

Internet Engineering Task Force (IETF)  
Request for Comments: 8352  
Category: Informational  
ISSN: 2070-1721

C. Gomez  
UPC  
M. Kovatsch  
ETH Zurich  
H. Tian  
China Academy of Telecommunication Research  
Z. Cao, Ed.  
Huawei Technologies  
April 2018

## Energy-Efficient Features of Internet of Things Protocols

### Abstract

This document describes the challenges for energy-efficient protocol operation on constrained devices and the current practices used to overcome those challenges. It summarizes the main link-layer techniques used for energy-efficient networking, and it highlights the impact of such techniques on the upper-layer protocols so that they can together achieve an energy-efficient behavior. The document also provides an overview of energy-efficient mechanisms available at each layer of the IETF protocol suite specified for constrained-node networks.

### Status of This Memo

This document is not an Internet Standards Track specification; it is published for informational purposes.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Not all documents approved by the IESG are candidates for any level of Internet Standard; see Section 2 of RFC 7841.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at <https://www.rfc-editor.org/info/rfc8352>.

## Copyright Notice

Copyright (c) 2018 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

|  |    |
|--|----|
| 1. Introduction . . . . .  | 3  |
| 1.1. Terminology . . . . .   | 3  |
| 2. Overview . . . . .  | 4  |
| 3. Medium Access Control and Radio Duty Cycling . . . . .                | 6  |
| 3.1. Techniques for Radio Duty Cycling . . . . .                         | 6  |
| 3.2. Latency and Buffering . . . . .                                     | 7  |
| 3.3. Throughput . . . . .  | 8  |
| 3.4. Radio Interface Tuning . . . . .                                    | 8  |
| 3.5. Packet Bundling . . . . .   | 8  |
| 3.6. Power Save Services Available in Example Low-Power Radios . . . . . | 8  |
| 3.6.1. Power Save Services Provided by IEEE 802.11 . . . . .             | 8  |
| 3.6.2. Power Save Services Provided by Bluetooth LE . . . . .            | 10 |
| 3.6.3. Power Save Services in IEEE 802.15.4 . . . . .                    | 11 |
| 3.6.4. Power Save Services in DECT ULE . . . . .                         | 12 |
| 4. IP Adaptation and Transport Layer . . . . .                           | 14 |
| 5. Routing Protocols . . . . .   | 15 |
| 6. Application Layer . . . . .   | 16 |
| 6.1. Energy-Efficient Features in CoAP . . . . .                         | 16 |
| 6.2. Sleepy Node Support . . . . .                                       | 17 |
| 6.3. CoAP Timers . . . . .   | 17 |
| 6.4. Data Compression . . . . .  | 18 |
| 7. Summary and Conclusions . . . . .                                     | 18 |
| 8. IANA Considerations . . . . .   | 19 |
| 9. Security Considerations . . . . .                                     | 19 |
| 10. References . . . . .   | 19 |
| 10.1. Normative References . . . . .                                     | 19 |
| 10.2. Informative References . . . . .                                   | 22 |
| Acknowledgments . . . . .  | 23 |
| Contributors . . . . .   | 24 |
| Authors' Addresses . . . . .   | 24 |

## 1. Introduction

Network systems for monitoring the physical world contain many battery-powered or energy-harvesting devices. For example, in an environmental monitoring system or a temperature and humidity monitoring system, there may not be always on and sustained power supplies for the potentially large number of constrained devices. In such deployment scenarios, it is necessary to optimize the energy consumption of the constrained devices. In this document, we describe techniques that are in common use at Layer 2 and at Layer 3, and we indicate the need for higher-layer awareness of lower-layer features.

Many research efforts have studied this "energy efficiency" problem. Most of this research has focused on how to optimize the system's power consumption in certain deployment scenarios or how an existing network function such as routing or security could be more energy efficient. Only few efforts have focused on energy-efficient designs for IETF protocols and standardized network stacks for such constrained devices [CLASS1-CoAP].

The IETF has developed a suite of Internet protocols suitable for such constrained devices, including IPv6 over Low-Power Wireless Personal Area Networks (6LoWPAN) [RFC6282] [RFC6775] [RFC4944], the IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL) [RFC6550], and the Constrained Application Protocol (CoAP) [RFC7252]. This document tries to summarize the design considerations for making the IETF constrained protocol suite as energy efficient as possible. While this document does not provide detailed and systematic solutions to the energy-efficiency problem, it summarizes the design efforts and analyzes the design space of this problem. In particular, it provides an overview of the techniques used by the lower layers to save energy and how these may impact on the upper layers. Cross-layer interaction is therefore considered in this document from this specific point of view. Providing further design recommendations that go beyond the layered protocol architecture is out of the scope of this document.

After reviewing the energy-efficient designs of each layer, we summarize the document by presenting some overall conclusions. Though the lower-layer communication optimization is the key part of energy-efficient design, the protocol design at the upper layers is also important to make the device energy efficient.

### 1.1. Terminology

Terms used in this document are defined in [RFC7228] [CNN-TERMS].

## 2. Overview

The IETF has developed protocols to enable end-to-end IP communication between constrained nodes and fully capable nodes. This work has expedited the evolution of the traditional Internet protocol stack to a lightweight Internet protocol stack. As shown in Figure 1 below, the IETF has developed CoAP as the application layer and 6LoWPAN as the adaption layer to run IPv6 over IEEE 802.15.4 [IEEE802.15.4] and Bluetooth Low Energy (also referred to as Bluetooth LE and BTLE), with the support of routing by RPL and efficient neighbor discovery by 6LoWPAN Neighbor Discovery (6LoWPAN-ND). 6LoWPAN is currently being adapted by the 6lo Working Group to support IPv6 over various other technologies, such as ITU-T G.9959 [G9959], Digital Enhanced Cordless Telecommunications Ultra Low Energy (DECT ULE) [TS102], Building Automation and Control Networks Master-Slave/Token-Passing (BACnet MS/TP) [MSTP], and Near Field Communication [NFC].

