

モバイル端末の消費電力削減のための BLE を用いた Wi-Fi ウェイクアップ制御方式の実装

田中 直也[†] 湯 素華[†] 小花 貞夫[†]

概要 : スマートフォンの使用可能時間の短さが問題になっており, その原因の一つとして Wi-Fi の使用がある. 最近では Social Networking Service(SNS)のように, データを受信するタイミングが不規則的なアプリケーションが増えており, その通信のために Wi-Fi を常に起動して電力を多く消費している. 筆者らは, モバイル端末の消費電力の改善を目的として, 電力消費の少ない Bluetooth Low Energy(BLE)を用いて必要ときにだけ Wi-Fi をウェイクアップさせることで, Wi-Fi の使用による消費電力の削減を行う方式を提案済みであり, アプリケーションによるプロトタイプの実装も既に行っている. 今回, 本方式を PC 上に実装し評価した結果, 従来の省電力モード(PSM)と比較して, 提案方式により消費電力を約 4 割削減できる可能性があることを確認した.

キーワード : Wi-Fi, ウェイクアップ制御, BLE.

Implementation of Wi-Fi Wake-Up Control Using BLE for Energy Saving in Mobile Device

NAOYA TANAKA[†] SUHUA TANG[†] SADA O OBANA[†]

Abstract: The short lasting time per charge of smart phones is a concern to users, and one of the causes is the frequent use of the embedded Wi-Fi module. With the widespread of new applications like social networking service (SNS), the Wi-Fi module in a smartphone is kept awake because packets arrive at irregular timing, which is unpredictable. To reduce energy consumption, we had proposed a Wi-Fi wake-up control method by using a low power BLE module as a wake-up radio, which activates the power-hungry Wi-Fi module on demand. We also made a prototype system, which, however, has limited effect because of the constraint of Android API. In this paper, we implemented this method in the driver level, and evaluated its performance on a testbed PC. Experimental results confirm that the proposed method can reduce energy consumption of a Wi-Fi module by about 40%, compared with the conventional power-save mode(PSM).

Keywords: Wi-Fi, Wake-up control, BLE.

1. はじめに

スマートフォンにおける問題の一つとして, 消費電力の増加による使用可能時間の短さがある. その原因としては, 画面表示, Wi-Fi, GPS, センサなどのデバイスの使用があり[1], 様々なアプローチで消費電力を抑える研究が行われている. ここでは, その中でも消費電力の大きい Wi-Fi に着目する.

近年, LINE に代表される SNS のような, データを受信するタイミングが不規則なアプリケーションが増えており, 待ち受けのために Wi-Fi を常に起動して電力を多く消費している場合がある. このため, 筆者らは先に, Bluetooth4.0 から殆どのスマートフォンで実装されている消費電力が小さい BLE(Bluetooth Low Energy)をウェイクアップ送受信機として用いて, 必要なときにだけ Wi-Fi を立ち上げるウェイクアップ制御によって消費電力を削減する制御方式を提案した[2]. このたび本方式を PC 上に実装した

ので, そのための予備実験, システムの設計・実装方法ならびに評価実験の結果について報告する.

2. 従来技術と先行研究

2.1 Wi-Fi の Power Save Mode(PSM)

Wi-Fi のアクセスポイント(AP)は定期的にビーコンを送信する. Wi-Fi 端末には, 通常モードと Power Save Mode(PSM)[3]と呼ばれる省電力モードがある. 通常モードでは, 端末は Wi-Fi を常時起動してビーコンを待ち受ける.

一方, PSM モードでは, アウェイク状態(データの送受信を行うことができる状態)とスリープ状態(データの送受信をしない低消費電力状態)を定期的に繰り返す. 端末がスリープ状態であるとき, 自端末宛のパケットの有無を確認するために定期的にアウェイク状態に移行し, ビーコンを受信する. 端末は全てのビーコンを受信するのではなく, ビーコンに含まれる情報から受信間隔を計算し, その間隔毎に受信する. ビーコンには自端末宛のデータの有無に関する

[†] 電気通信大学 大学院情報理工学研究所
Graduate School of Informatics and Engineering, The University of Electro-Communications.

