

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky  
a komunikačních technologií

DIPLOMOVÁ PRÁCE



# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

## FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

## ÚSTAV TELEKOMUNIKACÍ

DEPARTMENT OF TELECOMMUNICATIONS

## SPORTTESTER S ROZHRANÍM BLUETOOTH LE

SPORTTESTER WITH BLUETOOTH LE

### DIPLOMOVÁ PRÁCE

MASTER'S THESIS

### AUTOR PRÁCE

AUTHOR

**Bc. Vojtěch Blažek**

### VEDOUCÍ PRÁCE

SUPERVISOR

**Ing. Ondřej Krajsa, Ph.D.**

**BRNO 2020**

# Diplomová práce

magisterský navazující studijní obor **Telekomunikační a informační technika**

Ústav telekomunikací

**Student:** Bc. Vojtěch Blažek

**ID:** 186490

**Ročník:** 2

**Akademický rok:** 2019/20

**NÁZEV TÉMATU:**

## Sporttester s rozhraním Bluetooth LE

**POKYNY PRO VYPRACOVÁNÍ:**

Navrhněte a realizujte prototyp sporttesteru komunikujícího pomocí technologie Bluetooth LE. Sporttester bude využívat akcelerometru k měření počtu kroků a síly nárazu. Tyto údaje se budou přenášet do aplikace v mobilním telefonu. Při překročení prahové hodnoty síly nárazu se spustí vhodný alarm.

**DOPORUČENÁ LITERATURA:**

[1] BLE-Stack User's Guide for Bluetooth 4.2 [online]. Texas Instruments, 2016 [cit. 2019-09-18]. Dostupné z: <https://bit.ly/2kSEXUD>

[2] MACII, Enrico. Ultra low-power electronics and design. Dordrecht: Kluwer Academic Publishers, 2004, 273 s. ISBN 1-4020-8075-1

**Termín zadání:** 3.2.2020

**Termín odevzdání:** 1.6.2020

**Vedoucí práce:** Ing. Ondřej Krajsa, Ph.D.

**prof. Ing. Jiří Mišurec, CSc.**  
předseda oborové rady

**UPOZORNĚNÍ:**

Autor diplomové práce nesmí při vytváření diplomové práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

## **ABSTRAKT**

Tato diplomová práce se zabývá návrhem sporttesteru, což je zařízení schopné počítat kadenci uběhnutých kroků a kontrolovat síly nárazů nohy na zem při běhání. Působení dlouhodobých silných nárazů při běhání může způsobit i vážné únavové zlomeniny kyčle, lýtkové kosti či v oblasti chodidla. Sporttester má za cíl kontrolovat a signalizovat běžci silné nárazy, čímž napomáhá předcházet vzniku únavových zlomenin. Zmíněné funkce jsou vyhodnocovány na základě změny akcelerace měřené akcelerometrem. Vypočítané hodnoty kadence kroků, rychlosti běhu a síly nárazu jsou posílány skrze Bluetooth Low Energy do kompatibilní mobilní aplikace.

## **KLÍČOVÁ SLOVA**

sporttester, počítání kroků, síla nárazu, Bluetooth, běhání, zdraví, únavové zlomeniny

## **ABSTRACT**

This thesis deals with the design of a sporttester, which is a device capable of calculating the cadence of the taken steps and monitoring the impact force of foot on the ground while running. Exposure to long-term strong impacts while running can cause serious fatigue fractures of the hip, calf bone or the foot area. The purpose of the sporttester is to monitor and signal strong impacts to the runner, thus helping to prevent fatigue fractures. The aforementioned features are evaluated based on the change in acceleration measured by the accelerometer. The calculated values of step cadence, running speed and impact force are sent via Bluetooth Low Energy to a compatible mobile application.

## **KEYWORDS**

sporttester, counting steps, impact force, Bluetooth, running, health, fatigue fractures

BLAŽEK, Vojtěch. *Sporttester s rozhraním Bluetooth LE*. Brno, 2020, 63 s. Diplomová práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací. Vedoucí práce: Ing. Ondřej Krajsa, Ph.D.

## PROHLÁŠENÍ

Prohlašuji, že svou diplomovou práci na téma „Sporttester s rozhraním Bluetooth LE“ jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené diplomové práce dále prohlašuji, že v souvislosti s vytvořením této diplomové práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno .....

.....

podpis autora

## PODĚKOVÁNÍ

Rád bych poděkoval vedoucímu diplomové práce panu Ing. Ondřeji Krajsovi, Ph.D. za výborné odborné vedení, konzultace, trpělivost a podnětné návrhy k práci.

# Obsah

Úvod	11
<b>1 Sporttester</b>	<b>12</b>
<b>2 Bluetooth</b>	<b>13</b>
2.1 Bluetooth Classic	13
2.2 Bluetooth Low Energy	13
2.2.1 Základní technické parametry BLE	14
2.2.2 Omezení BLE	15
2.2.3 Výhody BLE	16
2.2.4 Architektura BLE	16
<b>3 Implementace Bluetooth Low Energy</b>	<b>26</b>
3.1 Použitý hardware a software	26
3.2 Postup implementace	27
3.2.1 Instalace	27
3.2.2 Import projektu	28
3.2.3 Nastavení BLE parametrů	29
3.2.4 Přidání profilu	30
<b>4 Návrh sporttesteru</b>	<b>34</b>
4.1 Hardwarové řešení	34
4.1.1 Akcelerometr BMA423	36
4.1.2 Návrh desky plošných spojů	36
4.2 Softwarové řešení	37
4.2.1 Výběr profilu a služeb	38
4.2.2 Nastavení parametrů	38
4.2.3 Algoritmus počítání kroků	40
4.2.4 Popis periodické funkce	41
4.3 Budoucí vývoj	45
4.3.1 Návrh měření tepové frekvence srdce	45
<b>5 Měření</b>	<b>47</b>
5.1 Měření průběhu akcelerace při běhu	47
5.2 Měření přesnosti počtu kroků	49
<b>6 Závěr</b>	<b>52</b>
<b>Literatura</b>	<b>53</b>

<b>Seznam symbolů, veličin a zkratk</b>	<b>58</b>
<b>A Schéma zapojení shieldu</b>	<b>60</b>
<b>B Návrh DPS shieldu</b>	<b>61</b>
<b>C Schéma zapojení sporttesteru</b>	<b>62</b>
<b>D Návrh DPS sporttesteru</b>	<b>63</b>



# Seznam obrázků

2.1	Kompatibilita typů Bluetooth . . . . .	14
2.2	Bluetooth Low Energy architektura . . . . .	17
2.3	Frekvenční spektrum Bluetooth Low Energy . . . . .	18
2.4	Stavy linkové vrstvy . . . . .	19
2.5	Bluetooth Low Energy 4.2 paket s L2CAP . . . . .	20
2.6	Hierarchie uvnitř Generic Attribute Profile . . . . .	23
2.7	Interval připojení a otrokovo zpoždění . . . . .	25
3.1	Stažení SDK v Code Composer Studio . . . . .	28
3.2	Import projektu z SDK . . . . .	29
3.3	Vztah mezi RSCP profilem a obsaženými službami. . . . .	30
4.1	Blokové schéma zapojení . . . . .	35
4.2	Vývojová deska LAUNCHXL-CC2640R2 a vyrobený shield . . . . .	37
4.3	Vývojový diagram programu . . . . .	39
4.4	Výsledky cvičného běhu z aplikace Ala Fitness . . . . .	41
4.5	Doporučené zapojení s diagramem pro AFE4420 a BH1790GLC-E2 . . . . .	46
5.1	Graf průběhů akcelerace . . . . .	49
5.2	Graf odchylek počtu kroků . . . . .	51

# Seznam tabulek

1.1	Porovnání vybraných sporttesterů . . . . .	12
5.1	Tabulka hodnot pro měření průběhu akcelerace kroků . . . . .	48
5.2	Tabulka hodnot měření počtu kroků pro chůzi . . . . .	50
5.3	Tabulka hodnot měření počtu kroků pro běh . . . . .	51

# Seznam výpisů

3.1	Definice datového typu attAttribut_t. . . . .	31
3.2	Tabulka atributů služby RSCS. . . . .	32
3.3	Úprava programu v simple_peripheral.c. . . . .	33
4.1	Periodická funkce sporttesteru – definice konstant. . . . .	42
4.2	Periodická funkce sporttesteru – část první. . . . .	43
4.3	Periodická funkce sporttesteru – část druhá. . . . .	44
4.4	Periodická funkce sporttesteru – část třetí. . . . .	45

# Úvod

Sporttestery mohou mít různou podobu a vykonávat různé funkce, ale všechny mají společný cíl. Tím je pomoci lidem vyhodnotit zatížení organismu při tělesné aktivitě. To může odhalit nedostatečný trénink nebo naopak problémy s přetěžováním organismu. Právě přetěžování může vést k vážným zdravotním problémům. Jedním ze způsobů, jak je sporttester schopen předejít možným problémům, je měření a vyhodnocování síly nárazů při dopadu nohy na zem. Příliš silné nárazy v dlouhodobém měřítku mohou způsobit únavové zlomeniny kostí v oblasti chodidla, kosti lýtkové nebo kyčle.

Tato práce se zabývá návrhem sporttesteru. Jako sporttester je v této práci chápáno zařízení, které je schopné měřit kadenci kroků a sílu nárazů při běhu na základě dat z akcelerometru. Změřené hodnoty následně sporttester posílá skrze technologii Bluetooth do mobilní aplikace. V návrhu sporttesteru je řešena jak hardwarová, tak i softwarová část. Hardwarová část obnáší výběr vhodných součástek pro logickou i napájecí část, návrh zapojení vybraných součástek a výrobu desky plošných spojů. Softwarová část řeší předání a zpracování naměřených hodnot z akcelerometru a také komunikaci sporttesteru s mobilní aplikací pomocí Bluetooth. To obnáší vhodné nastavení parametrů přenosu a výběr použitelného Bluetooth profilu a služeb, které v sobě přenášejí samotné hodnoty.

Tato práce je rozdělena do šesti kapitol. První kapitola objasňuje, co se pod pojmem sporttester běžně uvažuje a co pod tímto pojmem bude uvažováno v této práci. Jsou zde také zmíněny funkce navrhovaného sporttesteru. Tato kapitola je doplněna o tabulku, která porovnává různé parametry vybraných sporttesterů, které jsou, nebo se dříve objevili, na trhu. Kapitola druhá je zaměřena na technologii Bluetooth. Obsahuje vysvětlení rozdílů mezi Bluetooth Classic a Bluetooth Low Energy. Bluetooth Low Energy je poté detailněji rozebráno. Pochopení, jak pracuje Bluetooth Low Energy, je klíčové k správnému nastavení přenosových parametrů, které výrazně ovlivňují spolehlivost přenosu a celkovou spotřebu elektrické energie zařízení. Třetí kapitola obsahuje stručný návod, jak implementovat Bluetooth Low Energy do mikrokontroléru CC2640R2F. V třetí kapitole je také popsáno kde se nastavují přenosové parametry a jakým způsobem lze přidat vybraný Bluetooth profil. Čtvrtá kapitola obsahuje samotný návrh sporttesteru. Ten je rozdělen na softwarovou část, hardwarovou část a budoucí vývoj návrhu. Pátá kapitola obsahuje měření průběhu akcelerace při běhu a měření přesnosti počtu kroků. Poslední kapitolou je závěr práce, ve kterém jsou shrnuty dosažené výsledky.

# 1 Sporttester

Obecně se pod pojmem sporttester označuje zařízení, které je schopné měřit a analyzovat data o sportovní aktivitě. Nejčastěji se sporttestery zaměřují na běhání. Mezi základní funkce sporttesterů patří měření času, tepové frekvence, počtu kroků, výpočet spálených kalorií, sledování polohy pomocí GPS. Většina sporttesterů je také voděodolná. V této práci se pod pojmem sporttester bude uvažovat zařízení, které dokáže vypočítat počet kroků, které nositel sporttesteru uběhl nebo ušel, a také dokáže kontrolovat sílu nárazů při běhání. Změřená data nebudou sporttesterem přímo zobrazována, ale budou pomocí technologie Bluetooth posílána do mobilní aplikace. K měření zmíněných funkcí slouží akcelerometr, který je v sporttesteru obsažen. Akcelerometr je součástka, která měří akceleraci. Ze změny akcelerace je vyhodnocována síla nárazů i uběhnutý počet kroků za daný časový interval.

Funkce měření síly nárazů je v navrhovaném sporttesteru zahrnuta proto, protože nadměrné dopady při pravidelném běhání mohou mít za následek zranění. Opakované silné nárazy způsobují přetěžování kostí, což může vést až k únavovým zlomeninám. Únavové zlomeniny u běžců dlouhých tratí se nejčastěji vyskytují v kyčli, holení kosti a v oblasti chodidla. Jiné typy zranění, v důsledku silných nárazů při běhání, prokázány nebyli [1, 2]. Přehled několika dostupných sporttesterů naleznete v tabulce 1.1. Tato tabulka ukazuje cenu sporttesterů, jeho umístění na těle a podporované funkce.

Tab. 1.1: Porovnání vybraných sporttesterů (převzato z [3]).

Název	Cena	Umístění	Kadence kroků	Čas kontaktu se zemí	Síla nárazu	Srdeční tep
MilestonePod	25 \$	Bota	Ano	Ano	Ne	Ne
Running Dynamics v2	100 \$	Hrudník	Ano	Ano	Ne	Ano
TgForce	145 \$	Kotník	Ano	Ne	Ano	Ne
RunScribe	200 \$	Bota	Ano	Ano (nespolehlivě)	Ano	Ne
SHFT	300 \$	Bota, Hrudník	Ano	Ano (nespolehlivě)	Ne	Ano
Moxy	770 \$	Lýtka	Ano*	Ano*	Ne	Ano*

\* podporuje pouze jednu z označených možností podle nastavení

## 2 Bluetooth

Tato kapitola popisuje technologii Bluetooth. Je v ní objasněn rozdíl mezi Bluetooth Classic (BR/EDR) a Bluetooth Low Energy (BLE). Následovně je do větších detailů popsán BLE a jeho vrstevná architektura.

### 2.1 Bluetooth Classic

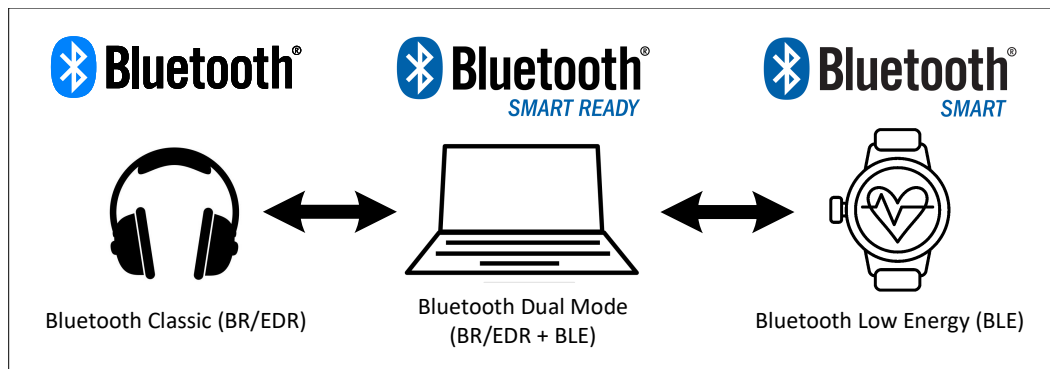
Bluetooth je standard pro bezdrátovou komunikaci používaný k výměně dat na krátké vzdálenosti mezi stacionárními i mobilními zařízeními. Původně byl koncipován jako bezdrátová alternativa pro standard RS-232, který slouží k sériové komunikaci po kabelovém vedení. První verze Bluetooth byla zveřejněna v roce 1994 společností Ericsson a byla pojmenována po dánském králi Haraldovi Modrozubovi [4].

Existují dva typy Bluetooth. Jeden je označován jako Bluetooth Classic (BR/EDR), který je používán například v bezdrátových reproduktorech, sluchátkách nebo informačních systémech pro automobily. Druhým typem je Bluetooth Low Energy (BLE). BLE byl poprvé představen v Bluetooth verzi 4.0 v roce 2010. Jeho hlavní oblast použití je v aplikacích, kde je spotřeba elektrické energie rozhodující a kde se přenáší většinou jen malé množství dat. Používá se převážně v zařízeních napájených z baterie, jako jsou například různé senzory, hlásiče, chytré hodinky či náramky [5].

Tyto dva typy Bluetooth standardu jsou vzájemně nekompatibilní, i když sdílejí stejnou značku, a dokonce i dokument, ve kterém jsou upřesněny. Zařízení pracující s Bluetooth Classic nemohou přímo komunikovat se zařízeními pracující s BLE. Z tohoto důvodu některá zařízení, jako jsou například chytré mobilní telefony, mají do sebe implementovány oba typy (nazývané také jako zařízení s duálním režimem) [6]. Toto jim umožňuje komunikovat s oběma typy Bluetooth. To je znázorněno na obrázku 2.1.

### 2.2 Bluetooth Low Energy

Bluetooth Low Energy je někdy označováno jako Bluetooth Smart. Prvotní návrh technologie BLE vytvořila firma Nokia pod názvem Wibree a v roce 2010 byla tato technologie odkoupena firmou Bluetooth. O následný vývoj se až do teď stará organizace pro normalizaci zvaná Bluetooth Special Interest Group (Bluetooth SIG) [7]. Na první pohled by se mohlo zdát, že BLE je obdoba klasického Bluetooth s nižší přenosovou rychlostí a nižší energetickou náročností. To však není pravda. Mezi BLE



Obr. 2.1: Kompatibilita typů Bluetooth.

a Bluetooth Classic jsou velké rozdíly z hlediska technických parametrů přenosu, implementace a aplikací, pro které jsou vyvinuty. BLE má se svými vlastnostmi velký potenciál v oblasti internetu věcí (anglicky Internet of Things, zkratka IoT), zatímco klasické Bluetooth se pro použití v IoT příliš nehodí [6].

### 2.2.1 Základní technické parametry BLE

Tato kapitola obsahuje pouze základní technické parametry týkající se Bluetooth Low Energy [4]. Těmito parametry jsou:

- BLE pracuje ve stejném frekvenčním rozsahu jako Bluetooth Classic, který se nachází v rozpětí od 2,40 do 2,48 GHz.
- Frekvenční spektrum je rozděleno do 40 kanálů, kde každý kanál má šířku 2 MHz.
- Maximální přenosová rychlost na fyzické vrstvě je 1 Mb/s pro BLE verze 4. Pro BLE 5 je maximální přenosová rychlost až 2 Mb/s.
- Maximální vzdálenost komunikujících zařízení se pohybuje v rozsahu 10 až 30 m v závislosti na okolním prostředí a nastaveném vysílacím výkonu (maximálně 10 mW pro BLE 4 a 100 mW pro BLE 5).
- Spotřeba elektrické energie se markantně liší v závislosti na implementaci samotné aplikace, která využívá BLE k přenosu dat. Spotřeba je také výrazně ovlivňována nastavením parametrů BLE přenosu a použitým hardwarem (HW). Maximální špičková spotřeba proudu samotného BLE chipsetu se běžně pohybuje kolem 17,5 mA [8].
- Zabezpečení komunikace v BLE je volitelné a záleží jen na samotném zařízení a vývojářích, zda implementují zabezpečení do své aplikace. Pokud se pro to rozhodnou, tak mají na výběr z několika úrovní zabezpečení (více na toto téma v kapitole 2.2.4 Architektura BLE na straně 16)

- Pro všechny operace šifrování v komunikaci je použita 128-bitová AES (Advanced Encryption Standard) šifra v režimu CCM (Counter with CBC-MAC).
- BLE je navrženo pro přenosy jen malého množství dat. Při implementování BLE v aplikaci, která vyžaduje přenosy velkého množství dat, je pak výrazně ohrožena nízká spotřeba energie. Optimální spotřeby elektrické energie je dosaženo pouze pokud je doba využití rádia co nejmenší.
- Všechny verze BLE jsou zpětně kompatibilní. Pokud však komunikují zařízení s různými verzemi, tak je možné využívat pouze funkce, které podporuje starší verze. Například při komunikaci zařízení podporující Bluetooth 4.2 BLE se zařízením podporující Bluetooth 5 BLE, tak není možné využívat funkce, které přinesla verze 5.

