count is incremented (ii) 7~23 bits: 1/8th millisecond count. This counter increments every time the microsecond counter wraps
IE [0 – n]	Array of information elements. There must be at least one IE

other information elements that are used for command and control. The MMC is a broadcast control packet that is for receipt only by devices that are members of the WUSB cluster. The host must use the broadcast cluster ID value in the DestAddr field of an MMC packet’s MAC, WiMedia MAC header. This technique identifies this packet transmission as a broadcast targeting all devices in a WUSB cluster and avoids potential confusion at non-WUSB devices in listening range of the host. The MMC payload must be encapsulated within a WiMedia MAC secure packet; however, its data payload is transmitted in plain text, thus using the security encapsulation for authentication purpose only.

A host is required to implement the WiMedia MAC protocol and to establish and maintain WUSB Channels by allocating sequences of private DRP Reservation in WiMedia MAC. A device may implement the full WiMedia MAC protocol; however, it is nominally only required to implement the WUSB protocol which operates within the WUSB Channel.

As mentioned above, a WUSB host and WUSB devices must include a DRP IE in their beacon frames to protect the WUSB Channel. When a WUSB host becomes active, it must choose a PHY channel to operate as a WUSB Channel. Once the host is beaconing, it then establishes a WUSB Channel by DRP reservation for WUSB data communications. In this case, WUSB host is the DRP Reservation Owner, and WUSB device is the DRP Reservation Target. Thus, WUSB device must be able to determine which MASs are available for communication with the WUSB host. Since there are various devices in the WUSB cluster, the WUSB device identifies the WUSB host’s DRP IE based on the following keys.

- (i) Reservation type field is private.
- (ii) Stream index field has the value of the WUSB Channel’s stream index.
- (iii) Owner DevAddr field is set to the WUSB Channel’s broadcast cluster ID.

The WUSB device identifies a cluster member’s DRP IE based on the following keys.

- (i) Reservation Type field is Private.
- (ii) Stream Index field has the value of the WUSB Channel’s stream index.
- (iii) Owner DevAddr field is set to the WUSB host’s DevAddr.

Figure 6 shows the current operation of DRP reservation in WUSB protocol. A WUSB host uses the GetStatus (MAS Availability) request shown in Figure 7 to retrieve a device’s MAS Availability information. A WUSB device that receives the GetStatus (MAS Availability) request from the WUSB host accumulates the information from its neighbors’ beacon about available MASs. Then, the WUSB device responds to the GetStatus (MAS Availability) request through the bmMASAvailability field in the GetStatus request. In the format of the GetStatus request, bmMASAvailability field is a 256-bit map, where each bit location corresponds to an MAS slot in the WiMedia D-MAC Layer superframe. A 1 B in a bit location means that the device is available for a reservation in the corresponding MAS slot. A 0 B indicates that the device is not available.

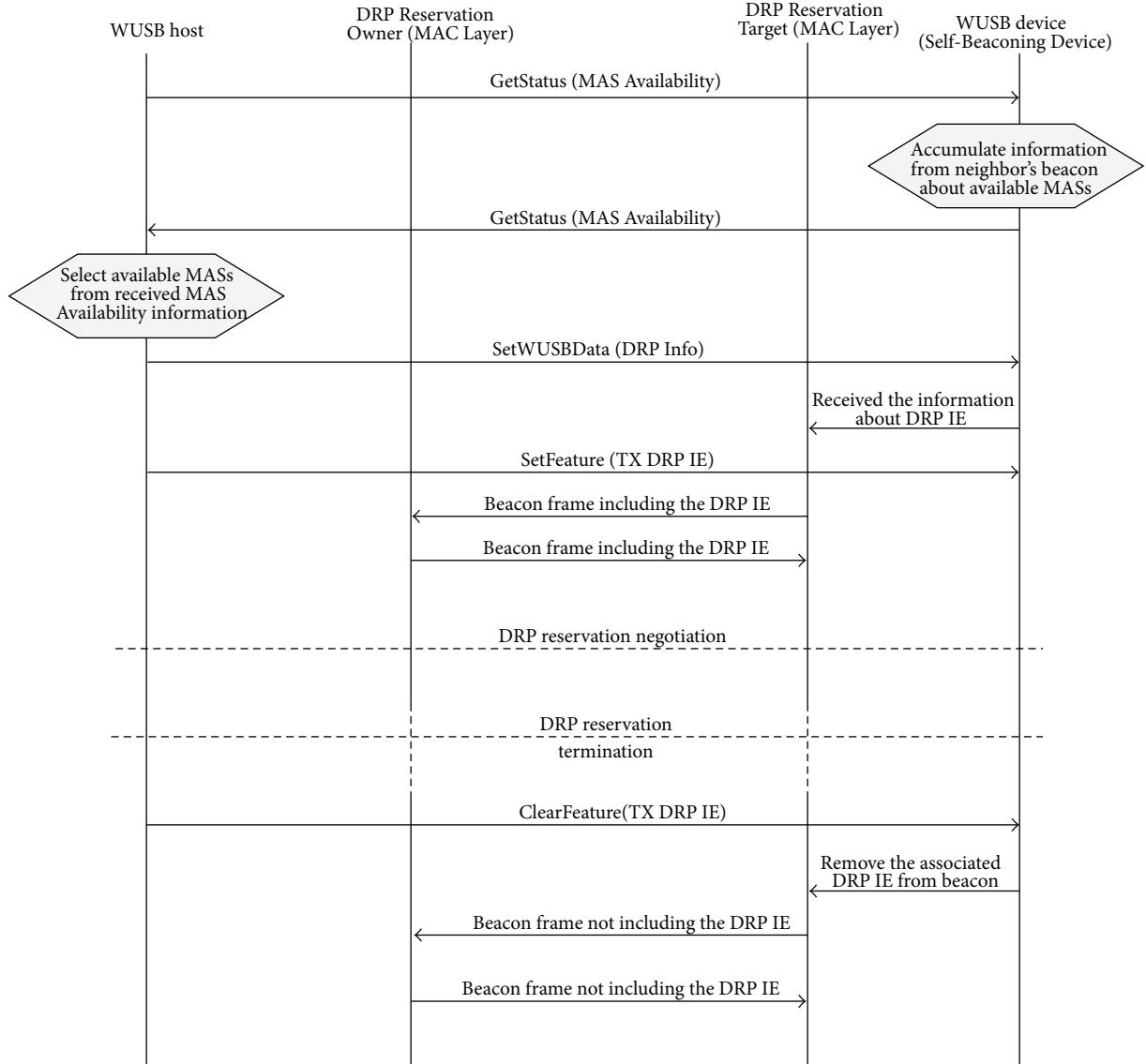


FIGURE 6: The current operation of DRP reservation in WUSB protocol.

bmRequest type (=10000000B)	bRequest (=GET_STATUS)	wValue (=0)	wIndex (=0004H)	wLength	bmMASAvailability (only returned by WUSB device)					
1 octet	*	1 octet	*	2 octets	*	2 octets	*	2 octets	*	32 octets

FIGURE 7: The format of the GetStatus (MAS Availability) request.

