

Zadání maturitní práce s obhajobou

**Číslo a název**:08 -fitness tracker

**Jméno žáka**: Karel Beran

**Konzultant**: Ing. Luděk Fedurca

**Opponent**: Bc. Jakub Malý

**Datum zadání**:2. prosinec 2022

**Datum odevzdání**:1. dubna 2023

**Doba obhajoby**:15 minut

**Zadání:**  
 Navrhněte a implementujte na ESP32 jednoduchý fitness tracker – počítadlo dřepů na základě

dat z integrovaného barometru. Zařízení bude obsahovat kalibrační rutinu pro nastavení horní a

dolní úrovně cviku a přehled výkonu z posledních 10 cvičení.

**Způsob zpracování:**

* **Tištěná forma**: rozsah dokumentace 15 – 20 stran textu; v obálce s chlopněmi nebo pevná vazba; součástí práce bude úvodní obálka, zadání práce, harmonogram a prohlášení o souhlasu se zadáním práce, samostatnosti zpracování práce a použitím legálního software
* **Digitální forma**: kopie práce a pracovní soubory, dokumentace, prezentace na přiloženém CD nebo DVD v papírové obálce s jednoduchým HTML rozcestníkem; soubory v alternativních formátech
* **Model projektu**: vytvořte funkční model, který bude simulovat zadání

**Počet vyhotovení:** 1

**Formální úprava práce:**  
 **Písmo:** velikost 12  
 **Font:** Calibri, Arial nebo Times New Roman (zvolený font dodržte v celé práci)  
 **Řádkování:** 1,5  
 **Vzdálenost mezi odstavci:** 6 b.  
 **Okraje:** horní a dolní 25 mm, levý (vnitřní) 40 mm, pravý (vnější) 20 mm  
 **Zarovnání odstavce:** do bloku  
 **Číslování stránek:** vpravo dolu  
 **Začátek hlavní kapitoly:** vždy na nové straně  
 Dodržení typografických pravidel hladké sazby

**Hodnocení:**

1. Splnění zadání
2. Plnění plánu práce a účast na konzultacích
3. Aktuálnost a přínosnost tématu
4. Odborná úroveň práce, kvalita zpracování práce, použité prostředky
5. Zpracování dokumentace – typografie, zdroje, struktura, rozsah…
6. Dodržení ČSN ISO 690 a ČSN ISO 690-2 – bibliografické citace dokumentů
7. Hodnocení modelu a prezentace

**Podpis žáka**  …………………………………………………………………………….

**Podpis konzultanta** …………………………………………………………………………….

**Podpis ředitele školy** …………………………………………………………………………….

| **Harmonogram práce MZ** |
| --- |

| **Třída:**  4. I |
| --- |
| **Studijní obor:** 18-20-M/01 Informační technologie |
| **Jméno studenta:** Karel Beran |
| **Konzultant:** Ing.Luděk Fedurca |
| **Číslo a název úlohy:** 08 -fitness tracker |

| **Plán práce** | |
| --- | --- |
| **Týden** | **Práce** |
| 16.1.-21.1 2023 | osnova závěrečné práce, náplň kapitol prezentace prototypu |
| 6.2.-10.2 2023 | prezentace funkčnosti minimálního prototypu, 50% dokumentace |
| 27.2.-3.3. 2023 | cvičná obhajoba, finální text k revizi |

| **Ve Štětí dne ………………………………………** |
| --- |
| **Podpis ………………………………………** |
|  |

| **Kontrola plnění plánu práce** | | |
| --- | --- | --- |
| **Datum** | **Poznámky** | **Podpis** |
|  |  |  |
|  |  |  |
|  |  |  |

**Prohlášení**

**Třída:** 4. I

**Studijní obor:** 18-20-M/01 Informační technologie

**Jméno žáka:** Karel Beran

**Konzultant:** Ing. Luděk Fedurca

**Číslo a název úlohy:** 08 -fitness tracker

***Čestné prohlášení o souhlasu se zadáním maturitní práce***

Prohlašuji, že jsem se seznámil s obsahem zadání maturitní práce. Souhlasím se zadaným tématem.[[1]](#footnote-0)

Ve Štětí dne 2. 12. 2022 Podpis: ……………………

***Čestné prohlášení o samostatnosti zpracování maturitní práce***

Prohlašuji, že jsem odevzdanou maturitní práci vypracoval samostatně a uvedl jsem všechny použité zdroje. Uvědomuji si, že prokáže-li se opak, může být má práce hodnocena jako nedostatečná.