### 2.2.2 Omezení BLE

Tato kapitola popisuje omezení technologie BLE, která se vztahují k datové propustnosti a rádiovému dosahu. Tyto parametry byly vybrány především proto, že to jsou dva hlavní parametry, které jsou uvažovány při výběru bezdrátové komunikační technologie pro vybranou aplikaci.

#### Datová propustnost

Datová propustnost je u BLE limitována rychlostí fyzického rádiového přenosu dat, což je rychlost, kterou rádio přenáší data. Tato rychlost závisí na použité verzi BLE. Pro verze Bluetooth 4.2 a starší je rychlost přenosu dat fixně nastavena na 1 Mb/s. Pro verze Bluetooth 5.0 a novější je rychlost přenosu závislá na použitém přenosovém režimu. Režim dlouhého dosahu umožňuje použití datové rychlosti 500 kb/s nebo 125 kb/s. Vysokorychlostní režim umožňuje využití datové rychlosti 2 Mb/s [9].

Přenosová rychlost na aplikační vrstvě je mnohem nižší oproti přenosové rychlosti na vrstvě fyzické. Tento rozdíl mezi rychlostmi je způsoben několika okolnostmi. Jedna z těchto okolností je fakt, že Bluetooth specifikace udává, že mezi vyslanými pakety musí být dodržena mezera dlouhá 150  $\mu$ s. Během této mezery neprobíhá přenos užitečných dat, a v důsledku toho se pak snižuje přenosová rychlost na aplikační vrstvě. Další okolností je velikost hlavičky paketu. Obecně platí, že čím větší je záhlaví, tím menší je rychlost na aplikační vrstvě vzhledem k rychlosti na fyzické vrstvě a Bluetooth není výjimkou. Někdy také nastává případ, kdy centrální zařízení požaduje od periferního zařízení vyslání datových paketů, i když periferie nemá žádná užitečná data k odeslání. Další možný faktor ovlivňující přenosovou rychlost je ztráta paketu. V případě ztráty paketu nebo jeho poškození při přenosu se paket opětovně posílá odesílatelem [4].



## Dosah

BLE bylo vytvořeno pro komunikaci na krátké vzdálenosti. Jeho radiový dosah je omezen několika faktory. Ten nejdůležitější je frekvence, na které probíhá rádiová komunikace. Bluetooth pracuje v pásmu 2,4 GHz a vysílání na této frekvenci je velice ovlivňováno kovovými předměty, zdi a vodou (tudíž i lidmi). S vzrůstající četností a velikostí takovýchto překážek v okolí klesá dosah signálu vysílaném na této frekvenci. Toto pásmo je také jedno z ISM (z anglického Industrial, Scientific and Medical) pásem, která jsou vyhrazena pro rádiové vysílání v oborech průmyslových, vědeckých a zdravotnických. V tomto pásmu pracují také některé standardy IEEE 802.11 (WiFi) [10]. Dalším důležitým faktorem je anténa, tedy její zisk, orientace vůči okolí, umístění vzhledem ke zbytku zařízení. Materiál, ze kterého je vyrobeno pouzdro celého zařízení, může výrazně ovlivnit dosah zařízení. Především pokud je použita vnitřní anténa [4].

### 2.2.3 Výhody BLE

Mezi hlavní výhody Bluetooth Low Energy patří především jeho nízká spotřeba energie, což vyplývá již z jeho názvu. Spotřeba energie je velmi závislá na nastavení přenosových parametrů a také na použitém hardwaru.

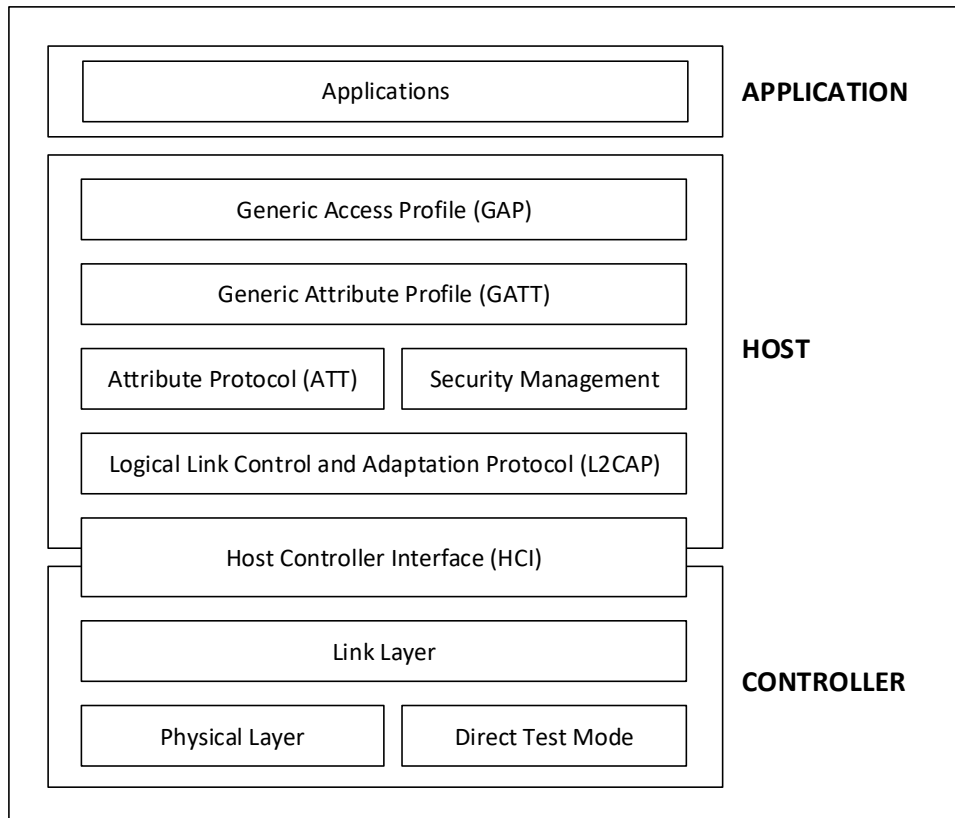
Další důležitá výhoda je jeho nízká pořizovací cena. U většiny bezdrátových protokolů je běžné, že přístup k specifikačním dokumentům je zpoplatněn (někdy až tisíce dolarů ročně). Přístup k specifikacím je v případě BLE zadarmo. Dále cena samotných BLE modulů a čipů je nižší v porovnání s podobnými technologiemi. Další výhodou je fakt, že BLE je zakomponováno ve většině moderních mobilních telefonů.

### 2.2.4 Architektura BLE

Tato kapitola se zabývá jednotlivými vrstvami architektury BLE, neboli tzv. BLE stackem. Do detailu budou popsány jen některé vrstvy, které jsou důležité pro pochopení funkčnosti BLE. Zbylé vrstvy budou popsány stručněji. Obrázek 2.2 naznačuje jednotlivé vrstvy a jejich vzájemné uspořádání a působení mezi sebou. Ukazuje také rozdělení vrstev do tří skupin [11, 12]:

- Aplikace (Application)
- Hostitel (Host):
  - Generic Access Profile (GAP)
  - Generic Attribute Profile (GATT)
  - Attribute Protocol (ATT)

- Řízení bezpečnosti (Security management)
- Logical Link Control and Adaptation Protocol (L2CAP)
- Host Controller Interface (HCI)
- Ovladač (Controller)
  - Host Controller Interface (HCI)
  - Linková vrstva (Link layer)
  - Fyzická vrstva (Physical layer)
  - Režim přímého testování (Direct test mode)

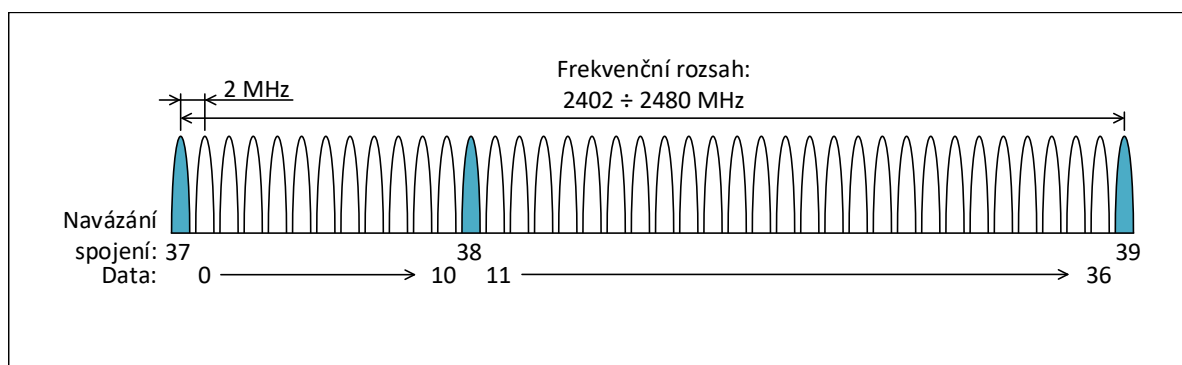


Obr. 2.2: Bluetooth Low Energy architektura.

Aplikační část není v BLE architektuře nijak specifikována a je pouze na vývoji jakou podobu bude tato vrstva mít. Tato vrstva přímo závisí na účelu použití celého zařízení a je ve vrstevné architektuře umístěna přímo nad vrstvami Generic Access Profile (GAP) a Generic Attribute Profile (GATT). V této vrstvě by měla být definována logika zpracování dat přijatých od jiných zařízení a také logika generování dat k odeslání [11].

## Fyzická vrstva

Fyzická vrstva se týká rádiového hardwaru používaného pro komunikaci a pro modulaci či demodulaci dat. BLE pracuje v bezlicenčním ISM pásmu ve frekvenčním rozsahu od 2402 MHz do 2480 MHz. Tento rozsah je rozdělen do 40 rádiových kanálů, kde 3 kanály jsou používány pro skenování (ang. scanning) a propagaci (ang. advertising) a zbylých 37 pro posílání dat. Střední frekvence sousedních kanálů jsou od sebe vždy vzdáleny 2 MHz [12]. Frekvenční spektrum a rádiové kanály můžete vidět na obrázku 2.3.



Obr. 2.3: Frekvenční spektrum Bluetooth Low Energy.

BLE používá Frequency Hopping Spread Spectrum (FHSS), což umožňuje oběma komunikujícím zařízením náhodně měnit nosnou frekvenci, na které jsou následně po určité době posílána data. Tato metoda výrazně zlepšuje spolehlivost přenosu a dovoluje zařízením vyhnout se frekvenčním kanálům, které jsou používány pro jinou komunikaci nebo jsou zarušeny [12].

Vysílací výkon rádia lze nastavit v rozsahu [11]:

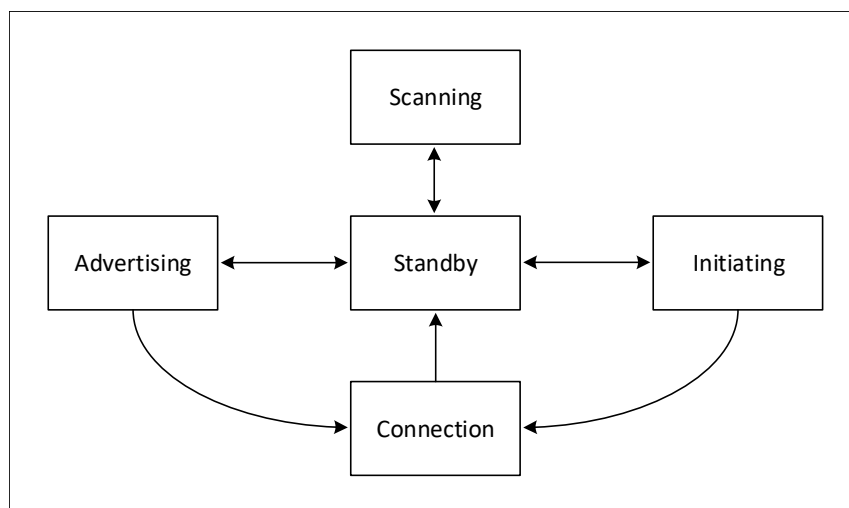
- Minimum je 0,01 mW (-20 dBm) pro všechny verze BLE,
- Maximum pro verzi BLE 5 a novější je 100 mW (20 dBm),
- Maximum pro verzi BLE 4.2 a starší je 10 mW (10 dBm).

## Linková vrstva

Linková vrstva je přímo propojena s fyzickou vrstvou a poskytuje vyšším vrstvám jistou míru abstrakce. Skrze HCI vrstvu také poskytuje možnost interakce s rádiem. Linková vrstva je zodpovědná především za výpočet cyklického redundantního součtu (CRC), generování náhodných čísel, šifrování a také za správu stavu, ve kterém se zařízení nachází. Zařízení se může nacházet vždy v jednom z těchto stavů:

- Standby – Výchozí stav, ve kterém rádio nic nevysílá ani nepřijímá.
- Advertising – Stav, ve kterém zařízení vysílá propagační pakety. Ty umožňují ostatním zařízením zjistit přítomnost zařízení v tomto stavu.
- Scanning – Stav, ve kterém zařízení vyhledává ostatní zařízení, která jsou ve stavu advertising.
- Initiating – Stav, který udává, že se zařízení ve stavu scanning snaží ustanovit spojení se zařízením ve stavu advertising, ale k samotnému spojení ještě nedošlo.
- Connection – Stav, ve kterém se nacházejí zařízení, která jsou spojena s dalšími zařízeními a dochází k pravidelné výměně dat.

Zařízení, které bylo ve stavu initiating a již ustanovilo spojení s jiným zařízením, přechází do stavu connection a stává se tzv. mistrem (master). Jiné zařízení, které bylo ve stavu advertising a spojilo se s jiným zařízením, přechází také do stavu connection a stává se tzv. otrokem (slave) [11]. Všechny možné stavy a jejich možné změny zobrazuje obrázek 2.4.



Obr. 2.4: Stavy linkové vrstvy.

### Host Controller Interface

Host Controller Interface (HCI) je protokol definovaný v Bluetooth specifikačním dokumentu. HCI umožňuje komunikaci mezi hostitelskou částí a částí ovladače. Tyto dvě části totiž mohou být fyzicky v jiných čipech a HCI zajišťuje jejich kompatibilitu. Vývojář zařízení poté může zvolit různé výrobce pro ovládací část i část

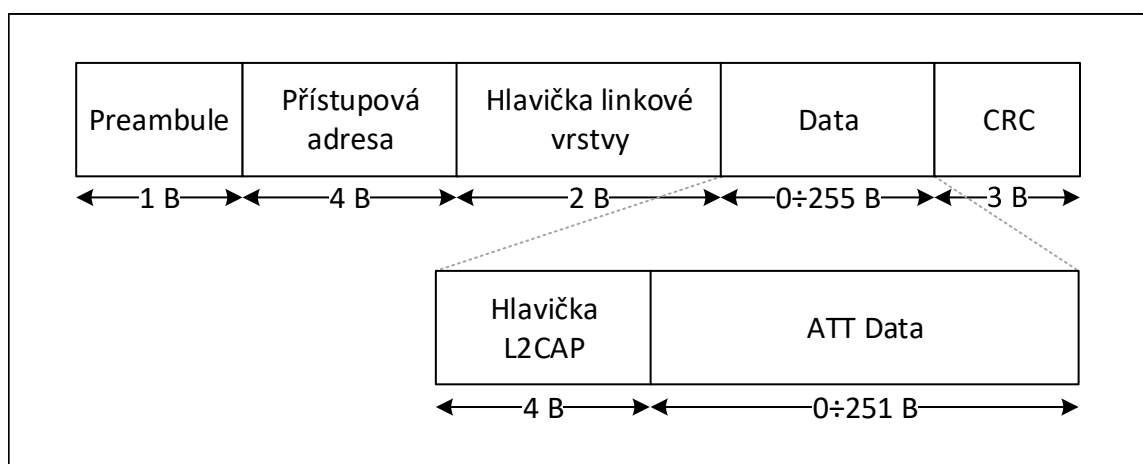
hostitele a může si být jistý, že díky HCI bude možná jejich vzájemná kooperace .

Pokud jsou zmiňované části ve fyzicky oddělených čipech, HCI je poté implementováno skrze fyzické rozhraní. Oficiálně jsou podporovány rozhraní UART (Universal Asynchronous Receiver-Transmitter), USB (Universal Serial Bus) a SDIO (Secure Digital Input Output). Pokud je ovládací a hostitelská část ve stejném chipu, tak se HCI vrstva realizuje pomocí logického rozhraní. Hlavním úkolem HCI vrstvy je přenášet příkazy z hostitele (ATT příkazy) k ovládací části a také přenášet události opačným směrem [4].

### Logical Link Control and Adaptation Protocol

Logical Link Control and Adaptation Protocol (L2CAP) tvoří jakýsi protokolový multiplex. Je to stejný protokol, jaký se používá u Bluetooth Classic. Má za úkol přebírat data z několika protokolů nad ním a sdružovat je do standardních Bluetooth paketů, které poté předává nižším vrstvám. V případě BLE přebírá data především od ATT vrstvy a vrstvy řízení bezpečnosti [11].

Stará se také o fragmentování a opětovné skládání paketů. Velké pakety od vyšších vrstev rozdělí na části, které se vejdu do prostoru vyhrazeného pro užitečná data v Bluetooth paketech. Maximální velikost užitečných dat závisí na verzi BLE. U verze 4.0 a 4.1 je maximální velikost užitečných dat rovna 27 B a pro verzi 4.2 je tato hodnota 251 B. Na straně příjemce tato vrstva přebírá několik Bluetooth paketů a rekonstruuje z nich pakety vyšší vrstvy. Zapouzdření dat ATT vrstvy do BLE paketu ukazuje obrázek 2.5.



Obr. 2.5: Bluetooth Low Energy 4.2 paket s L2CAP.

## Security Manager

Zabezpečení v BLE zajišťuje Security Manager vrstva. Tato vrstva využívá řadu protokolů a algoritmů potřebných k zajištění bezpečnosti. Hlavními funkcemi této vrstvy však jsou [13]:

- používání tzv. whitelistů, které umožňují filtrovat zařízení při skenování, propagaci i navazování spojení,
- vytvoření a uložení sdílených tajných klíčů mezi zařízeními (Diffieho-Hellmanův protokol s využitím eliptických křivek, zkratka ECDH),
- zajištění autentizace,
- šifrování dat vyměňovaných mezi zařízeními (128-bitová AES šifra),
- zajištění integrity dat pomocí digitálního podpisu dat.

## Attribute Protocol

Attribute Protocol definuje, jak jsou data předávána od serveru ke klientovi a také definuje uspořádání těchto dat. Server je zařízení, které odhaluje data v něm obsažena nebo ním řízena. Dále také přijímá příchozí příkazy a odesílá zpět odpovědi, oznámení a indikace. Klient je zařízení, které čte serverem odhalená data, řídí chování serveru, odesílá požadavky a přijímá oznámení a indikace.

Představme si příklad, kde máme zařízení, které je schopné měřit teplotu okolního prostředí, stav své baterie a čas, kdy byla teplota změřena. Takové zařízení by v chápání ATT protokolu byl server. Klient by byl třeba mobilní telefon, který se k teploměru připojuje a žádá po něm teplotu a další atributy.

