**SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE  
FAKULTA ELEKTROTECHNIKY A INFORMATIKY**

**Zadanie č.1**

**Aplikácia IoT**

**2024 Marek Trúchly**

Obsah

[1. Popis zadania 3](#_Toc168327736)

[2. Diagramy a využité protokoly 3](#_Toc168327737)

[3. Komplikácie a vysvetlenie serverovej časti 3](#_Toc168327738)

[4. Vysvetlenie Arduino kódu 6](#_Toc168327739)

[5. Užívateľský manuál 7](#_Toc168327740)

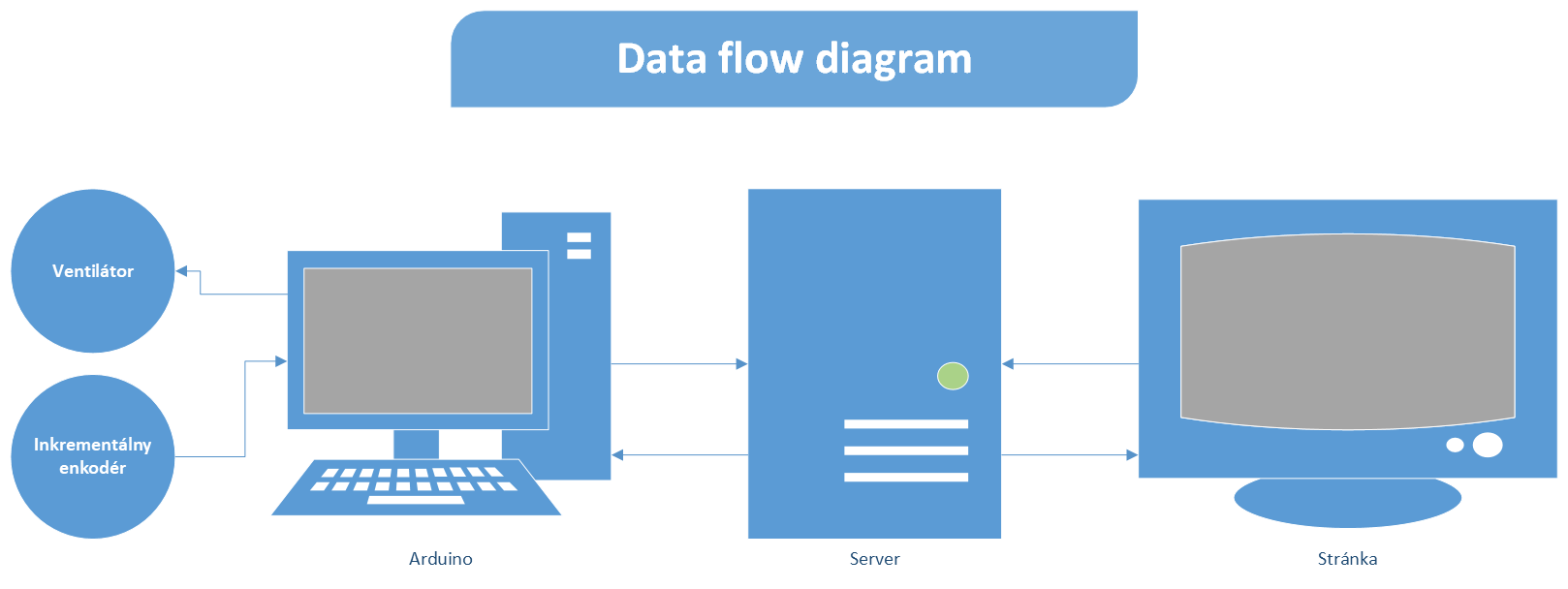
[6. Záver 8](#_Toc168327741)

# Popis zadania

V zadaní sme použili inkrementálny enkodér (E1171S-DAK3-2020-K01), 5 voltový ventilátor a arduino uno. V pôvodnom zadaní sme mali snímať otáčky enkodéra a riadiť motor. Vzhľadom k vysokému momentu, ktorý je potrebný k otočeniu enkodéra sme ustúpili od snímania otáčok a snímame iba natočenie enkodéra. Nasnímané otočenia enkodéra prevádzame do natočenia v uhloch a posielame ich na server pomocou sériovej linky. Na stránke ich potom vypisujeme a vykresľujeme.

