Fitness systém založený na IoT

Analýza a návrh riešenia

Bc. Viktor Pristaš

1Im 2019 – 2020

Abstrakt. V dnešnej dobe je ťažké nájsť oblasť, ktorej by sa nedotkli inteligentné systémy alebo IoT. Jednou z nich je posilňovňa, v ktorej nájdeme málo inteligentných zariadení. Cieľom práce je navrhnúť systém ktorý je schopný automaticky zaznamenávať cvičenie na niektorých vybraných strojoch pomocou jednoduchých mikrokontrolérov a senzorov.

Kľúčové slová: mikrokontrolér, internet vecí, posilňovanie, Arduino

1. Úvod

Internet vecí (alebo IoT) je dnes bez pochýb jedna z najrýchlejšie vyvíjajúcich sa vetiev. Niet divu, veď ponúka nekonečné možnosti ako zjednodušiť, zrýchliť alebo zefektívniť naše každodenné činnosti. S IoT (a s často k nemu pridruženým prívlastkom „smart“) sa bežne stretávame na rôznych miestach akými sú napríklad domácnosti (osvetlenie, biela technika, vysávač), mestá (doprava, odvoz odpadu), poľnohospodárstvo (zavlažovanie, postrekovanie) alebo energetika (spotreba plynu, elektriny, vody).

Najväčšou výhodou niektorých týchto systémov je, že dokážu vykonávať isté (hlavne manuálne) činnosti bez interakcie človeka. Kým pri klasickom vodomere je potrebné vyslať v pravidelných intervaloch pracovníka vodárne, aby odčítal spotrebu vody, inteligentné vodomery dokážu automaticky namerať a odoslať namerané údaje do centrálnej databázy. Navyše, pri tomto prístupe sa značne zníži aj pravdepodobnosť chyby.

Jedno z odvetví, ktoré ešte neprešlo touto revolúciou „zinteligentňovania“ sú fitness centrá. Sú síce dostupné fitness appky, no takmer všetky slúžia na sledovanie kardio cvičení pomocou senzorov v smartfónoch a smart hodinkách ako beh, cyklistika, plávanie a pod. Tieto aplikácie sú schopné uložiť dáta o vykonanej aktivite. Okrem toho umožňujú

používateľovi sledovať jeho osobné pokroky a tým ho motivovať. Ďalšie aplikácie, ktoré môžeme nájsť pre smartfóny, slúžia na uchovanie manuálne zapísaných údajov o cvičení.

Ako už bolo spomenuté vyššie, cieľom IoT systémov a zariadení je samostatné a automatické vykonávanie istých činností. To znamená, že užívateľ by sa nemal starať o to, aby počítal a zaznamenával počet opakovaní a použitú váhu (ako ani pri behaní nepočíta svoje kroky a nemeria svoj čas), ale toto všetko má na starosti samotný systém. V práci sa zameriame na získanie práve týchto dvoch údajov – použitá váha a počet opakovaní – keďže práve tieto dáta obsahuje (samozrejme okrem samotného cviku) každý poctivý denník cvičiaceho.

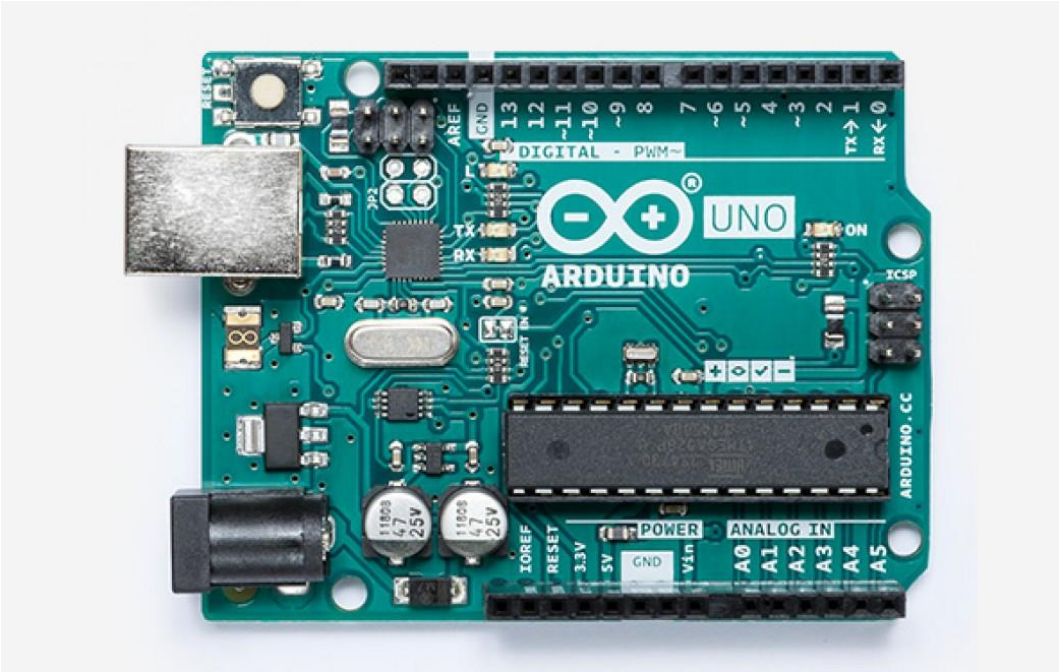
2. Mikrokontrolér a senzory

Základnou riadiacou jednotkou systémov IoT (ako aj mnohých iných zariadení) je malá riadiaca jednotka nazývaná mikrokontrolér. Mikrokontroléry majú integrované v jadre všetky základné prvky, ktoré mikroprocesor potrebuje k svojej činnosti:

