

Optimizări în conexiuni Bluetooth Low Energy pentru aplicații medicale

Conceptul de Bluetooth Low Energy (abreviat BLE) se bazează pe reducerea energiei consumate de device-urile ce necesită atât o autonomie ridicată cât și capacități puternice de comunicație cu alte dispozitive.

În timpul funcționării unui modul Bluetooth, majoritatea consumului de curent se datorează transmisiei și recepției de date prin radio. Astfel, BLE reușește să **maximizeze autonomia** prin a reduce transferul de date la un minim necesar susținerii unei conexiuni între două dispozitive atunci când nu sunt date utile de transmis (date de la senzori, notificări, etc). În cazul aplicației curente, BLE urmează o topologie client-server, unde clientul este, de cele mai multe ori, un telefon, un smart watch sau un laptop, iar serverul este dispozitivul ce colectează date din mediul înconjurător (senzori), le procesează și apoi **așteaptă** ca un client să le acceseze.

Device-ul prezentat poate fi în două stări de bază: Advertising și Connected.

- *Advertising*: Cât timp niciun client nu încearcă realizarea unei conexiuni, modulul Bluetooth va transmite periodic pachete cu un minim de informații despre identitatea sa, pentru a putea fi găsit de către un potențial client prin scanare.
- *Connected*: După ce un client realizează o conexiune, nu se vor mai transmite Advertising Packets, dar periodic se vor transmite pachete lipsite de informație utilă către clientul conectat, pentru a anunța faptul că device-ul este încă activ și prezent în rețea. Aceste pachete se numesc Connection (Interval) Packet.

Specificația BLE permite ca un server și un client să își stabilească un set de parametri de conexiune ce, în principal, descriu frecvența pachetelor *Connection Packet* cât și un Slave Latency, o caracteristică a serverului de a **ignora** un număr bine definit de pachete de conexiune (își va menține modulul **radio oprit** pentru o

perioada mai lungă) pentru a economisi energie. Deoarece clientul inițiază transmisiile de date, la un interval de timp prestabilit (ținând cont și de Slave Latency) cele două entități trebuie să poată comunica pentru a resincroniza conexiunea – serverul trebuie să aștepte un pachet de conexiune de la client căruia să îi răspundă.

În cadrul pachetelor *Advertising packet* se regăsesc date despre **serviciile** oferite de un server: un serviciu este o aplicație specifică a sistemului ce prezintă diverse **caracteristici** – structuri ce grupează datele utile pentru un acces mai eficient. Un server poate oferi mai multe servicii, spre exemplu: un serviciu de informații generice despre dispozitiv, un serviciu de monitorizare a nivelului bateriei și un serviciu „util” cu date colectate de la senzori. Datele stocate în caracteristici pot fi accesate de către client prin cereri, dar serverul poate, la orice moment, să notifice clientul că există date noi disponibile pe care ar trebui să le preia. Pe baza acestei funcționalități este construită aplicația descrisă mai jos.