る情報が含まれ、自端末宛のデータがある場合はアウェイク状態のままデータを待ち受け、無い場合はスリープ状態に移行する。LINEのようなアプリケーションでメッセージを受信する場合、そのデータがいつ来るか分からないので、PSMでも定期的にWi-Fiをウェイクアップしてビーコンを受信し続ける必要があり、その分の電力を消費する。

2.2 先行研究

ウェイクアップを用いたWi-Fiの省電力化に関する研究には、端末側からAPをウェイクアップする手法[5]と、AP側から端末をウェイクアップさせ、消費電力の大きいWi-Fiをスリープさせる手法[6][7]がある。[5]の場合、端末側からAPのウェイクアップに特殊な信号を送信するため、AP側に専用の受信装置を搭載する必要がある。[6]の場合、音波を使ってウェイクアップ信号を送信し、マイクで検知する。そのため、通信が音速で行われるために遅延が生じ、また、到達距離が短い等の欠点がある。[7]では、ZigBeeをウェイクアップ制御に使用するが、ZigBeeはBLEより消費電力が多く、また、ZigBeeはモバイル端末では普及していない。

3. 先に提案したBLEを用いたWi-Fi ウェイクアップ制御方式の概要

提案方式では、アクセスポイント(AP)はBLEを用いてビーコン情報に含まれる端末宛のデータの有無に関する情報を送信し、端末はBLEによりその情報を受信して、自端末宛のデータがあることが示されている場合、自端末のWi-Fiをウェイクアップさせてデータを受信する。データを受信する時以外にはWi-Fiをスリープ状態にする。これにより、データ待機時の消費電力の削減が期待できる。また、BLEによる情報の送信には、アドバタイズメントと呼ばれるBLEデバイス同士がお互いを発見するためのパケットを用いる。

図1に提案方式による通信の概要を、図2にWi-Fiのウェイクアップの流れを示す。前提として、APは電源が供給されており、Wi-Fiを常時起動しているものとする。

- 1) APは、各端末宛のデータの有無に関するビーコン情報をBLEのアドバタイズメント(以下、BLEビーコンと呼ぶ)を用いて送信する。
- 2) 各端末は、普段Wi-Fiをスリープ状態にして、APからBLEを介してBLEビーコンを受信する。
- 3) BLEビーコンを受信した端末は、ビーコン情報から自端末宛のデータが有るかどうを確認する。
 - a) 自端末宛のデータが有る場合、自端末のWi-Fiをウェイクアップして、データを受信し、同時にBLEがスリープ状態に移行する。データの受信が終了したら、Wi-Fiをスリープ状態にして、BLEビー