Data, která server odhaluje klientům se nazývají tzv. atributy. Ty mají přesně danou strukturu, která se skládá z popisovače (handle), typu, hodnoty a oprávnění atributu. Popisovač atributu je 16 bitů dlouhá adresa, kterou atributům přiřazuje server. Pomocí této adresy se klient odkazuje na daný atribut. Server ručí za unikátnost jednotlivých popisovačů v rámci daného spojení. V případě jiného spojení již předešlé popisovače neidentifikují stejné atributy. Adresy jsou vybírány z rozsahu 0x0001–0xFFFF, kde adresa 0x0000 je rezervována [11].

Typ atributu, někdy nazývaný jako univerzálně unikátní identifikátor (UUID), je 16 nebo 128 bitů dlouhé číslo. Atributy adoptované Bluetooth SIG skupinou mají délku 16 bitů, přičemž zbytek do 128 bitů mají vždy stejný. Příkladem může být UUID pro měření teploty, kde podtržená část je SIG adoptovaný atribut:

00002A1C-0000-1000-8000-00805F9B34FB

Pokud vývojář nechce použít UUID adoptované SIG skupinou, může si vytvořit vlastní. Vlastní UUID může být jakékoliv 128 bitů dlouhé číslo vyjma základového UUID (příklad zmíněný výše), které používá SIG pro jimi adoptované atributy. Hlavní výhodou používání adoptovaných atributů je snížení délky paketů, jelikož se pak přenáší pouze 16 bitů namísto 128 bitů [11].

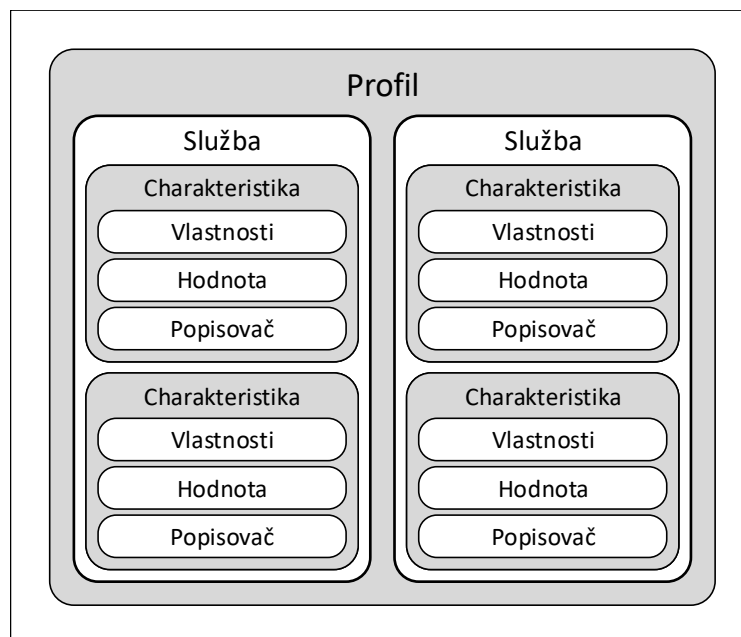
Oprávnění atributu udává, zda do daného atributu lze zapisovat, číst ho, zapnout na něm oznámení či indikaci. Všechna tato povolení nejsou zjistitelná přímo pomocí ATT, ale spíše skrze vyšší vrstvy (GATT nebo až aplikační vrstvu).

Existuje 6 možných operací, které lze v případě atributů provést:

- Příkazy
  - Jsou posílány klientem k serveru a nevyžadují odpověď.
- Požadavky
  - Jsou posílány klientem k serveru a vyžadují odpověď.
- Odpověď
  - Je posílána serverem ke klientovi po vyřízení požadavku.
- Oznámení
  - Je posíláno serverem ke klientovi za účelem upozornění klienta o změně konkrétní charakteristické hodnoty a není od klienta vyžadováno potvrzení.
- Indikace
  - Je posílána serverem ke klientovi za účelem upozornění klienta o změně konkrétní charakteristické hodnoty a od klienta je vyžadováno potvrzení.
- Potvrzení
  - Je posíláno klientem k serveru po úspěšném přijetí indikace.

## **Generic Attribute Profile**

Generic Attribute Profile (GATT) definuje jakou podobu má komunikace po navázání spojení. To obnáší především jak jsou data formátována, zabalena a předávána k ostatním zařízením. Pro přenos dat mezi dvěma BLE zařízeními je využit ATT protokol a také tzv. charakteristiky, služby a profily. Charakteristika je obdoba atributu z ATT protokolu. V případě GATT vrstvy se pojmy klient a server chápou stejně jako tomu bylo u ATT s tím rozdílem, že jedno zařízení může svojí roli měnit v závislosti na směru zrovna probíhající komunikace. Jednou může zařízení zastávat roli klienta, který si žádá informace od serveru a podruhé tomu může být naopak. Vzájemné uspořádání charakteristik a služeb v profilu ukazuje obrázek 2.6.



Obr. 2.6: Hierarchie uvnitř Generic Attribute Profile.

Profil sdružuje většinou několik služeb dohromady. Může být předdefinovaný od Bluetooth SIG nebo si vývojář může definovat vlastní profil. Služba dohromady sdružuje více charakteristik a jednotlivé služby jsou popisovány pomocí UUID, stejně jako tomu bylo v případě charakteristik nebo atributů. Charakteristiky zapouzdřují data, kterými mohou být konkrétní naměřené hodnoty získané ze senzorů [14].

Jako názorný příklad profilu může být Heart Rate Sensor Profile, který slouží ke sběru informací o frekvenci srdečního tepu z patřičných senzorů a jeho podrobný popis můžete nalézt v [15]. Tento profil se skládá ze služeb Heart Rate Service (definice dostupná v [16]) a Device Information Service (definice dostupná v [17]).

### Generic Access Profile

Generic Access Profile (GAP) je základní vrstva, která umožňuje BLE zařízením vzájemně spolupracovat. Poskytuje výchozí strukturu, které se jakákoliv BLE implementace musí držet, aby se zařízení navzájem mohli nalézt, vysílat data, ustanovit bezpečné spojení a provádět mnoho dalších funkcí způsobem, který je pro ostatní zařízení srozumitelný. Tato vrstva také výrazně ovlivňuje výslednou spotřebu elektrické energie zařízení, protože skrze tuto vrstvu se nastavují důležité parametry přenosu [4].

Než bude vysvětlen význam těch nejdůležitějších parametrů, je důležité si nejdříve



ujasnit jakou roli dané zařízení zastupuje. Zařízení může najednou vykonávat funkce z více rolí a je důležité zmínit, že role v GAP nejsou nijak spojeny s rolemi na GATT či ATT vrstvě. Role na GAP vrstvě jsou:

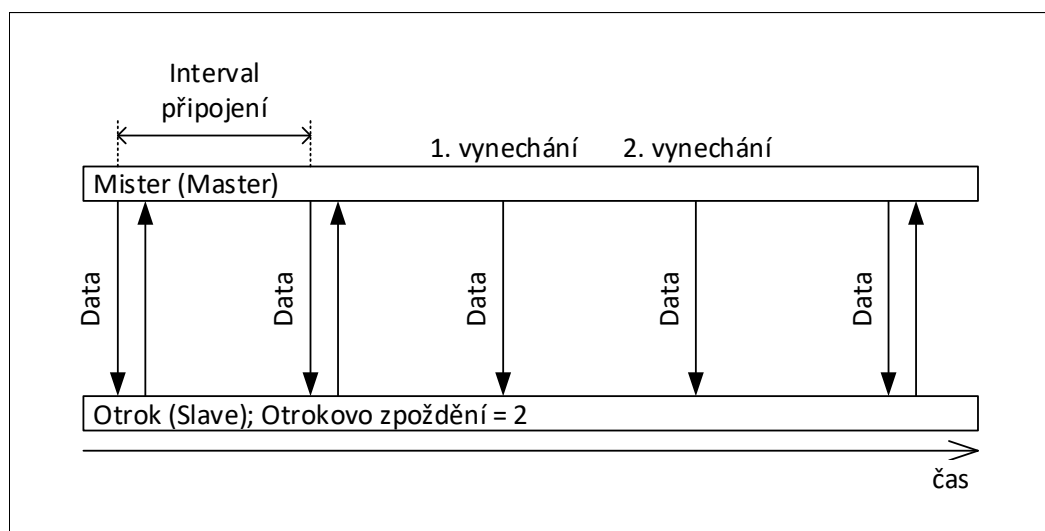
- Vysílač (Broadcaster)
  - Zařízení plní roli vysílače periodicky vysílá data, avšak nikdy žádná užitečná data nepřijímá. Tato data jsou vysílána již v propagačních paketech, a ne až při navázání spojení s příjemcem. Takové zařízení se nachází vždy ve stavu advertising (propagace) na linkové vrstvě.
- Pozorovatel (Observer)
  - Zařízení s touto rolí pouze přijímá data. Sbírá nejčastěji data od zařízení v roli vysílače. Takové zařízení se nachází vždy ve stavu scanning (skenování) na linkové vrstvě.
- Centrální (Central)
  - Tato role odpovídá roli mistra (master) na linkové vrstvě. Zařízení s centrální rolí je schopné navazovat několik spojení s periferními zařízeními a je vždy iniciátorem spojení. Tato role je většinou vykonávána zařízením s větším výpočetním výkonem (např. mobilní telefon, osobní počítač), protože navazování a udržování několika spojení současně vyžaduje silnější CPU a větší množství paměti.
- Periferní (Peripheral)
  - Tato role odpovídá roli otroka (slave) na linkové vrstvě. Zařízení vykonávající roli periferie vysílá propagační pakety, čímž se odhaluje centrálním zařízením a umožňuje navázání spojení mezi nimi. Periferní role nevyžaduje na svůj běh výkonný HW čímž se periferní zařízení stávají cenově velice dostupnými.

Chod zařízení ve stavu propagace nejvíce ovlivňuje propagační interval (advertising interval). Ten udává, jak často jsou vysílány propagační pakety. Má velký vliv na životnost baterie zařízení, a tudíž by měl být nastavován s rozvahou. Je ho možné nastavit v rozsahu od 20 ms do 10,24 s s krokem 625  $\mu$ s. Čím kratší je tento interval, tím je větší šance na zaznamenání paketu skenujícím zařízením, a tedy rychlejší navázání spojení. S rostoucím počtem odeslaných paketů roste také spotřeba elektrické energie a zařízení se stává méně úsporné. Spotřebu také velice ovlivňuje obsah propagačních paketů. Implementované profily a jejich služby většinou definují doporučené položky, které by měli být v propagačních paketech zahrnuty. Výsledná podoba propagačních paketů je však pouze na vývojáři [14].

Chod zařízení, nacházejících se ve stavu connection, jsou nejvíce ovlivňována parametry:

- Interval připojení (Connection interval)
  - Je to čas mezi událostmi, kdy se mezi spojenými zařízeními začínají odesílat data. Je možné ho nastavit v rozsahu od 7,5 ms do 4 s s krokem 1,25 ms. Tato hodnota je nastavena mistrem. Otrok může pouze informovat mistra o preferované hodnotě tohoto parametru, ale mistr otrokovu preferenci nemusí brát v potaz.
- Otrokově zpoždění (Slave latency)
  - Je to celočíselná hodnota udávající kolikrát může otrok neodpovědět na výzvu mistra. Mistr s otrokem udržuje spojení periodickými zprávami, které odesílá podle parametru intervalu připojení. Otrok s cílem úspory energie může v případě, kdy nemá žádná užitečná data k odeslání, vynechat odpovědi na mistrovi výzvy. Rozsah hodnot je od nuly do hodnoty (*ČasovýLimitDohledu/IntervalPřipojení*) - 1.
- Časový limit dohledu (Supervision timeout)
  - Tento parametr je používán k detekci ztráty spojení. Je definován jako maximální časový úsek mezi dvěma přijatými pakety. Pokud tento čas uběhne od posledního přijatého paketu, tak je spojení považováno za ztracené. Má rozsah od 100 ms do 32 s s krokem 10 ms.
  - Pro správný výběr časového limitu dohledu s ohledem na ostatní parametry slouží vzorec:

$$\text{ČasovýLimitDohledu} > (1 + \text{OtrokovoZpoždění}) \cdot \text{IntervalPřipojení} \cdot 2. \quad (2.1)$$



Obr. 2.7: Interval připojení a otrokově zpoždění.

## 3 Implementace Bluetooth Low Energy

Tato kapitola popisuje postup, jak implementovat BLE softwarový stack pro mikrokontrolér (ang. microcontroller unit, zkratka MCU) CC2640R2F od firmy Texas Instruments (TI) ve vývojovém prostředí (ang. Integrated Development Environment, zkratka IDE) Code Composer Studio (CCS). Ještě před samotným postupem je krátce představen HW a SW použitý k demonstrování postupu.

### 3.1 Použitý hardware a software

Mikrokontrolér CC2640R2F je v našem případě osazen na vývojovém kitu LAUNCHXL-CC2640R2 Rev: 1.0. Dále je použito vývojové prostředí CCS verze 9.2.0. Samotný BLE stack je součástí sady SimpleLink CC2640R2 Software Development Kit (SDK). Mikrokontrolér spolu s SDK tvoří periferní část Bluetooth komunikace. Jako centrální část může být použita jakákoliv mobilní aplikace, která správně implementuje vybraný Bluetooth profil (viz. kapitola 4.2.1). Taková aplikace samozřejmě musí být nainstalována na mobilním zařízení podporující Bluetooth protokol verze 4.2 nebo novější. Pro mobilní zařízení s operačním systémem Android je možné použít aplikaci Ala Fitness, která je dostupná ze stránek Google Play.

#### SimpleLink CC2640R2 SDK

SimpleLink CC2640R2 SDK je softwarový balíček poskytující základní softwarové komponenty, jako jsou BLE stack, TI-RTOS a ovladače pro případné periferie s kompletní dokumentací. TI-RTOS (Texas Instruments – Real Time Operating System) je preventivní, vícevláknový operační systém pracující v reálném čase obsahující ovladače, nástroje pro synchronizaci a plánování. Jeho jádro je přizpůsobená verze staršího jádra známého jako SYS/BIOS. V níže popsaném postupu je použit SimpleLink CC2640R2 SDK balíček verze 3.30.00.20, který obsahuje TI BLE stack verze 3.03.01.00 [18].

#### CC2640R2F

Zařízení CC2640R2F je bezdrátový MCU zaměřený na použití v nízkoenergetických aplikacích s Bluetooth 4.2 a 5. Je poháněn jádrem ARM Cortex-M3 s maximálním taktem až 48 MHz a spotřebou 61  $\mu\text{A}/\text{MHz}$ . Velmi nízká aktivita rádiové části a nízký odebíraný proud MCU v Standby módu (1,1  $\mu\text{A}$ ) spolu umožňují dosáhnout výborné životnosti baterie. V našem postupu je MCU osazen na vývojovém kitu LAUNCHXL-CC2640R2. Na desce se také nachází ladící sonda (ang. debug probe)

XDS110, která slouží k hledání chyb v programu pomocí krokování. Ladící sondu lze pomocí propojek kompletně odpojit od MCU [19].

## Code Composer Studio IDE

Code Composer Studio je vývojové prostředí, které podporuje kompletní portfolio mikrokontrolérů firmy TI. CCS zahrnuje sadu nástrojů používaných k vývoji a ladění aplikací. Obsahuje optimalizační kompilátor C/C++, editor zdrojového kódu, prostředí pro vytváření projektů, debugger, profiler a mnoho dalších funkcí. CCS je založen na volně dostupném vývojovém prostředí Eclipse [18].

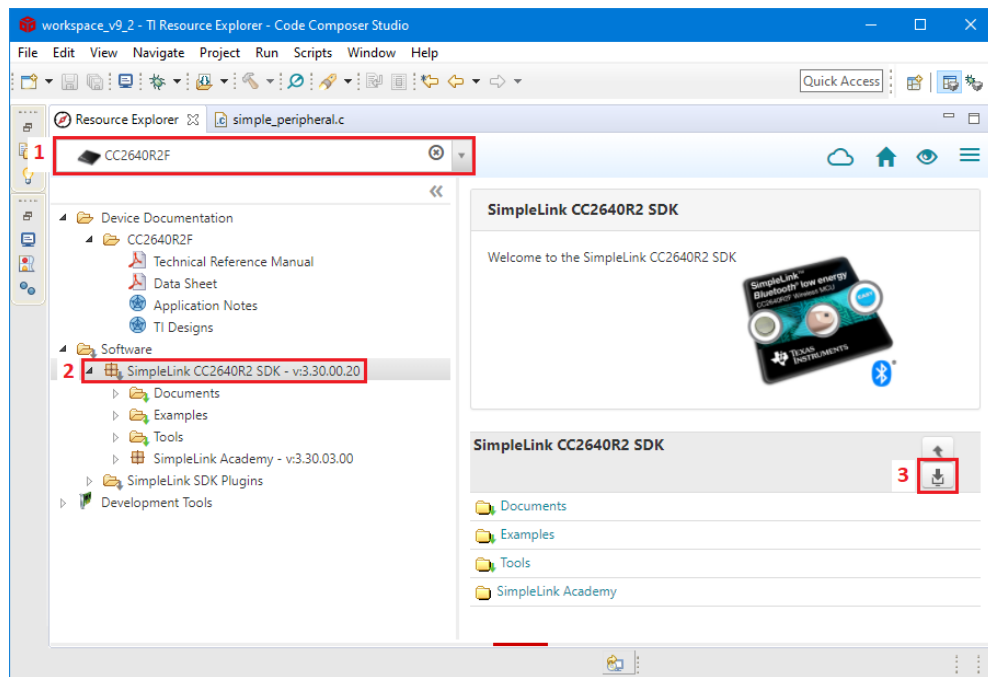
## 3.2 Postup implementace

Tento postup by měl pomoci urychlit počáteční fázi vývoje zařízení s BLE. Předpokládá se, že vývojář disponuje osobním počítačem s operačním systémem Windows, připojením k Internetu, a že vývojář má alespoň základní znalosti programovacího jazyka C, ve kterém je projekt napsán. Celý postup je rozdělen do čtyř podkapitol: Instalace, Import projektu, Nastavení BLE parametrů a Přidání profilu.

### 3.2.1 Instalace

Před samotnou instalací SDK je nutné stáhnout a nainstalovat CCS. Podrobný postup instalace CCS není součástí této práce. Může být spolu s podrobným manuálem celého CCS nalezen v *CCS User's Guide* v kapitole 3.2 [20]. Po úspěšné instalaci CCS se může přistoupit ke stažení a instalaci SDK. Stažení aktuální verze SDK je možné z oficiálních stránek firmy TI. Je doporučeno instalovat SDK i CCS do výchozích složek, které byli instalačním programem vybrány automaticky. Změna místa instalace může mít za důsledek nesprávné rozeznání SDK a jeho nefunkčnost. SDK je možné stáhnout a nainstalovat také přímo z CCS pomocí Resource Exploreru:

1. V horní liště CCS zvolte *View*→*Resource Explorer*
2. V okně Resource Explorer vlevo nahoře vyhledejte používaný mikrokontrolér či vývojový kit.
3. Po vyhledání se zobrazí možné softwarové balíčky, které jsou pro zvolený HW dostupné.
4. Dále už jen stačí vybrat požadované SDK a vpravo vedle jeho názvu kliknout na ikonu *Download and install*. Stažení SDK v CCS znázorňuje obrázek 3.1.