After the WUSB host that receives the WUSB device's response selects available MAs, it transmits a SetWUSBDData (DRP Info) request. The SetWUSBDData (DRP Info) is used to construct the DRP IE that the WUSB device transmits in its beacon. If the WUSB device does not have an existing DRP IE for this Wireless USB Channel, it simply adds the received DRP IE to its beacon. If the device has an existing DRP IE for this Wireless USB Channel, then it must replace the existing

DRP IE with the new DRP IE provided in this command payload. Figure 8 shows the format of SetWUSBDData (DRP Info) request.

The values of bmAttributes field are used to construct the DRP control field of a DRP IE. The Conflict Tie-Breaker bit is set to a random value of zero or one when a reservation request is made, and it is used for DRP conflict resolution. DRPIEData field is the DRP allocation blocks that must be

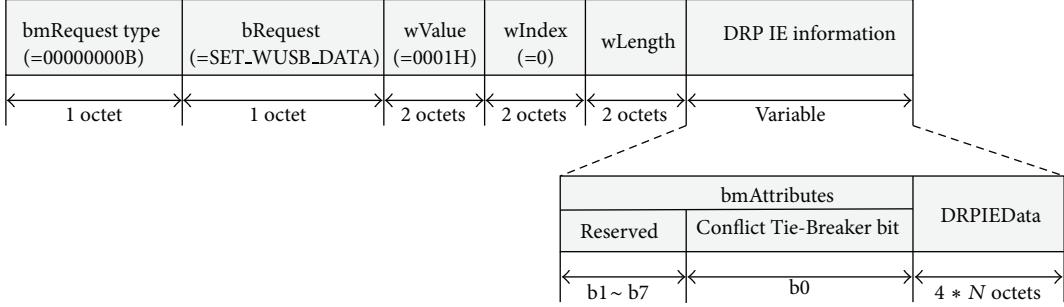


FIGURE 8: The format of the SetWUSBDATA (DRP Info) request.

bmRequest type (=00000000B)	bRequest (=SET_FEATURE)	wValue (=WUSB_DEVICE)	wIndex (=TX_DRPIE)	wLength (=0)
1 octet	1 octet	2 octets	2 octets	2 octets

FIGURE 9: The format of the SetFeature (TX_DRPIE) request.

included in the DRP IE transmitted by the device. If the WUSB host transmits the SetFeature (TX_DRPIE) request to the WUSB device, the WUSB device starts the transmission of its beacon including DRP IE set to the values in DRP IE information field. Figure 9 shows the format of SetFeature (TX_DRPIE) request.

If the WUSB host wants to terminate the DRP reservations with the WUSB device, it transmits the ClearFeature (TX_DRPIE) request. On receipt of this request from the WUSB host, the WUSB device removes the associated DRP IE from its own beacon. Figure 10 shows the format of ClearFeature (TX_DRPIE) request.

4. WUSB over WBAN Architecture for WHMS

WBAN slave devices which have received beacon from WBAN host schedule their receiving and transmitting operations according to the information delivered by the beacon. IEEE 802.15.6 WBAN superframe begins with a beacon period (BP) in which the WBAN hub performing the WUSB host's role sends the beacon. The data transmission period in each superframe is divided into the exclusive access phase 1 (EAP1), random access phase 1 (RAP1), type I/II access phase, EAP2, RAP2, type I/II access phase, and contention access phase (CAP) periods. The EAP1 and EAP2 periods are assigned through contention to data traffic with higher priorities. Further, the RAP1, RAP2, and CAP periods are assigned through contention to data traffic with lower priorities.

In the WBAN beacon frame of Figure 11, the Sender Address field is set to the IEEE MAC address of the WBAN hub sending the current beacon. The Beacon Period Length field is set to the length of the current beacon period (superframe) in units of allocation slots. It is set to 0 to encode a value of 256 allocation slots. The random access phase 1 (RAP1) and RAP2 start fields are set to the number

of the allocation slot that starts RAP1 and RAP2, respectively. The Beacon Shifting Sequence, Channel Hopping State, Next Channel Hop, and Inactive Duration fields are used for interference avoidance in WBAN wireless channel environment.

The IEEE 802.15.6 WBAN MAC systems have several MAC Capability options. Figure 12 shows current WBAN MAC Capability format standard. To successfully set up wireless communication links for WHMS, secure WUSB Channels should be encapsulated privately within a WBAN superframe. Furthermore, the WUSB Channel allocated privately in the IEEE 802.15.6 WBAN MAC enables the MMC scheduling between WUSB host and its several peripheral devices. In the current IEEE 802.15.6 WBAN standard, there are no available periods allocated for other heterogeneous networks.

In Figure 13 for a new WBAN MAC Capability format, a Private Period Allocation field is made by using one bit in the reserved bits in Figure 12. The Private Period Allocation field is set to one if the WBAN MAC supports private WUSB Channel allocations or set to 0 otherwise. If the Private Period Allocation field is set to 1, the RAP2 length field indicates the private WUSB Channel Length for MMC scheduling. Therefore, as shown in Figure 14, after receiving beacon from the WBAN host, non-WUSB WBAN slave devices enter into sleep mode during the RAP2 period. On the contrary, the WUSB/WBAN slave devices enter into active mode during the RAP2 period and they enter into sleep mode during the other periods. The RAP2 Length is determined by the WUSB/WBAN host according to priorities between WUSB transactions and WBAN traffic.

Figure 15 shows a new IEEE 802.15.6 superframe structure supporting the WUSB private channel allocation. The WUSB/WBAN host should transmit WUSB data without interference with WBAN data when a request for WUSB data transmissions occurs in the WUSB cluster. For this

bmRequest type (=0000000B)	bRequest (=CLEAR_FEATURE)	wValue (=WUSB_DEVICE)	wIndex (=TX DRP IE)	wLength (=0)
1 octet	1 octet	2 octets	2 octets	2 octets

FIGURE 10: The format of the ClearFeature (TX_DRPIE) request.