コンの受信を再開する。

- b) 自端末宛のデータが無い場合、BLEビーコンの受信を継続する。

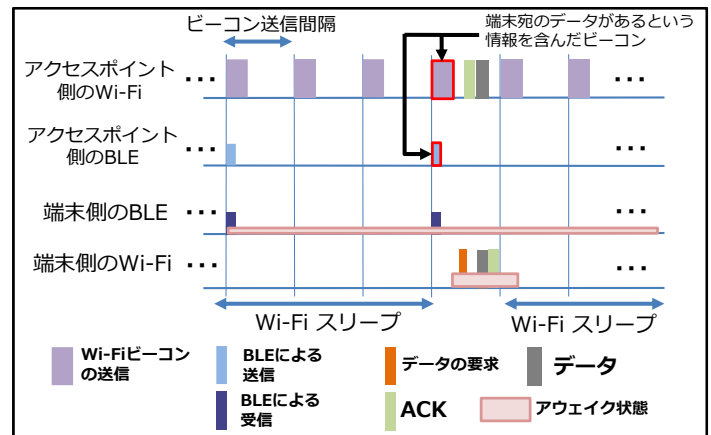


図1 提案方式の概要

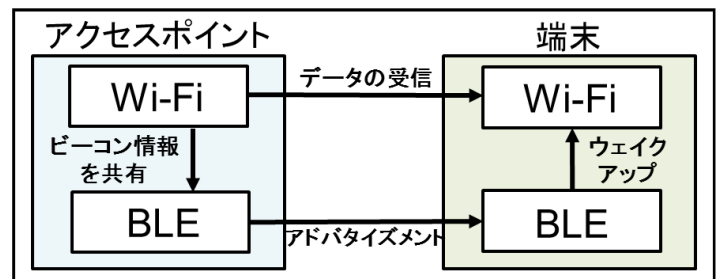


図2 提案方式によるウェイクアップの流れ

3.1 プロトタイプにおける電流測定

Android上で動作する提案方式を模したアプリケーション(プロトタイプ)を作成し、BLE搭載の端末(スマートフォン:LG G2 mini)上で動作させ、Wi-Fi通常モード、PSMモードの消費電流量を比較した。APから7KBのデータを一定間隔毎に送信した時の、提案方式と従来のWi-Fiのみの場合の、データを受信するのにかかる消費電流量を計算した。消費電流の測定は、それぞれ50回行い、平均を取った。電流の測定の際、テスターにはNR-2000を用いた。また、プロトタイプの場合、AndroidのAPIを用いているためにWi-Fiをスリープ状態にすることができず、Wi-Fiをスリープ/ウェイクアップの代わりにON/OFFしている。そのため、データ受信にかかる時間には、Wi-Fi ONの時間が含まれる。

図3に測定機器の接続構成を示す。表1に測定したアイドル状態と受信時の毎秒の消費電流を、表2にデータ受信にかかる時間を示す。

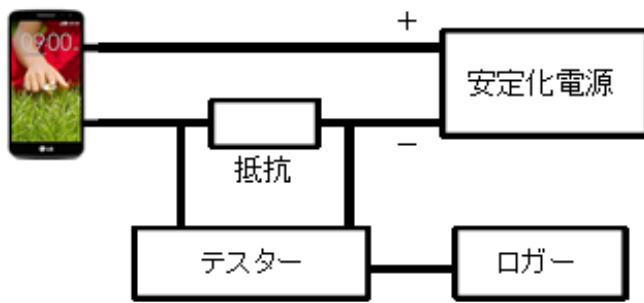


図 3 測定機器の接続構成

表 1 アイドル状態と受信時の消費電流

	アイドル状態[mA]	データ受信時[mA]
プロトタイプ(BLE)	7.02	51.97
Wi-Fi 通常	22.66	49.22
Wi-Fi PSM	11.27	49.22

表 2 データ受信にかかる時間

	データ受信にかかる時間[s]
プロトタイプ	3.42
Wi-Fiのみ	0.02

表 1 より、データ受信時の消費電流は本方式の方が多く、表 2 より、プロトタイプの方がデータ受信にかかる時間が長いことが分かる。これは、プロトタイプでは Wi-Fi を ON/OFF しており、それに伴った AP のスキャンなどの余分な動作が含まれるためと考えられる。表 3 にプロトタイプのデータ受信にかかる消費電流量(電流×時間)の内訳を示す。

表 3 データ受信にかかる消費電流量の内訳

各状態	かかる時間[s]	消費電流量[mAs]
Wi-Fi ON	1.18	65.74
スキャン	0.88	43.26
AP への接続	0.34	24.13
アイドル状態	0.65	14.73
データ受信	0.02	0.98
Wi-Fi OFF	0.34	24.94

4. 予備実験

4.1 Wi-Fi スリープ/ウェイクアップ制御

Wi-Fi のスリープ制御をアプリケーション層ではなく、無線 LAN ドライバ等の下位層で行うことで、AP との接続を維持した状態でスリープする事ができるため、表 3 にあるスキャンや接続等の動作を無くすことができ、Wi-Fi ON するのにかかる時間と消費電流を抑える事ができると考えられる。Linux 環境におけるデバイスドライバの書き換え

のために、デスクトップ PC(Ubuntu)、Wi-Fi MiniPCI カード (Ubiquiti SR71-E) とオープンソースの無線 LAN ドライバ (ath9k[8]) を用いた。効果の検証のために、Wi-Fi をスリープ状態にしてウェイクアップさせた時の電流を測定した。また、PC 全体ではなく Wi-Fi のみによる消費電流を測定するために、MiniPCI の延長ケーブルを用いて Wi-Fi カードを接続し、チップに電流を供給している部分を切断し抵抗を挟み測定を行った。