Ve Štětí dne 2. 12. 2022 Podpis: ………………………………

***Čestné prohlášení o použití legálního softwarového vybavení***

Prohlašuji, že veškeré programové vybavení, které bylo použito při řešení této maturitní práce, bylo užito v souladu s jeho licencí.

Ve Štětí dne 2.12.2022 Podpis: ……………………………………

# **Obsah**

[**Obsah 6**](#_heading=h.xgthspm9tryn)

[**Fitness tracker 8**](#_heading=h.ouhc3xu5oaob)

[Popis 8](#_heading=h.6rmkjtbny49w)

[Jak zařízení funguje 8](#_heading=h.rbohjxc8d8eu)

[**Hardware 9**](#_heading=h.setg1fqxn9wr)

[Zařízení M5 Stick C Plus 9](#_heading=h.wxgm9fcyot19)

[Displej TFT LCD 10](#_heading=h.c6j4povbw55w)

[Zakomponování do projektu 10](#_heading=h.o7iojbf71now)

[Baterie Lithium-Polymer 11](#_heading=h.tm4z0t4f4vad)

[Zakomponování do projektu 11](#_heading=h.lv0mp13a087x)

[Klady a zápory použití bezdrátové Wifi 12](#_heading=h.xmg0o9jkm2eb)

[Klady 12](#_heading=h.7khmgs3n1d9u)

[Zápory 12](#_heading=h.ko9ecphfa0gc)

[Senzor QMP6988 13](#_heading=h.d86nuydfog1c)

[Zakomponování do projektu 13](#_heading=h.vg0uv0kl5pos)

[Kabel typu Grove 14](#_heading=h.9cs4n3nrt5vk)

[**Zakomponování do projektu 14**](#_heading=h.e05loju0syzi)

[**Software 15**](#_heading=h.80cd3reg6fx4)

[Použité knihovny 15](#_heading=h.pilqfzu9q7mo)

[Připojení k Wifi 16](#_heading=h.2wh4lnpbvizm)

[Kalibrace (nastavení důležitých hodnot v programu) 16](#_heading=h.lwtsw3fmzv4h)

[Nastavení nejnižšího tlaku 16](#_heading=h.4yh558tcfvt5)

[Nastavení nejvyššího tlaku 16](#_heading=h.mteamlrb2liu)

[Podmínky které zamezí základním problémům: 17](#_heading=h.cnwb11tluc8a)

[Zjištění kde se nacházíme: 18](#_heading=h.e5yme1b2sgjc)

[Krok č.1 18](#_heading=h.m69ym9vwjlsc)

[Krok č.2 18](#_heading=h.v20a2fu4f20z)

[Ukládání dat na thingspeak 18](#_heading=h.jxr0yc1f4hdf)

[Za každým krokem se nachází podmínka 19](#_heading=h.pdciffcr4zhm)

[V neposlední řadě vypsání potřebných hodnot na displej: 19](#_heading=h.gs8fyjfolrxi)

[**Vývoj projektu 20**](#_heading=h.aepyhj58kq8)

[První verze 20](#_heading=h.5ekhgvs6r0fa)

[Druhá verze 20](#_heading=h.8x889vrtuhsu)

[Problémy v kódu 21](#_heading=h.h88e0434ga2v)

[**Proč jsem se rozhodl udělat zrovna tento projekt? 21**](#_heading=h.r6b5jsvxcnnw)

[**Zdroje: 22**](#_heading=h.a533q9oe0bi)

[Zdrojový kód 22](#_heading=h.7njmpezbauuh)

[M5StickC Plus 22](#_heading=h.3nqt5ld25ptw)

[Senzor 22](#_heading=h.m07n1r1l9ae6)

[Wifi 23](#_heading=h.5s3mdddmgjki)

[Thingspeak 23](#_heading=h.2k22ltu2j9o2)

[**Přílohy 24**](#_heading=h.h6wvrqbrzr51)

[obraz.1 24](#_heading=h.8xj8bxunm0vy)

[obraz.2 obraz.3 24](#_heading=h.npcp7rmoioa0)