- výpočtové jadro
- pamäť RAM
- pamäť s programom (ROM, EPROM, FLASH)
- časovače
- vstupno-výstupné porty (piny)

Výkon týchto zariadení je oproti bežným stolovým počítačom značne nižší, bežne obsahujú procesor s taktovacou frekvenciou na úrovni 8 MHz a operačnú pamäť okolo 32 kB. Na druhej strane, týmto parametrom zodpovedá aj ich cena, ktorá sa pohybuje v intervale 2 – 25 eúr.

Najznámejším výrobcom vývojových dosiek je bezpochybne spoločnosť Arduino. Vďaka ich open-hardware filozofií (všetky technické údaje mikrokontroléra sú voľne dostupné pre kohokoľvek) sa vyrábajú a vyvíjajú mnohé napodobeniny, vďaka čomu klesla ich cena na spomínanú hladinu.



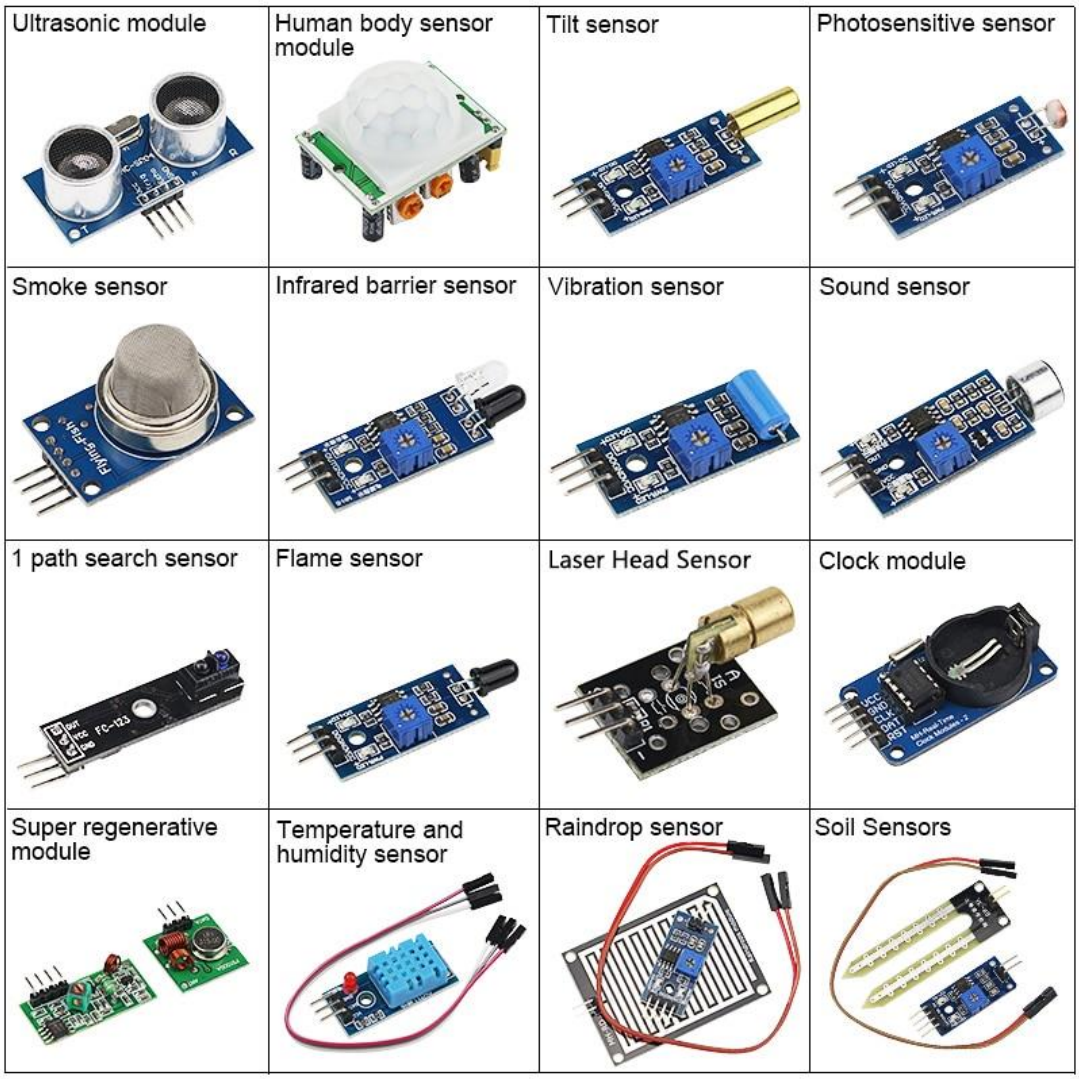
Obrázok 1 Najznámejšia doska od spoločnosti Arduino, model Uno

Väčšina dosiek Arduino používa mikročipy ATmega z rodiny 8-bitových AVR mikročipov postavených na základe RISC architektúry.