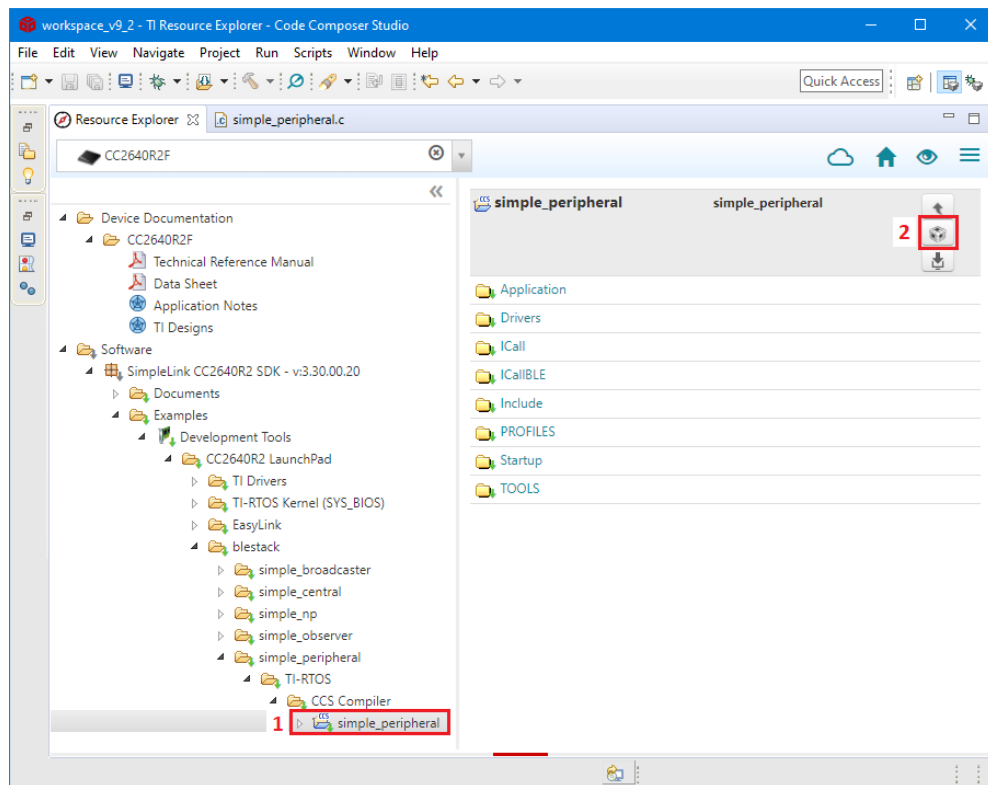


Obr. 3.1: Stažení SDK v Code Composer Studio.

### 3.2.2 Import projektu

V této části je ukázáno, jak importovat předpřipravené příkladové projekty. Vybrané projekty slouží jako dobrý počáteční bod vývoje. Podle toho, jakou roli na GAP vrstvě bude zařízení plnit, zvolíme projekt `Simple_broadcaster`, `Simple_observer`, `Simple_central` nebo `Simple_peripheral`. Dále v tomto postupu se bude pracovat s projektem `Simple_peripheral`. Tento projekt spolu s ostatními naleznete v Resource Exploreru v složce `Software` → `SimpleLink CC2640R2 SDK` → `Examples` → `Development Tools` → `CC2640R2 LaunchPad` → `ble-stack`. Po zvolení CCS projektu stačí jen kliknout na ikonu `Import to IDE`. Postup importování projektu `Simple_peripheral` ukazuje obrázek 3.2.

Po importu by měl být importovaný projekt vidět v Project Exploreru (`View` → `Project Explorer`) pod názvem `simple_peripheral_cc2640r2lp_app`. Spolu s projektem se importuje také potřebná knihovna v podobě druhého projektu s názvem `simple_peripheral_cc2640r2lp_stack_library`. Další informace o vytváření projektů a obecně o TI BLE stacku mohou být nalezeny v BLE-Stack User's Guide [21].



Obr. 3.2: Import projektu z SDK.

### 3.2.3 Nastavení BLE parametrů

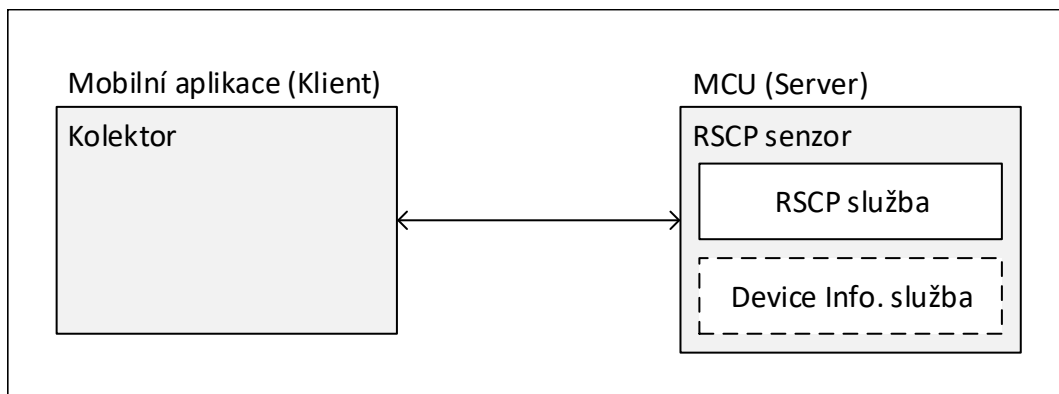
Nastavení BLE parametrů v případě Simple\_peripheral se provádí ve zdrojovém souboru simple\_peripheral.c, který se nachází ve složce Application. Veškeré hodnoty parametrů jsou definovány příkazem *#define* v horní části zdrojového kódu (přibližně řádky 90 až 110). Pro jejich změnu stačí přepsat odpovídající výchozí hodnotu na požadovanou hodnotu. Více o nastavování BLE parametrů naleznete v kapitole 4.2.2. Seznam parametrů a jejich výchozí hodnoty jsou následující:

- Propagační interval (Advertising interval)
  - *DEFAULT\_ADVERTISING\_INTERVAL* = 160 (s krokem 625  $\mu$ s, 160 = 100 ms)
- Interval připojení (Connection interval)
  - *DEFAULT\_DESIRED\_MIN\_CONN\_INTERVAL* = 80 (s krokem 1,25 ms, 80 = 100 ms)
  - *DEFAULT\_DESIRED\_MAX\_CONN\_INTERVAL* = 800 (s krokem 1,25 ms, 800 = 1000 ms)
- Otrokově zpoždění (Slave latency)
  - *DEFAULT\_DESIRED\_SLAVE\_LATENCY* = 0

- Časový limit dohledu (Supervision timeout)
  - `DEFAULT_DESIRED_CONN_TIMEOUT` = 1000 (s krokem 10 ms, 1000 = 10 s)

### 3.2.4 Přidání profilu

V této části implementace bude popsán postup přidání profilu Running Speed and Cadence Profile (RSCP, definice dostupná v [22]) do příkladového projektu. Přidání RSCP profilu se v zásadě neliší od přidávání jiných Bluetooth SIG adoptovaných profilů. Obsahem této práce není popis přidání profilu vytvořeného na míru. RSCP profil je tvořen službou Running Speed and Cadence Service (RSCS, definice dostupná v [23]), která je povinná, a službou Device Information Service (DIS, definice dostupná v [17]), která je volitelná a v implementaci RSCP profilu nemusí být obsažena. Specifikační dokument profilu RSCP dále udává nejrůznější požadavky na kolektor, senzor, navazování komunikace, zabezpečení a další. V tomto postupu je názorně ukázáno přidání pouze služby RSCS. Postup přidání služby DIS se výrazně neliší od postupu přidání služby RSCS. Vztah mezi RSCP profilem a službami, které obsahuje, znázorňuje obrázek 3.3.



Obr. 3.3: Vztah mezi RSCP profilem a obsaženými službami.

### Vytvoření zdrojového a hlavičkového souboru

Služba RSCS se definuje v samostatném zdrojovém a hlavičkovém souboru. V našem příkladě to jsou soubory `RSC_Service.c` a `RSC_Service.h`. Vytvoření těchto souborů je možné v CCS pod *File* → *New* → *Source File* nebo *Header File*. Po otevření okna stačí pouze zadat název souboru následovaný příponou `.c`, anebo `.h` v případě hlavičkového souboru. Zdrojový soubor obsahuje funkce pro práci se službou a především tabulku atributů, ve které je deklarován typ, práva a hodnoty jednotlivých

služeb a charakteristik. Tabulka atributů se skládá z prvků datového typu `attAttribute_t`. Definici datového typu `attAttribute_t` ukazuje výpis 3.1. Pro vytvoření obsahu zdrojového a hlavičkového souboru lze využít Example Service Generator, který je součástí SimpleLink™ Academy pro danou verzi SDK [24].

Tabulka atributů služby RSCS v sobě chronologicky obsahuje [23]:

- Deklaraci služby Running Speed and Cadence Service
- Deklaraci charakteristiky RSC Measurement
- Hodnotu charakteristiky RSC Measurement
- Deklaraci RSC Measurement Client Characteristic Configuration
- Deklaraci charakteristiky RSC Feature
- Hodnotu charakteristiky RSC Feature

Výpis 3.1: Definice datového typu `attAttribute_t`.

```

typedef struct
{
    uint8 len;           //Délka UUID (2 nebo 6 B)
    const uint8 *uuid;  //Ukazatel na UUID
} gattAttrType_t;

typedef struct attAttribute_t {
    gattAttrType_t type; //Typ atributu definovaný pomocí UUID
    uint8 permissions;  //Práva atributu (READ, WRITE)
    uint16 handle;      //Handler přiřazuje attribute server
    uint8* const pValue; //Ukazatel na hodnotu atributu
} gattAttribute_t;

```

Výpis 3.2 ukazuje vytvoření tabulky obsahující deklaraci služby RSCS. Deklarace služby RSCS (řádky 3 až 7) obsahuje nejdříve obecné UUID primární služby 0x2800 o velikosti 2 bajty. Dále definuje, že lze tuto deklaraci číst. Handler všech položek tabulky atributů je nulový, protože je atributům přiřazen atribut serverem. Poslední položka deklarace je specifické UUID vytvářené služby. V případě RSCS služby je toto UUID rovno hodnotě 0x1814. Zbylé atributy v tabulce se definují obdobným způsobem, a proto jsou ve výpisu vynechány. Seznam všech UUID je dostupný na oficiálních Bluetooth webových stránkách.



Výpis 3.2: Tabulka atributů služby RSCS.

```

static gattAttribute_t RSC_ServiceAttrTbl [] =
{
    { {ATT_BT_UUID_SIZE, primaryServiceUUID}, //2, 0x2800
      GATT_PERMIT_READ, //0x01
      0, //Handler
      (uint8_t *)&RSC_ServiceDecl //0x1814
    },
    ...
};

```

### Úprava programu

Po vytvoření potřebných souborů je dále potřeba upravit zdrojový soubor `simple_peripheral.c`. Nejdříve se musí ve vrchní části zahrnout hlavičkový soubor `RSC_Service.h` pomocí příkazu `#include`. Dále je vhodné přidat RSCS službu do propagačních paketů. Přítomnost podporovaných služeb v propagačních paketech umožňuje před připojením dvou BLE zařízení filtrovat jen ta zařízení, která podporují vyžadované služby. Následuje inicializace vytvořené RSCS služby příkazem `RSC_Service_AddService()`. Na závěr je potřeba deklarovat a inicializovat proměnné obsahující hodnoty charakteristik RSC Measurement a RSC Feature. Výše popsaný postup úpravy souboru `simple_peripheral.c` ukazuje výpis 3.3.

Výpis 3.3: Úprava programu v simple\_peripheral.c.

```

#include "RSC_Service.h" 1
... 2
static uint8_t advertData [] = 3
{ 4
... 5
    0x03, //Délka propagovaných dat 6
    GAP_ADTYPE_16BIT_MORE, //Typ propagovaných dat 7
    LO_UINT16(RSC_SERVICE_SERV_UUID), 8
    HI_UINT16(RSC_SERVICE_SERV_UUID) 9
}; 10
... 11
static void SimplePeripheral_init(void) 12
{ 13
... 14
    RSC_Service_AddService(selfEntity); 15
    uint8_t RSC_Service_RSC_Msr_initVal[RSC_SERVICE_RSC_MSR_LEN]= 16
        {0b00000000, 0b0000000000000000, 0b00000000}; 17
    RSC_Service_SetParameter(RSC_SERVICE_RSC_MSR_ID, 18
        RSC_SERVICE_RSC_MSR_LEN, 19
        RSC_Service_RSC_Msr_initVal); 20
    uint8_t RSC_Service_RSC_Ftr_initVal[RSC_SERVICE_RSC_FTR_LEN]= 21
        {0b0000000000000000}; 22
    RSC_Service_SetParameter(RSC_SERVICE_RSC_FTR_ID, 23
        RSC_SERVICE_RSC_FTR_LEN, 24
        RSC_Service_RSC_Ftr_initVal); 25
... 26
} 27

```

## 4 Návrh sporttesteru

V této kapitole je popsán návrh sporttesteru komunikujícího pomocí BLE. Sporttester obsahuje akcelerometr, který na základě změny akcelerace dokáže vypočítat celkový počet kroků, které nositel sporttesteru uběhl nebo ušel. Z celkového počtu kroků je následně vypočítána kadence kroků a aktuální rychlost. Tyto dvě hodnoty jsou posílány skrze BLE do chytrého mobilního telefonu, který je zobrazuje v kompatibilní mobilní aplikaci.

Další funkcí navrhovaného sporttesteru je měření síly nárazů. Tato funkce nám napomáhá do jisté míry kontrolovat styl běhu. Při překročení stanovené hranice síly nárazů je nositel upozorněn na nevhodný styl běhu krátkým rozsvícením LED diody. Zároveň s tím je do mobilní aplikace poslána maximální hodnota akcelerace, která byla naměřená právě v době překročení hranice síly nárazů.

Pro využití všech zmíněných funkcí je potřeba navrhnout vlastní mobilní aplikaci, pro kterou je již sporttester připraven. Návrh takovéto aplikace ovšem není součástí této práce. Současné mobilní aplikace třetích stran, podporující Bluetooth profil vybraný v kapitole 4.2.1, jsou schopny zobrazit pouze hodnoty kadence kroků a aktuální rychlost [22].

### 4.1 Hardwarové řešení

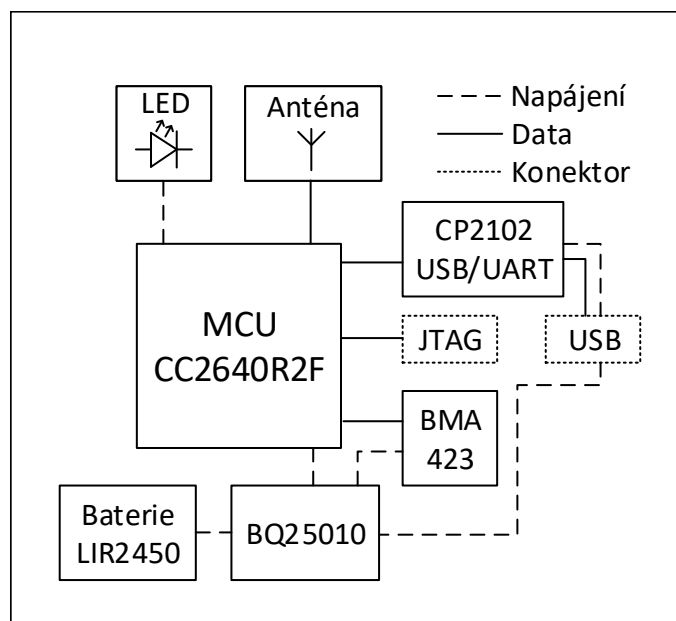
Pro hardwarovou realizaci navrhovaného sporttesteru je zapotřebí vybrat vhodný mikrokontrolér, BLE modul, akcelerometr, a jelikož bude zařízení připevněno na těle nositele sporttesteru, tak i bateriové napájení. Pro jednoduchost návrhu je většina součástí vybrána z nabídky firmy Texas Instruments.

Mikrokontrolér CC2640R2F je ideální volbou pro realizaci sporttesteru, jelikož má v sobě zabudovaný obvod pro bezdrátovou komunikaci podporující Bluetooth. Další velkou výhodou, oproti běžným MCU, je jeho nízká spotřeba energie, což je kritické u aplikací napájených z baterie. Z řady dostupných akcelerometrů je vybrán BMA423, který je, stejně jako vybraný MCU, velice úsporný a disponuje samostatnou výpočetní částí, která je schopná mikrokontroléru posílat již vypočítanou kadenci uběhnutých kroků. To napomáhá snížit spotřebu elektrické energie, jelikož není potřeba odesílat všechny vzorky akcelerace do MCU a až v něm z těchto vzorků počítat kadenci kroků [19, 25].

Pro napájecí část je potřeba vybrat integrovaný obvod (ang. integrated circuit, zkratka IC) pro stabilizaci napájecího napětí, IC pro nabíjení baterie a samotnou

baterii. Za účelem ještě větší úspory energie byla zvolena napájecí napěťová úroveň o hodnotě 1,8 V, která je podporována všemi použitými součástkami. Pro stabilizaci a snížení bateriového napětí na úroveň 1,8 V je zvolen IC BQ25010. Tento IC je schopen nejen snížit a stabilizovat napájecí napětí, ale také nabíjet baterie lithium-iontového (Li-Ion) typu [26]. Z dostupných Li-Ion baterií je vybrána dobíjecí knoflíková baterie LIR2450 od firmy Multicomp. Ta disponuje kapacitou 120 mAh a napětím 3,6 V [27]. Tato kapacita by měla, podle spotřeb jednotlivých součástek uvedených v datasheetech, stačit na aktivní používání sporttesteru po dobu 75 h. Po úplném vybití baterie je IC BQ25010 schopen baterii plně dobýt do 2 h. Pro signalizaci nevhodného běžeckého stylu je použita běžně dostupná SMD LED dioda.

Aby bylo možné do MCU nahrát program, tak je obvod doplněn o IC CP2102, což je USB/UART převodník. Ten zajišťuje převod dat z připojeného USB na UART komunikaci [28]. Pro dodržení správných napěťových úrovní mezi MCU a USB/UART převodníkem musí být mezi ně zapojen ještě překladač napěťových úrovní TXS0104EPWR [29]. Druhá možnost nahrání programu do MCU je za pomoci debugovacího a komunikačního JTAG portu. Zjednodušené blokové schéma můžete vidět na obrázku 4.1.



Obr. 4.1: Blokové schéma zapojení.

### 4.1.1 Akcelerometr BMA423

BMA423 je digitální akcelerometr od firmy Bosch Sensortec. Akceleraci je schopen měřit ve třech osách s rozlišením až 12 bitů. Obsahuje také samostatnou výpočetní část, která je schopna ze vzorků akcelerace přímo vyhodnocovat kroky, které běžec ušel nebo uběhl. Algoritmus počítání kroků je blíže popsán v kapitole 4.2.3. Měření akcelerace je možné v rozsahu od  $\pm 2$  g do  $\pm 16$  g a s frekvencí od 12,5 Hz do 1,6 kHz. Lze nastavit také průměrování zvoleného počtu posledních vzorků, čímž se sníží výstupní datový tok. Průměrování lze nastavit v rozsahu od 1 vzorku, kdy průměrování nemá žádný efekt, až do 128 vzorků. Komunikovat je schopen buď přes I2C sběrnici, anebo přes SPI komunikační rozhraní [25].