[obraz.4 obraz.6 25](#_heading=h.pe7e2t5rwp27)

[obraz.7 25](#_heading=h.uudspgx1679a)

[obraz č.5 26](#_heading=h.ud7b42cawcmr)

# **Fitness tracker**

## **Popis**

Jedná se o zařízení které se skládá z M5StickC Plus a senzoru na měření tlaku, kde senzor zachytává hodnoty se kterými zařízení následně pracuje, myšleno odesílání na webovou stránku nebo kalibraci a v neposlední řadě, výpis na samotný displej tohoto zařízení. Jak už jsem zmínil, data se posílají pomocí wifi na webovou stránku Thingspeak, na které se nacházejí grafy do kterých se poté ukládají hodnoty, aby jste po každém cvičení mohli vidět, nějaké ty výsledky.

## **Jak zařízení funguje**

Zaznamenává a poté vypíše počet kalorií, které spálíte a čas za který jste udělali dřep, ukládá se totiž nejvyšší tlak, nejnižší tlak a také aktuální tlak takže lehce můžete zjistit za jakou dobu jste se dostali do nejvyššího bodu a do nejnižšího bodu, má také funkci, která když dosáhnete 5 dřepů tak resetuje hodnoty, samozřejmě je na začátku kalibrace hodnot, která slouží proto, aby si to mohl každý nastavit pro sebe, protože někdo je vysoký a někdo zase nízký.

# **Hardware**

## **Zařízení M5 Stick C Plus**

M5StickC Plus je malý, ale výkonný mikrokontrolér a vývojová deska, která umožňuje vývojářům a nadšencům v oblasti IoT rychle a snadno vytvářet různé projekty. Tato deska je vybavena procesorem ESP32, který kombinuje v sobě WiFi, Bluetooth a další funkce.

Díky svému malému a kompaktnímu provedení je M5StickC Plus ideální pro vytváření projektů, které vyžadují mobilní, nebo přenosné řešení. Deska obsahuje vstup/výstup piny, USB-C port, tlačítka, displej a mnoho dalších funkcí. M5 StickER Plus také podporuje různé periferie, jako jsou senzory, displeje a další, což umožňuje uživatelům rozšířit jeho funkčnost.

Díky svému vestavěnému WiFi a Bluetooth modulu, M5StickC Plus umožňuje připojení k internetu a komunikaci s dalšími zařízeními, což umožňuje vytváření různých IoT aplikací. Kromě toho podporuje různé programovací jazyky, jako je například MicroPython, Arduino IDE nebo UIFlow, což umožňuje uživatelům vybrat si ten, který nejlépe odpovídá jejich potřebám.

M5StickC Plus lze využít v mnoha různých aplikacích, jako jsou například:

* Vytváření různých senzorových sítí pro monitorování klimatických podmínek nebo pro měření různých fyzikálních veličin.
* Vývoj chytrých zařízení pro domácnost, jako jsou například chytré zámky, termostaty, osvětlení a další.
* Vývoj chytrých náramků, hodinek nebo jiných nositelných zařízení pro sledování zdravotního stavu, fitness nebo jiných parametrů.
* Vytváření různých automatizovaných řešení pro průmysl, jako jsou například senzorické systémy pro sledování výrobních procesů, logistických operací a další.

Celkově lze říci, že M5StickC Plus nabízí výkonné a kompaktní řešení pro vývoj IoT aplikací a mnoho dalších projektů. viz [obraz.1](#_heading=h.8xj8bxunm0vy)

# 

## **Displej TFT LCD**

Tento displej má úhlopříčku 1,14 palce a rozlišení 135 x 240 pixelů. Displej má vysokou kvalitu obrazu a podporuje 65 tisíc barev.

* Tento displej se ovládá pomocí SPI rozhraní, což umožňuje rychlou a snadnou komunikaci s mikrokontrolérem, který je umístěn na desce.
* Díky integrovanému displeji lze na desce snadno zobrazovat různé informace a výstupy, jako jsou například hodnoty ze senzorů, stavové zprávy nebo uživatelské rozhraní pro ovládání.
* Je také podporován širokou škálou knihoven a nástrojů pro vývoj, což umožňuje snadnou integraci a rychlý vývoj prototypů a projektů. Displej TFT LCD na M5StickC Plus tedy nabízí výkonný a kompaktní způsob, jak zobrazovat informace a interagovat s uživatelem.

viz [obraz.1](#_heading=h.8xj8bxunm0vy)

### **Zakomponování do projektu**

TFT LCD displej používám na výpis hodnot se kterými poté pracuji , např. jaký je nejnižší a nejvyšší tlak, nebo počet dřepů, také čas za který jsem tyto dřepy udělal a nakonec kde se právě nacházím, jestli v pohybu nebo stojím, sedím.