図 4 に MiniPCI 延長ケーブル加工後の様子を、図 5 に測定回路の構成を示す。

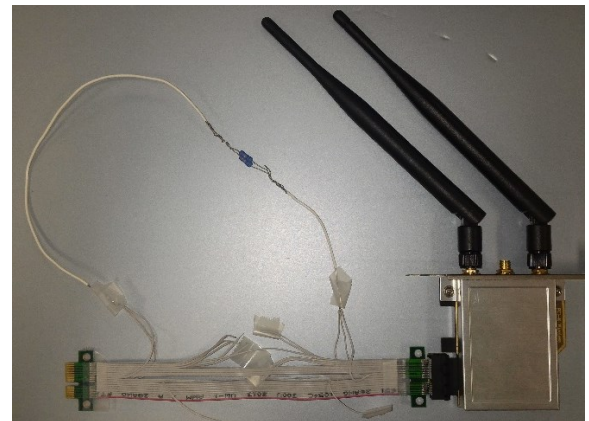


図 4 MiniPCI 延長ケーブルと Wi-Fi MiniPCI カード

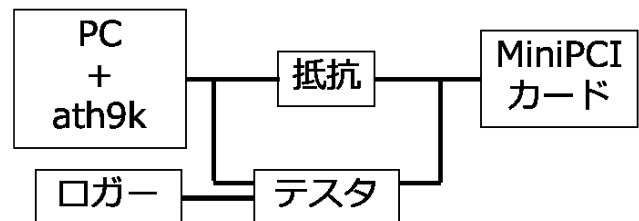


図 5 測定回路の構成

また、デバイスドライバのソースコードを書き換え、通常はスリープ時間が 100ms である所を、測定をやすくするために約 5s まで延長した。ソースコードの書き換えが実際のスリープ時間に反映されているかを確認するために、`printk` を用いてスリープ/ウェイクアップ時にメッセージを出力させるように書き換え、`dmesg` コマンドでタイムスタンプと共に表示することで、実際のスリープ時間が約 5s になることを確認した。文末に `NETWORK_SLEEP` と表示されているとき、Wi-Fi がスリープしたことを示し、`AWAKE` が表示されているときは、Wi-Fi がウェイクアップしたことを示す。図 6 に `dmesg` による出力を、図 7 にウェイクアップ時の電流波形を示す。

図 7 より、プロトタイプでは Wi-Fi ON 時に先に述べたような余分な動作によって電力を消費しているが、Wi-Fi をスリープ状態にすることで余分な動作が無くなっていることが分かる。

```
[ 132.040032] ath9k_ps_restore, NETWORK_SLEEP
[ 136.828160] ath9k_hw_setpower, AWAKE
[ 136.928235] ath9k_ps_restore, NETWORK_SLEEP
[ 141.804156] ath9k_hw_setpower, AWAKE
[ 141.904226] ath9k_ps_restore, NETWORK_SLEEP
[ 146.828107] ath9k_hw_setpower, AWAKE
```

図 6 dmesg による出力



図 7 ウェイクアップ時の電流波形
ON/OFF 制御(上), SLEEP/Wake-up 制御(下)