Na programovanie Arduino dosiek sa štandardne používa jazyk Arduino Programming Language, ktorého základom je jazyk C++. Je výhodné ho použiť, ak programátor nechce upravovať základné konfiguračné nastavenia, v opačnom prípade je možné použiť čistý C++. Takto získa programátor väčšiu kontrolu nad zariadením, a spravidla pamäťovo menej náročný kód za cenu ťažšieho programovania. Okrem jazyka Arduino Programming Language ponúka spoločnosť na prácu s (nie len Arduino) doskami inteligentné vývojové prostredie nazývané Arduino IDE.

Dosky Arduino (a aj iné vývojové dosky a mikrokontroléry) sú ľahko využiteľné na čítanie senzormi nameraných údajov. Senzory sú súčiastky schopné merať nejakú fyzikálnu veličinu a pretransformovať tieto dáta na elektrický signál, ktorý je možné prenášať a spracovať.

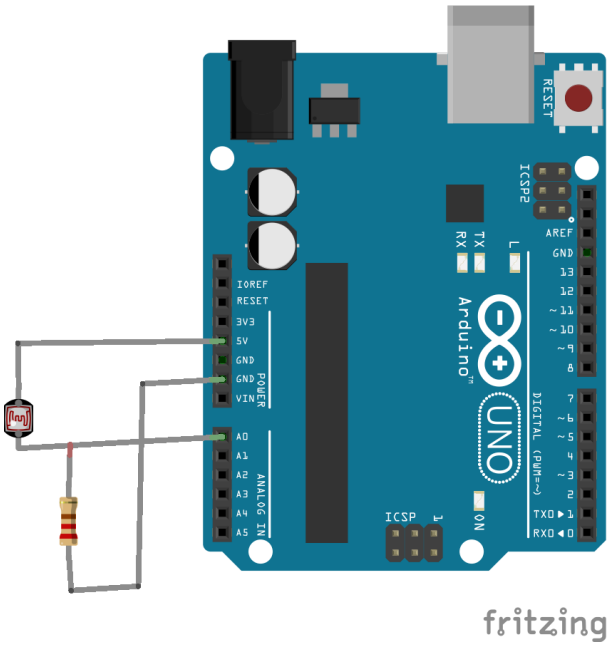
Na nasledujúcom obrázku sú znázornené najčastejšie používané senzory s Arduino doskami.



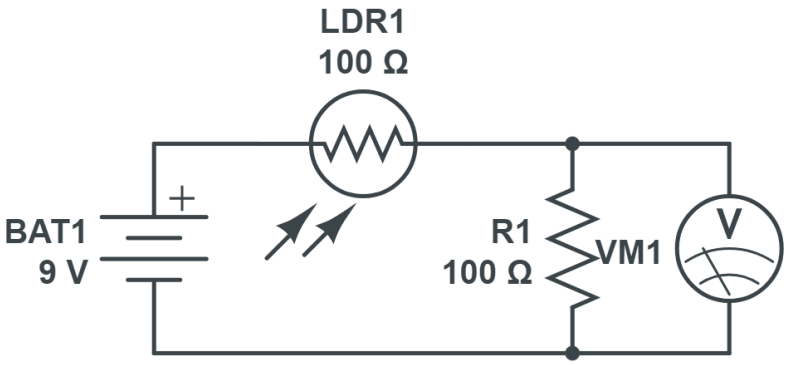
Obrázok 2 Niekoľko najčastejšie používaných senzorov

Ako bolo už vyššie spomenuté, mikrokontroléry sú vybavené vstupno-výstupnými pinmi (angl. general-purpose input/output, skratka GPIO). Tieto porty nemajú vopred definovanú funkcionalitu, je na programátorovi aby ich nakonfiguroval vzhľadom na použitie (prípadne prenechal konfiguráciu na vývojové prostredie). Ak je pin nastavený na vstupný, je schopný merať veľkosť napätia v obvode, v prípade analógového pinu jeho presnú hodnotu a v prípade digitálneho len jeho prítomnosť. Pin konfigurovaný ako výstupný dokáže generovať napätie 0 V alebo 5 V (pri istých doskách 3,3 V) ktoré zodpovedajú digitálnym hodnotám 0 a 1.

Ako ilustračný príklad je vhodné použiť jeden z najjednoduchších používaných senzorov – fotorezistor.