## **Baterie Lithium-Polymer**

* Baterie Lithium-Polymer (LiPo) je součástí vývojové desky M5StickC Plus. LiPo baterie se vyznačuje vysokou kapacitou a nízkou hmotností, což umožňuje vývojářům a výzkumníkům použít M5StickC Plus na mnoha místech, včetně těch, kde není přístup k napájení ze sítě.
* Baterie má kapacitu 120 mAh a je nabíjecí pomocí USB portu. Baterie také obsahuje ochranné funkce, jako je například ochrana proti přebíjení a vybíjení, což zvyšuje bezpečnost a životnost baterie.
* Díky baterii LiPo může M5StickC Plus pracovat samostatně bez připojení k zdroji napájení po dobu několika hodin v závislosti na zátěži. To umožňuje vývojářům a výzkumníkům používat M5StickC Plus v terénu, na místech s omezeným přístupem k napájení nebo pro mobilní aplikace.
* Baterie LiPo je běžně používána v mnoha typech zařízení, včetně mobilních telefonů, tabletů a přenosných herních konzolí. Díky své vysoké kapacitě a nízké hmotnosti je ideální pro použití v přenosných elektronických zařízeních, jako je M5StickC Plus.
* Baterie LiPo je také známá svou vysokou energetickou hustotou, což znamená, že obsahuje velké množství energie v malém prostoru. To umožňuje, aby M5StickC Plus byl velmi kompaktní a lehký, což je důležité pro mobilní aplikace a projekty, které vyžadují malé rozměry a nízkou hmotnost.

### **Zakomponování do projektu**

Jelikož potřebuji něco, co bude napájet moje zařízení aniž bych byl zrovna připojen k adaptéru nebo počítači, tak používám baterii(Lithium-Polymer) která vestavná a jak už jsem zmínil vydrží cca 2-3 hodiny, jelikož je docela nepraktické držet v ruce powerbanku a u toho dělat dřepy.

## **Klady a zápory použití bezdrátové Wifi**

M5StickC Plus je vývojová deska, která má integrovaný Wi-Fi modul a umožňuje bezdrátové připojení k internetu a k jiným síťovým zařízením. Wi-Fi modul na této desce podporuje standardy 802.11 b/g/n, což znamená, že podporuje frekvence 2,4 GHz a rychlosti přenosu až 150 Mbps.

### Klady

* **Pohodlnost:** Wi-Fi umožňuje bezdrátové připojení k internetu nebo k dalším síťovým zařízením bez nutnosti kabelového připojení, což zvyšuje pohodlí a flexibilitu použití.
* **Mobilita:** Bezdrátový Wi-Fi umožňuje připojení k internetu nebo k jiným síťovým zařízením odkudkoliv v dosahu signálu Wi-Fi, což umožňuje použití desky na různých místech.
* **Široká dostupnost:** Wi-Fi je široce rozšířená technologie a je k dispozici v mnoha místech, včetně domácností, kanceláří, veřejných míst a dalších.

### Zápory

* **Omezený dosah signálu:** Signál Wi-Fi má omezený dosah a může být narušen překážkami jako jsou stěny, předměty a další elektronická zařízení.
* **Zranitelnost:** Bezdrátové sítě jsou zranitelné vůči útokům hackerů, což může vést k ohrožení bezpečnosti a soukromí uživatele.
* **Rychlost přenosu dat:** Přenos dat přes Wi-Fi může být pomalejší než u kabelových připojení, což může ovlivnit rychlost a výkon aplikací a služeb používaných na M5StickC Plus.

Celkově lze říci, že používání bezdrátového Wi-Fi na M5StickC Plus má mnoho výhod, jako jsou pohodlnost, mobilita a široká dostupnost, ale také několik nevýhod, jako jsou omezený dosah signálu, zranitelnost a omezená rychlost přenosu dat.

