

Time Synchronous Bluetooth Sensor Networks

Jens Eliasson

Dept. of Computer Science and
Electrical Engineering
Luleå University of Technology
SE-971 87 Luleå, Sweden
Email: jens.eliasson@ltu.se

Magnus Lundberg

Dept. of Computer Science and
Electrical Engineering
Luleå University of Technology
SE-971 87 Luleå, Sweden
Email: magnus.lundberg@ltu.se

Per Lindgren

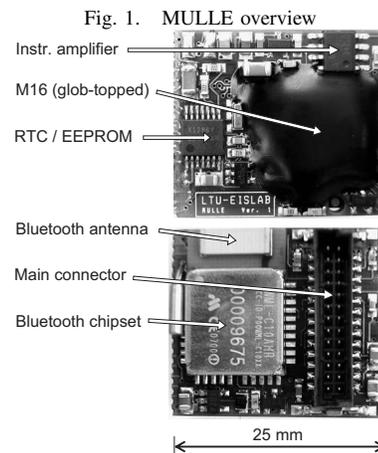
Dept. of Computer Science and
Electrical Engineering
Luleå University of Technology
SE-971 87 Luleå, Sweden
Email: per.lindgren@ltu.se

Abstract—Bluetooth-equipped wireless sensor nodes can be quickly integrated in small home networks. These networks can be utilized e.g. for surveillance, home monitoring and automation. Accurate time is an important factor for time-stamping of sensor data, encryption/authentication and it can also be used to implement time synchronous schemes for low power radio communication. We argue that IP-based time synchronization, such as various flavors of the NTP protocol, can be used with Bluetooth networks. This in combination with an activation schedule allows an efficient trade-off between energy consumption and communication delay, and provides easy integration with available infrastructure. The proposed approach in this paper is well suited for smaller wireless home networks, typically single-hop networks with access points that are always available. Our approach is verified by experiments performed on a COTS-based platform using Bluetooth.

I. INTRODUCTION

In battery powered wireless sensor devices, the wireless communication usually dominates the energy consumption [2], [3], [13]. Therefore, there is a need to minimize the number of transmissions and reduce the time required to either make a transmission or to be listening for incoming connections, as listening in some cases can be as costly as transmitting. The device used in this paper is a COTS (Commercial Off-The-Shelf) based system. The advantages with COTS-system is that the development time can be kept much shorter compared to ASIC (Application Specific Integrated Circuit) based systems, even if the ASIC systems can be made smaller and more cost efficient for large scale production.

In traditional Wireless Sensor Networks, communication between nodes are usually performed using a low delay, customized radio system. By the use of high precision time synchronization, it is possible to turn off the radio system when it is not used, without adding extra communication delay. As shown in many studies [13], [18], time synchronization solutions for traditional wired networks, such as IP-based synchronization protocols like NTP or SNTP, are inappropriate for wireless networks where the radio systems require fine granularity timing resolution for optimal performance. Instead proprietary hardware and software solutions are often utilized for high resolution time synchronization [20]. A clock error greater than the communication delay will severely degrade the performance and hence increase the power consumption. Bluetooth provides a synchronized high precision clock be-

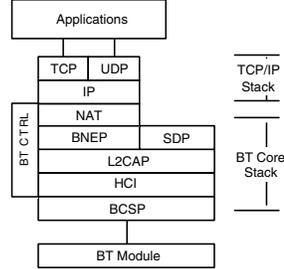


tween connected devices, that can be used for various low power modes [2]. This clock can however not be used in situations where the Bluetooth module is switched off or the link is disconnected. To overcome this problem, we propose to use a synchronized activation schedule based on TCP/IP. Each node in the network maintains an activation schedule, which contains information about when the node will activate its Bluetooth transceiver in the future. The required granularity of the schedule is in the range of hundreds of milliseconds, since the connection time of a Bluetooth link are normally counted in seconds. Since this granularity can be obtained using IP-based synchronization techniques, the need for a wake-up radio is eliminated. By embedding the time schedule into a Service Discovery Protocol, such as zeroconf [14], the potential power reductions of scheduled communication can be efficiently utilized.

