

# A Testbed Evaluation of the Scalability of IEEE 802.11s Light Sleep Mode

Marco Porsch, Thomas Bauschert

► **To cite this version:**

Marco Porsch, Thomas Bauschert. A Testbed Evaluation of the Scalability of IEEE 802.11s Light Sleep Mode. Thomas Bauschert. 19th Open European Summer School (EUNICE), Aug 2013, Chemnitz, Germany. Springer, Lecture Notes in Computer Science, LNCS-8115, pp.292-297, 2013, Advances in Communication Networking. <10.1007/978-3-642-40552-5\_28>. <hal-01497027>

**HAL Id: hal-01497027**

**<https://hal.inria.fr/hal-01497027>**

Submitted on 28 Mar 2017

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



# A Testbed Evaluation of the Scalability of IEEE 802.11s Light Sleep Mode

Marco Porsch and Thomas Bauschert

Chemnitz University of Technology,  
Reichenhainer Str. 70, 09126 Chemnitz, Germany  
{marco.porsch, thomas.bauschert}@etit.tu-chemnitz.de

**Abstract.** Wireless mesh networks have not yet made their breakthrough in consumer electronics, although they are well suited for use in home or on-the-go environments. Especially IEEE 802.11s is ideal for multimedia sharing on handheld devices as it uses regular Wi-Fi hardware, but provides connectivity without requiring an access point. To maintain proper battery runtime on mobile devices 802.11s introduces new power save schemes. Despite the increased complexity due to the meshed connectivity, these schemes allow efficient energy saving for the mesh nodes. In this paper we present a testbed implementation of the IEEE 802.11s power save schemes on off-the-shelf hardware and assess their effects on current consumption for the idle network case in detail.

**Keywords:** wireless mesh networks, IEEE 802.11s, power save, green networking, energy efficiency.

## 1 Introduction

Today's smartphone users are struggling with the low battery runtime of their devices, as standby times of just one day average have become quite common. As groundbreaking new battery technologies are yet to be found, increasing the energy efficiency is the task at hand. Concerning the power consumption for connectivity, Wi-Fi is already a better choice than cellular for its higher rate and overall better energy-efficiency under network load [1]. In the next generation of smartphones the Wi-Fi functionalities may be extended with the wireless mesh network amendment IEEE 802.11s, which—among other features—allows sharing multimedia contents without requiring the presence of an access point. 802.11s is a standard for wireless mesh networks based on regular Wi-Fi hardware as commonly found in today's laptops and mobile phones. It allows simple connectivity of user devices, similar to Wi-Fi Direct, but also allows meshed multi-hop connectivity when partaking in bigger networks. In IEEE 802.11s networks all mesh nodes are true peers and share the same behavioral rules. Concerning power save, this allows employing equally sophisticated power save schemes on all nodes. Of course there are also new issues; while in managed mode the client has to schedule only the access point's periodic beacon transmissions, in the meshed connectivity of 802.11s the power save scheme has to take all peers

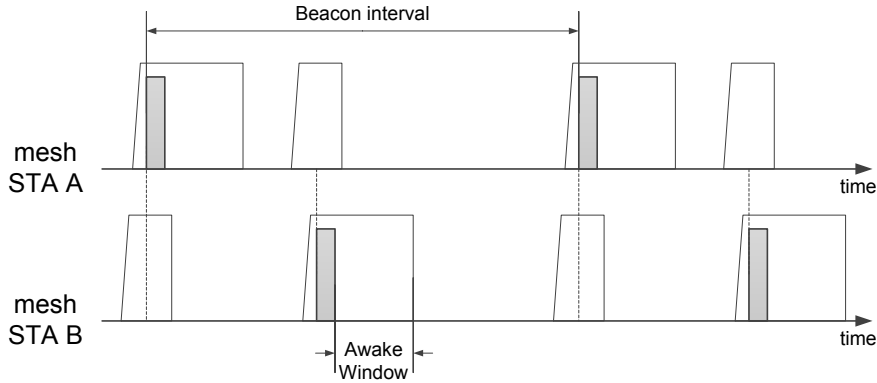
into account for its sleep schedule. In this paper we focus on the case of an idle network with no user data being sent or received. For typical user devices this case is very relevant: a smartphone rests in standby state in the user's pocket most of the time and just periodically checks for incoming mails; a laptop or tablet does not send or receive data while the user is reading a web page. Conserving energy in these situations is most crucial for proper battery runtime.