Obrázok 3 Zapojenie fotorezistoru k Arduinu



Obrázok 4 Schéma zapojenia fotorezistoru do obvodu

Na týchto schémach zapojenia vidno, ako sa zapája fotorezistor k analógovému pinu A0 mikrokontroléra. Mikrokontrolér meria v skutočnosti napätie medzi pinom A0 a pinom GND nad rezistorom R1 ako to vidno na pravom obrázku. Rezistor R1 je v zapojení potrebný, pretože základnou vlastnosťou voltmetra je, že sa zapája do obvodu paralelne. Meniacou sa intenzitou svetla sa mení aj napätie nad rezistorom R1 a teda nameria mikrokontrolér rôzne hodnoty.

3. Implementácia

Na zaznamenanie cvičenia, ako bolo spomenuté skôr, je potrebné sledovať okrem toho aký cvik sa cvičil dva základné atribúty:

- koľko opakovaní sa vykonalo
- akou váhou sa cvičilo

V práci sa zameriame na tzv. posilňovacie veže vybavené závažovými tehličkami, ktoré sa dajú jednoducho pripnúť pomocou kolíka.

Obrázok 5 Posilňovacia veža

3.1 Počítanie opakovaní

Počet opakovaní je z implementačného hľadiska počet koľkokrát sa tehličky zdvihli smerom hore a vrátili zas naspäť. Ich smer pohybu je jednoduchý: vertikálne smerom hore, následne vertikálne smerom dole. Počas analýzy sme skúmali niekoľko možných ciest ako detegovať tento pohyb a snažili sme sa nájsť čím presnejšiu no najmenej poruchovú.

3.1.1 Akcelerometer

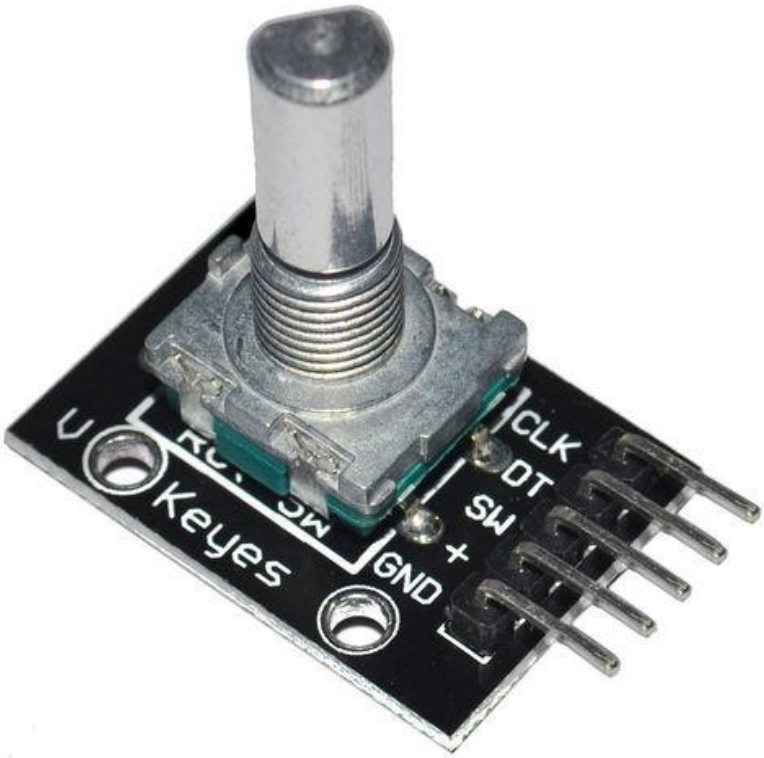
Z pohľadu senzorov je takýto pohyb najjednoduchšie merateľný akcelerometrom. Akcelerometer umiestnený na vrchu závaží je schopný detegovať smer pohybu a tým aj spočítať počet dvihnutí a položení. Okrem toho dokáže zmerať aj zrýchlenie, čo sa javí ako zaujímavý údaj na neskoršiu analýzu cvičenia (dynamika pri opakovaniach), ale ak nie je pohyb dostatočne plynulý alebo je príliš rovnomerný, môžu vzniknúť značné skreslenia a nepresné hodnoty.

3.1.2 Ultrazvukový senzor vzdialenosti

Iný pohľad na tento pohyb je, že tehličky sa každým dvihnutím priblížia ku vrchu posilňovacej veže a naopak pri položení sa oddialia. Toto pozorovanie evokuje možnosť použitia ultrazvukového senzoru vzdialenosti. Ten, umiestnený presne nad závažiami by meral vzdialenosť od vrchnej tehličky. Nevhodnosť tejto myšlienky spočíva v nepresnosti týchto senzorov pri prekročení meranej vzdialenosti niekoľkých desiatok centimetrov.