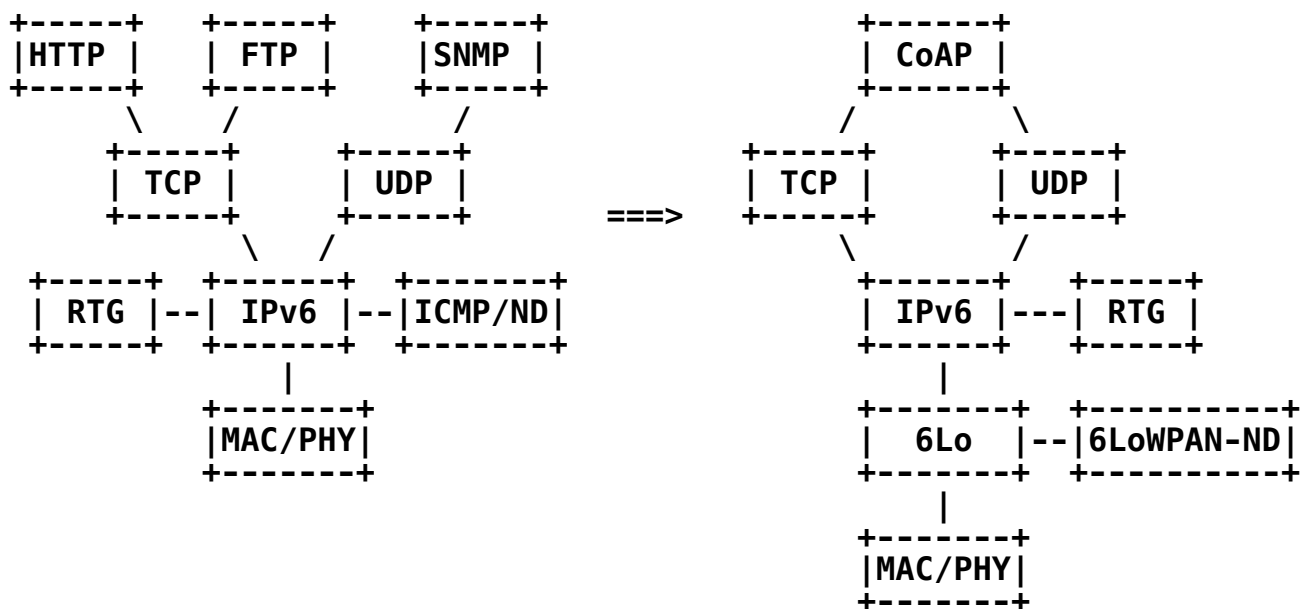


Figure 1: Traditional and Lightweight Internet Protocol Stack

There are numerous published studies reporting comprehensive measurements of wireless communication platforms [Powertrace]. As an example, below we list the energy-consumption profile of the most common operations involved in communication on a prevalent sensor node platform. The measurement was based on the Tmote Sky with ContikiMAC [ContikiMAC] as the Radio Duty Cycling algorithm. From this and many other measurement reports (e.g., [AN079]), we can see that the energy consumption of optimized transmission and reception

are in the same order. For IEEE 802.15.4 [IEEE802.15.4] and Ultra WideBand (UWB) links, transmitting may actually be even cheaper than receiving. It also shows that broadcast and non-synchronized communication transmissions are energy costly because they need to acquire the medium for a long time.

| Activity                              | Energy<br>(microjoules) |
|---------------------------------------|-------------------------|
| Broadcast reception                   | 178                     |
| Unicast reception                     | 222                     |
| Broadcast transmission                | 1790                    |
| Non-synchronized unicast transmission | 1090                    |
| Synchronized unicast transmission     | 120                     |
| Unicast TX to awake receiver          | 96                      |
| Listening (for 1000 ms)               | 63000                   |

Figure 2: Power Consumption of Common Operations Involved in Communication on the Tmote Sky with ContikiMAC

At the Physical layer, one approach that may reduce the energy consumption of a device that uses a wireless interface is based on reducing the device transmit power level, as long as the intended next hop(s) is still within range of the device. In some cases, if node A has to transmit a message to node B, a solution to reduce node A transmit power is to leverage an intermediate device, e.g., node C as a message forwarder. Let  $d$  be the distance between node A and node B. Assuming free-space propagation, where path loss is proportional to  $d^2$ , if node C is placed right in the middle of the path between A and B (that is, at a distance  $d/2$  from both node A and node B), the minimum transmit power to be used by node A (and by node C) is reduced by a factor of 4. However, this solution requires additional devices, it requires a routing solution, and it also increases transmission delay between A and B.

### 3. Medium Access Control and Radio Duty Cycling

In networks, communication and power consumption are interdependent. The communication device is typically the most power-consuming component, but merely refraining from transmissions is not enough to achieve a low power consumption: the radio may consume as much power in listen mode as when actively transmitting. This illustrates the key problem known as idle listening, whereby the radio of a device may be in receive mode (ready to receive any message), even if no message is being transmitted to that device. Idle listening can consume a huge amount of energy unnecessarily. To reduce power consumption, the radio must be switched completely off -- duty-cycled -- as much as possible. By applying duty cycling, the lifetime of a device operating on a common button battery may be on the order of years, whereas otherwise the battery may be exhausted in a few days or even hours. Duty cycling is a technique generally employed by devices that use the P1 strategy [RFC7228], which need to be able to communicate on a relatively frequent basis. Note that a more aggressive approach to save energy relies on the P0 (Normally-off) strategy, whereby devices sleep for very long periods and communicate infrequently, even though they spend energy in network reattachment procedures.

From the perspective of Medium Access Control (MAC) and Radio Duty Cycling (RDC), all upper-layer protocols, such as routing, RESTful communication, adaptation, and management flows, are applications. Since the duty-cycling algorithm is the key to energy efficiency of the wireless medium, it synchronizes transmission and/or reception requests from the higher layers.

MAC and RDC are not in the scope of the IETF, yet lower-layer designers and chipset manufacturers take great care to save energy. By knowing the behaviors of these lower layers, engineers can design protocols that work well with them. The IETF protocols to be discussed in the following sections are the customers of the lower layers.