## 

## Senzor QMP6988

* Je vyspělý mikroelektromechanický systém (MEMS) určený pro měření atmosférického tlaku a výšky. Tento senzor je vyráběn společností Bosch Sensortec a patří mezi nejpokročilejší senzory svého druhu.
* Využívá technologii piezorezistivního snímače a je vybaven vlastním digitálním signálovým procesorem (DSP). Tento senzor umožňuje měřit atmosférický tlak s vysokou přesností a rychlostí a je schopen detekovat změny atmosférického tlaku a výšky s vysokou citlivostí. Rozsah měření snímače QMP6988 je od 300 do 1100 hPa a jeho přesnost je ±3,9 Pa.
* Využívá se v různých aplikacích, jako je například měření teploty, průměru a rychlosti větru, měření výšky, sledování počasí a další. V kombinaci s dalšími senzory umožňuje senzor QMP 6988 vytvářet sofistikované systémy pro monitorování podmínek v prostředí.
* Senzor QMP 6988 se využívá v různých oblastech, jako jsou například průmysl, meteorologie, letectví, výzkum a mnoho dalších. Senzor QMP6988 je součástí různých zařízení a technologií, jako jsou například chytré telefony, hodinky, senzory v autě a další.

Celkově lze říci, že senzor QMP6988 je výkonný a vyspělý senzor pro měření atmosférického tlaku a výšky, který umožňuje vytvářet sofistikované systémy pro monitorování podmínek v prostředí.

### Zakomponování do projektu

Senzor QMP 6988 používám právě na sledování tlaku, jelikož počítám dřepy, tak potřebuji vědět jaký je nejnižší a nejvyšší tlak a také aktuální tlak, např.pro [kalibraci](#_heading=h.lwtsw3fmzv4h)  nebo pro zjištění [kde se to vlastně nacházíme](#_heading=h.e5yme1b2sgjc).

# 

## Kabel typu Grove

M5StickC Plus používá k připojení se Senzorem ENV.III kabel typu Grove. Grove kabel je speciální druh kabelu používaný pro propojení různých elektronických komponentů v rámci projektů založených na platformě Arduino a dalších mikrokontrolérech.

Konkrétně se jedná o kabel s 4 vodiči a konektory typu JST, který slouží k připojení Senzoru ENV.III k portu Grove na M5StickC Plus. Tento port je označen jako "G" a je umístěn na spodní straně M5StickC Plus, vedle portu pro nabíjení a datové připojení USB-C.

Při použití tohoto kabelu je nutné zajistit, aby byl správně připojen k Senzoru ENV.III i k M5StickC Plus. Na každém konci kabelu jsou totiž připojeny konektory typu JST, které slouží ke spojení s odpovídajícími konektory na Senzoru ENV.III a M5StickC Plus.

### Zakomponování do projektu

Kabelem typu Grove se propojuje M5StickC Plus se Senzorem QMP 6988.

# 

# 

# 

# **Software**

## **Použité knihovny**

* **M5StickCPlus.h** - tato knihovna poskytuje funkce a metody pro ovládání a konfiguraci vývojové desky M5StickC Plus, která je založena na mikrokontroléru ESP32. Obsahuje například funkce pro práci s displejem, tlačítky, LED diodami, Wi-Fi modulem a dalšími periferiemi na desce.
* **UNIT\_ENV.h** - tato knihovna slouží pro ovládání Senzoru ENV.III, což je kompaktní senzorová deska pro měření teploty, vlhkosti, tlaku a dalších parametrů v prostředí. Poskytuje funkce pro inicializaci senzoru, čtení dat a jejich zpracování.
* **stdio.h** - tato knihovna obsahuje základní funkce pro vstupní a výstupní operace v jazyce C. Zahrnuje například funkce pro čtení a zápis ze souborů, práci s řetězci a další.
* **stdbool.h** - tato knihovna obsahuje datový typ bool, který se používá pro ukládání logických hodnot true/false. Poskytuje také makra pro porovnávání logických hodnot.
* **WiFi.h** - tato knihovna umožňuje připojení k Wi-Fi sítím a poskytuje funkce pro konfiguraci a ovládání Wi-Fi modulu na desce ESP32. Obsahuje například funkce pro nastavení SSID a hesla sítě, připojení k síti a získání informací o síti.
* **HTTPClient.h** - tato knihovna slouží k vytváření HTTP požadavků a zpracování HTTP odpovědí. Obsahuje funkce pro odeslání požadavku na server, získání odpovědi a zpracování dat z odpovědi. Tato knihovna se často používá pro přístup k webovým API a dalším online službám.
* **time.h** - je součástí standardní knihovny jazyka C a obsahuje funkce pro práci s časem a datumem. Tato knihovna je velmi užitečná pro programování aplikací, které pracují s časem a datumem, jako jsou například kalendáře, časové plánovače a monitorovací systémy.