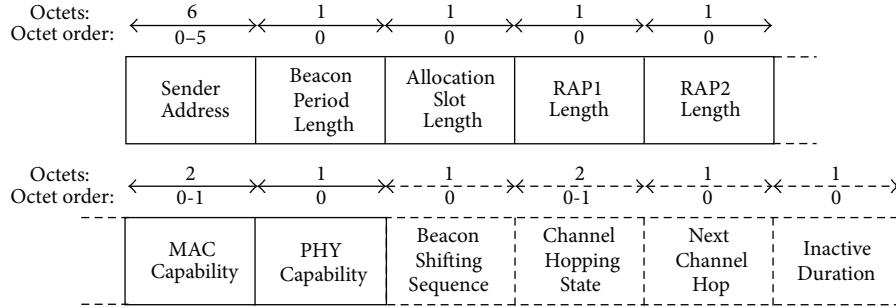


FIGURE 11: WBAN beacon frame format.

purpose, the WUSB/WBAN host has to allocate the WUSB private channels. Except for the RAP1 period, length of the other periods can be set to zero. By using this feature, the WBAN host which also performs the function of WUSB host allocates the WUSB private channels at the RAP2 period.

In the WUSB over WBAN architecture, in order to set up a wireless communication link to WHMS, secure WUSB Channels should be encapsulated within a WBAN superframe. This enables the MMC scheduling between WUSB host and its several peripheral devices without contention. Figure 16 shows the example topologies of the WUSB over WBAN architecture in both nonmedical and medical traffic environments. And Figure 17 explains A signaling flow for WUSB private channel reservation over WBAN protocol.

In this scenario, the user carries a portable or WHMS host device. This host device performs roles of the WUSB host and the WBAN hub simultaneously. Therefore, a “wearable” WUSB cluster and a WBAN cluster can be formed. The attached input sensor nodes perform the functions of localization-based input interfaces for WHMS’s healthcare monitoring. Furthermore, the attached wireless nodes comprise the peripherals of a wearable computer system, and the central WUSB host exchanges data with the outer peripherals of the WUSB slave devices.

5. Time-Synchronization and Localization Middleware in the Proposed WHMS

Many protocols have been reported for clock synchronization in WSNs, and these protocols all have some common basic approaches. Synchronization is achieved by exchanging clock (timestamp) information among nodes while reducing the effect of nondeterministic factors in message delivery. They can be classified into three types: receiver-receiver (R-R) synchronization, sender-receiver (S-R) synchronization, and one-way message dissemination.

In R-R synchronization, a node periodically broadcasts wireless beacon messages to its neighbors. The receivers use the message’s arrival time as a point of reference for comparing their clocks, and then they exchange the local timestamps at which they received the same broadcast message, as shown in Figure 18(a). The receivers finally compute their offset based on the difference in reception times to synchronize their clocks. Reference broadcast synchronization (RBS) [2] is a typical R-R synchronization protocol.

S-R synchronization is performed by a handshake protocol between a pair of nodes. Examples of synchronization protocols that employ this approach include timing-sync protocol for sensor networks (TPSN) and tiny-sync and minisync (TS/MS) [7]. Figure 18(b) shows a two-way message exchange mechanism in TPSN protocol. Node B sends a message at its local time T_1 , and Node A receives this packet at its local time T_2 . At time T_3 , Node A sends back an acknowledgment packet which contains the values of T_2 and T_3 . After receiving the packet at T_4 , Node B can calculate the clock offset (θ) and propagation delay (d) as in (1). Knowing the offset, Node B can synchronize its clock to Node B’s clock by adding θ to its current clock value.

This is expressed in the following equation:

$$\theta = \frac{(T_2 - T_1) - (T_4 - T_3)}{2}, \quad d = \frac{(T_2 - T_1) + (T_4 - T_3)}{2} \quad (1)$$

In one-way message dissemination, a reference node broadcasts its timing information to its neighbors, and they record the arrival times of the broadcast message, as shown in Figure 18(c). Collecting all the timestamps, each node can convert between the local hardware clock and the clock of the reference node by linear regression table. Flooding time-synchronization protocol (FTSP) [7] is a typical protocol utilizing one-way message dissemination scheme.

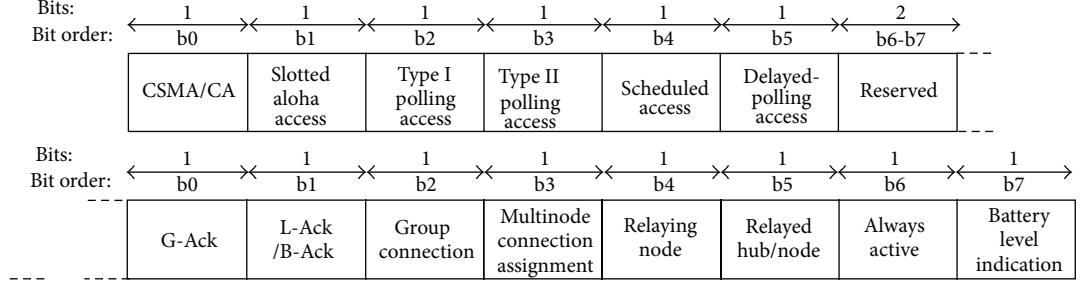


FIGURE 12: Current WBAN MAC Capability format standard.

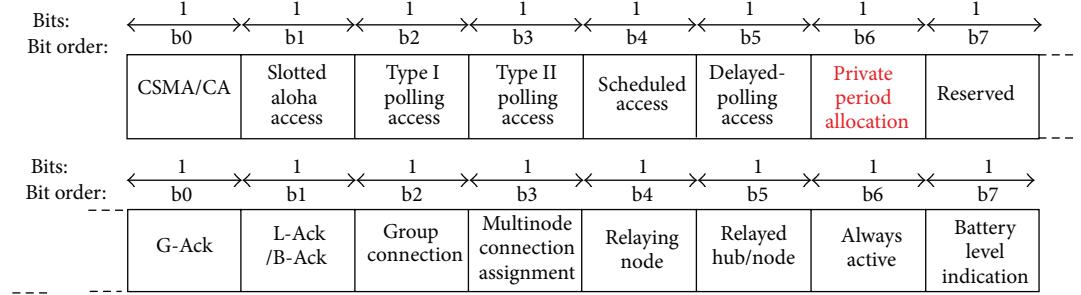


FIGURE 13: A new WBAN MAC Capability format for WUSB private channel allocation.