#### 3.1. Techniques for Radio Duty Cycling

This subsection describes three main RDC techniques. Note that more than one of these techniques may be available or can even be combined in a specific radio technology:

- a) Channel sampling: In this solution, the radio interface of a device periodically monitors the channel for very short time intervals (i.e., with a low duty cycle) with the aim of detecting incoming transmissions. In order to make sure that a receiver can correctly receive a transmitted data unit, the sender may

prepend a preamble of a duration at least the sampling period to the data unit to be sent. Another option for the sender is to repeatedly transmit the data unit instead of sending a preamble before the data unit. Once a transmission is detected by a receiver, the receiver may stay awake until the complete reception of the data unit. Examples of radio technologies that use preamble sampling include ContikiMAC, the Coordinated Sampled Listening (CSL) mode of IEEE 802.15.4e [IEEE802.15.4], and the Frequently Listening (FL) mode of ITU-T G.9959 [G9959].

- b) **Scheduled transmissions:** This approach allows a device to know the particular time at which it should be awake (during some time interval) in order to receive data. Otherwise, the device may remain in sleep mode. The decision on the times at which communication is attempted relies on some form of negotiation between the involved devices. Such negotiation may be performed per transmission or per session/connection. Bluetooth Low Energy (Bluetooth LE) is an example of a radio technology based on this mechanism.
- c) **Listen after send:** This technique allows a node to remain in sleep mode by default, then wake up and poll a sender (which must be ready to receive a poll message) for pending transmissions. After sending the poll message, the node remains in receive mode and is ready for a potential incoming transmission. After a certain time interval, the node may go back to sleep. For example, this technique is used in the Receiver Initiated Transmission (RIT) mode of IEEE 802.15.4e [IEEE802.15.4] and in the transmission of data between a coordinator and a device in the 2003 version of IEEE 802.15.4 [IEEE802.15.4].

### 3.2. Latency and Buffering

The latency of a data unit transmission to a duty-cycled device is equal to or greater than the latency of transmitting to an always-on device. Therefore, duty cycling leads to a trade-off between energy consumption and latency. Note that in addition to a latency increase, RDC may introduce latency variance since the latency increase is a random variable (which is uniformly distributed if duty cycling follows a periodic behavior).

On the other hand, due to the latency increase introduced by duty cycling, a sender waiting for a transmission opportunity may need to store subsequent outgoing packets in a buffer. This buffering would increase memory requirements and potentially incur queuing wait times. Such wait times would in turn contribute to packet transmission delay and increase the probability of buffer overflow, leading to losses.

### 3.3. Throughput

Although throughput is not typically a key concern in constrained-node network applications, it is indeed important in some services in such networks, such as over-the-air software updates or when off-line sensors accumulate measurements that have to be quickly transferred when there is an opportunity for connectivity.

Since RDC introduces inactive intervals in energy-constrained devices, it reduces the throughput that can be achieved when communicating with such devices. There exists a trade-off between the achievable throughput and energy consumption.

### 3.4. Radio Interface Tuning

The parameters controlling the radio duty cycle have to be carefully tuned to achieve the intended application and/or network requirements. On the other hand, upper layers should take into account the expected latency and/or throughput behavior due to RDC. The next subsection provides details on key parameters controlling RDC mechanisms, and thus fundamental trade-offs, for various examples of relevant low-power radio technologies.

### 3.5. Packet Bundling

Another technique that may be useful to increase communication energy efficiency is packet bundling. This technique, which is available in several radio interfaces (e.g., LTE and some 802.11 variants), allows for aggregation of several small packets into a single large packet. Header and communication overhead is therefore reduced.

### 3.6. Power Save Services Available in Example Low-Power Radios

This subsection presents power save services and techniques used in a few relevant examples of wireless low-power radios: IEEE 802.11 [IEEE802.11], Bluetooth LE, and IEEE 802.15.4 [IEEE802.15.4]. For a more detailed overview of each technology, the reader may refer to the literature or to the corresponding specifications.

#### 3.6.1. Power Save Services Provided by IEEE 802.11

IEEE 802.11 [IEEE802.11] defines the Power Save Mode (PSM) whereby a station may indicate to an Access Point (AP) that it will enter a sleep mode state. While the station is sleeping, the AP buffers any frames that should be sent to the sleeping station. The station wakes up every listen interval (which can be a multiple of the beacon interval) in order to receive beacons. The AP signals, by means of a beacon field, whether there is data pending for the station or not.



If there are not frames to be sent to the station, the latter may get back to sleep mode. Otherwise, the station may send a message requesting the transmission of the buffered data and stay awake in receive mode.

IEEE 802.11v [IEEE802.11] further defines mechanisms and services for power save of stations/nodes that include Flexible Multicast Service (FMS), Proxy ARP advertisement, extended sleep modes, and traffic filtering. Upper-layer protocol's knowledge of such capabilities, provided by the lower layer, enables better interworking.

These services include:

**Proxy ARP:** The Proxy ARP capability enables an Access Point (AP) to indicate that the non-AP station (STA) will not receive ARP frames. The Proxy ARP capability enables the non-AP STA to remain in power save mode for longer periods of time.

**Basic Service Set (BSS) Max Idle Period Management:** Enables an AP to indicate a time period during which the AP does not disassociate a STA due to non-receipt of frames from the STA. This supports improved STA power saving and AP resource management.

**FMS:** A service in which a non-AP STA can request a multicast delivery interval longer than the Delivery Traffic Indication Message (DTIM) interval for the purposes of lengthening the period of time a STA may be in a power save state.

**Traffic Filtering Service (TFS):** A service provided by an AP to a non-AP STA that can reduce the number of frames sent to the STA by dropping individually addressed frames that do not match traffic filters specified by the STA.

Using the above services provided by the lower layer, the constrained nodes can achieve either client-initiated power save (via TFS) or network-assisted power save (Proxy ARP, BSS Max Idle Period, and FMS).

Upper-layer protocols should synchronize with the parameters such as FMS interval and BSS MAX Idle Period so that the wireless transmissions are not triggered periodically.

### 3.6.2. Power Save Services Provided by Bluetooth LE

Bluetooth LE is a wireless low-power communications technology that is the hallmark component of the Bluetooth 4.0, 4.1, and 4.2 specifications [Bluetooth42]. BTLE has been designed for the goal of ultra-low power consumption. IPv6 can be run over Bluetooth LE networks by using a 6LoWPAN variant adapted to BTLE [RFC7668].