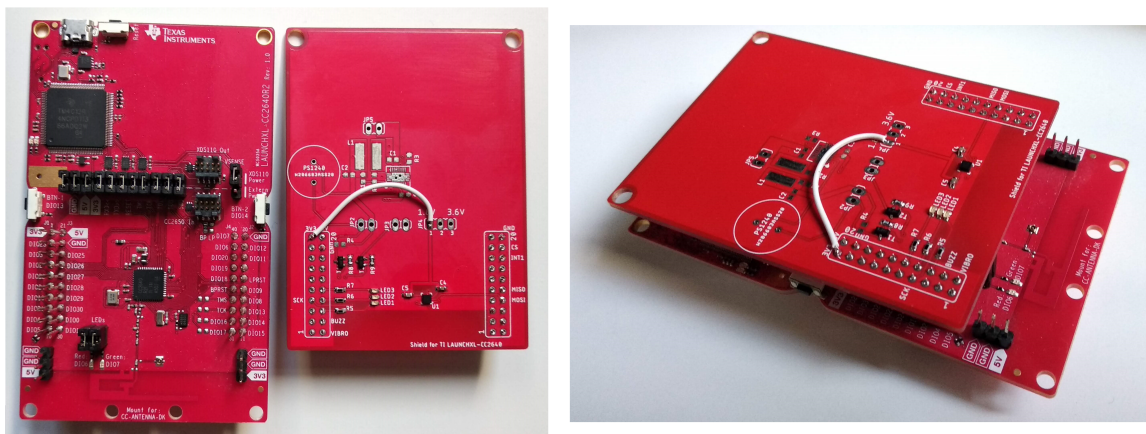
Pro použití v sporttesteru je akcelerometr nastaven následovně:

- rozlišení: 12 b,
- rozsah akcelerace:  $\pm 16$  g,
- frekvence měření: 200 Hz,
- průměrování: 1 vzorek,
- komunikační rozhraní: SPI.

### 4.1.2 Návrh desky plošných spojů

Pro připojení všech zmíněných součástek je zapotřebí zhotovit návrh zapojení obvodu a na jeho základě vyrobit desku plošných spojů (DPS). Pro snadnější vývoj sporttesteru byl nejdříve použit vývojový kit LAUNCHXL-CC2640R2, na kterém je osazen MCU CC2640R2F spolu s debuggerem TI XDS110. Pro tento vývojový kit byl vyroben tzv. shield, což je v zásadě DPS doplněna o další součástky, která se připojí přes dostupné konektory k vývojovému kitu. Takovýmto způsobem dostane vývojový kit další funkce, které mu shield umožní vykonávat. Sporttster shield obsahuje akcelerometr BMA423, doplněný o nezbytné pasivní součástky, LED diodu a dva dvacetipinové konektory, které zajišťují propojení s vývojovým kitem, a má rozměry  $78,1 \times 58,5$  mm. Na obrázku 4.2 je vidět vyrobený shield společně s vývojovým kitem LAUNCHXL-CC2640R2. Kompletní schéma zapojení a návrh DPS shieldu se nachází v přílohách A a B na konci této práce.

Ve výsledném návrhu sporttesteru již není použit vývojový kit společně s shieldem. Jedná se o kompletně samostatnou DPS o rozměrech  $62,2 \times 44,5$  mm, která obsahuje všechny součástky uvedené na začátku této kapitoly. Schéma zapojení a návrh DPS výsledného sporttesteru naleznete v přílohách C a D.



Obr. 4.2: Vývojová deska LAUNCHXL-CC2640R2 a vyrobený shield.

Z logických důvodů není možné na tělo nositele připevnit samotnou DPS bez další ochrany. Mohlo by dojít ke zničení DPS a připojených součástek, ke zkratu na DPS a nošení takového sporttesteru by pro nositele určitě nebylo pohodlné. Umístění DPS do vhodné krabičky malých rozměrů by vyřešilo zmíněné problémy. Z dostupných krabiček byla vybrána krabička od výrobce New Age Enclosures s označením S-272008, která má rozměry  $69,9 \times 50,8 \times 20,3$  mm [30].

Aby bylo možné spolehlivě měřit kadenci běhu a sílu nárazu, je nutné vhodně zvolit místo upevnění sporttesteru na těle nositele. Běžnými místy, kde se sporttestery upevňují, jsou hrudník, kotník, holeně a v některých případech i přímo v botě v podobě vložky do bot. S ohledem na případné možnosti měření tepové frekvence je příhodné upevnit zařízení přímo na pokožce. Umístění na pase či hrudníku je nevhodné, jelikož by měřená změna akcelerace byla mnohem menší nežli změna akcelerace, která by byla změřena při upevnění zařízení na končetině nositele. Z těchto důvodů vyplývá, že nejvhodnějším místem pro upevnění navrženého sporttesteru je spodní část holeně. Podle umístění sporttesteru je také důležité vybrat vhodný způsob upevnění. Pro upevnění na holeni se dá použít elastický pásek, pás suchého zipu případně kombinace obojího. Z uvedených způsobů upevnění je preferovaný pás suchého zipu, jelikož je to nejpevnější způsob upevnění. Čím pevněji je zařízení upevněno na těle nositele, tím přesnější jsou hodnoty změřené akcelerace [31].

## 4.2 Softwarové řešení

Softwarové řešení je postaveno na příkladovém projektu Simple\_peripheral ze softwarového balíčku SimpleLink CC2640R2 SDK od firmy TI. Tento projekt používá

operační systém pracující v reálném čase (TI-RTOS) a plně funkční BLE verze 4.2 [19]. Projekt samozřejmě ve výchozí podobě nevykonává funkce sporttesteru, a proto je dále upraven z pohledu implementovaných BLE profilů, služeb a nastavených parametrů. Je také doplněn o kód připravující změřené hodnoty z akcelerometru k odeslání skrze BLE. Všechn SW je psán v programovacím jazyce C a pro jeho úpravu, testování a nahrávání je použito vývojové prostředí Code Composer Studio. Chod programu znázorňuje vývojový diagram na obrázku 4.3, kde tečkované šipky značí přesah do BLE procesu.

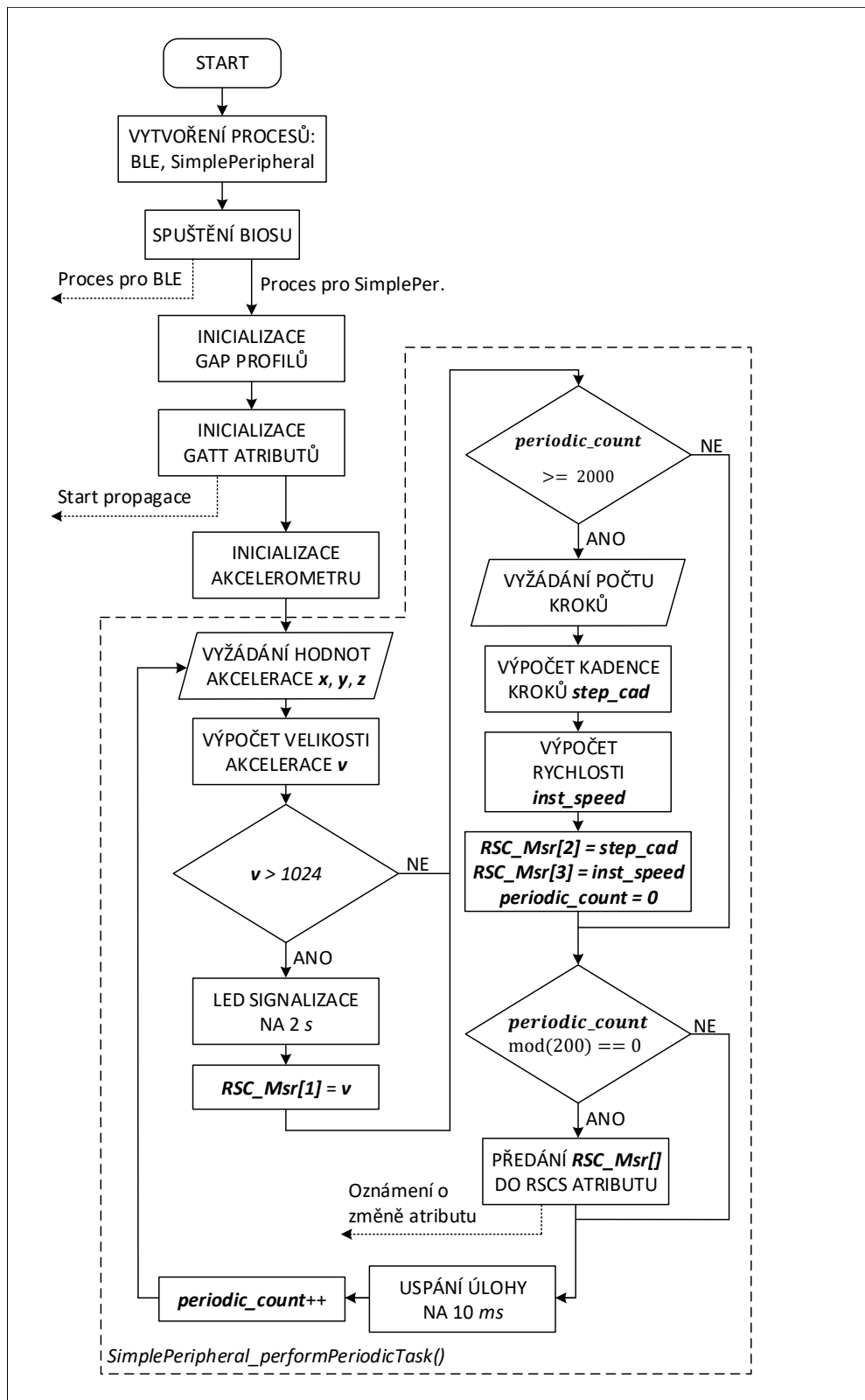
### 4.2.1 Výběr profilu a služeb

Při výběru profilu, který by pomáhal předávat naměřené hodnoty do mobilní aplikace, existují dvě základní možnosti. Jedna je vytvořit si vlastní profil, a tudíž i služby. To by ale vyžadovalo také vlastní mobilní aplikaci, což by mělo za následek větší finanční prostředky potřebné na vývoj. Jednodušším řešením je využít volně dostupnou aplikaci, která podporuje některý z vhodných adoptovaných profilů. Skupina SIG adoptovala řadu profilů a služeb, které mají uplatnění v zdravotnictví a sportu (více v *GATT Specifications*). Pouze jeden profil plně vyhovuje našim potřebám a je zároveň podporován různými běžeckými aplikacemi. Jedná se o profil Running Speed and Cadence Profile [22]. Tento profil zahrnuje služby RSCS a DIS. Služba RSCS dále obsahuje 2 povinné charakteristiky – RSC Measurement a RSC Feature. Charakteristika RSC Measurement se skládá z 10 bajtů. První bajt obsahuje tzv. flagy, které značí dostupnost hodnot dalších funkcí, které jsou definovány v charakteristice RSC Feature. Druhý a třetí bajt obsahují hodnotu aktuální rychlosti v metrech za sekundu. Kadence kroků v krocích za minutu je obsažena v čtvrtém bajtu [32].

### 4.2.2 Nastavení parametrů

Výchozí hodnoty BLE parametrů v příkladovém projektu nejsou příliš vhodné pro funkce sporttesteru a z hlediska úspory energie také nejsou dobře zvoleny. Propagační interval 100 ms zaručuje poměrně rychlé připojení bez zdlouhavého čekání, ale výrazně zvyšuje spotřebu zařízení při častém navazování spojení. Vhodnější by bylo zvolit tuto hodnotu vyšší, přibližně kolem 500 ms. Při nastavování tohoto parametru se vždy dělá kompromis mezi pohodlím uživatele (čekání na připojení) a úsporností zařízení při připojování k mobilnímu telefonu.

Jelikož je naše zařízení periferní, další parametry jsou jen upřednostňované a o skutečném nastavení parametrů rozhoduje centrální zařízení (v našem případě mobilní telefon). Při nastavování intervalu připojení se provádí kompromis mezi četností



Obr. 4.3: Vývojový diagram programu.



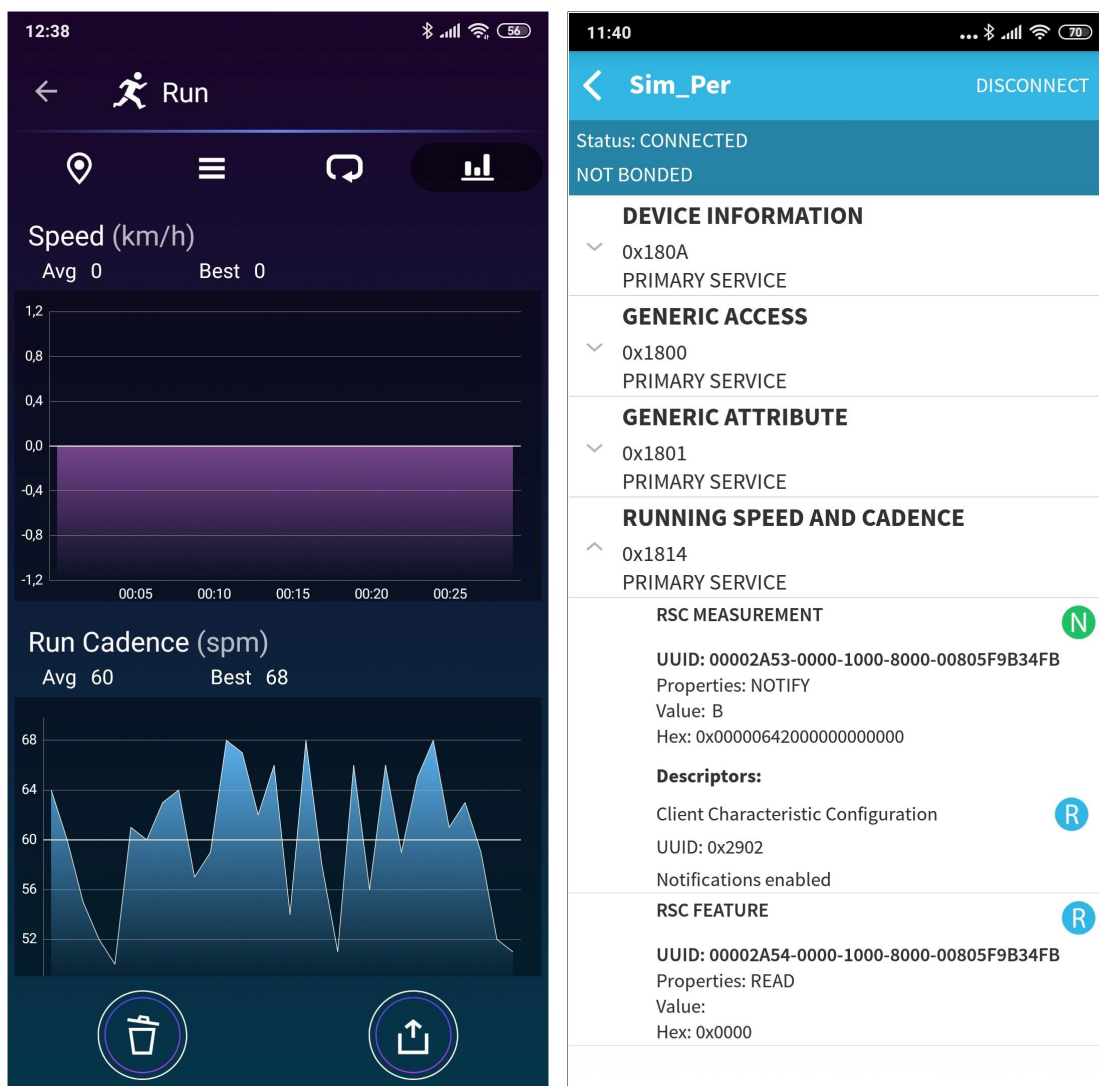
odesílání zpráv a úsporou zařízení. Výchozí upřednostňovaná hodnota je nastavena v rozsahu od 100 ms do 1000 ms. Při nastavování tohoto parametru se musí brát v potaz zobrazovací možnosti aplikace a také požadavky uživatele. Pokud uživateli stačí vědět pouze průměrnou hodnotu kadence běhu za dobu např. 5 minut, tak je možné nastavit interval připojení blíže k maximální hodnotě 4 s. Na druhou stranu, pokud aplikace vykresluje graf změny kadence v čase, tak je vhodné nastavit interval připojení na menší hodnotu, např. 1 s, aby bylo dost hodnot v grafu a nedocházelo k velkému zkreslení průměrováním. Parametry otrokového zpoždění a časový limit dohledu je vhodné nastavit podle vzorců uvedených v kapitole 2.2.4.

Pro testování komunikace a nastavených parametrů byl navržený sporttester připojen ke kompatibilní aplikaci Ala Fitness. Po připojení byl proveden test v podobě spuštění cvičného běhu. Sporttester byl nastaven tak, aby posílal každou vteřinu náhodně generovanou hodnotu kadence běhu v rozsahu od 50 do 70 kroků za minutu. Aplikace zobrazuje spolu s kadencí běhu také další parametry, které v našem testu nebyly sledovány a jsou většinou nulové. Obrázek 4.4 ukazuje výsledky cvičného běhu, který trval 30 s. Průměrná a maximální kadence běhu je, společně s grafem kadence kroků v závislosti na čase, zobrazena v levé části obrázku. Správnost nastavení BLE profilu potvrzuje také servisní aplikace s názvem BLE Scanner, která je na obrázku 4.4 vpravo.

### 4.2.3 Algoritmus počítání kroků

Algoritmus zajišťující počítání kroků je obsažen přímo v akcelerometru BMA423. Tento algoritmus definuje dvě základní funkce – detektor kroků a počítadlo kroků. Funkce detektoru kroků hlásí každý detekovaný krok nositele při chůzi či běhu s nízkým zpožděním. Tato funkce splňuje požadavky, které jsou také kladeny na algoritmy počítání kroků v Android 4.4 a vyšší. Jeden z požadavků udává, že odchylka mezi změřeným a skutečným počtem kroků může být maximálně 10 % [33].

Počítadlo kroků hlásí počet kroků, které uživatel provedl od první aktivace funkce. Součet událostí hlášených detektorem kroků se může lišit od hodnoty počtu kroků hlášené počítadlem kroků. To je způsobeno dodatečnou validací kroků při následném zpracování. Při takové validaci může někdy dojít k rozhodnutí, že krok nahlášený detektorem je chybný a takový krok poté není zahrnut do celkového počtu kroků. Navrhnutý sporttester přímo pracuje jen s funkcí počítadla kroků, jelikož pouze vyčítá registr obsahující celkový počet kroků. Na základě hodnoty tohoto registru se dále v MCU odvozují hodnoty rychlosti běhu a kadence kroků.



Obr. 4.4: Výsledky cvičného běhu z aplikace Ala Fitness.

#### 4.2.4 Popis periodické funkce

Tato podkapitola obsahuje podrobný popis programového kódu periodické funkce *SimplePeripheral\_performPeriodicTask*. V této funkci jsou napsány veškeré periodicky se opakující operace, které sporttester vykonává. Mezi tyto funkce patří vyžádání vzorku akcelerace z akcelerometru, kontrola překročení hranice síly nárazů, předání vypočítaných hodnot do BLE profilu a další.

Výpis 4.1 obsahuje definice často používaných číselných konstant, které se vyskytují v periodické funkci. *PER\_EVT\_PERIOD* určuje, po jaké době se opět zavolá periodická funkce *SimplePeripheral\_performPeriodicTask*. Pro spolehlivé fungování

funkce sledování síly nárazů byla tato proměnná zvolena na hodnotu 10 ms. Od této hodnoty se také odvíjí frekvence vyčítání vzorků akcelerace  $f_{\text{akc}}$ , která je rovna:

$$f_{\text{akc}} = \frac{1}{\text{PER\_EVT\_PERIOD}} = \frac{1}{0,01} = 100 \text{ Hz} . \quad (4.1)$$

Proměnná *STEP\_PERIOD* udává, po kolika milisekundách se vyžádá celkově uběhnutý počet kroků z akcelerometru. *ACCEL\_THRESHOLD* je hranice výsledné akcelerace s jednotkou  $\frac{1}{128}g$ . Hodnota 1024 tedy odpovídá akceleraci 8 g. Dále *AVG\_STRIDE\_LGTH* obsahuje hodnotu délky průměrného kroku v centimetrech, která se rovná 175 cm [34, 35]. Poslední proměnná *LED\_ON\_RATIO* nepřímo udává velikost doby  $t_{\text{led}}$ , po kterou se rozsvítí LED dioda v případě překročení hranice síly nárazů. Tato doba je rovna:

$$t_{\text{led}} = \frac{\text{STEP\_PERIOD}}{\text{LED\_ON\_RATIO}} = \frac{20000}{10} = 2000 \text{ ms} . \quad (4.2)$$

Výpis 4.1: Periodická funkce sporttesteru – definice konstant.