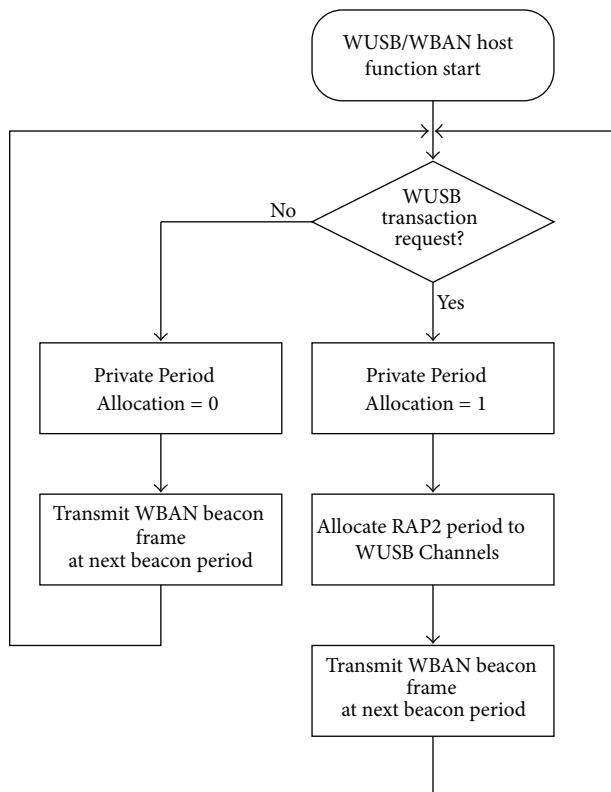


FIGURE 14: WBAN beacon signaling framework (BSF) for WUSB private channel allocation.

A clock is a device that measures time. It generally consists of a periodic component such as an oscillator and a counting component. The characteristics of the oscillator and the counter define the clock's behavior. The frequency of the oscillator controls the rate at which the clock advances. Since it is impossible to create oscillators that oscillate at exactly the same rate, every clock advances at a different rate in real world.

The time reported by a clock at some ideal time t is written as $C(t)$. We will write $C_A(t)$ as the time given by the local clock of Node A. The difference between the time of an ideal clock and a given clock is said to be the offset, $\theta(t)$, which is defined as in (2). The relative offset between Node A and Node B from the viewpoint of Node B, $\theta_B^A(t)$, is defined in (2).

The oscillator in a clock produces periodic pulses. The difference between the rate these pulses are produced at and the rate an ideal clock counts the desired interval is called the skew (frequency offset). Let $\epsilon(t)$ denote the skew as in (2). The clock skew is the slope of the change in offset compared to the local clock. The slope of the relative offset $\theta_B^A(t)$ is relative skew $\epsilon_B^A(t)$. This value means the skew of Node A's timer from the viewpoint of Node B's timer and is defined as in (2).

Typical quartz oscillators exhibit frequency offsets on the order of a few parts per million (ppm). For example, a 20 ppm oscillator will introduce an error of maximum 72 ms in 1 hour. For the remainder of the paper, we adopted the linear clock skew model so that all the notations of skew (ϵ) are used without the term of time (t). Although the skew may be time varying [7], modeling a frequency offset as a piecewise linear

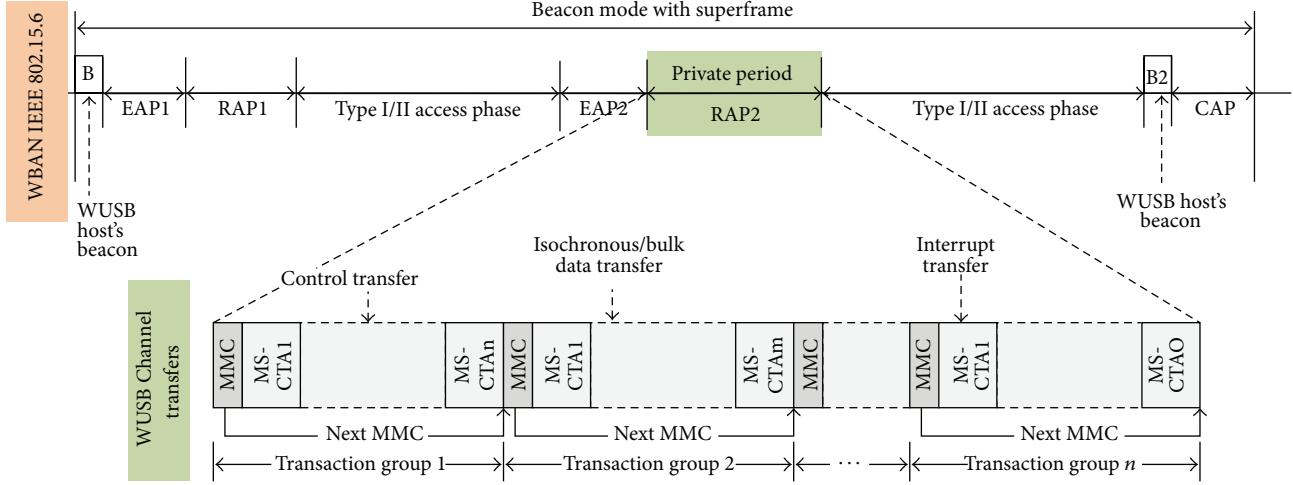


FIGURE 15: IEEE 802.15.6 superframe structure for WUSB private channel allocation.

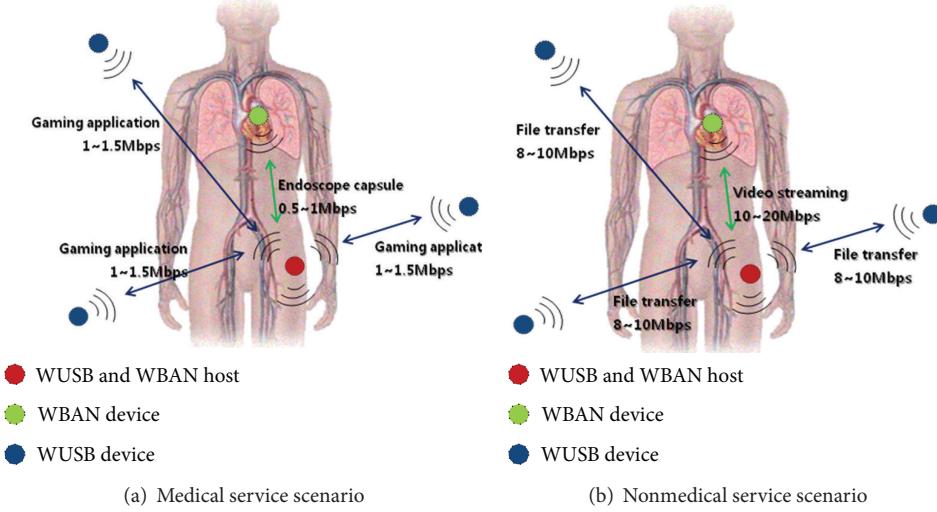


FIGURE 16: WUSB over WBAN architecture.