3.1.3 Rotačný enkodér

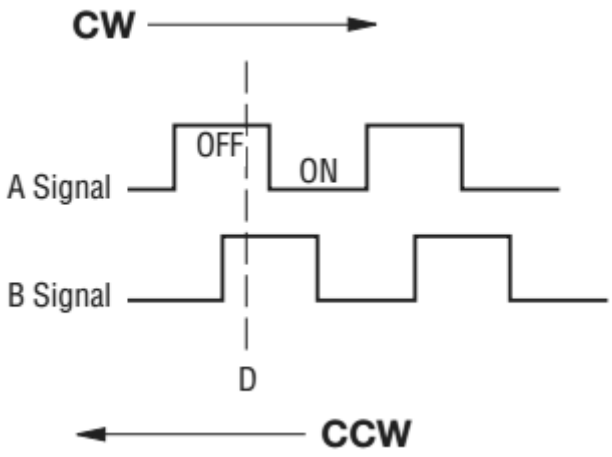
Ďalšou skúmanou možnosťou je sledovať otáčanie kladky, na ktorej sú zavesené závažia. Smer pohybu závaží je jednoznačne identifikovateľný smerom otáčania. Na zistenie otáčania slúži napríklad súčiastka zvaná rotačný enkodér (rotačný snímač, angl. rotary encoder). Je to súčiastka, ktorá umožňuje zistiť polohu pri otáčaní, stretnúť sa s ňou môžeme pri autorádiách alebo pri guľôčkových myšiach. Nemá krajnú polohu, to znamená, že je ním možné otáčať do oboch strán ľubovoľne veľakrát.



Obrázok 6 Rotačný enkodér

Pri otáčaní enkodéra dochádza k pravidelnej zmene dvoch digitálnych signálov, ktoré sú voči sebe mierne posunuté. Pri otáčaní do smeru hodinových ručičiek je možné

namerať na kanáloch (A,B) hodnoty v nasledujúcom poradí: (0, 0), (1, 0), (1, 1), (0, 1), (0, 0)... Analogicky budú tieto hodnoty pri opačnom smere v opačnom poradí. Týmto spôsobom sa dá ľahko detegovať nielen otáčanie ale taktiež jeho smer.

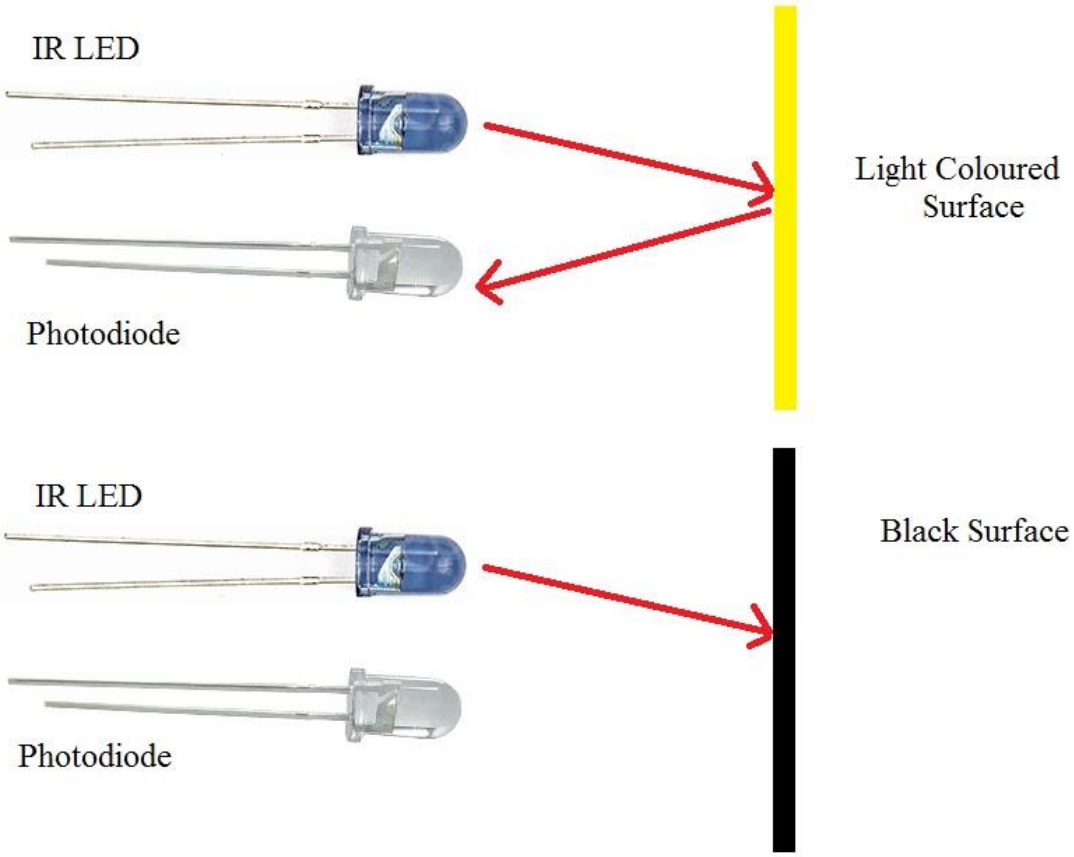


Obrázok 7 Fungovanie rotačného enkodéra

Problém pri tomto riešení spočíva v tom, že táto súčiastka musí byť v priamom kontakte s mechanickými, rotujúcimi časťami stroja, čo môže ľahko viesť k ich rýchlemu opotrebeniu alebo kazeniu. Hardvérová časť tohto systému musí byť odolná voči silným otrasom, aby sa minimalizoval počet potenciálne poruchových miest, čo ale toto riešenie nespĺňa.