II. THE MULLE

The experimental platform used during the writing of this paper is the MULLE, see Figure I, a minimal embedded system with a bare die Renesas M16C/62 micro-controller [4], with 20 kB of RAM running at 10 MHz, and a Mitsumi Bluetooth module [5] as the main components. The MULLE has a number of digital and analog I/O's, on-board instrumentation amplifier and a real-time clock with non-volatile EEPROM. The MULLE system is described in more detail in

Fig. 2. MULLE software architecture



[1]. Previous papers by the authors, see [2] and [3], described various techniques to characterize the power consumption under representative modes of operation, and proposed ideas to minimize system energy consumption. To further enhance the ability to save energy, two more protocol layers were added to the lwBT Bluetooth stack. These two are BCSP [8], and BNEP [9]. BNEP is a replacement for the RFCOMM and PPP layers in the old stack, see [3]. With the introduction of these two new layers, more efficient low power modes can be introduced in the system as shown in sections 2.2 and 2.3.

A. System Architecture

The MULLE software stack, see Figure II, shows the usage of standardized protocols, such as TCP/IP and Bluetooth. This approach eliminates proprietary communication and radio solutions and thus allows interoperability with existing infrastructures; GPRS, Internet or Bluetooth equipped devices. The two main components are lwIP [6] and lwBT [7], lightweight IP- and Bluetooth-stacks, respectively. The MULLE does not use any real-time operating system, instead a minimal hardware abstraction layer (HAL) is used.

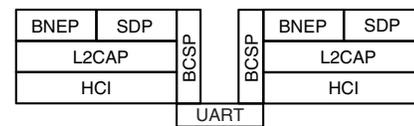
B. BlueCore Serial Protocol

BCSP [8] is used for communication between the micro-controller and the Bluetooth module over UART or USB. It is more advanced than H4, the standard protocol for HCI communication between a host and a Bluetooth module. BCSP handles retransmission of lost packets, an important feature when using low power modes on both the micro-controller and the Bluetooth module since packets can be lost during sleep or wakeup periods. BCSP also uses a Link Establishment procedure to detect if one of its peers reboots. Link Establishment is also used to synchronize the micro-controller and the Bluetooth module during booting. BCSP allows access to any of the stack layers inside the Bluetooth module (see Figure 3), and it can also act as a 2-way communication link between the micro-controller and the CPU inside the Bluetooth module, enabling applications executing on the Bluetooth device to communicate with the micro-controller. BCSP is a proprietary protocol developed by CSR, and can, for now, only be used with Bluetooth modules equipped with CSR's chipsets.

C. Bluetooth Network Emulation Protocol

BNEP [9] is used to transmit Ethernet frames over the L2CAP layer in the Bluetooth software stack. The Bluetooth

Fig. 3. BCSP access to the software stack inside a CSR Bluetooth module



profile using BNEP is the PAN (Personal Area Networking) profile. PAN gives both an increase in bandwidth and lower complexity than using the old LAP profile with PPP (previously used on the MULLE). PPP also has timers that periodically transmits keep-alive packets between its peers. These packets will not be transmitted when a link is in Hold or Park mode, and will cause a time-out, so PPP cannot be used to save power or allow more than 7 devices in a piconet. BNEP however does not rely on timers, and can therefore be used with Bluetooth low-power Hold and Park mode, and is also more suitable for larger networks. A device using BNEP can have any of the following three roles: PANU (PAN User), GN (Group Ad-hoc Network) or NAP (Network Access Point). PANU is the role a node has when it is a client, while GN and NAP are the two roles that are providing a network service. The difference between GN and NAP is that a GN does not provide a connection to any additional networks while NAP acts as a bridge to other networks, such as Ethernet or GPRS.

D. Requirements for power saving

To save power, parts of the system are powered down when they are not used, e.g. the analog system, the Bluetooth module or the attached sensor(s). The power supply can for this purpose be controlled by the micro-controller to switch off unused parts of the system. One problem that arises is that the different parts do not wake up instantaneously, a short time is required for them to boot up after being in a low power mode. This is especially true for both the micro-controller and the Bluetooth module. Therefore, the two parts must inform each other when they are able to receive data. By the usage of the Link Establishment protocol provided by BCSP, this problem can be solved. Even more flexibility is added by the usage of interrupts from the Bluetooth module to inform the micro-controller about occurring events.