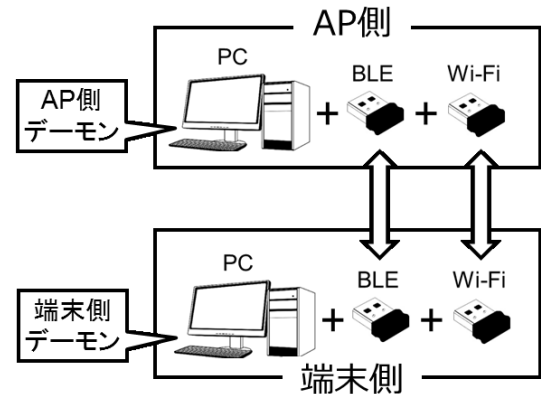


図 8 システムの構成

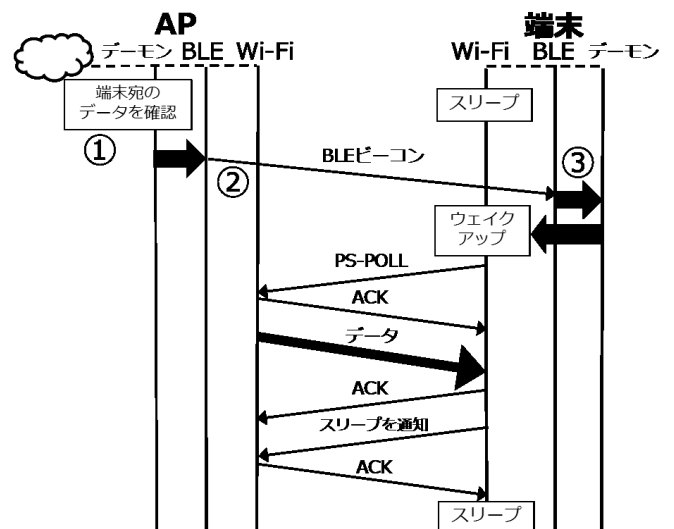


図 9 システムの動作概要

5. システムの設計と実装

AP と端末として、2 台のデスクトップ PC に、それぞれ提案方式の動作を行うデーモン(AP 側デーモン, 端末側デーモン)を実装した. 図 8 にシステムの構成を示す. また, 図 9 にシステムの動作の流れを示す.

- 1) AP は, Wi-Fi をスリープしている端末(スリープ端末)宛のデータをバッファに入れる際に, スリープ端末の MAC アドレスを取得し, AP 側デーモンへ渡す.
- 2) AP 側デーモンは, MAC アドレスをアドバタイズメントに付加し, BLE ビーコンとしてブロードキャストする. 端末側デーモンは BLE で BLE ビーコンのスクランを行い, BLE ビーコンを受信すると, BLE ビーコンに付加されている MAC アドレスが自端末と一致するかをチェックし, もし一致すれば Wi-Fi をウェイクアップさせて AP に PS-POLL(端末がウェイクアップしたことを AP に伝える信号)を送信する.

5.1 無線 LAN ドライバの書き換え

Wi-Fi PSM の仕様では, ネットワーク側よりスリープ端末宛のデータが AP に届いたとき, AP はバッファにそのデータを蓄え, スリープ端末がアウェイク状態に移行した際にその端末に対して一括して送信する. そのため, 本システムの実装では, AP がスリープ端末宛のデータをバッファに格納するタイミングでスリープ端末のアドレスを取得し, BLE ビーコンを端末に送信するようにする.

無線 LAN ドライバのカーネルモジュールである mac80211(backports-3.13-1/net/mac80211)を調査した所, tx.c 内の関数 ieee80211_tx_h_unicast_ps_buf で PSM 端末宛のデータをバッファに格納する関数 skb_queue_tail を用いてパケットの操作を行っていることを確認した.

BLE ビーコンを適切なタイミングで送信するために, mac80211 を書き換え, スリープ端末宛のデータを skb_queue_tail を用いてバッファに格納した直後に端末の MAC アドレスを読み, ユーザスペースで動作する AP 側デーモンと共有するために, proc ファイルシステムに書き込むようにした.

5.2 AP 側デーモン

AP 側デーモンは、proc ファイルシステムから mac80211 が書き込んだスリープ端末の MAC アドレスを読む。その MAC アドレスを BLE のアドバタイズメントに書き込み、BLE ビーコンとしてブロードキャストする。アドバタイズメントのフォーマット[9]は図 10 のようになっており、AD Data(最大 30 オクテット)のフィールドにビーコン情報を書き込む。本システムでは、AD Data の先頭 4 オクテットに BLE ビーコンであることを示すフラグ(BLE ビーコンフラグ)を書き込み、その後続けて 6 オクテットの MAC アドレスを書き込む。BLE の制御には Bluetooth のプロトコルスタックである BlueZ[10]を、PC の AP 化には hostapd[11]を用いた。

