

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ FAKULTA INFORMAČNÍCH TECHNOLOGIÍ



DOKUMENTACE K PROJEKTU DO PŘEDMĚTU SEN

Krokoměr

Bc. Lukáš Pelánek, xpelan03@stud.fit.vutbr.cz

1 Úvod

Cílem tohoto projektu bylo naimplementovat krokoměr pomocí malého jednodeskového počítače Arduino a akcelerometru MMA7455. V tomto dokumentu rozeberu zvolený algoritmus a implementaci krokoměru. V závěru rozeberu dosažené výsledky.

2 Krokoměr

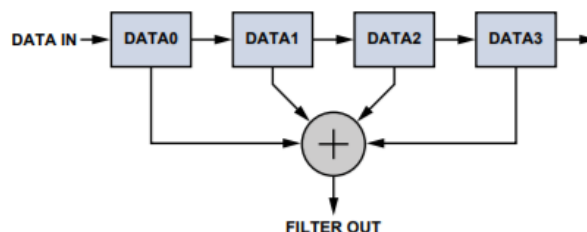
Krokoměr neboli pedometr je přenosný elektronický, elektromechanický nebo pouze mechanický přístroj, který slouží k počítání kroků. Krokoměr funguje na principu mechanického senzoru na počítání kroků. První přístroje využívaly k zaznamenání kroku mechanického zařízení. Současné přístroje využívají MEMS (Micro-Electro-Mechanical systems), které mají vnitřní senzory a software k detekci a zaznamenání počtu kroků [1].

3 Algoritmus

Pro implementaci krokoměru jsem se rozhodl zvolit algoritmus, který popisuje Neil Zhao ve své práci [2]. Tento algoritmus pracuje se všemi třemi osami, protože krokoměr se většinou nachází v neznámé poloze. Při pohybu má typicky jedna ze tří os větší periodické změny v akceleraci než ostatní bez ohledu na to, v jaké poloze se krokoměr nachází. Tento algoritmus využívá dynamický threshold pro detekci kroku. Jednotlivé části algoritmu jsou popsány v následujících kapitolách.

3.1 Digitální filtr

První částí algoritmu je digitální filtr (Obr. 1), který vstupní signál vyhladí. Skládá se ze 4 registrů, do kterých se postupně uloží 4 následující hodnoty z akcelerometru. Tyto hodnoty jsou následně sečteny.



Obrázek 1: Digitální filtr [2]

3.2 Dynamický threshold

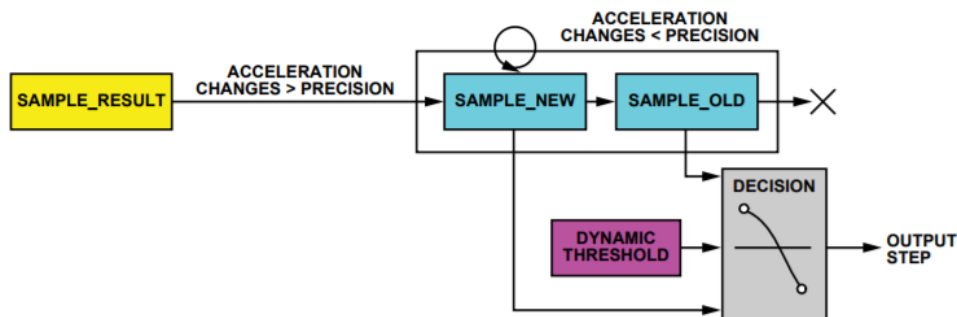
Systém periodicky aktualizuje maximální a minimální hodnoty všech tří os každých 50 vzorků. Průměrná hodnota $((\text{Max} + \text{Min}) / 2)$ se nazývá dynamický threshold. Tento threshold je použit pro následujících 50 snímků při rozhodování, zda došlo ke kroku.

3.3 Posuvné registry

Algoritmus využívá 2 posuvné registry *sample_new* a *sample_old*. Jakmile dorazí nová data na vstup, tak registr *sample_new* je bezpodmínečně přesunut do registru *sample_old*. Přesun do registru *sample_new* je podmíněný. Pokud změna v akceleraci je větší, než definovaná přesnost, tak nejnovější vstupní data jsou přesunuta do registru *sample_new*. V opačném případě zůstane registr *sample_new* nezměněný. Tyto registry slouží k odstranění vysokofrekvenčního šumu.

3.4 Detekce kroku

Krok je definován jako negativní propad v akceleraci. Pokud je hodnota registru *sample_old* větší než *threshold* a současně hodnota registru *sample_new* menší než *threshold*, tak je inkrementován čítač, který měří počet kroků. Rozhodovací logika je znázorněna na obrázku 2. Krok je vždy detekován z osy, u které došlo k největší změně v akceleraci.



Obrázek 2: Rozhodovací logika [2]

3.5 Časové okno

Algoritmus využívá tzv. časové okno. Předpokládá, že lidé se mohou pohybovat maximálně 5 kroků za vteřinu a minimálně 1 krok za 2 vteřiny. Z toho vyplývá, že interval mezi platnými kroky je $<0,2; 2>$ sekund. Všechny kroky, které jsou mimo toto časové okno, jsou zahozeny.

4 Implementace

Při implementaci jsem vycházel z algoritmu, který je popsán v kapitole 3. Pro snadnější práci s akcelerometrem jsem využil volně dostupnou knihovnu *MMA_7455.h*¹.

Prvním krokem je inicializace zařízení a akcelerometru. Akcelerometr je potřeba nejprve kalibrovat. Osy jsou kalibrovány na následující hodnoty:

- Osa x = 0
- Osa y = 0
- Osa z = 64

Následuje hlavní smyčka programu, ve které je výše popsán algoritmus implementován. Přesnost, u které dojde k přesunu do registru *sample_new* jsem zvolil 25. S touto hodnotou lze experimentovat pro optimalizaci senzitivity krokoměru. Na konci hlavní smyčky je proces na 20 ms přerušen k optimalizaci frekvence.

5 Závěr

V tomto projektu jsem implementoval krokoměr pomocí malého jednodeskového počítače Arduino a akcelerometru MMA7455. Při testování jsem ověřil funkčnost krokoměru, která splňuje všechny body zadání. Pro optimalizaci senzitivity krokoměru lze manipulovat s definovanou přesností, která se mi subjektivně jevila jako nejlepší při hodnotě 25.

¹ <https://code.google.com/archive/p/mma-7455-arduino-library>

6 Literatura

[1] Krokomeř [online]. [cit. 2017-11-22]. Dostupné z: <https://cs.wikipedia.org/wiki/Krokomeř>

[2] Neil Zhao - Full-Featured Pedometer Design Realized with 3-Axis Digital Accelerometer, Analog Dialogue 44-06, June 2010