<b>#define</b> PER_EVT_PERIOD	10	1
<b>#define</b> STEP_PERIOD	20000	2
<b>#define</b> ACCEL_THRESHOLD	1024	3
<b>#define</b> AVG_STRIDE_LGTH	175	4
<b>#define</b> LED_ON_RATIO	10	5

Ve výpisu 4.2 můžete vidět první část kódu periodické funkce. Jako první v sobě obsahuje deklaraci lokálních proměnných. Přestože RSCP profil udává, že hodnota kadence kroků může být maximálně 8 bitů velká, tak v kódu periodické funkce má proměnná *step\_cadence*, která se používá k ukládání kadence kroků, datový typ *uint16\_t* (obdoba datového typu *unsigned short*), což je 16 bitů velké kladné číslo. Důvod zvolení 16 bitové proměnné jsou pro názornost popsány až společně s počítáním kadence kroků uvedené ve výpisu 4.3. Dále je definována struktura *bma4\_accel*. Ta obsahuje proměnné *x*, *y* a *z*, které jsou datového typu *uint16\_t* a slouží k ukládání hodnot akcelerace pro všechny tři osy akcelerometru. Globální proměnná *periodic\_count* je také datového typu *uint16\_t* a je v ní zaznamenáno, kolikrát se již zavolala periodická funkce od posledního výčtu počtu kroků. Za pomoci této proměnné je z času *PER\_EVT\_PERIOD* odvozena doba *STEP\_PERIOD*, ze které je dále odvozen výše zmiňovaný čas  $t_{\text{led}}$ .

Po zvýšení *periodic\_count* o jedničku se dále zavolá funkce *bma4\_read\_accel\_xyz*, která zajistí vložení hodnot akcelerace z akcelerometru do proměnné *accel\_data*. Tato funkce je součástí ovladače dostupného pro akcelerometr BMA423, který byl zakomponován do upraveného projektu *Simple\_peripheral*. Dále následuje výpočet

výsledné velikosti akcelerace. Jelikož hodnoty akcelerace z jednotlivých os vlastně tvoří složky výsledného vektoru akcelerace  $\vec{v}$ , tak se velikost výsledné akcelerace vypočítá pomocí vzorce pro výpočet velikosti trojrozměrného vektoru:

$$|\vec{v}| = \sqrt{x^2 + y^2 + z^2} . \quad (4.3)$$

Hodnota velikosti výsledného vektoru je uložena do proměnné *sum\_accel\_data*, která je následně porovnávána v podmínce uvedené na řádce 9 výpisu 4.2. Pokud je hodnota této proměnné větší než hodnota proměnné *ACCEL\_THRESHOLD*, tak se globální booleovská proměnná *threshold\_exceeded* překlopí na hodnotu *true* a velikost výsledné akcelerace je předána do proměnné *RSC\_Service\_RSC\_Msr\_initVal*, která slouží k ukládání hodnot charakteristiky RSC Measurement.

Výpis 4.2: Periodická funkce sporttesteru – část první.

```

static void SimplePeripheral_performPeriodicTask(void){ 1
    uint16_t step_cadence = 0; 2
    struct bma4_accel accel_data = {.x=0, .y=0, .z=0}; 3
    periodic_count++; 4
    bma4_read_accel_xyz(&accel_data, &bma423); 5
    sum_accel_data = sqrt(accel_data.x*accel_data.x 6
                          + accel_data.y*accel_data.y 7
                          + accel_data.z*accel_data.z); 8
    if(sum_accel_data > ACCEL_THRESHOLD){ 9
        threshold_exceeded = true; 10
        RSC_Service_RSC_Msr_initVal[4] = 11
            (sum_accel_data & 0xFF00) >> 8; 12
        RSC_Service_RSC_Msr_initVal[5] = 13
            sum_accel_data & 0x00FF; 14
    } 15
    ... 16

```

V další části programu periodické funkce, uvedené ve výpisu 4.3, se zkontroluje zda hodnota proměnné *periodic\_count* je rovna nebo větší než poměr proměnných *STEP\_PERIOD* a *PER\_EVT\_PERIOD*. Tato podmínka vlastně jen kontroluje, zda již uplynula doba *STEP\_PERIOD* od posledního splnění této podmínky. Pokud již uběhla, tak se proměnná *periodic\_count* vynuluje. Dále se předchozí hodnota počtu kroků uloží do lokální proměnné *temp\_step\_count* a do globální proměnné *step\_count* se uloží aktuální hodnota počtu kroků, která je vyžádána od akcelerometru. Pomocí těchto dvou proměnných a hodnoty *STEP\_PERIOD* se vypočítá kadence kroků *K* podle vzorce (4.4):

$$K = \frac{(\text{step\_count} - \text{temp\_step\_count}) \cdot 60000}{\text{STEP\_PERIOD}} \text{ [krok/min]} . \quad (4.4)$$

Při počítání kadence kroků takovýmto způsobem může dojít k případu, kdy je výsledná hodnota větší než 255, a tudíž musí mít proměnná *step\_cadence*, do které se výsledek ukládá, velikost alespoň 16 bitů. Vzhledem k tomu, že kadence nemůže být záporná, tak je proměnná *step\_cadence* datového typu *uint16\_t*. V RSCS službě je ovšem definováno, že pro přenos kadence kroků může být využito maximálně 8 bitů. Z toho důvodu je pomocí podmínky na řádku 8 zajištěno přepsání hodnot větších než 255 právě na hodnotu 255. Takovýmto způsobem nedojde v přetečení povolených 8 bitů pro přenos kadence. Na řádku 10 se poté uloží do proměnné *inst\_speed* okamžitá rychlost, která se vypočítá z průměrné délky kroku *AVG\_STRIDE\_LGTH* podle vzorce (4.5):

$$inst\_speed = \frac{(step\_count - temp\_step\_count) \cdot AVG\_STRIDE\_LGTH \cdot 10}{STEP\_PERIOD} [m/s] . \quad (4.5)$$

Po výpočtu okamžité rychlosti se společně s kadencí kroků vloží tyto hodnoty do příslušných pozic v proměnné *RSC\_Service\_RSC\_Msr\_initVal*.

Výpis 4.3: Periodická funkce sporttesteru – část druhá.

...	1
<b>if</b> (periodic_count >= STEP_PERIOD / PER_EVT_PERIOD){	2
periodic_count = 0;	3
uint16_t temp_step_count = step_count;	4
bma423_step_counter_output(&step_count, &bma423);	5
step_cadence =	6
((step_count - temp_step_count) * 60000) / (STEP_PERIOD);	7
<b>if</b> (step_cadence > 255)	8
step_cadence = 255;	9
uint8_t inst_speed =	10
((step_count - temp_step_count) * AVG_STRIDE_LGTH * 10) / STEP_PERIOD;	11
RSC_Service_RSC_Msr_initVal[1] = inst_speed;	12
RSC_Service_RSC_Msr_initVal[2] = 0;	13
RSC_Service_RSC_Msr_initVal[3] = step_cadence;w	14
}	15
...	16

Na konci periodické funkce, uvedené ve výpisu 4.4, se nachází podmínka kontrolující zda je splněno, že je proměnná *periodic\_count* dělitelná celočíselným výrazem  $\frac{STEP\_PERIOD / PER\_EVT\_PERIOD}{LED\_ON\_RATIO}$ . To se dá docílit pomocí funkce modulo, resp. pomocí zbytku po celočíselném dělení, což ukazuje rovnice (4.6):

$$periodic\_count \bmod \frac{STEP\_PERIOD}{PER\_EVT\_PERIOD \cdot LED\_ON\_RATIO} = 0 . \quad (4.6)$$

V podstatě se jedná o kontrolu uplynutí doby  $t_{led}$ , která byla vypočítána v rovnici (4.2). Při splnění této podmínky se dále kontroluje, zda je booleovská proměnná *threshold\_exceeded* rovna *true*. Pokud ano, tak se přepíše na hodnotu *false* a je rozsvícena signalizační LED dioda pomocí funkce *PIN\_setOutputValue*, která je součástí obecného PIN a GPIO ovladače pro MCU CC2640R2F. V případě, že se *threshold\_exceeded* rovná hodnotě *false*, tak je pouze vypnuta LED dioda opět pomocí funkce *PIN\_setOutputValue*. Na závěr dojde k předání hodnot obsažených v proměnné *RSC\_Service\_RSC\_Msr\_initVal* do RSCS služby, čímž se vyvolá aktualizace BLE oznámení a nové hodnoty jsou ihned poslány až do připojené mobilní aplikace.

Výpis 4.4: Periodická funkce sporttesteru – část třetí.

```

...
    if(periodic_count%
        ((STEP_PERIOD/PER_EVT_PERIOD)/LED_ON_RATIO)==0){
    if(threshold_exceeded == true){
        threshold_exceeded = false;
        PIN_setOutputValue(PinHandle, IOID_28, 1);
    }else
        PIN_setOutputValue(PinHandle, IOID_28, 0);
    RSC_Service_SetParameter(RSC_SERVICE_RSC_MSR_ID,
                            RSC_SERVICE_RSC_MSR_LEN,
                            &RSC_Service_RSC_Msr_initVal);
    }
} //Konec SimplePeripheral_performPeriodicTask

```

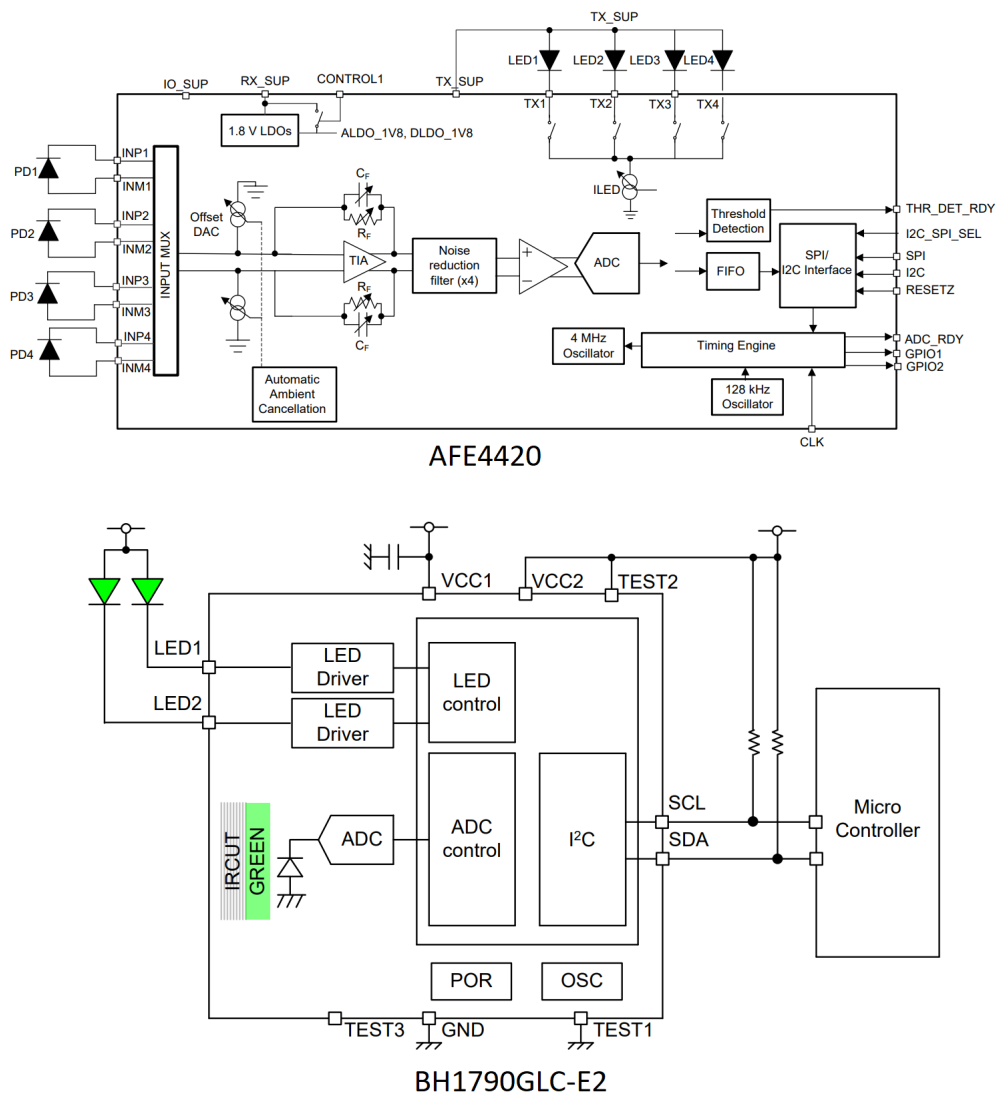
## 4.3 Budoucí vývoj

Vývoj tohoto sporttesteru by mohl dále pokračovat jak v SW části, tak z pohledu hardwaru. Mohly by se přidat další funkcionality, jako je například měření tepové frekvence. Dále by se veličiny okamžitá rychlost a kadence kroků mohli počítat s plovoucí desetinnou čárkou, což by pomohlo zpřesnit jejich výsledné hodnoty. V současnosti jsou tyto veličiny počítány v celočíselné rovině. Dále by se mohla optimalizovat úspora energie zařízení za účelem delší výdrže baterie.

### 4.3.1 Návrh měření tepové frekvence srdce

Měření tepové frekvence za běhu je možné provést buď měřením časové změny elektrického potenciálu způsobeného srdeční aktivitou, nebo pomocí měření změny prokrvenosti krve pod kůží. K měření elektrického potenciálu se používají hrudní pásy.

Připojení hrudního pásu k našemu zařízení by však bylo obtížné a jeho nošení bývá často nepohodlné pro nositele. Lepším řešením je použití optického snímače, který prosvěcuje pokožku pod sporttesterem. Světlo vyzařované do pokožky se částečně odrazí zpět do senzoru. Se změnou prokrvenosti pokožky se mění množství zpět odraženého světla. Tyto změny jsou pak vyhodnocovány za účelem zjištění tepové frekvence srdce. Tato metoda je sice méně přesná, ale je stále dostačující pro použití v sporttesterech [36]. Použitelné senzory pro optické měření srdečního tepu jsou například AFE4420 od výrobce TI nebo BH1790GLC-E2 od ROHM Semiconductor. Jejich doporučené zapojení s blokovým diagramem můžete vidět na obrázku 4.5.



Obr. 4.5: Doporučené zapojení s diagramem pro AFE4420 a BH1790GLC-E2 (převzato z [37, 38]).

## 5 Měření

### 5.1 Měření průběhu akcelerace při běhu

Měření průběhu akcelerace má za cíl co nejdetailněji změřit změny akcelerace při běhu. Výstupy tohoto měření by měli pomoci při nastavování parametrů, které ovlivňují přesnost a spolehlivost funkce kontroly síly nárazů. Těmito parametry jsou perioda vyčítání akcelerace (*PER\_EVT\_PERIOD*) a hraniční hodnota akcelerace (*ACCEL\_THRESHOLD*). Průběhy akcelerace byly změřeny pro tři různé tvrdé styly běhu – lehký, střední a tvrdý. Při lehkém stylu běhu noha dopadá na špičku chodidla a pata běžce se sotva dotýká země. Všechn náraz při dopadu nohy na zem je odpružen lýtkovým svalstvem. Při středním stylu běhu na zem dopadá špička následovaná patou. Při tvrdém stylu běhu na zem jako první tvrdě dopadá pata.

Očekávanými výsledky jsou grafy průběhů akcelerace jednotlivých kroků a maximální hodnoty akcelerace kroků pro 3 různé styly běhu. Z těchto grafů budou vyvozeny zmiňované parametry takovým způsobem, aby při tvrdém stylu běhu byl běžci pomocí LED diody signalizován nevhodný styl běhu, zatímco při ostatních dvou stylech běhu by tato signalizace neměla být aktivní. Při nastavování hraniční hodnoty akcelerace bude přihlíženo také k odborné literatuře.

### Výsledky

V tabulce 5.1 jsou zobrazeny maximální hodnoty akcelerace kroku, což je vždy největší hodnota vzorku akcelerace naměřené akcelerometrem v průběhu jednoho kroku. V zásadě to je vždy vzorek změřený po dopadu nohy na zem. Některé vzorky akcelerace pro tvrdý styl běhu dosahují hodnot přesahující 8 g. Právě hodnota akcelerace 8 g byla v článku [39] zvolena jako hraniční hodnota pro pozvání sledovaných běžců do rekvalifikačního programu. Tento program měl za cíl snížit zatížení dolních končetin vybraných jednotlivců. Tento článek také uvádí, že hodnota akcelerace 8 g byla o jednu standardní odchylku nad průměrem 171 nezraněných běžců z probíhající studie prováděné v laboratoři pro analýzu pohybu na University of Delaware. Na základě tohoto článku a hodnot maximální akcelerace kroků tvrdého stylu běhu byl zvolen parametr hraniční hodnota akcelerace na hodnotu 1024, což odpovídá hodnotě 8 g.

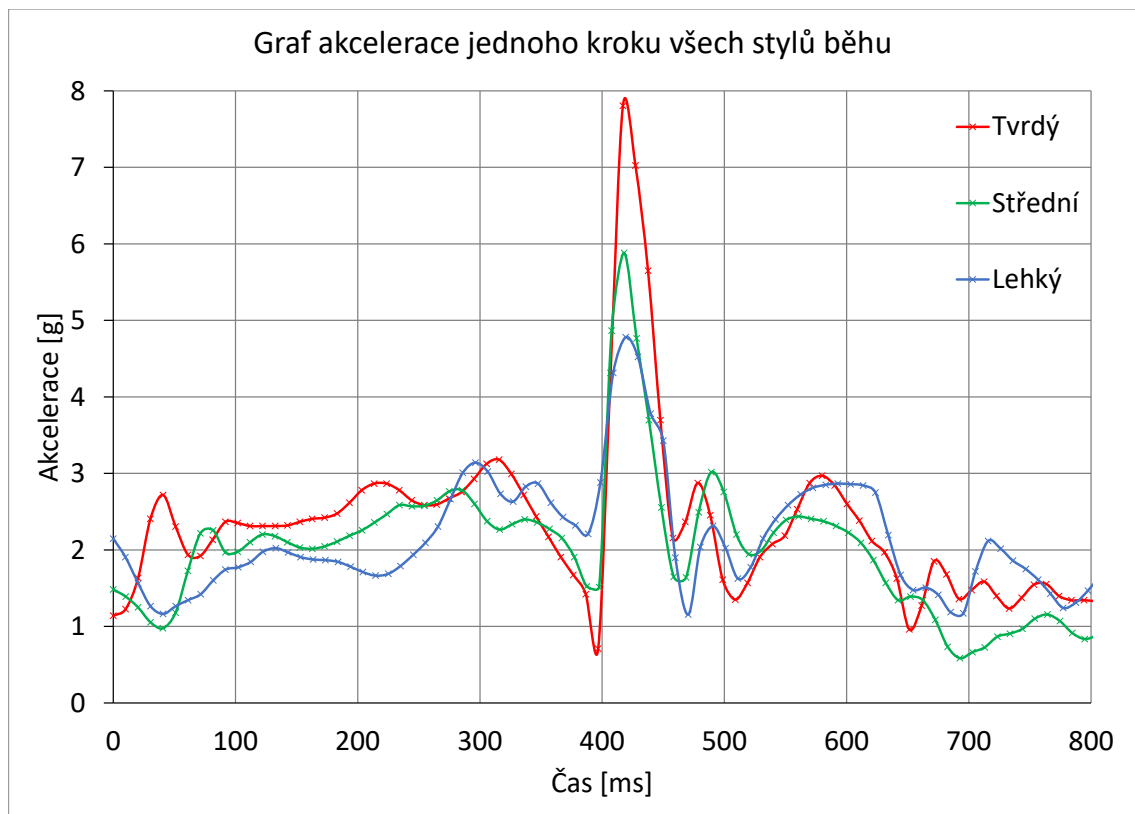
Graf na obrázku 5.2 obsahuje průběh akcelerace jednoho kroku pro všechny tři styly běhu. Při měření byla perioda vyčítání akcelerace zvolena na hodnotu 10 ms. Aby bylo možné spolehlivě kontrolovat překročení hranice síly nárazů, tak je velice



důležitě, aby bylo naměřeno dostatečné množství vzorků. V případě, kdy by byla perioda zvolena příliš velká, resp. změřené množství vzorků za sekundu příliš malé, tak by mohlo docházet k nezaznamenání dominantního pulzu průběhu akcelerace, který v grafu začíná v čase 400 ms. Vzhledem k trvání tohoto pulsu přibližně po dobu 50 ms, tak je při periodě vyčítání vzorků 10 ms naměřeno 5 vzorků akcelerace z tohoto pulzu. To je dostatečné množství pro kontrolu překročení hranice síly nárazů, a tudíž je perioda vyčítání vzorků 10 ms vhodně zvolena.