# Diagramy a využité protokoly

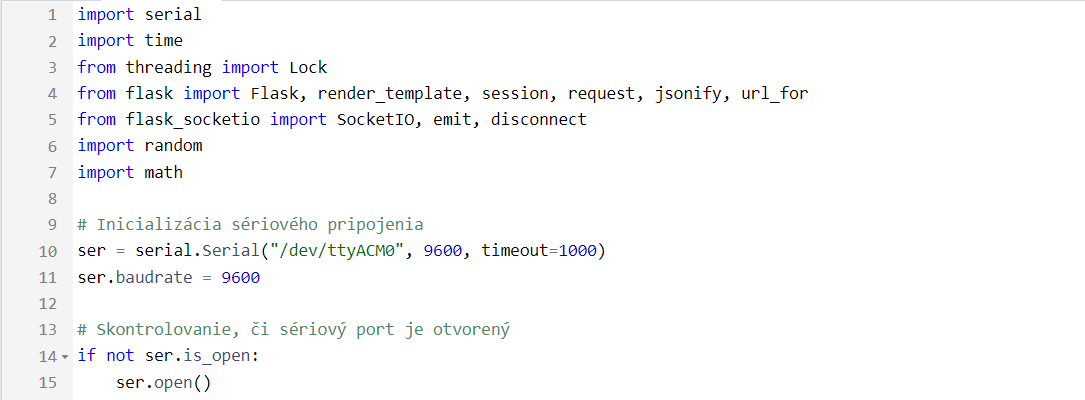
V rámci zadanie sme vypracovali aj UML diagram, ktorý vidíme na obrázku 1. Diagram ukazuje prúdenie dát a komunikačné kanály.



Obr.1 UML diagram

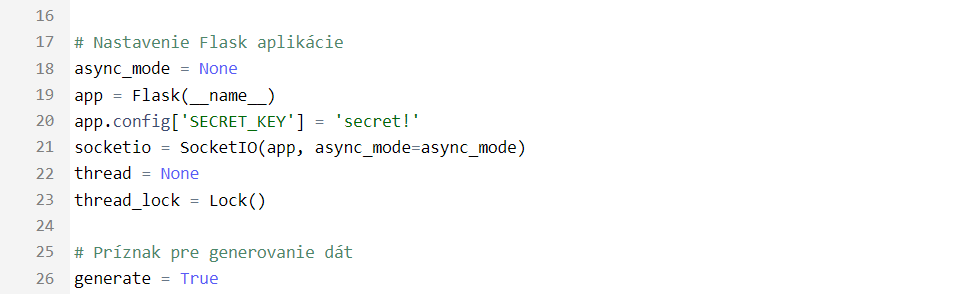
Ventilátor je ovládaný PWM signálom, ktorý je generovaný arduinom unom. Inkrementálny enkodér sme zapájali podľa návodov na internete. Arduino prepočítava signály na uhol natočenia a cez full duplex linku posiela uhol natočenia do servera. Server je na počítači, na ktorom beží Linux vo virtual boxe. Stránka sa otvára ako lokálny hosť v Linuxe. Server so stránkou komunikuje cez websokety.