viz [obraz.4](#_heading=h.pe7e2t5rwp27)

## 

## **Připojení k Wifi**

* K wifi se moje zařízení připojuje pomocí tohoto kódu, musíte předem definovat jaké je heslo a jméno wifi jinak vás to samozřejmě nepřipojí, v tomto kódu je také podmínka, že dokud nebudete připojeni k wifi budou se vypisovat pouze tečky, tím pádem jde snadno zjistit jestli jste připojeni nebo ne.viz [obraz.6](#_heading=h.pe7e2t5rwp27)

## **Kalibrace** (nastavení důležitých hodnot v programu)

* Zde se nastavují základní hodnoty které potřebujeme, tedy nejvyšší a nejnižší tlak, s těmito hodnoty budeme poté pracovat, po nastavení těchto hodnot můžeme zjistit ne jednu věc. Víme třeba v jakém rozmezí se budeme pohybovat a taky se z toho dá zjistit, jestli dřepy děláte správně nebo ne.viz [obraz.5](#_heading=h.ud7b42cawcmr)

### **Nastavení nejnižšího tlaku**

* Na začátku zmáčknete tlačítko A(s názvem M5), to nastaví nejnižší tlak ”což znamená moment kdy stojíte”, doporučuji držet zařízení v ruce na úrovni krku. Hodnota HPF(hodnota\_prumer\_first) je proměnná do které se uloží hodnota se kterou bude kód pracovat po nastavení obou tlaků.

### **Nastavení nejvyššího tlaku**

* Jako druhý krok zmáčkneme tlačítko B(pravé postranní tlačítko), toto tlačítko nastaví nejvyšší tlak ” moment když jste v dřepu”, doporučuji držet zařízení v ruce na úrovni krku. Hodnota HPS(hodnota\_prumer\_second) je proměnná do které se uloží hodnota se kterou bude kód pracovat za moment.

### **Podmínky které zamezí základním problémům:**

* např. pokud bude nejnižší tlak mít vyšší hodnotu než tlak nejvyšší, tak se kalibrace zopakuje, jelikož jste nejspíš udělali chybu a prohodili tlačítka. Také zde můžeme vidět podmínku, která zamezí tomu aby hodnoty byly nulové, vím sice že to zní na první pohled jako absurdní, ale může se to stát.
* Zde jak jsem už zmínil máme hodnoty HPF a HPS, které používám abych zjistil rozdíl tlaků(myšleno jaký rozdíl je mezi tlaky). Mám tu také podmínky, které zjistí jestli je rozdíl tlaků záporné číslo nebo ne. Sice by někdo mohl říct že je to vlastně k ničemu když mám podmínky které zamezí aby nejnižší tlak byl větší než nejvyšší.Mám tu také nějakou tu toleranci, jelikož né všem dřepy jdou. Je tu také příkaz round(ToleranceTlaku) což zaokrouhlí číslo, které poté použiji. Také tu mám proměnnou start díky které program zjistí že jsme na začátku.

## **Zjištění kde se nacházíme:**

* Jako první tu máme podmínku která se vypíše za předpokladu, že budeme stát s nějakou tou tolerancí, jelikož jsme na samém začátku(viz. [Krok č.1](#_heading=h.m69ym9vwjlsc)).

### **Krok č.1**

* Podmínka, která pokud je start = pravda “true”,tak se zapnou stopky a vypíše se text který vám oznamuje, že se nacházíte na začátku(stojíte) a zároveň se nastaví proměnná middle, pomocí které zjistí program že jsem už začátkem úspěšně prošli a následuje [Krok č.2](#_heading=h.v20a2fu4f20z), kde je potřeba právě podmínka middle.