function of time is a reasonable step as it changes relatively slow. If we refer the operating frequency of Node B's clock to f_B , f_A is expressed as in (2) and the relative skew ε_B^A have another form of definition as follows:

$$\begin{aligned} \theta(t) &= t - C(t), \quad \theta_B^A(t) = C_A(t) - C_B(t), \\ \varepsilon(t) &= \frac{1}{C'(t)} - 1, \\ \varepsilon_B^A(t) &= \frac{C'_A(t)}{C'_B(t)} - 1 = \frac{\varepsilon_B(t) + 1}{\varepsilon_A(t) + 1} - 1, \\ f_A &= f_B(1 + \varepsilon_B^A), \quad \varepsilon_B^A = \frac{f_A}{f_B} - 1. \end{aligned} \quad (2)$$

In the context of WSNs, time synchronization refers to the problem of synchronizing clocks across a set of sensor nodes which are connected to one another over single-hop or multi-hop wireless networks. The time-synchronization problem in

WSNs generally involves two steps. The first is synchronizing the nodes in the network to one common absolute time by adjusting the clock offset among the nodes, and the second is correcting the clock skew relative to a certain standard frequency. The latter is because the imperfections in the quartz crystal and the environmental conditions cause different clocks to run at slightly different frequencies. The effect of clock skew is the main reason that clock offset keeps drifting away. Adjusting clock skew guarantees long-term reliability of synchronization and it reduces networkwide energy consumption in synchronization protocols.

Figure 19 shows the synchronization process of Node B's clock to synchronize with Node A's clock. We assume that the two clocks have the identical initial value for the sake of simplicity. Since each oscillator has its unique clock frequency, the clock offset between the two nodes keeps increasing. If Node B determines the relative offsets $\theta_B^A(t)$ and executes offset compensation at t_1, t_2, t_3 , and t_4 , the corresponding

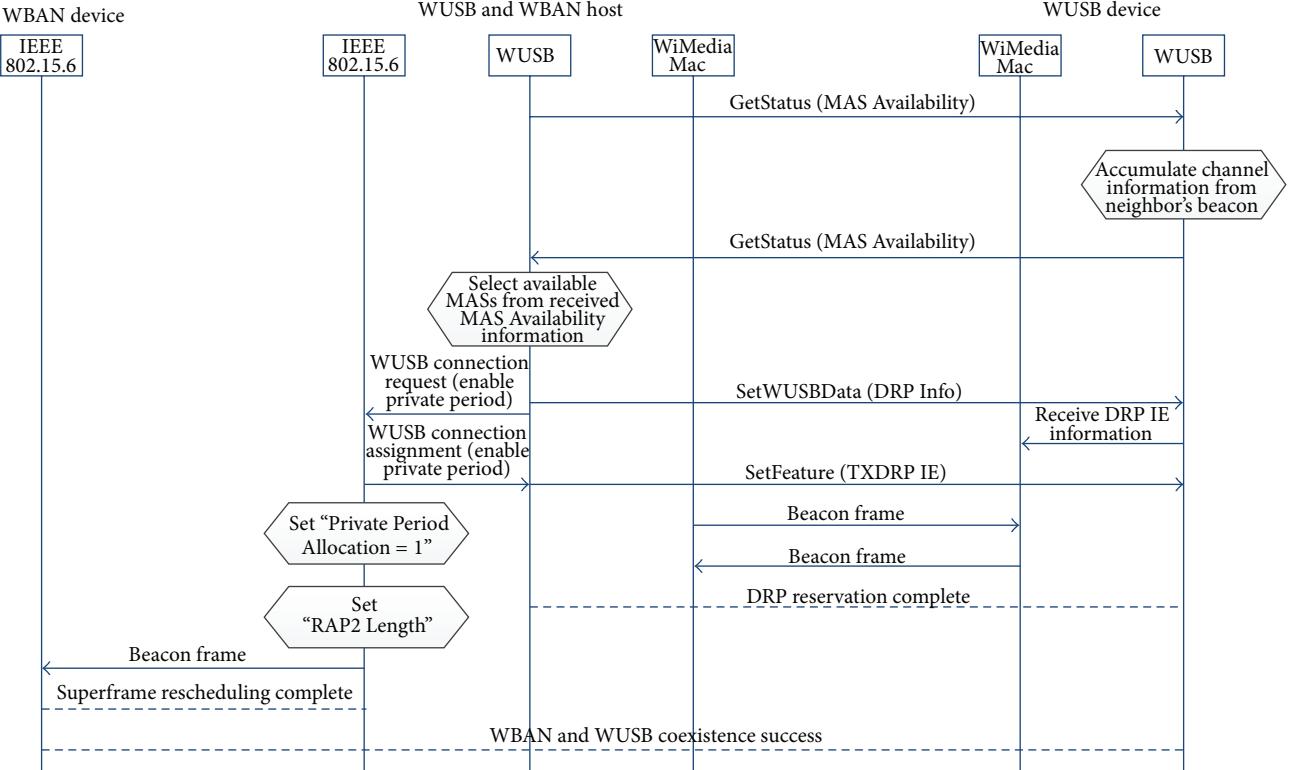


FIGURE 17: A signaling flow for WUSB private channel reservation over WBAN protocol.

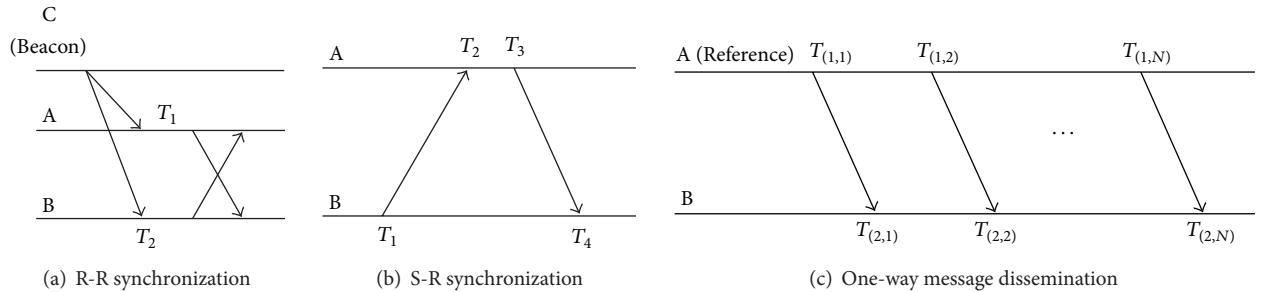


FIGURE 18: Basic synchronization approaches.