Key contributions of this paper are testbed measurements of the IEEE 802.11s power save algorithm's effects on current consumption and an analysis of its scalability with respect to an increasing number of peers. The paper is structured as follows: section 2 gives an introduction on the main concepts of IEEE 802.11s power save before section 3 shows and interprets measurements done in our mesh testbed. Subsequently, section 4 concludes and summarizes this paper.

## 2 IEEE 802.11s Power Save

The 802.11s power save schemes are defined in the IEEE 802.11 standard for wireless LAN [2] which, in its 2012 revision, incorporates the 802.11s amendment of 2011. All 802.11 family protocols have a common approach to power save; for saving energy the radio is suspended in times of no activity. With the complex receiver baseband processing disabled the device power consumption is reduced drastically. This makes it more useful than transmission power control as it especially allows idle nodes to conserve energy. Also, without any receiver interrupts, the CPU may spend more time in low power sleep states. The radio will be reactivated for sending a packet or for a scheduled receipt, e.g. of a neighbor beacon.

To reliably schedule doze and wakeups in any of the 802.11 family power save schemes, a tight synchronization is required. The time reference is given by maintaining a local clock and transmitting beacon frames periodically at the TBTT (target beacon transmission time). Dissimilar to wireless LAN ad-hoc and managed mode, mesh mode does not shift the TBTT to a common time; instead, it focuses on keeping the arbitrary offsets between the TSF and TBTT of neighboring nodes constant in the presence of imperfect reference clocks. With the time references synchronized, a power save schedule may be set up to doze and wakeup according to the neighbor TBTTs. Again, the IEEE 802.11s schemes differ from those of managed and ad-hoc mode. Other than waking up at the singular IBSS or access point TBTT in mesh mode there are multiple wakeup events given by multiple peer TBTTs and the own beacon transmission. Also while infrastructure and ad-hoc WLAN employ just a singular power mode towards all neighbors, in IEEE 802.11s the power mode is set for each peer link individually by both parties. This power mode is not a binary representation of power save being on or off; instead three power modes are defined: active, light sleep and deep sleep mode. A link in active mode may receive packets anytime and therefore the radio cannot be suspended at all. A peer link in light sleep mode will only receive packets at certain times. This allows suspending the radio while there are no receipts scheduled on any link. A link in deep sleep mode behaves similarly but the amount of wakeups—and also the possible performance—is further reduced.

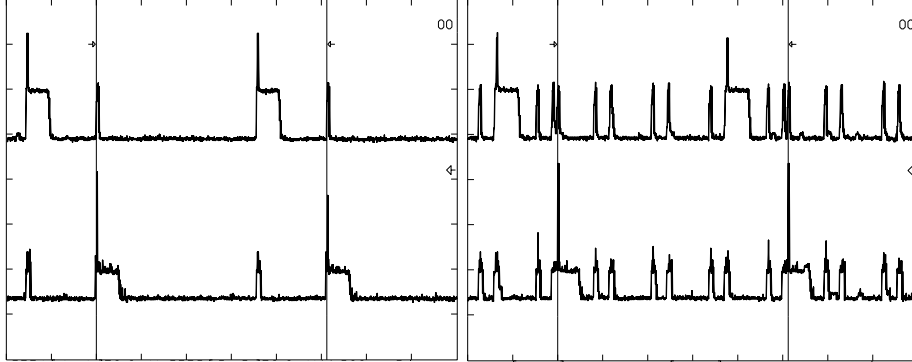


**Fig. 1.** Wakeup scheduling of two peers in light sleep mode towards each other

Links operating in deep sleep mode require special routines for synchronization and frame forwarding which are left undefined in the 802.11 standard. Therefore, we will focus solely on light sleep mode in the remainder of this paper. The link-specific power mode currently used is announced in unicast frames towards the corresponding peer. If any of a node's links is in power save mode, this is also indicated in broadcast frames. The standard leaves the rules defining which power mode to choose for a link up to the implementation.

Fig. 1 illustrates the behavior of a peer link in light sleep mode; here both peers A and B are in light sleep mode towards each other. In this example no further links are established and the sleep/doze schedule is solely determined by the own and the single peer's TBTT. After waking up to transmit its own beacon, node A stays awake for the awake window duration, which is advertised in an information element within the beacon. During this time interval peer B may transmit frames to A. After the awake window passed, node A resumes its doze. B wakes up just before the TBTT of its light sleep peer A to receive its beacon frame. Equally, peer A will later wake up before B's TBTT to receive its beacon. This process repeats periodically with the beacon interval. Beacon frames may contain a TIM (traffic indication map) information element with information about buffered frames, which may be polled during the awake window as part of a subsequent Peer Service Period. For more information on traffic forwarding on mesh links in light sleep mode we would like to refer the interested reader to [3] where we examine this topic using testbed measurements or to [4] where similar research has been conducted by Alam et al. using ns-2 simulations.