Tab. 5.1: Tabulka hodnot pro měření průběhu akcelerace kroků.

Krok číslo	Maximální akcelerace kroku [g]		
	Lehký styl	Střední styl	Tvrký styl
1	4,48	5,76	7,51
2	4,71	5,92	8,21
3	4,78	5,20	8,27
4	4,63	5,74	7,68
5	5,06	5,59	8,59
6	5,13	5,22	7,69
7	4,75	6,13	7,80
8	4,55	5,88	7,52
9	4,75	5,37	7,87
10	4,54	5,53	7,60
Arit. průměr	4,74	5,63	7,87



Obr. 5.1: Graf průběhů akcelerace.

## 5.2 Měření přesnosti počtu kroků

Toto měření mělo za cíl zjistit, jak přesně je navrhnutý sporttester schopen měřit kroky, které nositel uběhl nebo ušel. Zároveň se sporttesterem byla pro porovnání použita i jiná zařízení umožňující měření kroků. Z dostupných zařízení byl vybrán fitness náramek Xiaomi Mi Band 3 a mobilní telefon Xiaomi Redmi Note 8 Pro s operačním systémem Android 9 Pie. Xiaomi Mi Band 3 byl v průběhu měření nastaven do režimu Treadmill, protože to je jediný dostupný režim, který měří kroky v reálném čase. Na mobilním telefonu byla spuštěna aplikace Google Fit. Všechna zařízení byla při měření na běžci připevněna současně. Jako testovací dráha sloužila přímá část asfaltové cyklotrasy bez znatelného převýšení či nerovností. Měření probíhalo takovým způsobem, že běžec nejdříve chůzí ušel přesně 200 kroků po testovací dráze. Následně byly ze všech zařízení zapsány hodnoty naměřených kroků. Podobným způsobem probíhalo měření pro běh. Běžec nejdříve v poklusu uběhl 200 kroků po testovací dráze, a následně byly zapsány hodnoty ze všech zařízení. Takovýto průběh měření byl opakován celkem čtyřikrát.

Očekávanými výsledky tohoto měření jsou hodnoty počtu kroků, které zařízení naměřila během jednotlivých opakování měření. Z těchto hodnot budou vypočítány průměry a z průměrů následně odchylky od referenční hodnoty 200 kroků. Tyto odchylky nám poskytnou lepší představu o přesnosti měření kroků, které běžec ušel a uběhl.

## Výsledky

Tabulka 5.2 obsahuje všechny hodnoty počtu kroků  $ch_i$ , které zařízení naměřila při chůzi. Tabulka 5.3 obsahuje všechny hodnoty počtu kroků  $b_i$  pro běh. Z těchto hodnot byly vypočítány aritmetické průměry  $\overline{ch}$  a  $\overline{b}$  zvláště pro daný styl pohybu a zařízení, které dané hodnoty naměřilo. Z aritmetických průměrů byla vypočítána odchylka  $\delta$  od referenční hodnoty  $r = 200$  pro každé zařízení podle vzorců (5.1) a (5.2):

$$\delta_{ch} = \left| \left( 1 - \frac{\overline{ch}}{r} \right) \right| \cdot 100 \text{ [%]} \quad (5.1)$$

$$\delta_b = \left| \left( 1 - \frac{\overline{b}}{r} \right) \right| \cdot 100 \text{ [%]} \quad (5.2)$$

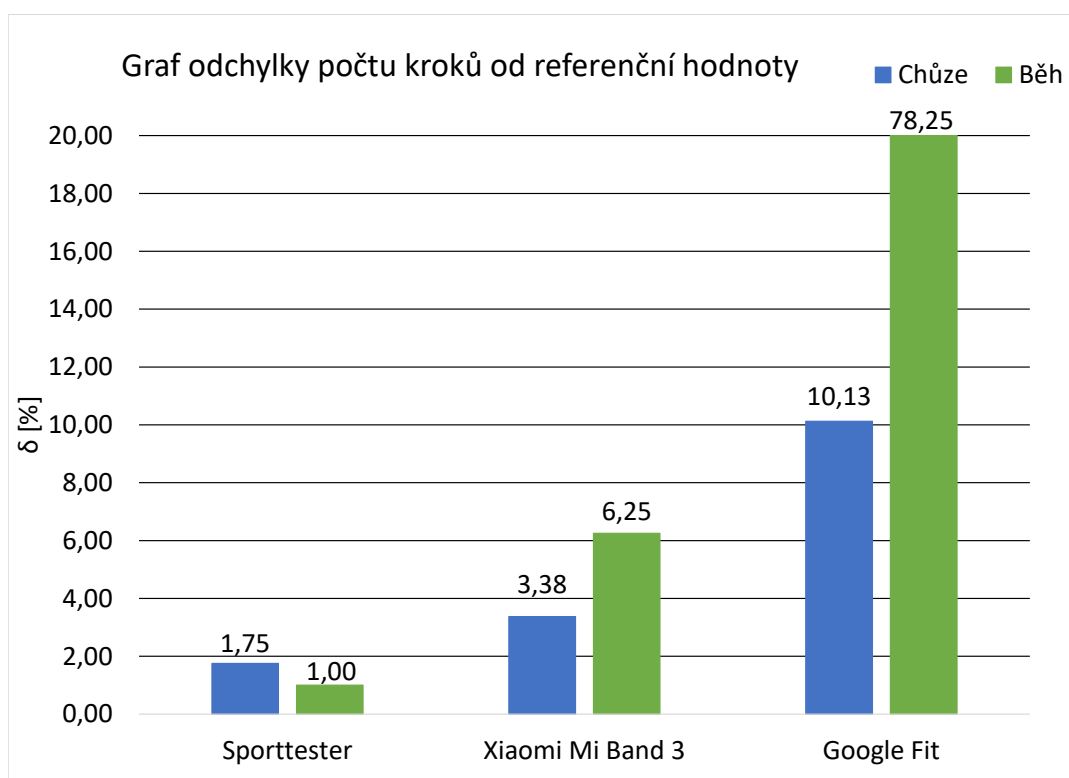
Z obrázku 5.2, který obsahuje odchylky všech zařízení, je jasně vidět, že navržený sporttester dosahuje nejmenší odchylky ze všech testovaných zařízení. Odchylka navrženého sporttesteru dosahuje hodnoty 1,75 % při chůzi a 1,00 % při běhu. Fitness náramek Xiaomi Mi Band 3 se také vyznačuje nepříliš velkou odchylkou s hodnotami 3,38 % při chůzi a 6,25 % při běhu. Mobilní aplikace Google Fit se ovšem prokázala být velice nespolehlivá. Její odchylky přesahují hraniční dovolenou hodnotu 10 %, která je uvedena v požadavcích na algoritmy počítání kroků v Android 4.4 a vyšší (viz *source.android.com*) [33]. Takto velké odchylky mohou být způsobeny nevhodným upevněním mobilního telefonu při pohybu. Během měření byl mobilní telefon umístěn v boční kapse sportovních kalhot. Aplikace ovšem nijak neupřesňuje, jakým způsobem by měl být telefon upevněn na těle běžce.

Tab. 5.2: Tabulka hodnot měření počtu kroků pro chůzi.

Název zařízení	$ch_1$	$ch_2$	$ch_3$	$ch_4$	Arit. průměr $\overline{ch}$ [krok]	Odchylka $\delta_{ch}$ [%]
Sporttester	198	198	192	198	196,50	1,75
Mi Band 3	208	211	206	202	206,75	3,38
Google Fit	168	176	201	174	179,75	10,13

Tab. 5.3: Tabulka hodnot měření počtu kroků pro běh.

Název zařízení	$b_1$	$b_2$	$b_3$	$b_4$	Arit. průměr $\bar{b}$ [krok]	Odchylka $\delta_b$ [%]
Sporttester	200	202	192	198	198,00	1,00
Mi Band 3	207	217	208	218	212,50	6,25
Google Fit	45	48	47	34	43,50	78,25



Obr. 5.2: Graf odchylek počtu kroků.

## 6 Závěr

Cílem této práce bylo navrhnout a realizovat sporttester, který je schopen komunikovat pomocí technologie Bluetooth Low Energy. BLE je potřeba, aby zařízení bylo schopné komunikovat s chytrým mobilním telefonem, na kterém je nainstalovaná kompatibilní mobilní aplikace. Ta má za účel shromažďovat a zobrazovat naměřené hodnoty uživateli.

Realizace sporttesteru je rozdělena na hardwarovou a softwarovou část. V HW části bylo potřeba vybrat především vhodný mikrokontrolér, BLE modul, akcelerometr a bateriové napájení. Byl vybrán mikrokontrolér CC2640R2F, který zároveň slouží i jako BLE modul, což částečně ulehčuje výslednou implementaci. Z řad dostupných akcelerometrů byl vybrán akcelerometr BMA423, který disponuje funkcí měření počtu kroků, díky čemuž nemusí být do mikrokontroléru posílány všechny vzorky akcelerace a až v něm počítán počet uběhnutých kroků. To velice snižuje množství posílaných dat mezi akcelerometrem a mikrokontrolérem, čímž klesá také spotřeba energie. Bateriové napájení bylo realizováno pomocí lithium-iontové baterie LIR2450 a integrovaného obvodu s označením BQ25010, který je schopen snížit a stabilizovat bateriové napětí na zvolené pracovní napětí 1,8 V. Tento integrovaný obvod zároveň umožňuje nabíjení připojené baterie.

V SW části bylo třeba vyřešit implementaci BLE komunikace a napsat program, který by předával vypočítanou kadenci kroků z akcelerometru do příslušného BLE profilu. Kompletní program, včetně BLE implementace, byl napsán v programovacím jazyce C. Komunikace mezi MCU a akcelerometrem probíhala pomocí SPI rozhraní, jehož implementace byla realizována díky dostupným ovladačům. K realizaci BLE komunikace byl využit příkladový projekt od firmy Texas Instruments. Ten však obsahuje pouze BLE profily a služby, se kterými nejsou běžné fitness mobilní aplikace kompatibilní. Byl tedy namísto nich použit profil adoptovaný Bluetooth SIG skupinou s názvem Running Speed and Cadence Profile. Tento profil umožňuje posílat do připojené mobilní aplikace údaje o kadenci uběhnutých kroků a o aktuální rychlosti. Dalším klíčovým aspektem v rámci BLE implementace bylo nastavení přenosových parametrů. Vhodné nastavení těchto parametrů závisí na BLE omezeních a na požadavcích uživatele, což je blíže popsáno v kapitole 4.2.2. Na závěr byla provedena dvě měření. První měření, s názvem Měření průběhu akcelerace při běhu, bylo zaměřené na zjištění změn akcelerace sporttesteru v průběhu jednoho kroku. To pomohlo lépe odhadnout parametry ovlivňující přesnost a spolehlivost funkce kontroly síly nárazů. Druhé měření prokázalo, že navržený sporttester je schopen velice přesně měřit počty kroků s odchylkou do 2 %.

## Literatura

- [1] HRELJAC, Alan, Robert N. MARSHALL a Patria HUME. Evaluation of lower extremity overuse injury potential in runners. *Medicine and Science in Sports and Exercise*. LWW, 2012.
- [2] SPIKER, Andrea M., Sameer DIXIT a Andrew J. COSGAREA. Triathlon: running injuries. *Sports medicine and arthroscopy review*. LWW, 2000.
- [3] SAVAGE, Jonathan. Running Sensors. *Fellrnr* [online]. USA, 1997 [cit. 2020-05-19]. Dostupné z: [https://fellrnr.com/wiki/Running\\_Sensors](https://fellrnr.com/wiki/Running_Sensors)
- [4] AFANEH, Mohammad. *Intro to Bluetooth Low Energy: The Easiest Way to Learn Ble* [online]. Amazon Digital Services LLC - KDP Print US, 2018, 136 s. [cit. 2020-05-19]. ISBN 9781790198153. Dostupné z: <https://books.google.cz/books?id=0UhjvwEACAAJ>
- [5] CZURAK, Piotr, Cezary MAJ, Michal SZERMER a Wojciech ZABIEROWSKI. Impact of Bluetooth low energy on energy consumption in Android OS. *2018 XIV-th International Conference on Perspective Technologies and Methods in MEMS Design (MEMSTECH)*. IEEE, 2018, 2018, 255-258. DOI: 10.1109/MEMSTECH.2018.8365745. ISBN 978-1-5386-5881-9. Dostupné také z: <https://ieeexplore.ieee.org/document/8365745/>
- [6] DRÁPEL, Roman. *Autentizace v IoT*. Brno, 2017. Bakalářská práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací. Vedoucí práce Petr Dzurenda.
- [7] KUBŮ, Jiří. *Elektronický zabezpečovací systém s prvky IoT*. Brno, 2017. Diplomová práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací. Vedoucí práce Ondřej Krajsa.
- [8] KAMATH, Sandeep a Joakim LINDH. Measuring bluetooth low energy power consumption. *Texas instruments application note AN092, Dallas*. 2010.
- [9] COLLOTTA, Mario, Giovanni PAU, Timothy TALTY a Ozan K. TONGUZ. Bluetooth 5: A concrete step forward toward the IoT. *IEEE Communications Magazine*. IEEE, 2018.
- [10] 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 Std 802.11-2016 (Revision of IEEE Std 802.11-2012)*. 2016, 1-3534. DOI: 10.1109/IEEESTD.2016.7786995.

- [11] BLUETOOTH SIG PROPRIETARY. *BLUETOOTH SIG. Bluetooth Specification Version 5.1*. 2019.
- [12] DECUIR, Joe. Bluetooth 4.0: low energy. *Cambridge Silicon Radio SR plc*. Cambridge, UK, 2010.
- [13] MIKE, Ryan. Bluetooth: With Low Energy Comes Low Security. In: *7th USE-NIX Workshop on Offensive Technologies* [online]. Washington, D.C.: UNISEX, 2013 [cit. 2020-05-19]. Dostupné z: <https://www.usenix.org/conference/woot13/workshop-program/presentation/Ryan>
- [14] TOWNSEND, Kevin, Carles CUFÍ, Robert DAVIDSON a AKIBA. *Getting Started with Bluetooth Low Energy: Tools and Techniques for Low-Power Networking* [online]. Přepřacované vydání. O'Reilly Media, 2014 [cit. 2020-05-19]. ISBN 149190058X, 9781491900581. Dostupné z: <https://books.google.cz/books?id=AIR7AwAAQBAJ>
- [15] Medical Devices Working Group: Hearth Rate Profile. *BLUETOOTH PROFILE SPECIFICATION* [online]. 2011 [cit. 2020-05-19]. Dostupné z: [https://www.bluetooth.org/docman/handlers/downloaddoc.ashx?doc\\_id=239865](https://www.bluetooth.org/docman/handlers/downloaddoc.ashx?doc_id=239865)
- [16] Medical Devices Working Group: Hearth Rate Service. *BLUETOOTH SERVICE SPECIFICATION* [online]. 2011 [cit. 2020-05-19]. Dostupné z: [https://www.bluetooth.org/docman/handlers/downloaddoc.ashx?doc\\_id=239866](https://www.bluetooth.org/docman/handlers/downloaddoc.ashx?doc_id=239866)
- [17] GPA Working Group: Device Information Service. *BLUETOOTH SERVICE SPECIFICATION* [online]. 2011 [cit. 2020-05-19]. Dostupné z: [https://www.bluetooth.org/docman/handlers/downloaddoc.ashx?doc\\_id=244369](https://www.bluetooth.org/docman/handlers/downloaddoc.ashx?doc_id=244369)
- [18] TEXAS INSTRUMENTS INCORPORATED. *SimpleLink Academy for SimpleLink CC2640R2 SDK*. Texas, USA. Dostupné také z: [http://dev.ti.com/tirex/explore/content/simplelink\\_academy\\_cc2640r2sdk\\_4\\_10\\_01\\_00/modules/blestack/ble\\_01\\_custom\\_profile/ble\\_01\\_custom\\_profile.html#example-service-generator](http://dev.ti.com/tirex/explore/content/simplelink_academy_cc2640r2sdk_4_10_01_00/modules/blestack/ble_01_custom_profile/ble_01_custom_profile.html#example-service-generator)
- [19] TEXAS INSTRUMENTS INCORPORATED. *CC2640R2F SimpleLink Bluetooth low energy Wireless MCU datasheet* [online]. Texas, USA, 2016. Dostupné také z: <http://www.ti.com/lit/ds/symlink/cc2640r2f.pdf>
- [20] TEXAS INSTRUMENTS INCORPORATED. *Code Composer Studio User-s Guide* [online]. Texas, USA [cit. 2020-05-19]. Dostupné z: [https://downloads.ti.com/ccs/esd/documents/users\\_guide/index.html](https://downloads.ti.com/ccs/esd/documents/users_guide/index.html)