# Komplikácie a vysvetlenie serverovej časti



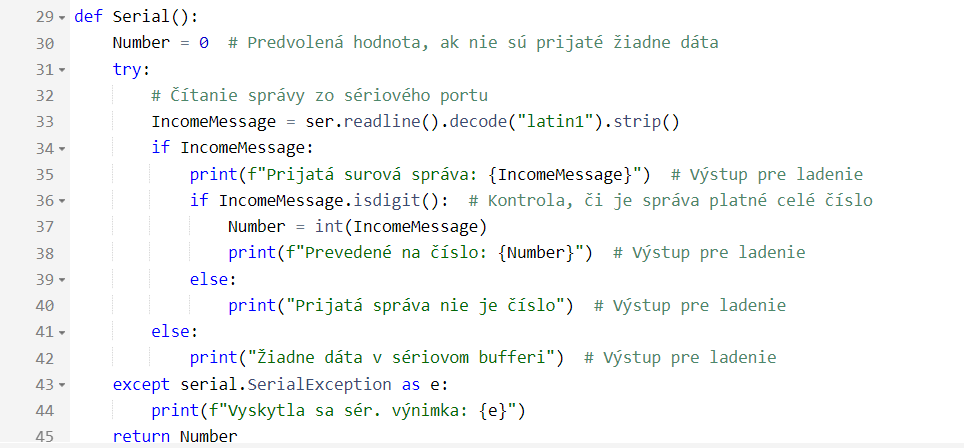
Obr.2 Importovanie knižníc a inicializácia serverovej časti kódu

Na serverovej časti sme mimo knižníc importovaných počas cvičení importovali knižnicu „serial“. Táto knižnica obsahuje funkcie, cez ktoré dokážeme komunikovať po sériovej linke s arduinom.



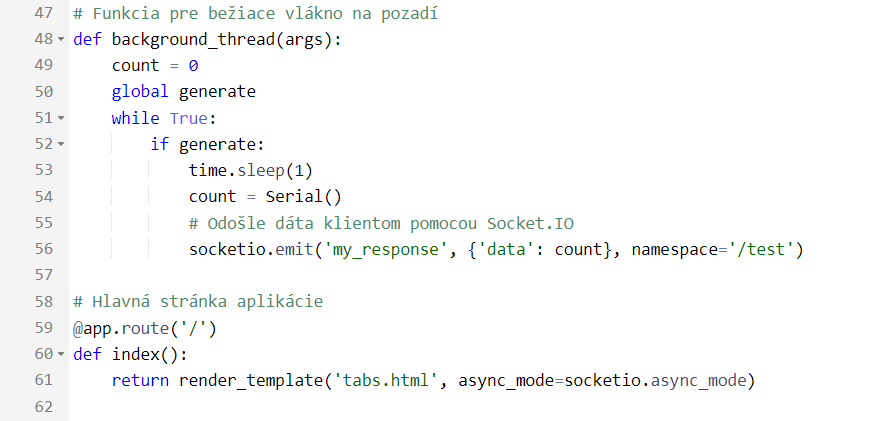
Obr.3 Nastavenie flasku a vlákna

V rámci kódu na obrázku 3 sme nič nemenili vzhľadom k cvičeniam.



Obr.4 Funkcia sériovej komunikácie

Ak sa pozrieme na kód na obrázku 4 môžeme vidieť prekomplikovane riešenie komunikácie s arduinom. Je to z toho dôvodu, že sme sa pokúšali debugovať kód. Mali sme závadu, pri ktorej nám sériová komunikácia na strane servera čítala iba nuly. Ošetrovali a zisťovali sme problémy. Skontrolovali sme arduino, kábel, premenné až kým sme zistili, že problém bol pri synchronizácií. K zisteniu sme prišli až po konzultácií s cvičiacim.



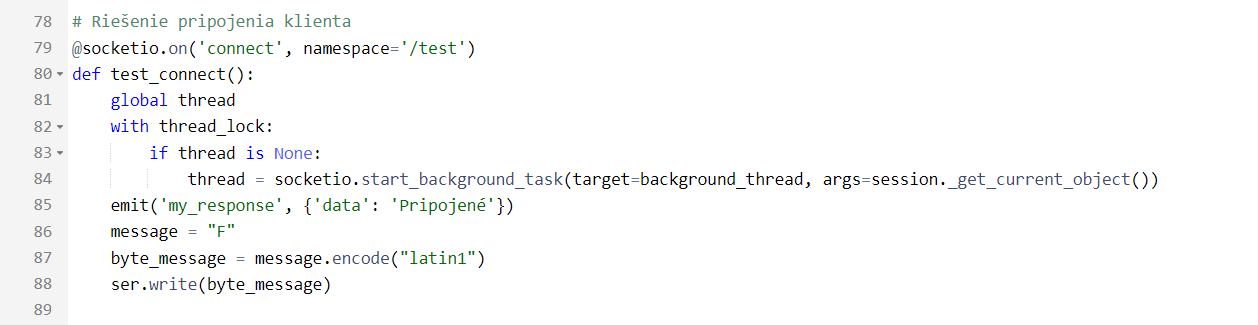
Obr.5 Vlákno bežiace na pozadí

V rámci kódu na obrázku 5 sme nič nemenili vzhľadom k cvičeniam.