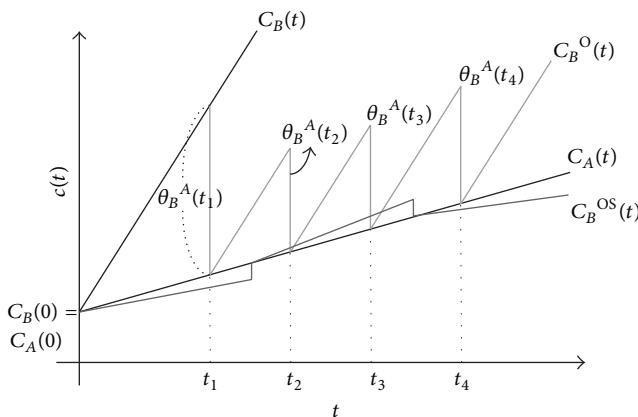


FIGURE 19: Offset compensation and skew compensation.

synchronized clock will be $C_B^O(t)$. Since the offset compensation does not take the impact of clock drift into account, $C_B^O(t)$ keeps the same varying rate and drifts away from $C_A(t)$. If the synchronization (offset compensation) interval becomes longer, the synchronization error also becomes larger. The solution for improving synchronization accuracy is to decrease the synchronization period, but frequent resynchronization will introduce too much communication overhead. If the relative clock skew $\epsilon_B^A(t)$ is accurately estimated, the corresponding synchronized clock can be $C_B^{OS}(t)$ as in Figure 19. $C_B^{OS}(t)$ approximately approaches $C_A(t)$ although it does not completely synchronize with $C_A(t)$. However, it is difficult to estimate the skew accurately due to various factors including short-term effects such as supply voltage, temperature, and humidity changes and long-term

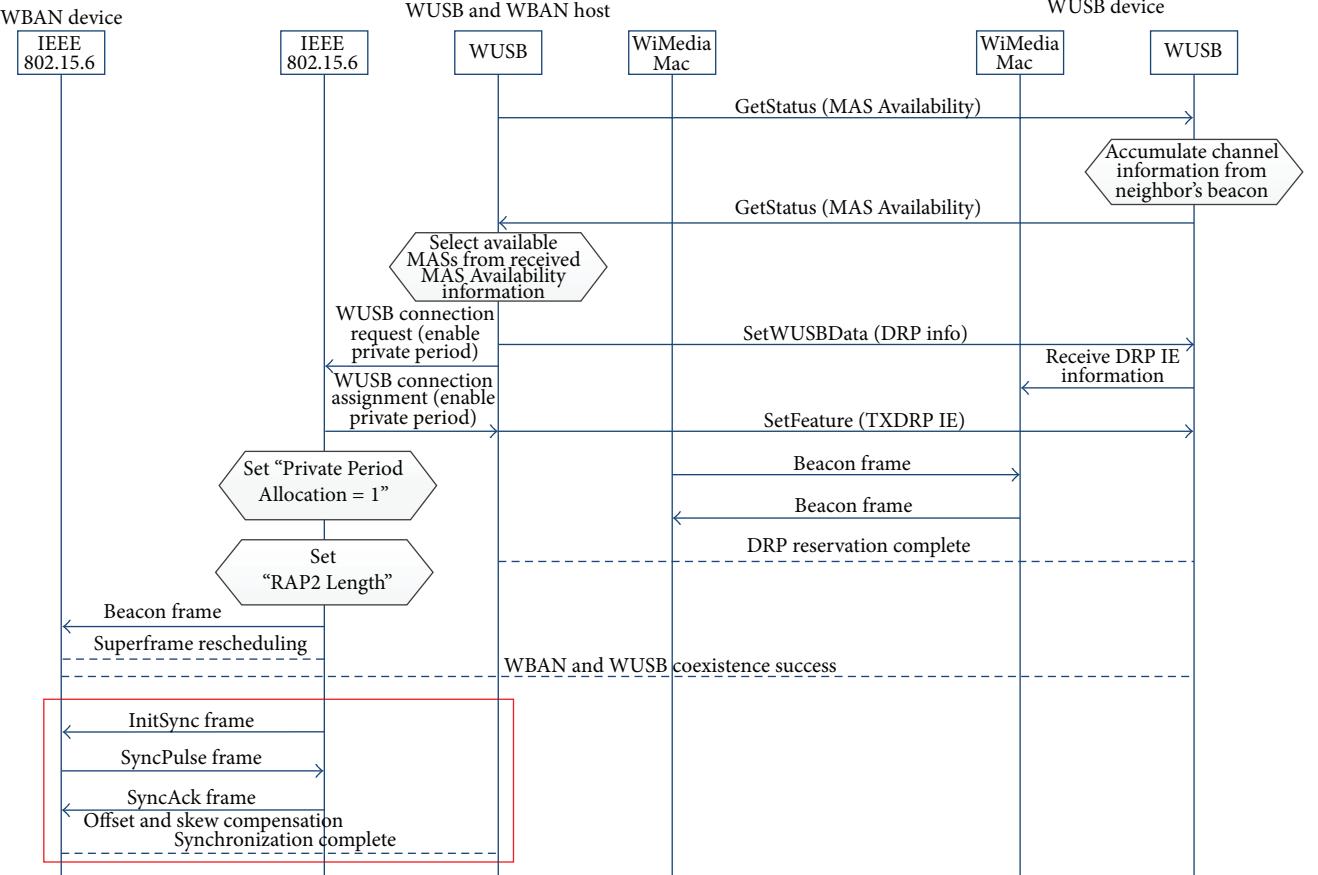


FIGURE 20: Procedures for time synchronization in WUSB over WBAN protocol.

effects such as crystal aging. Offset compensation enables short-term time synchronization, while skew compensation facilitates long-term synchronization. Therefore, these two compensation techniques should be employed together, so that the time-synchronization algorithm with high accuracy and moderate communication overhead can be realized. By using this time-synchronization technique, procedures for time-synchronization in WUSB over WBAN protocol are proposed in Figure 20.

We have built a WUSB over WBAN test bed for the WHMS workspace shown in Figure 21. The WBAN host uses the standard time difference of arrival technique by observing the time lag between the arrival of the signals and estimates its distance from each WUSB/WBAN biosensor device. The estimated distances are passed to the context-aware WHMS server, which computes the location of the WBAN biosensor device using the distances. WHMS application service is requested by providing the WUSB/WBAN host with the user's location information.