### **Krok č.2**

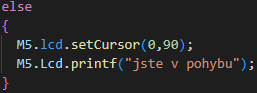
* Tady v té podmínce program zjišťuje, jestli jsme v dřepu, samozřejmě zase s nějakou tou tolerancí. Můžeme tu také vidět podmínku, která pokud je middle = pravda ”true”, tak vás textem upozorní že jste v dřepu a zároveň nastaví proměnné start = “false”, end = “true”, aby se nestalo že mezitím co budeme stát nám to vypsalo že jsme na začátku když jsme už na konci, proto musí být proměnná end = “true”, jinak bychom se vlastně nikam nedostali.

### Ukládání dat na thingspeak

* Tady máme podmínku end o které už jsem hovořil, v této podmínce se také nachází další podmínka, která znamená, pokud máme 5 nebo více dřepů, vypíše se počet dřepů a kalorie spálené za celé cvičení a uloží se aktuální čas. Také se resetují hodnoty a budete muset znovu kalibrovat, jelikož jedno cvičení máte za sebou a další můžete začít. Také se data pomocí HTTPClient.h, posílají na thingspeak, kde se ukládají do grafů(čas za cvičení, spálené kalorie, počet dřepů). Tyto data se posílají pouze když máte hotový daný počet dřepů. Ukládají se do grafů: field1(časomíra cvičení), field2(spálené kalorie) a field3(počet dřepů za cvičení). Počet kalorií zjistíme když vynásobíme kg, MET(jak moc intenzivně cvičíte na stupnici 1-8) a 0.0175, tím zjistíme spálené kalorie za minutu. Poté to vydělíme 60 a vynásobíme časem za který jsme udělali daný počet dřepů. Časomíru zjistíme tím že, vypočteme čas mezi začátkem a koncem, abychom zjistili počet sekund, které nám trvalo udělat daný počet dřepů. Počet dřepů zjistíme tím že přičítáme každý jeden dřep a na konci nám vyjde kolik jsme jich vlastně udělali.

### **Za každým krokem se nachází podmínka**

* Která, pokud žádná z podmínek v koloběhu dřepu není pravdivá, vypíše se že jste v pohybu, což je samozřejmě pravda.Jelikož kdyby tato podmínka nebyla, tak bychom z displeje nedokázali zjistit jestli jsme v pohybu, vím sice že by hodně lidí mohlo říci že přece vím jestli se jsem v pohybu nebo ne, ale tady jde především o kód jestli je napsán dobře a správně funguje. Protože **špatný kód nám nikdy fungovat správně nebude**.



## V neposlední řadě vypsání potřebných hodnot na displej**:**

* nejnižšího a nejvyššího tlaku
* aktuálního tlaku
* rozdílu tlaků
* tolerance tlaku
* počet dřepů

viz [obraz č.7](#_heading=h.uudspgx1679a)

# **Vývoj projektu**

## **První verze**

Začal bych tím že kód byl zbytečně dlouhý a zamotaný, takže jsem se v něm dots často nevyznal ani já sám, což znamenalo katastrofu, protože si dokážete představit že pokud se v tom nevyzná člověk který to píše, tak se v tom nevyzná ani člověk co to čte.Kód měl dost chyb např.nebyla v něm podmínka která by zamezila tomu že pokud stojíte, tak vám to bude připisovat dřepy i když je ve skutečnosti ani vlastně neděláte.

## **Druhá verze**

A to finální má kratší kód než první jelikož jsem se v tom ztrácel, tak jsem si řekl že to přepíši a zkusím udělat více srozumitelné a přehledné, jsem velmi rád že jsem se rozhodl kód přepsat, protože teď je přehledný a s pomocí tutoriálu nebo alespon nějakého vysvětlení se v něm dá lehce zorientovat. Jelikož kalibruji při začátku každé rutiny, tak můžu počítat, jak časomíru, tak kalorie , tím pádem se dozvím kolik jsem udělal dřepů, za jak dlouho a kolik jsem při tom spálil kalorii, což si myslím že je docela úspěch.Také posílám data na web.stránku thingspeak, kde se hodnoty zapisují do grafů, tím pádem můžete vidět kolik jste spálili kalorií a nebo jak dlouho vám trvalo celé cvičení.