Bluetooth LE networks comprise a master and one or more slaves, which are connected to the master. The Bluetooth LE master is assumed to be a relatively powerful device, whereas a slave is typically a constrained device (e.g., a Class 1 device).

Medium access in Bluetooth LE is based on a Time-Division Multiple Access (TDMA) scheme that is coordinated by the master. This device determines the start of connection events in which communication between the master and a slave takes place. At the beginning of a connection event, the master sends a poll message, which may encapsulate data, to the slave. The latter must send a response, which may also contain data. The master and the slave may continue exchanging data until the end of the connection event. The next opportunity for communication between the master and the slave will be in the next connection event scheduled for the slave.

The time between consecutive connection events is defined by the `connInterval` parameter, which may range between 7.5 ms and 4 s. The slave may remain in sleep mode from the end of its last connection event until the beginning of its next connection event. Therefore, Bluetooth LE is duty-cycled by design. Furthermore, after having replied to the master, a slave is not required to listen to the master (and thus may keep the radio in sleep mode) for `connSlaveLatency` consecutive connection events. `connSlaveLatency` is an integer parameter between 0 and 499 that should not cause link inactivity for more than `connSupervisionTimeout` time. The `connSupervisionTimeout` parameter is in the range between 100 ms and 32 s.

Upper-layer protocols should take into account the medium access and duty-cycling behavior of Bluetooth LE. In particular, `connInterval`, `connSlaveLatency`, and `connSupervisionTimeout` determine the time between two consecutive connection events for a given slave. The upper-layer packet generation pattern and rate should be consistent with the settings of the aforementioned parameters (and vice versa). For example, assume `connInterval` = 4 seconds, `connSlaveLatency` = 7 seconds, and `connSupervisionTimeout` = 32 seconds. With these settings, communication opportunities between a master and a slave will occur during a given interval every 32 seconds. Duration of the interval will depend on several factors, including number of

connected slaves, amount of data to be transmitted, etc. In the worst case, only one data unit can be sent from master to slave (and vice versa) every 32 seconds.

### 3.6.3. Power Save Services in IEEE 802.15.4

IEEE 802.15.4 [IEEE802.15.4] is a family of standard radio interfaces for low-rate, low-power wireless networking. Since the publication of its first version in 2003, IEEE 802.15.4 [IEEE802.15.4] has become the de facto choice for a wide range of constrained-node network application domains and has been a primary target technology of various IETF working groups such as 6LoWPAN [RFC6282] [RFC6775] [RFC4944] and 6TiSCH [ARCH-6TiSCH]. IEEE 802.15.4 [IEEE802.15.4] specifies a variety of related Physical layer (PHY) and MAC layer functionalities.

IEEE 802.15.4 [IEEE802.15.4] defines three roles called device, coordinator, and Personal Area Network (PAN) coordinator. The device role is adequate for nodes that do not implement the complete IEEE 802.15.4 [IEEE802.15.4] functionality and is mainly targeted for constrained nodes with a limited energy source. The coordinator role includes synchronization capabilities and is suitable for nodes that do not suffer severe constraints (e.g., a mains-powered node). The PAN coordinator is a special type of coordinator that acts as a principal controller in an IEEE 802.15.4 [IEEE802.15.4] network.

IEEE 802.15.4 [IEEE802.15.4] defines two main types of networks depending on their configuration: beacon-enabled and non-beacon-enabled networks. In the first network type, coordinators periodically transmit beacons. The time between beacons is divided in three main parts: the Contention Access Period (CAP), the Contention Free Period (CFP), and an inactive period. In the first period, nodes use slotted Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) for data communication. In the second one, a TDMA scheme controls medium access. During the idle period, communication does not take place, and thus the inactive period is a good opportunity for nodes to turn the radio off and save energy. The coordinator announces in each beacon the list of nodes for which data will be sent in the subsequent period. Therefore, devices may remain in sleep mode by default and wake up periodically to listen to the beacons sent by their coordinator. If a device wants to transmit data, or learns from a beacon that it is an intended destination, then it will exchange messages with the coordinator (and thus consume energy). An underlying assumption is that when a message is sent to a coordinator, the radio of the coordinator will be ready to receive the message.

The beacon interval and the duration of the active portion of the beacon interval (i.e., the CAP and the CFP), and thus the duty cycle, can be configured. The parameters that control these times are called `macBeaconOrder` and `macSuperframeOrder`, respectively. As an example, when IEEE 802.15.4 [IEEE802.15.4] operates in the 2.4 GHz PHY, both times can be (independently) set to values in the range between 15.36 ms and 251.6 s.

In the beaconless mode, nodes use unslotted CSMA/CA for data transmission. The device may be in sleep mode by default and may activate its radio to either i) request to the coordinator whether there is pending data for the device, or to ii) transmit data to the coordinator. The wake-up pattern of the device, if any, is out of the scope of IEEE 802.15.4 [IEEE802.15.4].

Communication between the two ends of an IEEE 802.15.4 [IEEE802.15.4] link may also take place in a peer-to-peer configuration, whereby both link ends assume the same role. In this case, data transmission can happen at any moment. Nodes must have their radio in receive mode and be ready to listen to the medium by default (which for battery-enabled nodes may lead to a quick battery depletion) or apply synchronization techniques. The latter are out of the scope of IEEE 802.15.4 [IEEE802.15.4].

The main MAC layer IEEE 802.15.4 [IEEE802.15.4] amendment to date is IEEE 802.15.4e. This amendment includes various new MAC layer modes, some of which include mechanisms for low energy consumption. Among these, the Time-Slotted Channel Hopping (TSCH) is an outstanding mode that offers robust features for industrial environments, among others. In order to provide the functionality needed to enable IPv6 over TSCH, the 6TiSCH Working Group was created. TSCH is based on a TDMA schedule whereby a set of timeslots are used for frame transmission and reception, and other timeslots are unscheduled. The latter timeslots may be used by a dynamic scheduling mechanism, otherwise, nodes may keep the radio off during the unscheduled timeslots, thus saving energy. The minimal schedule configuration specified in [RFC8180] comprises 101 timeslots; 95 of these timeslots are unscheduled and the timeslot duration is 15 ms.