The context-aware WHMS application server computes the location (x, y, z) of the WBAN biosensor device using the distances through the trilateration method, as illustrated in Figure 22. Suppose the location of the i th WBAN biosensor device is denoted by b_i , and the estimated distance between the i th WBAN biosensor device and the WUSB/WBAN host is d_i . Then, we made the following equation for the i th WBAN

biosensor device as in (3). This equation is solved using Newton-Raphson method [8]. The initial location (x_0, y_0, z_0) of the WUSB/WBAN host is set to the center of the workspace. Now consider the following:

$$(x - b_{ix})^2 + (y - b_{iy})^2 + (z - b_{iz})^2 = d_i^2. \quad (3)$$

6. Results and Discussion

Performance of the proposed WHMS scheme is evaluated through NS2 simulations with WBAN PHY/MAC simulation parameters [9–11]. The network size is 5 m * 5 m. Maximum 20 devices are randomly deployed into this workspace area. WBAN frame size is fixed to 4095 bytes. Figure 23 shows throughputs of a non-WUSB WBAN device according to each UWB PHY data rate for each m_{In} situation. The m_{In} parameter indicates the occupation ratio of WUSB traffic for the entire WBAN superframe length. In Figure 23, the larger m_{In} ratio of WUSB traffic reduces throughput of a non-WUSB WBAN device more. This result is caused by the reason that the increase of m_{In} ratio enlarged length of the RAP2 period allocated for WUSB traffic in the WBAN superframe.

In Figure 24, UWB PHY data rate of devices is fixed to 480 Mbps. As shown in Figure 24, throughput of a non-WUSB WBAN device does not vary largely according to the frame size after the frame size exceeds a certain threshold.

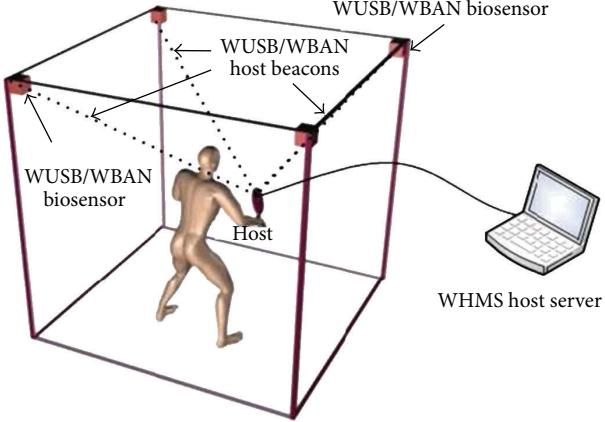


FIGURE 21: Test bed of a WHMS workspace.

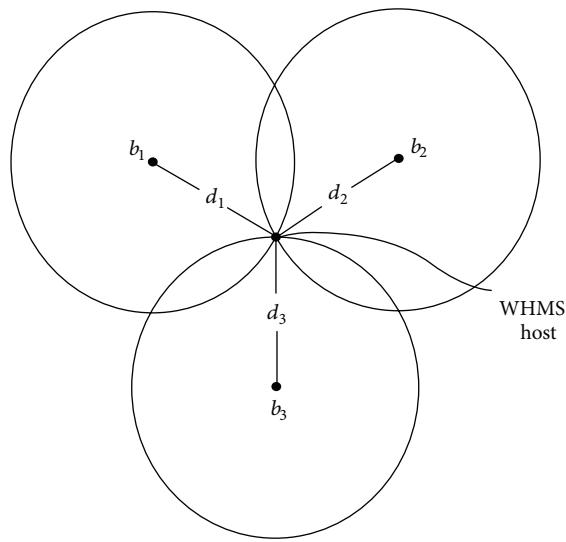


FIGURE 22: Trilateration method.

Because the WUSB transactions in RAP2 period are performed privately without contention, the increase of frame size does not cause the increase of frame collision probability. But the throughput of a non-WUSB WBAN device decreases extremely and proportionally according to m_{In} ratio of WUSB traffic.

To validate our time-synchronization middleware, the WBAN biosensor nodes use the 32.768 kHz real-time clock, and the frequency tolerance is set to ± 20 ppm. We performed measurements on the WHMS test bed. A WBAN biosensor node acts as the reference node. Another WUSB/WBAN host node acts as the sink node which is connected to WHMS server and gathers information about the other WBAN biosensor nodes' synchronization error values. We measured the synchronization errors between the reference WBAN biosensor node and the other WBAN biosensor nodes.

We built a network which consists of 20 WBAN biosensor nodes. To measure the synchronization precision, we make each sensor node report the clock offset (θ_i) to the WUSB/WBAN host sink node whenever it performs offset

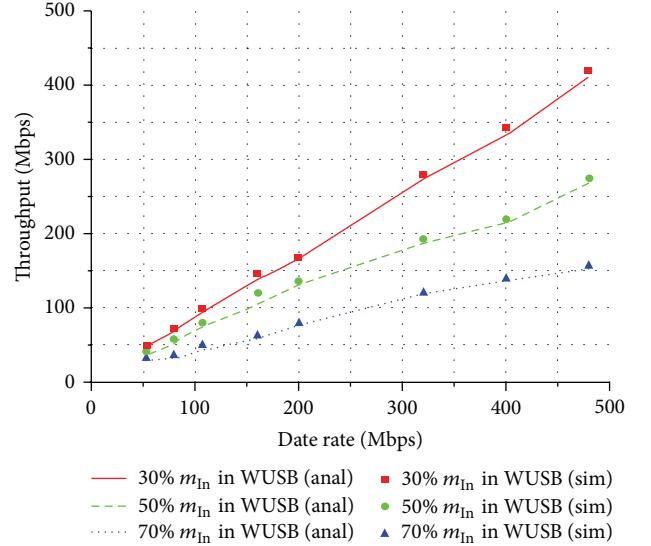


FIGURE 23: Throughput of a non-WUSB WBAN device according to each UWB PHY data rate.