3.1.4 Optický rotačný enkodér

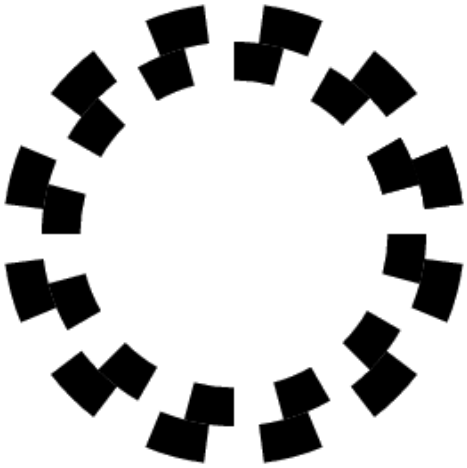
Optický rotačný enkodér funguje podobne ako vyššie predstavený mechanický, pričom pri jeho otáčaní nedochádza k fyzickému kontaktu so súčiastkou. Jeho hlavným hardvérovým komponentom je senzor infračerveného svetla, aký sa často používa v robotike ako senzor sledovania čiary. Princíp jeho fungovania spočíva v tom, že senzor vyžaruje infračervené svetlo a meria, koľko svetla sa odráža späť.



Obrázok 8 Infračervený senzor

Čierna farba (ako je dobre známe) pohlcuje svetlo, teda sa od neho odrazí oveľa menej ako od povrchov inej farby. Senzor musí byť umiestnený na cca. 1-3 cm od snímanej plochy a v závislosti od množstva odrazeného svetla vráti hodnotu v rozmedzí 0-1023 (ako je to obvyklé pri analógových senzoroch Arduina).

V praxi je potrebné nasledovať fungovanie mechanického rotačného enkodéra, t. j. vytvoriť dva signály, ktoré budú mierne posunuté. Toto vieme zabezpečiť tým, že dva takéto senzory budú snímať vytvorenú čierno bielu vzorkovnicu v tvare kruhu.



Obrázok 9 Vzorkovnica v tvare kruhu

Jeden zo senzorov bude snímať vonkajšie sekvenciu čierno bielych políčok, pričom ten druhý tú vnútornú. Nie je ťažké vidieť, že signál (po prevedení analógových hodnôt na digitálne) generovaný týmito senzormi sa zhoduje so signálom aký bol na obrázku 7.

Táto vzorkovnica umiestnená na kladke spolu so senzormi vo vhodnej vzdialenosti značne vylepšuje predošlé riešenie, keďže pri ňom nedochádza k otáčaniu/pohybu senzora a tak je odolnejší voči fyzickému poškodeniu.

3.2 Počítanie váhy

Druhým základným problémom v tejto práci je zistenie váhy akou sa vykonali opakovania. Meranie samotnej hmotnosti je v tomto prípade takmer nemožné, keďže váhy ako senzory pre mikrokontroléry sa nepoužívajú. Riešením by mohlo byť použitie silomera, ktorý by mohol do istej miery nahradiť funkciu váhy, no má značné obmedzenia. Po prvé je náročné jeho umiestnenie na vhodné miesto, aby nebránilo cvičiacemu pri manipulovaní so závažiami, po druhé, tieto prístroje potrebujú pravidelnú kalibráciu. Aj pri bežnom cvičení sa používajú závažia v rozmedzí cca. 10 – 120 kg, čo by ľahko mohlo spôsobiť rozladenie senzora, navyše, keďže ide o relatívne veľké hmotnosti, jeho životnosť (pri zachovaní priaznivej ceny, čo je tiež jeden z cieľov tohto systému) výrazne klesá.

Ak sa pozrieme na problém z inej stránky, to, čo v skutočnosti potrebujeme detegovať, je miesto, kde sa oddelili od seba tehličky. Teda ak zistíme, že k oddeleniu

došlo medzi tehličkami *i* a *i+1*, to znamená, že sa cvičí s prvými i tehličkami. Poznajúc ich nemenné hmotnosti sa jednoducho vypočíta ich celková hmotnosť.

3.2.1 Ohmov zákon

Ohmov zákon definuje vzťah medzi elektrickým prúdom, elektrickým odporom a elektrickým napätím. Jeho definícia znie nasledovne:

Elektrický prúd pretekajúci v uzavretom elektrickom obvode je priamo úmerný napätiu zdroja a nepriamo úmerný elektrickému odporu obvodu.