The previously mentioned CSL and RIT are also 802.15.4e modes designed for low energy.

#### 3.6.4. Power Save Services in DECT ULE

DECT Ultra Low Energy (DECT ULE) is a wireless technology building on the key fundamentals of traditional DECT / Cordless Advanced Technology - internet and quality (CAT-iq) [EN300] but with specific changes to significantly reduce the power consumption at the expense

of data throughput [TS102]. DECT ULE devices typically operate on special power-optimized silicon but can connect to a DECT Gateway supporting traditional DECT/CAT-ig for cordless telephony and data as well as the DECT ULE extensions. IPv6 can be run over DECT ULE by using a 6LoWPAN variant [RFC8105].

DECT defines two major roles: the Portable Part (PP) is the power constrained device while the Fixed Part (FP) is the Gateway or base station in a star topology. Because TDMA/FDMA and Time-Division Duplex (TDD) using dynamic channel allocation for interference, DECT operates in license-free and reserved frequency bands. It provides good indoor (~50 m) and outdoor (~300 m) coverage. It uses a frame length of 10 ms divided into 24 timeslots, and it supports connection-oriented packet data and connection-less services.

The FP usually transmits a so-called dummy bearer (beacon) that is used to broadcast synchronization, system, and paging information. The slot/carrier position of this dummy bearer can automatically be reallocated in order to avoid mutual interference with other DECT signals.

At the MAC level, DECT ULE communications between FP and PP are initiated by the PP. An FP can initiate communication indirectly by sending a paging signal to a PP. The PP determines the timeslot and frequency in which the communication between FP and PP takes place. The PP verifies the radio timeslot/frequency position is unoccupied before it initiates its transmitter. An access-request message, which usually carries data, is sent to the FP. The FP sends a confirm message, which also may carry data. More data can be sent in subsequent frames. A MAC-level automatic retransmission scheme significantly improves the reliability of data transfer. A segmentation and reassembly scheme supports transfer of larger, higher-layer Service Data Units (SDUs) and provides data integrity checks. The DECT ULE packet data service ensures data integrity, proper sequencing, and duplicate protection but not guaranteed delivery. Higher-layer protocols have to take this into consideration.

The FP may send paging information to PPs to trigger connection setup and indicate the required service type. The interval between paging information to a specific PP can be defined in the range of 10 ms to 327 s. The PP may enter sleep mode to save power. The listening interval is defined by the PP application. For short sleep intervals (below ~10 seconds), the PP may be able to retain synchronization to the FP dummy bearer and only turn on the receiver during the expected timeslot. For longer sleep intervals, the PP can't keep synchronization and has to search for, and resynchronize to, the FP dummy bearer. Hence, longer sleep intervals reduce the average

energy consumption but add an energy consumption penalty for acquiring synchronization to the FP dummy bearer. The PP can obtain all information to determine paging and acquire synchronization information in a single reception of one full timeslot.

Packet data latency is normally 30 ms for short packets (below or equal to 32 octets), however, if retry and back-off scenarios occur, the latency is increased. The latency can actually be reduced to about 10 ms by doing energy consuming Received Signal Strength Indication (RSSI) scanning in advance. In the direction from FP to PP, the latency is usually increased by the used paging interval and the sleep interval. The MAC layer can piggyback commands to improve efficiency (reduce latency) of higher-layer protocols. Such commands can instruct the PP to initiate a new packet transfer in N frames without the need for resynchronization and can listen to paging or instruct the PP to stay in a higher duty-cycle paging detection mode.

The DECT ULE technology allows a per-PP configuration of paging interval, MTU size, reassembly window size, and higher-layer service negotiation and protocol.

#### 4. IP Adaptation and Transport Layer

6LoWPAN provides an adaptation layer designed to support IPv6 over IEEE 802.15.4 [IEEE802.15.4]. 6LoWPAN affects the energy-efficiency problem in three aspects, as follows.

First, 6LoWPAN provides one fragmentation and reassembly mechanism, which is aimed at solving the packet size issue in IPv6 and could also affect energy efficiency. IPv6 requires that every link in the Internet have an MTU of 1280 octets or greater. On any link that cannot convey a 1280-octet packet in one piece, link-specific fragmentation and reassembly must be provided at a layer below IPv6 [RFC8200]. 6LoWPAN provides fragmentation and reassembly below the IP layer to solve the problem. One of the benefits from placing fragmentation at a lower layer such as the 6LoWPAN layer is that it can avoid the presence of more IP headers because fragmentation at the IP layer will produce more IP packets, each one carrying its own IP header. However, performance can be severely affected if, after IP layer fragmentation, then 6LoWPAN fragmentation happens as well (e.g., when the upper layer is not aware of the existence of the fragmentation at the 6LoWPAN layer). One solution is to require that the higher layers have an awareness of the lower-layer features and generate small enough packets to avoid fragmentation. In this regard, the Block option in CoAP can be useful when CoAP is used at the application layer [RFC7959].

Secondly, 6LoWPAN swaps computing with communication. 6LoWPAN applies compression of the IPv6 header. Subject to the packet size limit of IEEE 802.15.4 [IEEE802.15.4], a 40-octet-long IPv6 header and an 8 or 20-octet-long UDP and TCP header will consume even more packet space than the data itself. 6LoWPAN provides IPv6 and UDP header compression at the adaptation layer. Therefore, a lower amount of data will be handled by the lower layers, whereas both the sender and receiver will spend more computing power on the compression and decompression of the packets over the air. Compression can also be performed at higher layers (see Section 6.4).

Finally, the 6LoWPAN Working Group developed the energy-efficient Neighbor Discovery called 6LoWPAN-ND, which is an energy-efficient replacement of the IPv6 ND in constrained environments. IPv6 Neighbor Discovery was not designed for non-transitive wireless links, as its heavy use of multicast makes it inefficient and sometimes impractical in a low-power and lossy network. 6LoWPAN-ND describes simple optimizations to IPv6 Neighbor Discovery, its addressing mechanisms, and duplicate address detection for Low-Power Wireless Personal Area Networks and similar networks. However, 6LoWPAN-ND does not modify Neighbor Unreachability Detection (NUD) timeouts, which are very short (by default three transmissions spaced 1 second apart). NUD timeout settings should be tuned to take into account the latency that may be introduced by duty-cycled mechanisms at the link layer or the alternative, less impatient NUD algorithms should be considered [RFC7048].