図 11 に BLE ビーコンのフォーマットを示す。

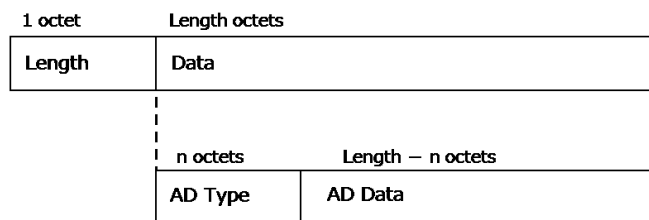


図 10 アドバタイズメントのフォーマット

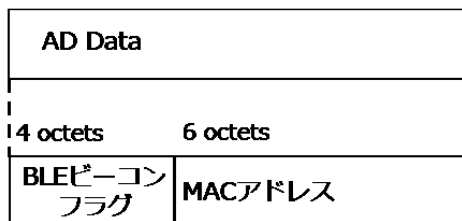


図 11 BLE ビーコンのフォーマット

5.3 端末側デーモン

端末側デーモンは、BLE のアドバタイズメントを受信し、受信したアドバタイズメントの AD Data を読み、BLE ビーコンフラグがあるかどうかを確認する。BLE ビーコンフラグがある場合、書き込まれている MAC アドレスを抽出し、自端末の MAC アドレスと比較する。アドレスが一致する場合、端末側の Wi-Fi を立ち上げる。Wi-Fi が立ち上がると、従来の PSM と同様に、自動的に PS-POLL(端末がウェイクアップしたことを AP に伝える信号)やデータのやり取りを行い、その後再び Wi-Fi はスリープ状態へと移行する。BLE の制御には同様に BlueZ を用いる。

5.4 動作確認

デバイスドライバを書き換え、端末側 Wi-Fi のスリープ時間を約 10s まで延長し、PSM 時と本システムを動作させたそれぞれの状態で AP 側から端末側に対して ping を 5 回送信し、本システムにおける端末側の Wi-Fi がウェイクアップして ping を受信することができるかを確認した。また、

端末側の Wi-Fi が他のパケットではなく確認のための ping でウェイクアップしたということが分かるように、他のパケットがない環境で行った。表 4 に ping 応答時間を示す。

表 4 ping 応答時間

PSM 時の応答時間[ms]	本システムの応答時間[ms]
1095	152
3693	261
4318	166
8014	101
3028	211

表 4 より、本システムでは平均して約 180ms で端末側から ping の応答が返っていることがわかる。また、PSM の場合には、スリープ時間を 10s にしているため 0~10s の時間がかかっていることが分かる。この結果から、本システムが提案方式通りに動作して、BLE を用いて端末側の Wi-Fi をウェイクアップできていることを確認できた。

6. 評価実験と考察

6.1 評価実験

予備実験と同様の測定方法で、作成したシステムと周期 100ms の PSM 時において、端末がデータ(1 回の ping)を受信した際の電流を測定した。図 12 に PSM と提案方式でデータを受信した時の電流波形を示す。

図 12 のそれぞれにおいて、中央部分がデータを受信している時にあたる。本システム動作時には、Wi-Fi がスリー

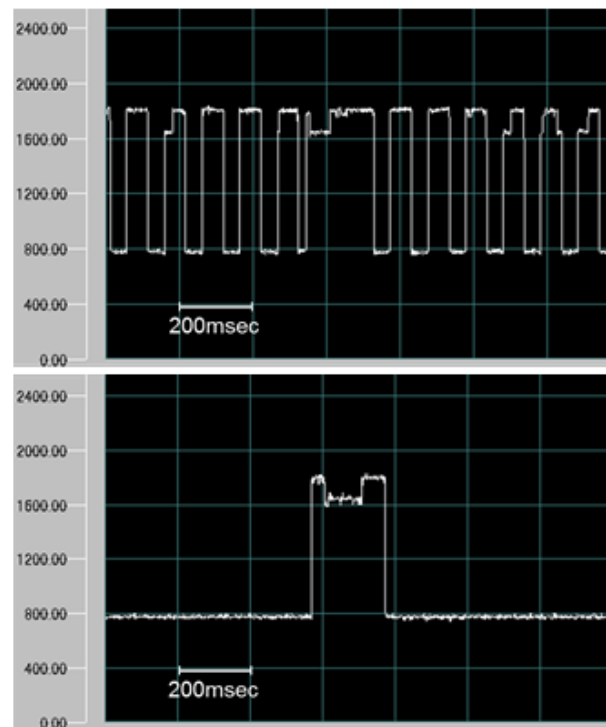


図 12 PSM 時(上)とシステム動作時(下)の電流波形

プの状態から、BLE ビーコンを受信して Wi-Fi が立ち上がりデータを受信し、その後にスリープ状態に戻る様子が確認できる。本システムの場合、データを受信していない時には常にスリープ状態になっている。電流の測定に用いたロガーを使用し、PSM と提案方式のデータ待機時(データ受信時以外の部分)の消費電流を計算した。表 5 にそれぞれの消費電流を示す。