- [21] TEXAS INSTRUMENTS INCORPORATED. *BLE-Stack User-s Guide* [online]. Texas, USA [cit. 2020-05-19]. Dostupné z: [http://software-dl.ti.com/simplelink/esd/simplelink\\_cc2640r2\\_sdk/3.20.00.21/exports/docs/blestack/ble\\_user\\_guide/html/ble-stack-3.x-guide/index.html](http://software-dl.ti.com/simplelink/esd/simplelink_cc2640r2_sdk/3.20.00.21/exports/docs/blestack/ble_user_guide/html/ble-stack-3.x-guide/index.html)
- [22] Sports and Fitness Working Group: Running Speed and Cadence Profile. *BLUETOOTH PROFILE SPECIFICATION* [online]. 2012 [cit. 2020-05-19]. Dostupné z: [https://www.bluetooth.org/docman/handlers/downloaddoc.ashx?doc\\_id=261266](https://www.bluetooth.org/docman/handlers/downloaddoc.ashx?doc_id=261266)
- [23] Sports and Fitness Working Group: Running Speed and Cadence Service. *BLUETOOTH SERVICE SPECIFICATION* [online]. 2012 [cit. 2020-05-19]. Dostupné z: [https://www.bluetooth.org/docman/handlers/downloaddoc.ashx?doc\\_id=261265](https://www.bluetooth.org/docman/handlers/downloaddoc.ashx?doc_id=261265)
- [24] TEXAS INSTRUMENTS INCORPORATED. *SimpleLink Academy for SimpleLink CC2640R2 SDK*. Texas, USA. Dostupné také z: [http://dev.ti.com/tirex/explore/content/simplelink\\_academy\\_cc2640r2sdk\\_4\\_10\\_01\\_00/modules/blestack/ble\\_01\\_custom\\_profile/ble\\_01\\_custom\\_profile.html#example-service-generator](http://dev.ti.com/tirex/explore/content/simplelink_academy_cc2640r2sdk_4_10_01_00/modules/blestack/ble_01_custom_profile/ble_01_custom_profile.html#example-service-generator)
- [25] BOSCH. *BMA423 - Data Sheet*. 2019. Dostupné také z: [https://www.bosch-sensortec.com/media/boschsensortec/downloads/motion\\_sensors\\_2/accelerometers\\_3/bma423/bst-bma423-ds000.pdf](https://www.bosch-sensortec.com/media/boschsensortec/downloads/motion_sensors_2/accelerometers_3/bma423/bst-bma423-ds000.pdf)
- [26] TEXAS INSTRUMENTS INCORPORATED. *SINGLE-CHIP CHARGER AND DC/DC CONVERTER IC FOR PORTABLE APPLICATIONS (bq2501x)* [online]. Texas, USA, 2004 [cit. 2020-05-19]. SLUS615B. Dostupné z: <https://www.ti.com/lit/ds/symlink/bq25010.pdf?&ts=1589962848931>
- [27] PREMIER FARNELL. *LIR2450 Datasheet* [online]. V1.1. 2011 [cit. 2020-05-19]. Dostupné z: <http://www.farnell.com/datasheets/1475807.pdf>
- [28] SILICON LABORATORIES INC. *CP2102/9: SINGLE-CHIP USB-TO-UART BRIDGE* [online]. Rev. 1.8. USA, 2017 [cit. 2020-05-19]. Dostupné z: <https://www.silabs.com/documents/public/data-sheets/CP2102-9.pdf>
- [29] TEXAS INSTRUMENTS INCORPORATED. *TXS0104E 4-Bit Bidirectional Voltage-Level Translator for Open-Drain and Push-Pull Applications* [online]. Texas, USA, 2006 [cit. 2020-05-19]. SCES651H. Dostupné z: <https://www.ti.com/lit/ds/symlink/txs0104e.pdf?&ts=1589966880329>



- [30] NEW AGE ENCLOSURES. *Small Utility - S-272008 Datasheet* [online]. 2004 [cit. 2020-05-19]. Dostupné z: [https://www.newageenclosures.com/images/ecommm/products/files/272008\\_r2\\_1.pdf](https://www.newageenclosures.com/images/ecommm/products/files/272008_r2_1.pdf)
- [31] SHEERIN, Kelly R., Duncan REID a Thor F. BESIER. The measurement of tibial acceleration in runners-A review of the factors that can affect tibial acceleration during running and evidence-based guidelines for its use. *Gait & Posture*. United Kingdom, 2019, **67**. DOI: 10.1016/j.gaitpost.2018.09.017. ISSN 09666362. Dostupné také z: <https://linkinghub.elsevier.com/retrieve/pii/S0966636218305290>
- [32] BLUETOOTH® SPECIFICATION. *Bluetooth Architectural Review Board: GATT Specification Supplement* [online]. 2017 [cit. 2020-05-19]. Dostupné z: [https://www.bluetooth.org/docman/handlers/downloadaddoc.ashx?doc\\_id=429632](https://www.bluetooth.org/docman/handlers/downloadaddoc.ashx?doc_id=429632)
- [33] BOSCH SENSORTEC. *BMA423 Application note - Wearable feature set* [online]. 2019 [cit. 2020-05-19]. Dostupné z: [https://www.bosch-sensortec.com/media/boschsensortec/downloads/application\\_notes\\_1/bst-mas-an032-00.pdf](https://www.bosch-sensortec.com/media/boschsensortec/downloads/application_notes_1/bst-mas-an032-00.pdf)
- [34] SLOAN, Jim. The Average Inches Per Stride in Running. *Houston Chronicle* [online]. Hearst Newspapers [cit. 2020-05-19]. Dostupné z: <https://livehealthy.chron.com/average-inches-per-stride-running-8064.html>
- [35] SMETANA, Josef. Krátký nebo delší krok? *Běhej.com* [online]. 2012 [cit. 2020-05-19]. Dostupné z: <https://www.behej.com/media/files/other/52-53.pdf>
- [36] TAKANO, Chichiro a Yuji OHTA. Heart rate measurement based on a time-lapse image. *Medical engineering and physics*. Elsevier, 2007.
- [37] TEXAS INSTRUMENTS INCORPORATED. *AFE4420 Ultra-Small, Integrated AFE With FIFO for Multi-sensor Wearable, Optical Heart-rate Monitoring and Bio-sensing* [online]. Texas, USA, 2018 [cit. 2020-05-19]. SBAS910A. Dostupné z: <https://www.ti.com/lit/ds/symlink/afe4420.pdf?ts=1590050252810>
- [38] ROHM SEMICONDUCTOR. *Optical Sensor for Heart Rate Monitor IC BH1790GLC* [online]. Rev.004. 2017 [cit. 2020-05-19]. Dostupné z: [https://fscdn.rohm.com/en/products/databook/datasheet/ic/sensor/pulse\\_wave/bh1790glc-e.pdf](https://fscdn.rohm.com/en/products/databook/datasheet/ic/sensor/pulse_wave/bh1790glc-e.pdf)

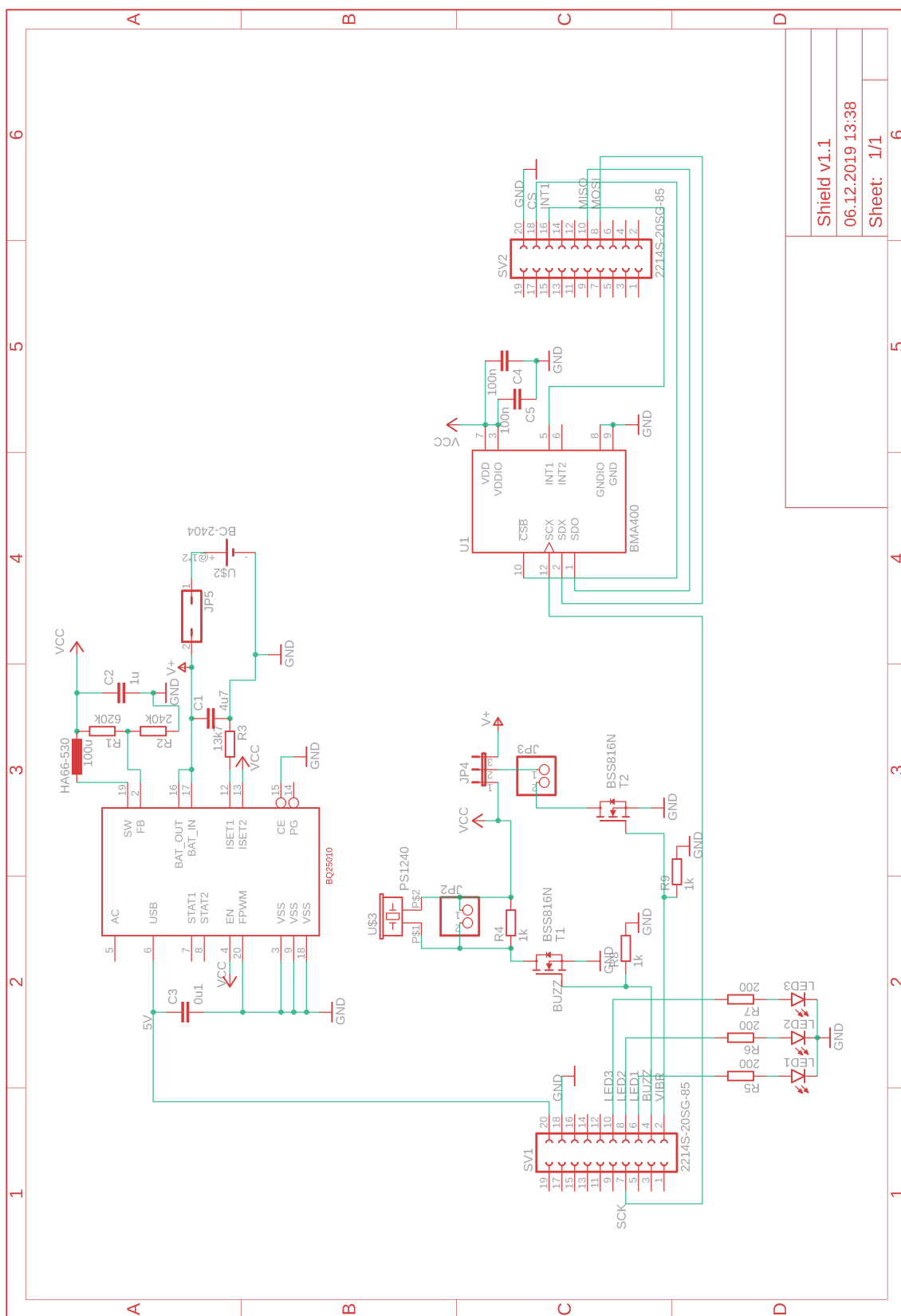
- [39] CROWELL, Harrison P. a Irene S. DAVIS. Gait retraining to reduce lower extremity loading in runners. *Clinical biomechanics*. Elsevier, 2011, 78-83.

# Seznam symbolů, veličin a zkratek

$\delta_b$	Odchylka počtu kroků pro běh
$\delta_{ch}$	Odchylka počtu kroků pro chůzi
$\bar{b}$	Průměr hodnot počtu kroků pro běh
$b_i$	Hodnoty počtu kroků pro běh
$f_{akc}$	Frekvence vyčítání vzorků akcelerace
$\overline{ch}$	Průměr hodnot počtu kroků pro chůzi
$ch_i$	Hodnoty počtu kroků pro chůzi
$K$	Kadence kroků
$t_{led}$	Doba rozsvícení LED diody
$\vec{v}$	Výsledný vektor akcelerace
<b>AES</b>	Advanced Encryption Standard
<b>ATT</b>	Attribute Protocol
<b>BLE</b>	Bluetooth Low Energy
<b>BR/EDR</b>	Basic Rate/ Enhanced Data Rate
<b>CBC-MAC</b>	Cipher Block Chaining Message Authentication Code
<b>CCM</b>	Counter with CBC-MAC
<b>CCS</b>	Code Composer Studio
<b>CPU</b>	Central Processing Unit
<b>CRC</b>	Cyclic Redundancy Check
<b>DIS</b>	Device Information Service
<b>DPS</b>	Deska Plošných Spojů
<b>ECDH</b>	Elliptic-curve Diffie-Hellman
<b>FHSS</b>	Frequency Hopping Spread Spectrum
<b>GAP</b>	Generic Access Profile
<b>GATT</b>	Generic Attribute Profile
<b>GPS</b>	Global Positioning System
<b>HCI</b>	Host Controller Interface
<b>HW</b>	Hardware
<b>I2C</b>	Inter Integrated Circuit
<b>IC</b>	Integrated Circuit
<b>IDE</b>	Integrated Development Environment
<b>IEEE</b>	Institute of Electrical and Electronics Engineers
<b>IoT</b>	Internet of Things
<b>ISM</b>	Industrial, Scientific and Medical
<b>L2CAP</b>	Logical Link Control and Adaptation Protocol
<b>LED</b>	Light Emitting Diode
<b>Li-Ion</b>	Lithium-Iont

<b>MCU</b>	Mikrokontrolér
<b>RSC</b>	Running Speed and Cadence
<b>RSCP</b>	Running Speed and Cadence Profile
<b>RSCS</b>	Running Speed and Cadence Service
<b>SDIO</b>	Secure Digital Input Output
<b>SDK</b>	Software Development Kit
<b>SIG</b>	Special Interest Group
<b>SPI</b>	Serial Peripheral Interface
<b>SW</b>	Software
<b>TI</b>	Texas Instruments
<b>TI-RTOS</b>	Texas Instruments - Real Time Operating System
<b>UART</b>	Universal Asynchronous Receiver-Transmitter
<b>USB</b>	Universal Serial Bus
<b>UUID</b>	Univerzálně Unikátní Identifikátor
<b>WiFi</b>	Wireless Fidelity

# A Schéma zapojení shieldu

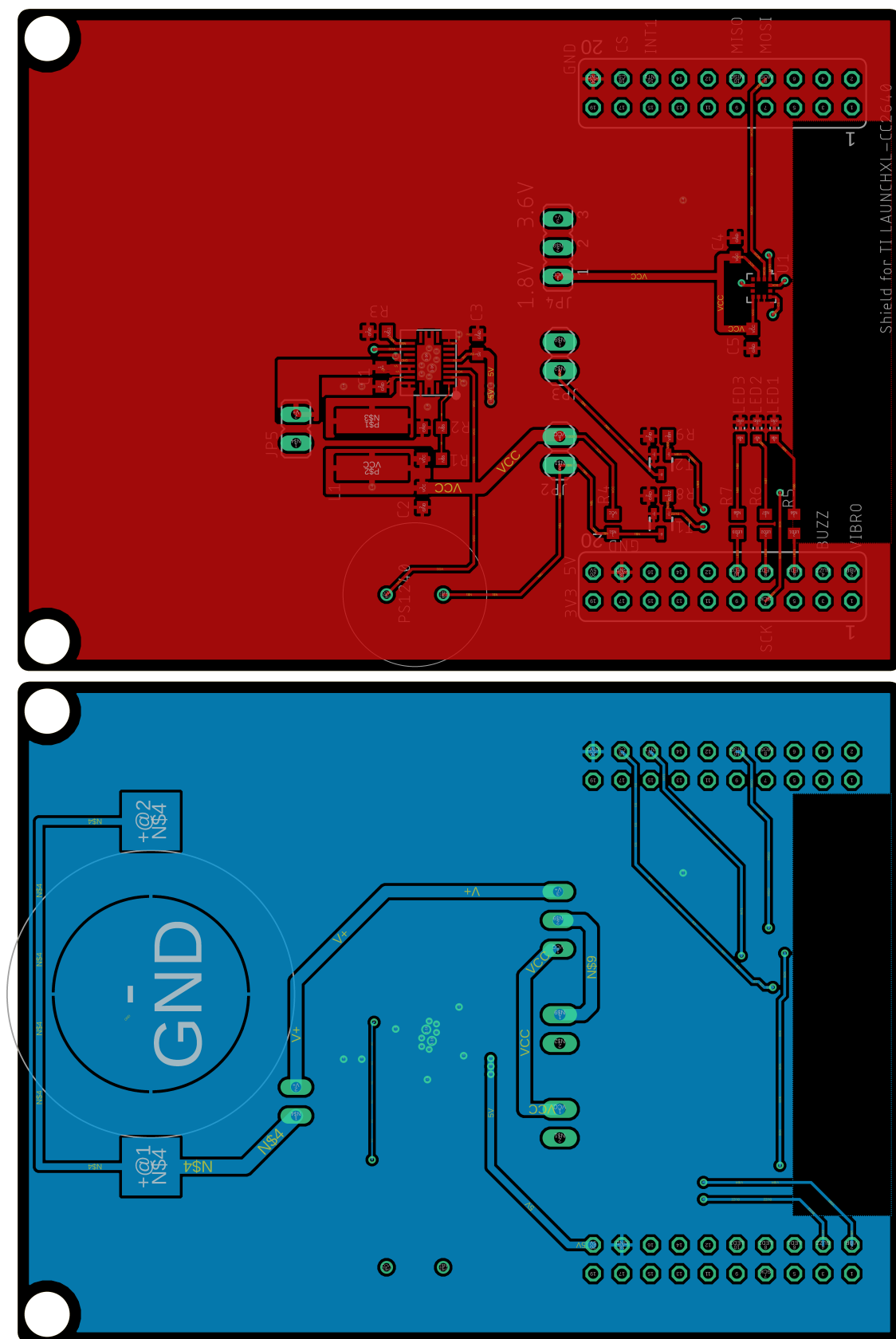


Shield v1.1

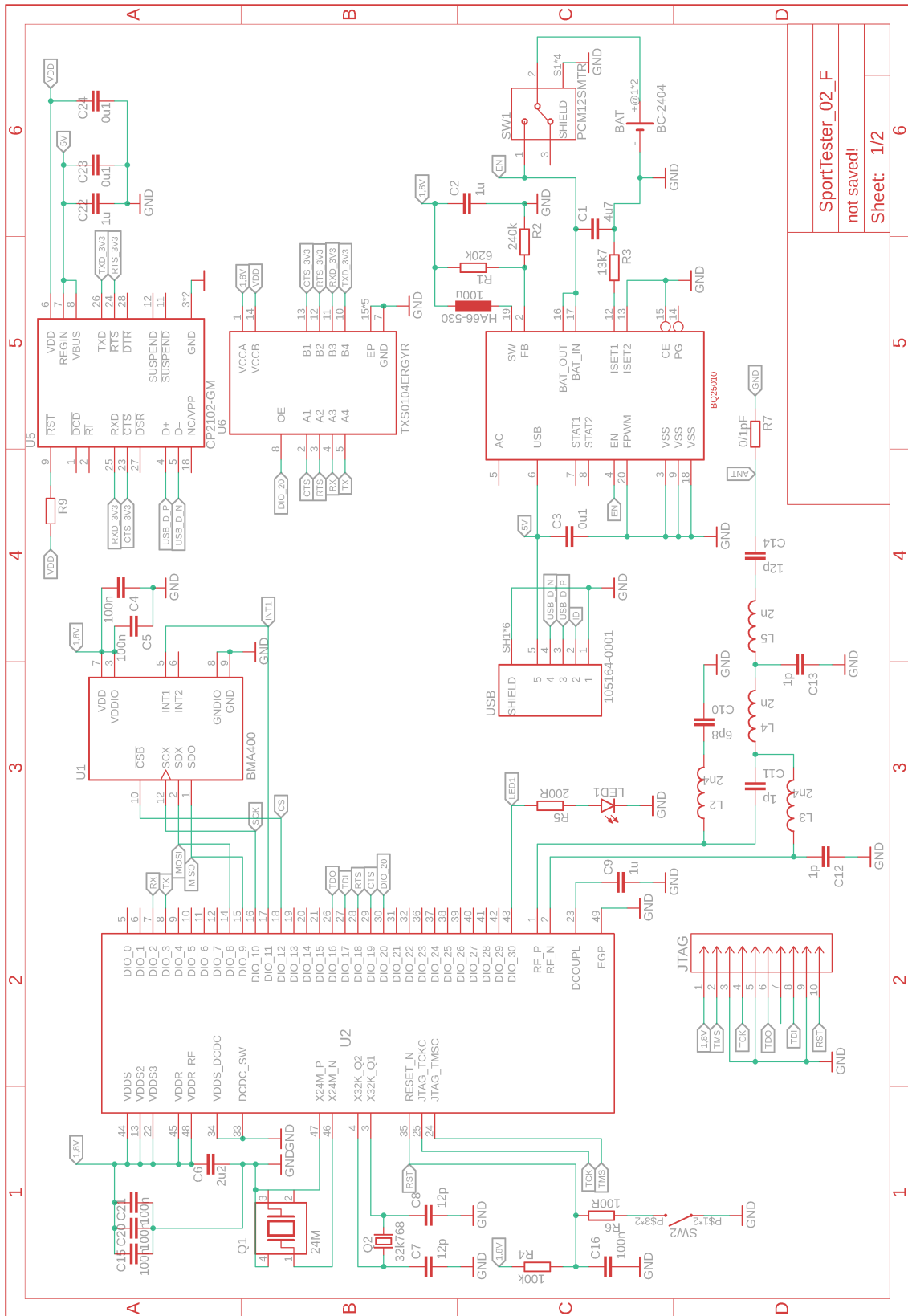
06.12.2019 13:38

Sheet: 1/1

## B Návrh DPS shieldu



# C Schéma zapojení sporttesteru



SportTester\_02\_F  
not saved!  
Sheet: 1/2

## D Návrh DPS sporttesteru