III. MODES OF OPERATION

In the case of the MULLE system, the most energy consuming part is the Bluetooth communication [2], [3]. The ideas proposed here have a potential to allow reductions on energy consumption, while keeping the system response time low. This is accomplished by a combination of two operating modes that have been used previously on the MULLE, *passive* and *active* mode. The new mode, called *time-synchronous* uses low granularity time synchronization between nodes and an activation schedule to inform the ad-hoc network of when each node will activate the Bluetooth module in the future.

TABLE I
 DELAY VS. ENERGY CONSUMPTION DURING 24 H. WITH A ONE MINUTE
 LISTEN CYCLE EACH PERIOD

Mode	Energy [J]	Delay [h]
<i>Active</i>	0.9	24
<i>ts 1 hour</i>	3.2	1
<i>ts 30 min</i>	5.5	0.5
<i>ts 10 min</i>	15	0.16
<i>Passive</i>	69	0

A. Modes for power saving

Below follows an explanation of each mode, with advantages and disadvantages outlined.

Active

The node decides when to make a connection, either to a mobile phone, LAN access point or to another node. This is the most energy saving mode since the node never needs to waste energy by having the Bluetooth module on, except from periodic transmissions. Even if this is the best mode from an energy point of view, it is the mode with the highest delay since an outside user cannot obtain sensor data until the MULLE decides to make a connection. A connection is typically performed when a timer expires or data buffers starts to fill.

Passive

The node acts as a server, and listens continuously for incoming connections from users, or other nodes in active mode. This is the most energy consuming mode since the Bluetooth module needs to be powered on in listening mode the whole time in order to accept incoming connections. This mode gives the lowest delay since the MULLE is always ready to deliver its data to a user at any given time.

Time-synchronous

This mode uses a clock to control when the Bluetooth module is active, i.e. it is a combination of the first two modes using synchronized system clocks. The result is a compromise between energy consumption and delay. A node in active mode that is programmed to transmit data one time each day can for example enter passive mode a short period every hour. This will give a slightly increased energy consumption but the delay decreases from 24 hours to 1 hour. This requires that the system clocks are synchronized and that the clock drift is sufficiently low. A requirement is also that all other devices that wishes to access the node must have their clocks synchronized, as well as being notified what time the node will enter passive mode. A notification scheme is therefore used between all nodes to exchange information about their activation schedule. The MULLE uses an Xicor X1288 Real-time clock [16] with programmable alarms to maintain time synchronization. The listening duty cycle are typical in tens of seconds. This is needed due to the long upper bounds on Bluetooth Inquiry Scans. This mode is

also suitable for situations where a user wants to see if a node is alive or not, without the need to make a time and energy consuming connection. It is sufficient to make periodic Inquiry Scans to see if nodes activates their Bluetooth modules according to their schedule.

B. Scheduled activations

With the addition of a clock and a notification scheme of when a node will be online in the future, energy consumption can be reduced while keeping the total delay low. One standard that is suitable is vCal [17], which is commonly used by a large number of applications. vCal documents are both human and machine readable, and can be transferred either by the Bluetooth Object Push Profile, or downloaded from a web server. By also integrating the schedule into a Service Discovery Protocol, the network will not only know what kind of services a node has, but also which operating mode it is in and when these services will become available the next time. The protocol recommended by the BNEP specification [9] for ad-hoc networks is zeroconf, but other protocols may also be used. See Table I for a summary of delay versus energy consumption for a MULLE set to transmit its data one time per day. Measurements based on the characterization setup used previously, see [2], shows the system energy consumption on a few typical intervals. The reduction in delay is induced by the Bluetooth module entering a listening mode for a period of one minute. The increase in energy is caused by the slightly higher energy consumption of the module in listening state.