IPv6 underlies the higher-layer protocols, including both TCP/UDP transport and applications. By design, the higher-layer protocols do not typically have specific information about the lower layers and thus cannot solve the energy-efficiency problem.

The network stack can be designed to save computing power. For example, the Contiki implementation has multiple cross-layer optimizations for buffers and energy management, e.g., the computing and validation of UDP/TCP checksums without the need of reading IP headers from a different layer. These optimizations are software implementation techniques and are out of the scope of the IETF and the LWIG Working Group.

## 5. Routing Protocols

RPL [RFC6550] is a routing protocol designed by the IETF for constrained environments. RPL exchanges messages periodically and keeps routing states for each destination. RPL is optimized for the many-to-one communication pattern (where network nodes primarily send data towards the border router) but has provisions for any-to-any routing as well.

The authors of the Powertrace tool [Powertrace] studied the power profile of RPL. Their analysis divides the routing protocol into control and data traffic. The control plane carries ICMP messages to establish and maintain the routing states. The data plane carries any application that uses RPL for routing packets. The study has shown that the power consumption of the control traffic goes down over time in a relatively stable network. The study also reflects that the routing protocol should keep the control traffic as low as possible to make it energy friendly. The amount of RPL control traffic can be tuned by setting the Trickle [RFC6206] algorithm parameters (i.e., Imin, Imax, and k) to appropriate values. However, there exists a trade-off between energy consumption and other performance parameters such as network convergence time and robustness.

RFC 6551 [RFC6551] defines routing metrics and constraints to be used by RPL in route computation. Among others, RFC 6551 specifies a Node Energy object that allows to provide information related to node energy, such as the energy source type or the estimated percentage of remaining energy. Appropriate use of energy-based routing metrics may help to balance energy consumption of network nodes, minimize network partitioning, and increase network lifetime.

## 6. Application Layer

### 6.1. Energy-Efficient Features in CoAP

CoAP [RFC7252] is designed as a RESTful application protocol that connects the services of smart devices to the World Wide Web. CoAP is not a chatty protocol. It provides basic communication services such as service discovery and GET/POST/PUT/DELETE methods with a binary header.

Energy efficiency is part of the CoAP protocol design. CoAP uses a fixed-length binary header of only four bytes that may be followed by binary options. To reduce regular and frequent queries of the resources, CoAP provides an observe mode in which the requester registers its interest of a certain resource and the responder will report the value whenever it was updated. This reduces the request/response round trips while keeping information exchange an ubiquitous service; an energy-constrained server can remain in sleep mode during the period between observe notification transmissions.

Furthermore, [RFC7252] defines CoAP proxies that can cache resource representations previously provided by sleepy CoAP servers. The proxies themselves may respond to client requests if the



corresponding server is sleeping and the resource representation is recent enough. Otherwise, a proxy may attempt to obtain the resource from the sleepy server.

CoAP proxy and cache functionality may also be used to perform data aggregation. This technique allows a node to receive data messages (e.g., carrying sensor readings) from other nodes in the network, perform an operation based on the content in those messages, and transmit the result of the operation. Such operation may simply be intended to use one packet to carry the readings transported in several packets (which reduces header and transmission overhead), or it may be a more sophisticated operation, possibly based on mathematical, logical, or filtering principles (which reduces the payload size to be transmitted).

## 6.2. Sleepy Node Support

Beyond these features of CoAP, there have been a number of proposals to further support sleepy nodes at the application layer by leveraging CoAP mechanisms. A good summary of such proposals can be found in [SLEEPY-DEVICES], while an example application (in the context of illustrating several security mechanisms) in a scenario with sleepy devices has been described [CRYPTO-SENSORS]. Approaches to support sleepy nodes include exploiting the use of proxies, leveraging the resource directory [CoRE-RD], or signaling when a node is awake to the interested nodes. Recent work defines publish-subscribe and message queuing extensions to CoAP and the resource directory in order to support devices that spend most of their time asleep [CoAP-BROKER]. Notably, this work has been adopted by the CoRE Working Group.

In addition to the work within the scope of CoAP to support sleepy nodes, other specifications define application-layer functionality for the same purpose. The Lightweight Machine-to-Machine (LwM2M) specification from the Open Mobile Alliance (OMA) defines a queue mode whereby an LwM2M Server queues requests to an LwM2M Client until the latter (which may often stay in sleep mode) is online. LwM2M functionality operates on top of CoAP.

oneM2M defines a CoAP binding with an application-layer mechanism for sleepy nodes [oneM2M].

## 6.3. CoAP Timers

CoAP offers mechanisms for reliable communication between two CoAP endpoints. A CoAP message may be signaled as a confirmable (CON) message, and an acknowledgment (ACK) is issued by the receiver if the CON message is correctly received. The sender starts a

Retransmission Timeout (RT0) for every CON message sent. The initial RT0 value is chosen randomly between 2 and 3 s. If an RT0 expires, the new RT0 value is doubled (unless a limit on the number of retransmissions has been reached). Since duty cycling at the link layer may lead to long latency (i.e., even greater than the initial RT0 value), CoAP RT0 parameters should be tuned accordingly in order to avoid spurious RT0s that would unnecessarily waste node energy and other resources. On the other hand, note that CoAP can also run on top of TCP [RFC8323]. In that case, similar guidance applies to TCP timers, albeit with greater motivation to carefully configure TCP RT0 parameters since [RFC6298] reduced the default initial TCP RT0 to 1 second, which may interact more negatively with duty-cycled links than default CoAP RT0 values.

#### 6.4. Data Compression

Another method intended to reduce the size of the data units to be communicated in constrained-node networks is data compression, which allows to encode data using fewer bits than the original data representation. Data compression is more efficient at higher layers, particularly before encryption is used. In fact, encryption mechanisms may generate an output that does not contain redundancy, making it almost impossible to reduce the data representation size. In CoAP, messages may be encrypted by using Datagram Transport Layer Security (DTLS) or TLS when CoAP over TCP is used, which is the default mechanism for securing CoAP exchanges.