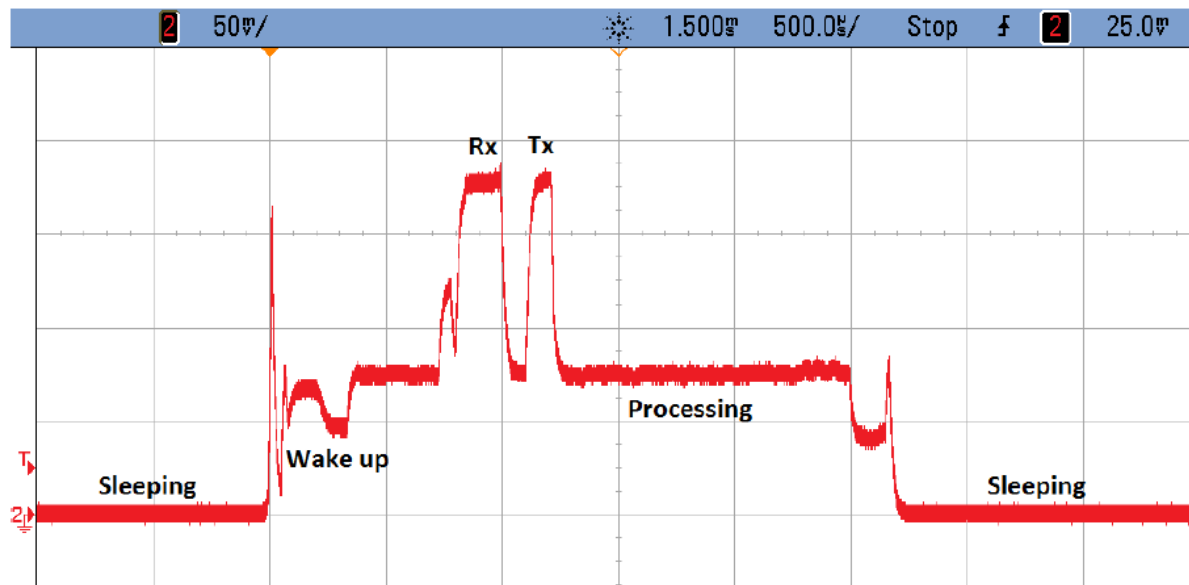
BLE în dispozitive medicale

De cele mai multe ori, un dispozitiv electronic medical este un sistem de achiziție a unor date despre pacient: puls, temperatură, presiune arterială, etc. Pentru această clasă de dispozitive, pe primul loc ca importantă constructivă se află autonomia: cu cât un device poate fi folosit mai mult cu o singură încărcare, cu atât se pot colecta mai multe date folositoare.

Pentru optimizarea consumului de curent, am ales să construiesc un algoritm care, pe baza indicelui de semnal radio RSSI, decide dacă un set de date colectate anterior ar trebui transmise către client (telefon) pentru procesare sau dacă ar trebui amânata transmisia până când condițiile radio sunt mai bune (distanța între cele două radio-uri este mai mică – când telefonul pacientului este aproape de device). Astfel, device-ul se poate afla în două stări de funcționare, odată conectat la client:

1. Colectare locală de date, într-un buffer (în limita memoriei interne), cât timp clientul este la o distanță ce ar face transmiterea setului de date ineficient și costisitor. Se transmit doar pachetele periodice de conexiune.
2. Transmiterea setului de date către client prin notificări succesive până la golirea buffer-ului de date stocate, în cazul în care condițiile de conectivitate radio sunt optime. Se transmit succesiv toate datele stocate.

Pentru a transmite date la o distanță mai mare, intuitiv, se va consuma mai mult curent. La puterea minimă de transmisie, cu raza minimă de acțiune, modulul radio consumă **5.5mA**. Pentru a transmite la putere maximă, pentru a maximiza distanța efectivă, acesta consumă **16mA**. Cât timp se recepționează date, modulul radio consumă un curent constant de aproximativ **13mA**.



1. Curentul consumat într-o transmisie de putere maximă

Având în vedere numărul mare de seturi de date stocate în buffer, este clar că se dorește minimizarea curentului consumat în perioada de transmitere a datelor prin radio. Pentru un interval de conexiune de 7,5ms cu transmisie la putere maximă (figura 1), se consumă aproximativ 8.53mA. Pentru aceeași conexiune, la putere minimă se va consuma aproximativ 7.75mA. Într-o singură transmisie se pot transmite maxim 6 pachete, deci consumul va fi maxim 51.18mA și minim 46.5mA.

Pentru a transmite date colectate cu o rată de 10Hz, e nevoie ca o transmisie să se producă la fiecare 0.6 secunde. Consumul mediu ar fi, astfel:

$$\frac{\{(600 - 7.5)ms * sleepCurrent\} + [7.5ms * connectionCurrent]}{600}$$

În cazul de economie maximă a energiei, consumul mediu ar fi 0.582mA, iar în cazul transmisiei la putere maximă ar fi 0.64mA. Pentru o baterie tipică Litium CR2032 de 220mAh, diferența ca autonomie a device-ului ar fi de aproximativ **35 ore**. (378 ore vs 343 ore). Această optimizare aduce, deci, un beneficiu de **minim 10% timp de funcționare în plus**, în aplicația studiată.

Bibliografie:

<https://devzone.nordicsemi.com/b/blog/posts/how-different-ble-packet-types-influence-throughput>

<https://devzone.nordicsemi.com/b/blog/posts/nrf51-current-consumption-for-common-scenarios>

<https://devzone.nordicsemi.com/f/nordic-q-a/1657/how-to-minimize-current-consumption-for-ble-application-on-nrf51822>