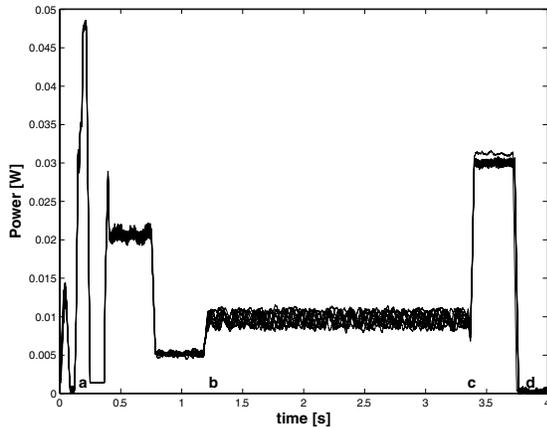
C. Extra costs for Bluetooth module shutdown

When the Bluetooth module is not used, it will enter a low power mode, Deep Sleep, which reduces the power consumption to 225 μ W. To reduce this unnecessary consumption, the Bluetooth module can be powered down between its active periods. This way the power consumption can be reduced, but it also causes the need to boot up the Bluetooth module before it is used. This procedure takes time and costs some additional energy. Measurements performed, see Figure 4, indicates that the boot time for a WML C-10 is around 1.3 seconds, and the total energy consumption is 46 mJ. This corresponds to nearly 3 minutes in Deep Sleep, which leads to the conclusion that it is better to stay in Deep Sleep if the duty cycle is less than 3 minutes. In Figure 4, a reset is performed at (a). The module has performed a complete boot at (b) and after a configurable delay at (c), the module prepares for Deep Sleep. At (d), the module is in Deep Sleep mode.

IV. TIME SYNCHRONIZATION

Time synchronization between nodes within an ad-hoc network, as well as with databases on the public Internet can be achieved by a number of different techniques. Three protocols based on IP are: TP - Time Protocol [10], the more advanced NTP - Network Time Protocol [11] and a simplified version of NTP, Simple NTP [12]. SNTP requires less computations than NTP, and is hence more suitable for embedded systems. Many embedded systems use wake-up radio systems [13] in parallel

Fig. 4. WML-C10 boot characteristics(10 overlaid curves)



with the normal radio used for data communication. That gives more flexibility and accuracy within the sensor network, however, by having extra radio circuits, the size and cost of the system increases. Another benefit by using standardized protocols such as (S)NTP is that any IP-enabled system, i.e. smart phones or computers, can also function both as servers and clients, and finally; high precision time servers on the Internet can be utilized in an efficient way. A sensor node equipped with a GPS receiver can also act as a high accuracy (S)NTP clock server for all other nodes in the local network. When a MULLE has received a timestamp, it will program the Xicor Real-Time Clock [16] with the correct time information. This approach increases accuracy and allows the MCU to be switched off without the need for re-synchronization.

V. SUMMARY

By a combination of state of the art COTS components, efficient usage of low power modes and a distributed time scheduling mechanism, it is possible to reduce the stand-by power consumption drastically. Our approach allows to efficiently trade off decreased power consumption against the cost of increased delay. Standardized time synchronization protocols, such as NTP or SNTP, are not as accurate for sensor networks as dedicated hardware solutions, but have the advantages of utilizing time servers on the Internet, and simplifies interoperability with other non-sensor nodes, for example users with computers or smart phones. For embedded systems where the size must be kept minimal, the extra space required by a time synchronization / wake-up radio may be unacceptable. The major problem with the proposed approach is that the network must be aware of each node's time schedule, and that the periods where the Bluetooth is in listening mode must be kept fairly long, in order of tens of seconds due to the long upper bound on Bluetooth Inquiry scans. Our characterization model applied on the MULLE, see Table I, indicates that a reduction in delay from 24 to 1 hour, a decrease of 24 times, leads to only 2.5 times higher energy consumption. We also see that by allowing a delay of

30 minutes (compared to *passive mode*), we can reduce the energy consumption by 80%.

VI. FUTURE WORK

One topic that needs to be investigated is to seamlessly integrate the time scheduling mechanism into a standard Service Discovery Protocol, preferably zeroconf [14]. This approach will enable users on the Internet as well as in the local network to both receive and set the duty cycle of sensor nodes. The usage of GPS based time synchronization is another way of obtaining correct time, and finally the precision of (S)NTP over high delay radio links (GPRS, UMTS), must be further investigated. The usage of synchronous Bluetooth data packets (SCO) with low delay and jitter may be used to decrease the clock error between nodes. A mechanism for a sensor node to cache the time schedules of other nodes in its vicinity may also prove useful.