An issue of the IEEE 802.11s power save algorithm is its scalability concerning the number of peers; with more peers more wakeups are necessary when links are in light sleep mode. Theoretically, this should not be much of an issue, as the time needed to receive a mesh beacon is typically very low compared to the beacon interval. But to the actual receive time there are additional delays due to CSMA channel access and the delays for powering up the receiver's radio, processing the beacon and resuming doze.



**Fig. 2.** Doze/wakeup cycle of nodes with one peer link in light sleep mode

**Fig. 3.** Doze/wakeup cycle of nodes with seven peer links in light sleep mode

### 3 Testbed Implementation and Results

In order to evaluate the power save algorithm of IEEE 802.11s we chose to test it under real-life conditions in a testbed. In cooperation with cozybit Inc. [5] we implemented the routines necessary in the Linux kernel's Wi-Fi stack [6]. Most of these are made open source and are available in current Linux distributions. Given suitable hardware, the full range of the 802.11s power save schemes can be implemented in software. In our testbed we used two types of off-the-shelf Wi-Fi routers using the "ath9k" device driver: Netgear WNDR3800 and TP-Link TL-MR3020. In our test setup we measured the overall device power consumption which also includes unrelated power sinks like CPU, Ethernet controller and others. We used a  $1\Omega$  shunt resistor in line with a lab voltage supply of 12V for the WNDR3800 and 5V for the TL-MR3020. The measurements are plotted with a Tektronix TLS-216 digital oscilloscope and logic analyzer. Unless defined differently all plots use a sampling rate of 1kS/s and a scale of 50mV/div vertical and 200ms/div horizontal. To reduce transients and noise effects all plots shown use an averaging factor of 10.

Fig. 2 shows the doze and wakeup cycle during one beacon interval for a single peer link with both nodes in light sleep mode. Clearly visible is the sharp spike in power consumption when the beacon frame is transmitted followed by the awake window. The spike of the lower plot serves as trigger source for the oscilloscope. Both nodes are configured with an awake window duration of 100TU and a beacon interval of 1000TU. It can be seen that just before one node transmits its beacon, the peer node wakes up to receive it and goes back to doze just afterwards. When compared to Fig. 1 it is visible that the testbed implementation precisely follows the standard's mandate. The upper plot is the WNDR3800 with current levels of 337mA in doze and 390mA in awake state, the lower plot corresponds to the TL-MR3020 with 108mA and 139mA readings. The area under the graphs between the drawn reference bars corresponds to the current consumption per beacon interval of both devices. Given that the network is idle, this curve will periodically repeat for each beacon interval. The oscilloscope readings show that the WNDR3800 consumes 352,81mAs

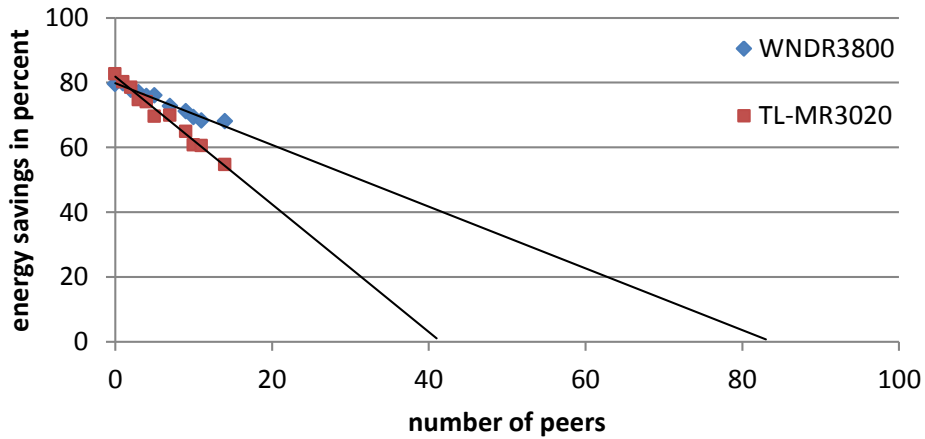


Fig. 4. Energy savings for increasing number of peers with common beacon interval