## **Problémy v kódu**

Můj největší problém byl, s nastavováním a kalibrací tlaků, sice to může znít jako prkotina ale bylo to dost těžké, jelikož jsem chtěl kalibrovat až moc často což by způsobilo že by se mohl program rozbít, tak jsem se rozhodl že stačí kalibrovat pouze na začátku což je samozřejmě logické, jelikož mi to úplně stačí, proto abych zjistil jak daleko jsou od sebe nejnižší a nejvyšší tlak, a také abych přidal nějakou tu toleranci, protože tlak není pořád stejný na jednom místě, tlak se pořád mění i když stojíte na místě, což znamená že i kdyby byl můj kód sebelepší, kód by nefungoval.

# **Proč jsem se rozhodl udělat zrovna tento projekt?**

Zajímalo mě totiž, jak vlastně tyto náramky fungují, ale nebudu lhát dalo mi to docela dost práce ale myslím si že to za to stálo, jelikož program funguje jak má a dělá přesně to co chci.

# **Zdroje:**

## Zdrojový kód

[www1] Dokumentace Fitness tracker

[online]

dostupné z : <https://github.com/Rick538/fitness-tracker>[cit. 14.3.2023]

## M5StickC Plus

[www2] m5stickc-plus-esp32-pico-mini-iot-development-kit

[online]

dostupné z :

<https://shop.m5stack.com/products/m5stickc-plus-esp32-pico-mini-iot-development-kit?variant=35275856609444> [cit. 14.3.2023]

## Senzor

[www3] Dokumentace senzoru ENV.III

[online]

dostupné z : <https://botland.store/unit-extension-modules/19836-sensor-for-pressure-humidity-and-temperature-env-iii-sht30-qmp6988-unit-extension-module-for-m5stack-development-modules-6972934172976.html> [cit. 14.3.2023]

## Wifi

[www4] Wi-Fi

[online]

dostupné z : <https://cs.wikipedia.org/wiki/Wi-Fi> [cit. 14.3.2023]

## Thingspeak

[www5] Dokumentace grafů

[online]

dostupné z : <https://thingspeak.com/channels/2044897> [cit. 14.3.2023]

# **Přílohy**

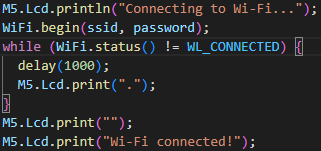
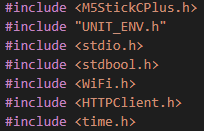
## obraz.1



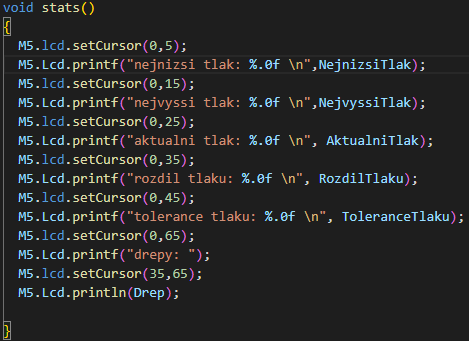
## obraz.2 obraz.3

 ****

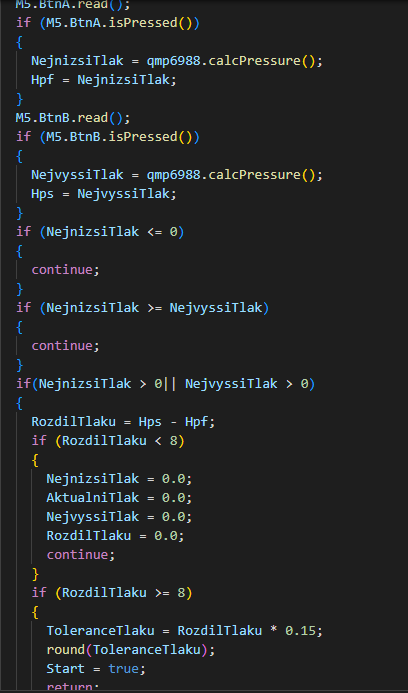
## obraz.4 obraz.6



## obraz.7



## obraz č.5



1. V případě nesouhlasu se zadáním maturitní práce se v den zadávání žák písemně obrátí na ředitele   
   VOŠ, SPŠ, SOŠS a CR s odůvodněním svého nesouhlasu. [↑](#footnote-ref-0)