VII. CONCLUSIONS

The usage of small battery powered wireless sensor nodes have great potential for home automation, monitoring and security. Two important requirements are low power consumption and low communication delay. The ideas proposed in this paper, achieves a good compromise between these two requirements, while enabling interoperability with existing infrastructure. The usage of standardized protocols, such as Bluetooth and TCP/IP, makes integration with local IP-based networks as well as the Internet possible. With the combination of clock synchronization and a time activation schedule, communication delays and energy consumption can be optimized. The activation schedule makes the behavior of each node in the sensor network more predictable and delays can easily be estimated. The proposed approach is well suited for applications such as home monitoring and surveillance, Personal/Body Area Networks and other situations where easy integration with available infrastructure is an important requirement.

REFERENCES

- [1] J. Johansson, M. Völker, J. Eliasson, Å. Östmark, P. Lindgren, "MULLE: A Minimal Sensor Networking Device - Implementation and Manufacturing challenges", In *IMAPS Nordic Conference 2004 International Microelectronics and Packaging Society*
- [2] M. Lundberg, J. Eliasson, L. Svensson, P. Lindgren, "Context Aware Power Optimizations of Wireless Embedded Internet Systems", In *IEEE IMTC 2004 Instrumentation and Measurement Technology Conference*
- [3] M. Lundberg, J. Eliasson, J. Allan, J. Johansson, P. Lindgren, "Power Characterization of a Bluetooth-Equipped Sensor Node" In *Workshop on Real-World Wireless Sensor Networks*, Stockholm 2005
- [4] M16C/62M Renesas Technology Corporation <http://www.renesas.com/eng/>, March 2005
- [5] Mitsumi Bluetooth Module WML-C10 AHR <http://www.mitsumi.com/>, May 2005
- [6] A. Dunkels. lwIP - A lightweight IP Stack <http://www.sics.se/~adam/lwip/>, March 2005
- [7] C. Öhult. lwBT - A lightweight Bluetooth Stack <http://www.csee.ltu.se/~conny/lwbt/>, March 2005
- [8] BCSP - BlueCore Serial Protocol <http://www.csrsupport.com/index.html>, March 2005
- [9] BNEP - Bluetooth Network Emulation Protocol <http://www.bluetooth.org/>, March 2005

- [10] RFC 868 - Time Protocol
<http://www.faqs.org/rfcs/rfc868.html> , March 2005
- [11] RFC 1305 - Network Time Protocol (version 3)
<http://www.faqs.org/rfcs/rfc1305.html> , March 2005
- [12] RFC 2030 - Simple Network Time Protocol (version 4)
<http://www.faqs.org/rfcs/rfc2030.html> , March 2005
- [13] J. Elson, K. Römer, "Wireless Sensor Networks - A New regime for Time Synchronization". In *Proceedings of The First Workshop on Hot Topics in Networks* Princeton, New Jersey. October 28-29 2002
- [14] zeroconf - Zero Configuration Networking
<http://www.zeroconf.org> , May 2005
- [15] SDP - Bluetooth Service Discovery Protocol
<http://www.bluetooth.org/spec/> , June 2005
- [16] Xicor X1288 - 2 Wire Real-Time Clock
<http://www.intersil.com/data/xicor/x1288.pdf> , June 2005
- [17] RFC 2445 - Internet Calendaring and Scheduling Core Object Specification
<http://www.ietf.org/rfc/rfc2445.txt> , June 2005
- [18] X. Yang, Nitin H. Vaidya, "A Wakeup Scheme for Sensor Networks: Achieving Balance between Energy Saving and End-to-End Delay". In *10'th IEEE Real-Time and Embedded Technology and Applications Symposium 2004*
- [19] J. Elson, D. Estrin, "Time Synchronization for Wireless Sensor Networks" In *Proceedings of the 2001 International Parallel and Distributed Processing Symposium (IPDPS)*
- [20] R. Casas, et al, "Synchronization in Wireless Sensor Networks Using Bluetooth" In *The Third International Workshop on Intelligent Solutions in Embedded Systems* Hamburg University of Technology, Hamburg May 20 2005