V matematickom zápise:

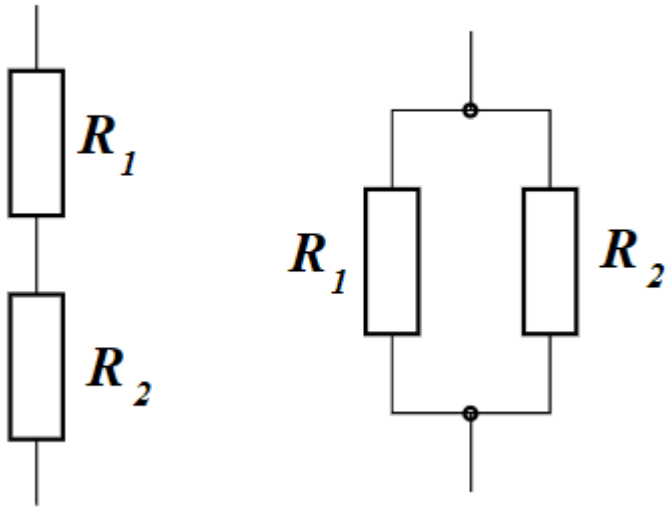
$$I = \frac{U}{R}$$

kde I je veľkosť prúdu, U veľkosť napätia a R veľkosť rezistencie.

Laickejšou rečou by sme mohli jednoducho povedať, že pri konštantnom napätí, s rastúcim odporom klesá veľkosť prúdu v obvode. Teda využijúc schopnosť Arduina merať elektrické napätie vieme zistiť veľkosť (a tým aj zmenu) odporu ktorá je obvode (keď veľkosť vychádzajúceho prúdu a tiež napätia z jeho pinov je konštantná).

3.2.2 Paralelné a sériové zapojenie odporov v obvode

Paralelné zapojenie je zapojenie odporov (alebo iných súčiastok) v elektrickom obvode pod sebou. Na rozdiel od sériového zapojenia, pri ktorom sú súčiastky zapojené v jednom rade za sebou, paralelný obvod obsahuje uzly, v ktorých sa vodiče vetvia a odpory sú umiestnené v rôznych vetvách.



Obrázok 10 Sériové a paralelné zapojenie odporov

Výsledný odpor sériovo zapojených odporov R_1, ... R_n je rovný súčtu ich odporov, teda:

$$R = \sum_{i=1}^{n} R_i = R_1 + R_2 + \cdots + R_n$$

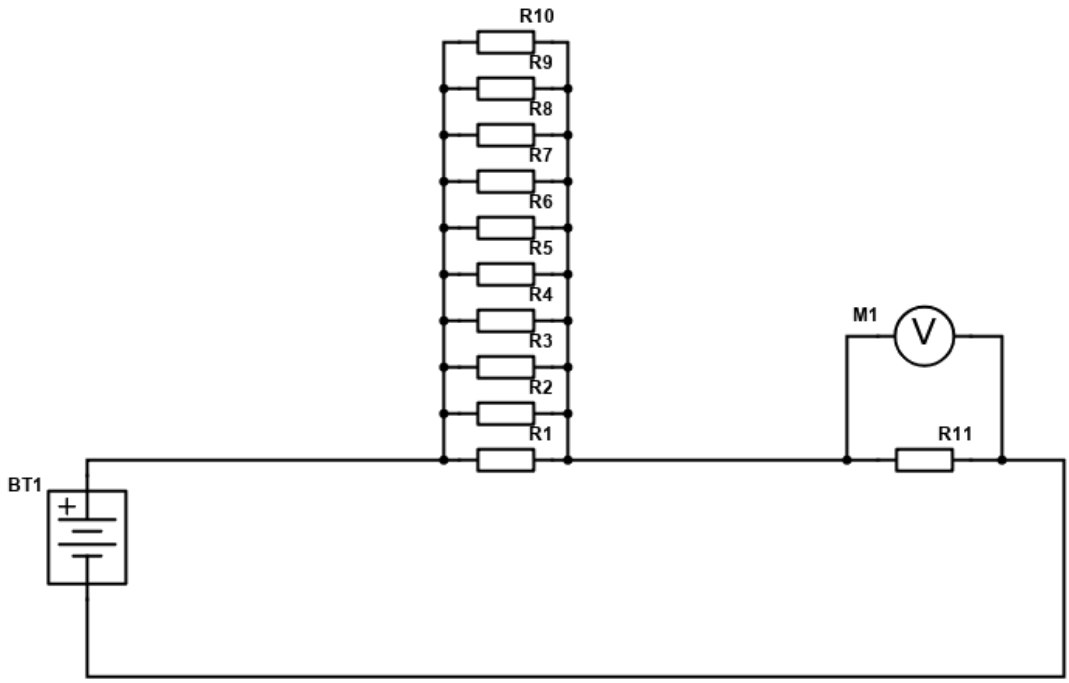
Z Ohmovho zákona je možné odvodiť aj súčet paralelne zapojených odporov R_1, ... R_n pre ktorý platí, že prevrátená hodnota výsledného odporu je daná ako súčet prevrátených hodnôt jednotlivých odporov:

$$\frac{1}{R} = \sum_{i=1}^{n} \frac{1}{R_1} + \frac{1}{R_2} + \cdots + \frac{1}{R_n}$$

Výsledný odpor pri paralelnom zapojení je teda stále menší ako odpor ktorejkoľvek vetvy.