#### 7. Summary and Conclusions

We summarize the key takeaways of this document:

- a. Internet protocols designed by the IETF can be considered the customer of the lower layers (PHY, MAC, and duty cycling). To reduce power consumption, it is recommended that Layer 3 designs should operate based on awareness of lower-level parameters rather than treating the lower layer as a black box (see Sections 4, 5, and 6).
- b. It is always useful to compress the protocol headers in order to reduce the transmission/reception power. This design principle has been employed by many protocols in the 6Lo and CoRE Working Groups (see Sections 4 and 6).
- c. Broadcast and non-synchronized transmissions consume more than other TX/RX operations. If protocols must use these ways to collect information, reduction of their usage by aggregating similar messages together will be helpful in saving power (see Sections 2 and 6.1).

- d. Saving power by sleeping as much as possible is used widely (Section 3).

## 8. IANA Considerations

This document has no IANA actions.

## 9. Security Considerations

This document discusses energy-efficient protocol design and does not incur any changes or challenges on security issues besides what the protocol specifications have analyzed.

## 10. References

### 10.1. Normative References

#### [Bluetooth42]

Bluetooth Special Interest Group, "Core Version 4.2", available from "Legacy Core Specifications", December 2014, <<https://www.bluetooth.com/specifications/bluetooth-core-specification/legacy-specifications>>.

#### [EN300]

ETSI, "Digital Enhanced Cordless Telecommunications (DECT); Common Interface (CI); Part 1: Overview", ETSI EN 300 175-1 V2.6.1, July 2015, <[https://www.etsi.org/deliver/etsi\\_en/300100\\_300199/30017501/02.06.01\\_60/en\\_30017501v020601p.pdf](https://www.etsi.org/deliver/etsi_en/300100_300199/30017501/02.06.01_60/en_30017501v020601p.pdf)>.

#### [G9959]

ITU-T, "Short range narrow-band digital radiocommunication transceivers - PHY, MAC, SAR and LLC layer specifications", ITU-T Recommendation G.9959, January 2015, <<http://www.itu.int/rec/T-REC-G.9959>>.

#### [IEEE802.11]

IEEE, "IEEE Standard for Information technology-- Telecommunications and information exchange between systems Local and metropolitan area networks--Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications", IEEE 802.11, DOI 10.1109/IEEESTD.2016.7786995, <<http://ieeexplore.ieee.org/document/7786995/versions>>.

- [IEEE802.15.4] IEEE, "IEEE Standard for Low-Rate Wireless Networks", IEEE 802.15.4, DOI 10.1109/IEEESTD.2016.7460875, <<https://standards.ieee.org/findstds/standard/802.15.4-2015.html>>.
- [MSTP] ANSI/ASHRAE, "Addenda: BACnet -- A Data Communication Protocol for Building Automation and Control Networks ANSI/ASHRAE Addenda an, at, au, av, aw, ax, and az to ANSI/ASHRAE Standard 135-2012", July 2014, <<https://www.ashrae.org/technical-resources/standards-and-guidelines/standards-addenda/addenda-to-standard-135-2012>>.
- [NFC] NFC Forum, "NFC Logical Link Control Protocol", Technical Specification, Version 1.3, March 2016.
- [oneM2M] oneM2M, "oneM2M - Published Specifications", <<http://www.onem2m.org/technical/published-documents>>.
- [RFC4944] Montenegro, G., Kushalnagar, N., Hui, J., and D. Culler, "Transmission of IPv6 Packets over IEEE 802.15.4 Networks", RFC 4944, DOI 10.17487/RFC4944, September 2007, <<https://www.rfc-editor.org/info/rfc4944>>.
- [RFC6206] Levis, P., Clausen, T., Hui, J., Gnawali, O., and J. Ko, "The Trickle Algorithm", RFC 6206, DOI 10.17487/RFC6206, March 2011, <<https://www.rfc-editor.org/info/rfc6206>>.
- [RFC6282] Hui, J., Ed. and P. Thubert, "Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks", RFC 6282, DOI 10.17487/RFC6282, September 2011, <<https://www.rfc-editor.org/info/rfc6282>>.
- [RFC6298] Paxson, V., Allman, M., Chu, J., and M. Sargent, "Computing TCP's Retransmission Timer", RFC 6298, DOI 10.17487/RFC6298, June 2011, <<https://www.rfc-editor.org/info/rfc6298>>.
- [RFC6550] Winter, T., Ed., Thubert, P., Ed., Brandt, A., Hui, J., Kelsey, R., Levis, P., Pister, K., Struik, R., Vasseur, JP., and R. Alexander, "RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks", RFC 6550, DOI 10.17487/RFC6550, March 2012, <<https://www.rfc-editor.org/info/rfc6550>>.

- [RFC6551] Vasseur, JP., Ed., Kim, M., Ed., Pister, K., Dejean, N., and D. Barthel, "Routing Metrics Used for Path Calculation in Low-Power and Lossy Networks", RFC 6551, DOI 10.17487/RFC6551, March 2012, <<https://www.rfc-editor.org/info/rfc6551>>.
- [RFC6775] Shelby, Z., Ed., Chakrabarti, S., Nordmark, E., and C. Bormann, "Neighbor Discovery Optimization for IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs)", RFC 6775, DOI 10.17487/RFC6775, November 2012, <<https://www.rfc-editor.org/info/rfc6775>>.
- [RFC7228] Bormann, C., Ersue, M., and A. Keranen, "Terminology for Constrained-Node Networks", RFC 7228, DOI 10.17487/RFC7228, May 2014, <<https://www.rfc-editor.org/info/rfc7228>>.
- [RFC7252] Shelby, Z., Hartke, K., and C. Bormann, "The Constrained Application Protocol (CoAP)", RFC 7252, DOI 10.17487/RFC7252, June 2014, <<https://www.rfc-editor.org/info/rfc7252>>.
- [RFC7668] Nieminen, J., Savolainen, T., Isomaki, M., Patil, B., Shelby, Z., and C. Gomez, "IPv6 over BLUETOOTH(R) Low Energy", RFC 7668, DOI 10.17487/RFC7668, October 2015, <<https://www.rfc-editor.org/info/rfc7668>>.
- [RFC8200] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", STD 86, RFC 8200, DOI 10.17487/RFC8200, July 2017, <<https://www.rfc-editor.org/info/rfc8200>>.
- [TS102] ETSI, "Digital Enhanced Cordless Telecommunications (DECT); Ultra Low Energy (ULE); Machine to Machine Communications; Part 2: Home Automation Network (phase 2)", ETSI TS 102 939-2 V1.1.1, March 2015, <[https://www.etsi.org/deliver/etsi\\_ts/102900\\_102999/10293902/01.01.01\\_60/ts\\_10293902v010101p.pdf](https://www.etsi.org/deliver/etsi_ts/102900_102999/10293902/01.01.01_60/ts_10293902v010101p.pdf)>.