表 5 データ待機時の消費電流

	データ待機時の消費電流[mA]
PSM	1356
本システム	788

6.2 考察

表 5 より、本方式は PSM と比べてデータ待機時の消費電力が約 4 割少ないことが分かる。PSM はデータ待機時にも定期的にビーコンを受信するために消費電力が高くなる。本方式では代わりに BLE を用いて BLE ビーコンを受信し、Wi-Fi をスリープさせているため消費電力を削減できる。本方式はデータ待機時に BLE ビーコンを受信するために BLE を起動しているため、その消費電流を加算する必要があるが、BLE におけるアイドル状態の消費電力は $550 \mu A$ 程であるため[12]、本システムの場合にはほぼ無視することができる値である。また、スマートフォンの場合においても、予備実験の PSM のアイドル時の消費電流は 11mA であるため、効果が期待できる。

本システムでは端末がデータを受信した後のスリープ制御は、従来の PSM と同様に一定期間データが来なければ Wi-Fi をスリープさせるという動作のままになっている。スリープ制御を行い、データ受信直後に Wi-Fi をスリープさせることでさらなる消費電力削減が可能であると考えられる。

7. おわりに

本稿では、モバイル端末の消費電力削減のために先に筆者らが提案した、BLE を用いた Wi-Fi ウェイクアップ制御方式をアプリケーションとして Android 上に実装した際の Wi-Fi ウェイクアップ時の遅延を解消することを目的に、Wi-Fi の制御をデバイスドライバレベルで行った場合の電流測定を行い、ウェイクアップ時の遅延が起きないことを確認した。また、2つの PC 上でそれぞれ AP と端末として動作するデーモンとして提案方式を実装し、動作確認を行った上で、評価実験として電流測定を行い、PSM と比較した場合、提案方式によりデータ待機時の消費電力を約 4 割削減できる可能性があることを示した。

今後は、本システムにおいてパケット受信後に直ちにスリープさせる事や、アダプタイズメントの送信周期の最適化による遅延の解消と消費電力削減効果の向上、より多く

の端末が AP に接続する場合のために、MAC アドレスではなく AP との接続 ID(1 ビット)を用いて端末を識別する等の改良を行い、本方式を使用したことによる通信の遅延の有無や、スマートフォン等の端末における提案方式の効果を検証する予定である。

また、端末に対して送られるデータが少量の場合には、端末は、Wi-Fi を起動せずに BLE を用いてデータを受信するなどの、消費電力削減の効果をより高めるための工夫を行う。

参考文献

- [1] A. Carroll and G. Heiser, "An analysis of power consumption in a smartphone," in Proc. USENIXATC, 2010, p. 21.
- [2] Naoya Tanaka, Suhua Tang, Sadao Obana, "モバイル端末の消費電力削減のための Bluetooth4.0(BLE)を用いた Wi-Fi ウェイクアップ制御方式の提案," DICO 2015.
- [3] Andrew J. Pyles, Xin Qi, Gang Zhou, Matthew Keally and Xue Liu, "SAPSM Smart Adaptive 802.11 PSM for Smartphones," UbiComp '12, pp. 11-20, 2012.
- [4] I. Demirkol, C. Ersoy, and E. Onur, "Wake-up receivers for wireless sensor networks: Benefits and challenges," IEEE Wireless Commun. Mag., vol. 16, no. 4, pp. 88-96, Aug. 2009.
- [5] S. Tang, H. Yomo, and Y. Takeuchi, "Optimization of Frame Length Modulation-Based Wake-up Control for Green WLANs," IEEE Transactions on Vehicular Technology, Vol.64, Issue.2, pp.768-780, 2014.
- [6] Mostafa Uddin and Tamer Nadeem, "A2PSM: Audio assisted Wi-Fi power saving mechanism for smart devices," ACM HotMobile'13, USA, Feb 2013.
- [7] Yifan Zhang and Qun Li, "HoWiES: A holistic approach to ZigBee assisted Wi-Fi energy savings in mobile devices," INFOCOM 2013, pp.1366-1374, USA, Apr 2013.
- [8] ath9k, wireless.wiki.kernel.org/en/users/drivers/ath9k.
- [9] Bluetooth Core Specification 4.2, www.bluetooth.com/specifications/adopted-specifications.
- [10] BlueZ, www.bluez.org/.
- [11] hostapd, w1.fi/hostapd/.
- [12] Measuring Bluetooth Smart Power Consumption, www.ti.com/lit/an/swra478a/swra478a.pdf.