3.2.3 Spočítanie tehličiek pomocou rezistorov

Znalosti o Ohmovom zákone a výpočte výsledného odporu paralelných odporov využijeme na zistenie počtu dvíhaných tehličiek nasledovne. Na každú tehličku pripojíme rezistor nejakej veľkosti, pričom ich zapojíme do obvodu paralelne. Schému zapojenia pre 10 závažových tehličiek zobrazuje nasledujúci obrázok:



Obrázok 11 Schéma zapojenia odporov

Odpory musia byť zapojené tak, aby v prípade, keď sú tehličky položené na mieste, existovalo medzi rezistormi elektrické spojenie. Naopak ak je horných *n* tehličiek vo vzduchu, teda sa nimi cvičí, tak sú v obvode zapojené len všetky ostatné rezistory, v modelovom príklade spodných *10 – n* odporov.

Ak sú teda známe veľkosti rezistorov, je možné podľa predošlého vzorca vypočítať ich výsledný odpor. Voltmeter M1 znázornený na obrázku 11 je v skutočnosti nami používaný mikrokontrolér, ktorý ako už bolo spomenuté vyššie je schopný merať vchádzajúce napätie. Odpojením niekoľkých odporov z vrchu zapojenia sa zmení výsledný odpor, a tým takisto aj napätie merané mikrokontrolérom.

3.2.3.1 Rozloženie odporov

Na to aby sme vedeli zistiť presný počet odporov je potrebné poznať ich presné hodnoty. V praxi sa budeme snažiť aby postupným odoberaním odporov od vrchu klesal ich výsledný odpor čo najlineárnejšie.

Zaveďme si označenie pre modelový príklad s 10 paralelne zapojenými rezistormi $R_1, \dots, R_{10}$ ako je to vidno na obrázku. Nech R_i' je hodnota výsledného odporu spomínaných 10 odporov, ak sú v obvode postupne zdola zapojené odpory $R_1, R_2, \dots, R_{10-i}$.

- $\frac{1}{R_0'} = \frac{1}{R_1} + \frac{1}{R_2} + \cdots + \frac{1}{R_{10}}$
- $\frac{1}{R_1'} = \frac{1}{R_1} + \frac{1}{R_2} + \cdots + \frac{1}{R_9}$
- $\frac{1}{R_2'} = \frac{1}{R_1} + \frac{1}{R_2} + \cdots + \frac{1}{R_8}$
- ...

Nech pre $i \in \{2, 3, \dots, 10\}$ označuje premenná R_i^k rozdiel $R_i^k - R_{i+1}^k$, kde k je kód konfigurácie odporov. Pre všetky rôzne konfigurácie k nájdeme najmenšiu hodnotu zo všetkých R_i^k a označíme ju R_{min}^k. Cieľom je nájsť konfiguráciu s maximálnou hodnotou pre R_{min}^k, keďže v tej konfigurácií majú hodnoty R_i^k najmenší rozptyl.

3. Záver

V článku sme popísali návrh riešenia hardvérovej časti pre systém ktorý je schopný zaznamenávať cvičenie v posilňovni. Zamerali sme sa na konkrétny druh strojov, ktorými sú posilňovacie veže so závažovými tehličkami, na ktorých sledujeme počet vykonaných opakovaní a váhu ktorá bola dvíhaná. Hardvér sme navrhli tak, aby bol odolný voči poškodeniu spôsobeného otrasmi alebo nevhodnou manipuláciou so strojom, pričom sme použili jednoduché senzory, elektrické súčiastky a mikrokontrolér.

V ďalšej časti sa budeme zaoberať vývojom mobilnej aplikácie a taktiež backendu, pomocou čoho bude možné automaticky ukladať údaje o vykonanom cvičení do databázy a sledovať históriu. Systém by mal do plnohodnotne nahradiť denníky, ktoré si cvičiaci vedú a navyše bude obohatený o rôzne štatistické ukazovatele pre lepší prehľad a orientáciu. Je navrhnutý dostatočne všeobecne, a je možné z neho vydolovať aj dodatočné dáta ohľadom dynamiky alebo správnosti vykonania cviku.

Literatúra

1. Binbin Yong et al., IoT-based intelligent fitness system, Journal of Parallel and Distributed Computing, Volume 118, Part 1, 2018, pp 14-21.
2. A. Jain, A Smart Gym Framework: Theoretical Approach, 2015 IEEE International Symposium on Nanoelectronic and Information Systems, Indore, 2015, pp. 191-196.
3. L. Mhatre and N. Rai, Integration between wireless sensor and cloud, 2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), Palladam, 2017, pp. 779-782.
4. S. Fitzgerald, M. Shiloh, T. Igoe, Arduino Projects Book, Arduino LLC, 2012
5. Radovici, A.; Culic, I.; Vaduva, A., „Lecture 1: Introduction to IoT," [Online]. Available: https://ocw.cs.pub.ro/courses/iot/courses/01.
6. Arduino, „Arduino Uno Rev3" [Online]. Available: https://store.arduino.cc/arduino-uno-rev3.