## 10.2. Informative References

- [AN079] Kim, C., "Measuring Power Consumption of CC2530 With Z-Stack", Application Note AN079, SWRA292, September 2012, <<http://www.ti.com/lit/an/swra292/swra292.pdf>>.
- [ARCH-6TISCH] Thubert, P., "An Architecture for IPv6 over the TSCH mode of IEEE 802.15.4", Work in Progress, draft-ietf-6tisch-architecture-13, November 2017.
- [CLASS1-CoAP] Kovatsch, M., "Implementing CoAP for Class 1 Devices", Work in Progress, draft-kovatsch-lwig-class1-coap-00, October 2012.
- [CNN-TERMS] Bormann, C., Ersue, M., Keranen, A., and C. Gomez, "Terminology for Constrained-Node Networks", Work in Progress, draft-bormann-lwig-7228bis-02, October 2017.
- [CoAP-BROKER] Koster, M., Keranen, A., and J. Jimenez, "Publish-Subscribe Broker for the Constrained Application Protocol (CoAP)", Work in Progress, draft-ietf-core-coap-pubsub-04, March 2018.
- [ContikiMAC] Dunkels, A., "The ContikiMAC Radio Duty Cycling Protocol", SICS Technical Report T2011:13, December 2011, <<http://soda.swedishict.se/5128/>>.
- [CoRE-RD] Shelby, Z., Koster, M., Bormann, C., Stok, P., and C. Amsuess, Ed., "CoRE Resource Directory", Work in Progress, draft-ietf-core-resource-directory-13, March 2018.
- [CRYPTO-SENSORS] Sethi, M., Arkko, J., Keranen, A., and H. Back, "Practical Considerations and Implementation Experiences in Securing Smart Object Networks", Work in Progress, draft-ietf-lwig-crypto-sensors-06, February 2018.
- [Powertrace] Dunkels, A., Eriksson, J., Finne, N., and N. Tsiftes, "Powertrace: Network-level Power Profiling for Low-power Wireless Networks", SICS Technical Report T2011:05, March 2011, <<http://soda.swedishict.se/4112/>>.

- [RFC7048] Nordmark, E. and I. Gashinsky, "Neighbor Unreachability Detection Is Too Impatient", RFC 7048, DOI 10.17487/RFC7048, January 2014, <<https://www.rfc-editor.org/info/rfc7048>>.
- [RFC7959] Bormann, C. and Z. Shelby, Ed., "Block-Wise Transfers in the Constrained Application Protocol (CoAP)", RFC 7959, DOI 10.17487/RFC7959, August 2016, <<https://www.rfc-editor.org/info/rfc7959>>.
- [RFC8105] Mariager, P., Petersen, J., Ed., Shelby, Z., Van de Logt, M., and D. Barthel, "Transmission of IPv6 Packets over Digital Enhanced Cordless Telecommunications (DECT) Ultra Low Energy (ULE)", RFC 8105, DOI 10.17487/RFC8105, May 2017, <<https://www.rfc-editor.org/info/rfc8105>>.
- [RFC8180] Vilajosana, X., Ed., Pister, K., and T. Watteyne, "Minimal IPv6 over the TSCH Mode of IEEE 802.15.4e (6TiSCH) Configuration", BCP 210, RFC 8180, DOI 10.17487/RFC8180, May 2017, <<https://www.rfc-editor.org/info/rfc8180>>.
- [RFC8323] Bormann, C., Lemay, S., Tschofenig, H., Hartke, K., Silverajan, B., and B. Raymor, Ed., "CoAP (Constrained Application Protocol) over TCP, TLS, and WebSockets", RFC 8323, DOI 10.17487/RFC8323, February 2018, <<https://www.rfc-editor.org/info/rfc8323>>.
- [SLEEPY-DEVICES]  
Rahman, A., "Sleepy Devices: Do we need to Support them in CORE?", Work in Progress, draft-rahman-core-sleepy-nodes-do-we-need-01, February 2014.

## Acknowledgments

Carles Gomez has been supported by the Spanish Government, FEDER, and the ERDF through projects TEC2012-32531 and TEC2016-79988-P.

The authors would like to give thanks for the review and feedback from a number of experts in this area: Carsten Bormann, Ari Keranen, Hannes Tschofenig, Dominique Barthel, Bernie Volz, and Charlie Perkins.

The text of this document was improved based on an IESG document editing session during IETF 87. Thanks to Ted Lemon and Joel Jaeggli for initiating and facilitating this editing session.

## Contributors

Jens T. Petersen, RTX, contributed the section on power save services in DECT ULE.

## Authors' Addresses

Carles Gomez  
Universitat Politecnica de Catalunya  
C/Esteve Terradas, 7  
Castelldefels 08860  
Spain

Email: [carlesgo@entel.upc.edu](mailto:carlesgo@entel.upc.edu)

Matthias Kovatsch  
ETH Zurich  
Universitaetstrasse 6  
Zurich, CH-8092  
Switzerland

Email: [ietf@kovatsch.net](mailto:ietf@kovatsch.net)

Hui Tian  
China Academy of Telecommunication Research  
Huayuanbeilu No. 52  
Beijing, Haidian District 100191  
China

Email: [tianhui@ritt.cn](mailto:tianhui@ritt.cn)

Zhen Cao (editor)  
Huawei Technologies  
China

Email: [zhencao.ietf@gmail.com](mailto:zhencao.ietf@gmail.com)