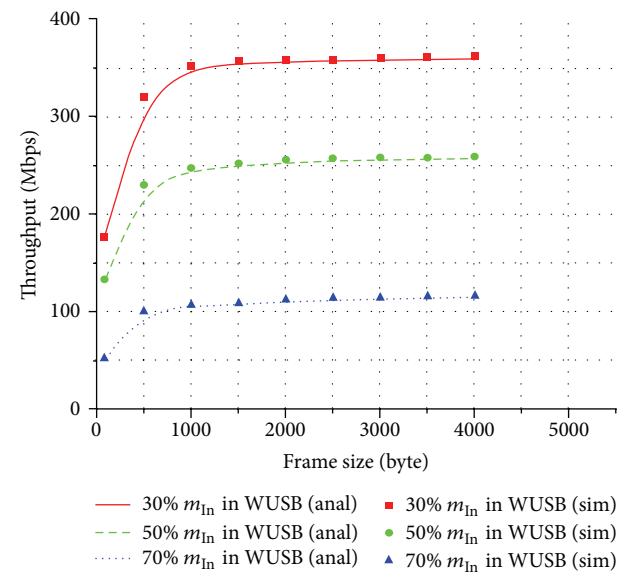


FIGURE 24: Throughput of a non-WUSB WBAN device according to each frame size.

compensation. This clock offset expresses the time difference between its own clock and the clock of reference node at the moment just before offset compensation. However, this value does not coincide with the synchronization error exactly since it may introduce some measurement errors. For the offset compensation, we adopt the S-R synchronization scheme of two-way message exchanges in TPSN. The WUSB/WBAN host sink node controls the initiation of two-way message exchanges and its interval. Once initialized by the WUSB/WBAN host sink node, the synchronization

TABLE 2: Estimated distances at 100 cm.

Distance (cm)	Frequency (2000 times in total)
99.9674	11
100.001	327
100.035	946
100.069	62
100.104	573
100.138	75
99.071	6

TABLE 3: Statistical analysis at 100 cm.

Statistics	Value
Mean	100.041 cm
Variance	0.00154603 cm
Standard deviation	0.03931958 cm

procedure of exchanging two-way message can be driven to the whole network.

For checking the effect of clock skew, we allowed WBAN biosensor nodes to perform only offset compensation as in the original TPSN. Every WBAN biosensor node performed offset compensation at the interval of 30 minutes and 1 hour. The data reported from each WBAN biosensor node are averaged according to message hop count. Figure 25 shows this result of the average synchronization error. It is checked that the synchronization errors weakly tend to increase with respect to message hop count from the WUSB/WBAN host sink node and that the effect of clock skew on synchronization error becomes considerable as time passes.

Finally, to validate our localization middleware, the distances between the WBAN biosensor nodes and the WUSB/WBAN host node are obtained in an asynchronous mode. All WBAN biosensor nodes use a 16-bit timer, and the location of a WBAN biosensor node is computed at 10 Hz. The accuracy of the distance estimation has been tested. For a fixed position of the WUSB/WBAN host, the distance from a WBAN biosensor node is measured by hand. Then, the distance is estimated in the proposed middleware framework. Such an estimation is done for 2,000 times. Tables 2 and 3 show the statistics for the measured distances of 100 cm. This distance test proves that our localization middleware can be successfully integrated with WHMS applications.

7. Conclusion

In this paper, we propose time-synchronization and localization middlewares platform built on WUSB over IEEE 802.15.6 WBAN hierarchical protocol. Our test results prove that they can be well integrated and lead to a new type of natural interface for the wearable health-monitoring systems (WHMS). In the current implementation, the location data are computed at 10 Hz. For fully supporting real-time applications of WHMS, more effort should be made to increase

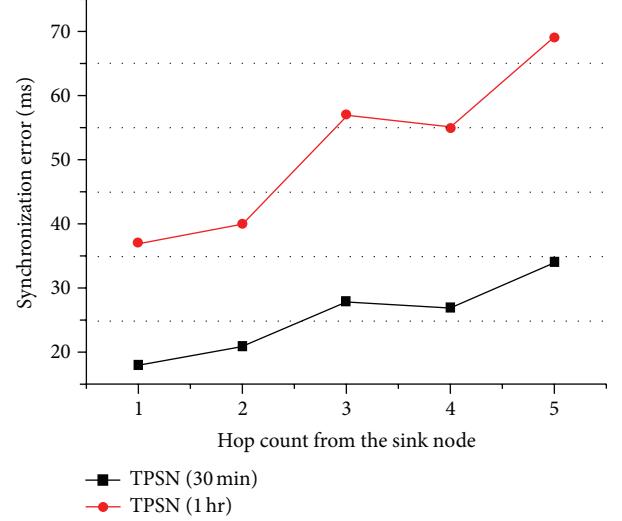


FIGURE 25: Synchronization error by offset and skew compensation.

the performance. Further, many applications may benefit not only from location awareness but also from orientation awareness built on the precise time synchronization. To fulfill such needs, the overall performance should be continuously upgraded.

Acknowledgments

This research was supported in part by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (MEST) (2010-0002366) and in part by Midcareer Researcher Program through NRF Grant funded by the MEST (2011-0016145).

References

- [1] A. Pantelopoulos and N. G. Bourbakis, "A survey on wearable sensor-based systems for health monitoring and prognosis," *IEEE Transactions on Systems, Man and Cybernetics C*, vol. 40, no. 1, pp. 1–12, 2010.
- [2] M. Patel and J. Wang, "Applications, challenges, and prospective in emerging body area networking technologies," *IEEE Wireless Communications*, vol. 17, no. 1, pp. 80–88, 2010.
- [3] IEEE 802.15 WPAN Task Group 6 Body Area Networks (BAN). IEEE, <http://www.ieee802.org>, 2010.
- [4] Certified Wireless USB 1.1. USB-IF, <http://www.usb.org>, 2010.
- [5] J. W. Kim, K. Hur, J. Park, and D. S. Eom, "A distributed MAC design for data collision-free wireless USB home networks," *IEEE Transactions on Consumer Electronics*, vol. 55, no. 3, pp. 1337–1343, 2009.
- [6] R. Palit, A. Singh, and K. Naik, "An architecture for enhancing capability and energy efficiency of wireless handheld devices," *International Journal of Energy, Information and Communications*, vol. 2, pp. 117–136, 2011.
- [7] H. Dai and R. Han, "TSync: a lightweight bidirectional time synchronization service for wireless sensor networks," *ACM*

- SIGMOBILE Mobile Computing and Communications Review*, vol. 8, pp. 125–139, 2004.
- [8] N. B. Priyantha, A. Chakraborty, and H. Balakrishnan, “Cricket location-support system,” in *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking (Mobicom '00)*, pp. 32–43, August 2000.
 - [9] N. Karthikeyan, V. Palanisamy, and K. Duraiswamy, “Performance comparison of broadcasting methods in Mobile Ad Hoc Network,” *International Journal of Future Generation Communication and Networking*, vol. 2, pp. 47–58, 2009.
 - [10] K.-I. Kim, “Adjusting transmission power for real-time communications in wireless sensor networks,” *Journal of Information and Communication Convergence Engineering*, vol. 10, pp. 21–26, 2012.
 - [11] M. Mana, M. Feham, and B. A. Bensaber, “SEKEBAN (secure and efficient key exchange for wireless body area network),” *International Journal of Advanced Science and Technology*, vol. 12, pp. 45–60, 2009.