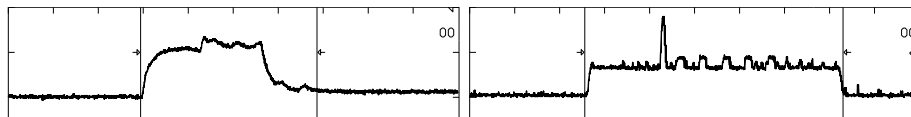


Fig. 5. Peer TBTT wakeup consumption WNDR3800

Fig. 6. Peer TBTT wakeup consumption TL-MR3020

per beacon interval, the TL-MR3020 112,59mAs respectively. The generally higher energy consumption of the WNDR3800 is due to the more complex design of this router model with its additional hardware and our measurement setup where we measure the overall device power consumption. Fig. 3 shows a similar setup as Fig. 2 but here both routers each have seven mesh peer links established. It is obvious that the current consumption is increased due to the further wakeups necessary; now the current consumption per beacon interval is 358,04mAs for the WNDR3800 and 118,16mAs for the TL-MR3020. In case the wakeups do not overlap, it can be expected that the energy savings will decrease linearly with increasing number of light sleep peers as each peer adds another wakeup per beacon interval. We can also expect that with a sufficient number of peers the doze ratio will decrease until the device is practically active full-time. It should be noted that wakeups may randomly overlap, thus reducing the overall awake time. We will subsequently assume the worst case scenario for power save which corresponds to the case of no overlapping wakeups.

Fig. 4 shows the calculated effectiveness of the power save algorithm in relation to the number of established peer links. In the diagram the energy savings due to the power save algorithm are shown as percent value of the difference between the active mode consumption and the consumption with the Wi-Fi device off. The lower limit of 0% energy savings is given by the consumption in active state when the radio is never put to doze, while 100% energy savings would mean that the Wi-Fi functionality draws no power at all. In the graph a situation with zero peers achieves the maximum energy savings of around 80% for both devices. Here no additional wakeups are per-

formed except for transmitting the own beacon every 1000TU and the subsequent awake window of 100TU in our setup. The remaining consumption of the Wi-Fi chipset is due to the mentioned beacon transmission and awake window, but also due to the now activated MAC coprocessor and host interface. One interesting observation is that the power save effectiveness decreases more rapidly for the TL-MR3020 router. The linear interpolation shows that the WNDR3800 would be effectively awake full-time with approximately 84 peers, the TL-MR3020 with 42 peers respectively. The reason for this effect may already be seen in Fig. 3 where the TL-MR3020 in the lower plot showed wider consumption spikes for its wakeups. Fig. 5 and Fig. 6 analyze the characteristics of a peer TBTT wakeup in more detail using a higher sampling rate of 50kS/s and a horizontal scale of 4ms/div. Here we measure about 12ms from waking up until resuming doze for the WNDR3800 and 23ms for the TL-MR3020. This is due to the different hardware of both router models with different revisions of the built-in Wi-Fi chipset and different processing power of the CPU.

## 4 Conclusion

In this paper we have analyzed and evaluated the IEEE 802.11s power saving schemes in terms of scalability. By implementing these schemes in the Linux kernel's Wi-Fi stack, we provide energy consumption measurements of off-the-shelf hardware. Our results show that, even for the not power-optimized router hardware, considerable energy savings are achieved. We analyzed that—unless TBTTs randomly overlap—the efficiency of the power save algorithm decreases linearly with the number of light sleep peers. The reason for this decrease is that the power-saving node has to wake up from doze for each light sleep peer link individually. The slope of the efficiency decrease depends on the hardware employed. In future work will aim to solve these scalability issues while staying in compliance with the 802.11 standard. Furthermore, we will address open implementation issues of links in deep sleep mode and intend to perform measurements with respect to energy-efficiency in bigger networks.

## References

1. Castignani, G., Montavont, N., Lampropulos, A.: Energy Considerations for a Wireless Multi-homed Environment, EUNICE, Lecture Notes in Computer Science 6955, pp. 181-192, Jan 2011
2. IEEE Std. 802.11™-2012, Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. IEEE Computer Society (May 2012)
3. Porsch, M., Bauschert, T., A Testbed Analysis of the Effects of IEEE 802.11s Power Save on Mesh Link Performance, EUNICE, volume 7479 of Lecture Notes in Computer Science, page 1-11. Springer, (2012)
4. Alam, M.N., Jäntti, R., Knecht, J., Nieminen, J., Performance Study of IEEE 802.11s PSM in FTP-TCP, Vehicular Technology Conference (VTC Fall), 2012 IEEE, pp. 1 – 5
5. cozybit inc., [www.cozybit.com](http://www.cozybit.com)
6. Linux Kernel wireless driver API for SoftMAC devices, mac80211 kernel module source code, <http://git.kernel.org/cgi/linux/kernel/git/torvalds/linux.git/tree/net/mac80211>