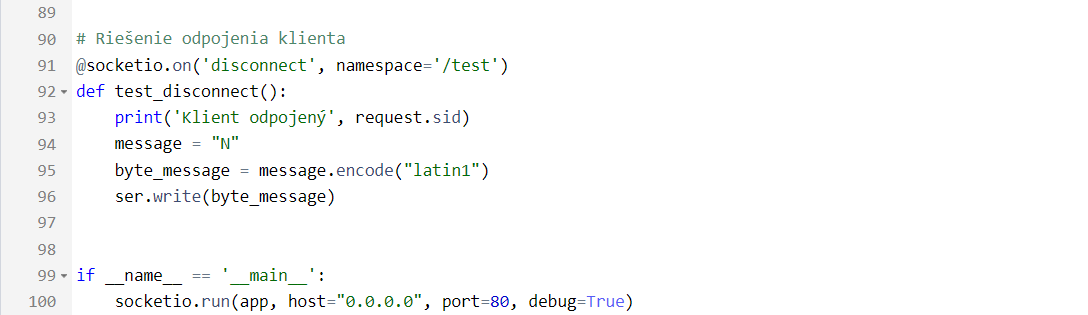
Obr.6 Vybavovanie žiadosti zo strany klienta

V rámci kódu na obrázku 7 sme nič nemenili vzhľadom k cvičeniam.



Obr. 7 Pripojenie ku klientovi

V kóde v obrázku 7 sme doplnili originálnu funkciu o pár riadkov, ktoré posielajú cez sériovú linku do arduina písmeno „F“. Písmeno „F“ zapína a písmeno „N“ vypína ventilátor. Spracovanie písmena si ukážeme v ďalšej kapitole.

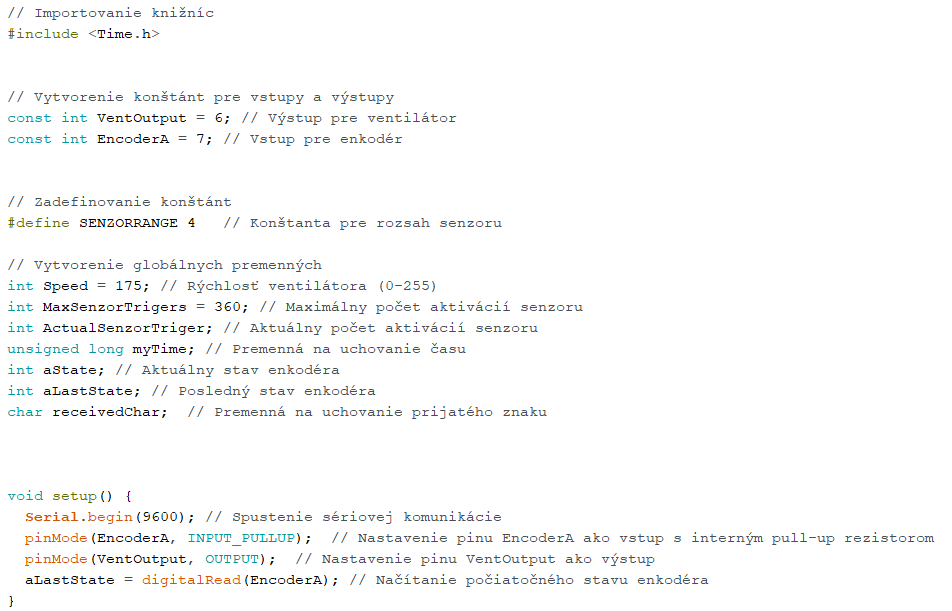


Obr.8 Odpojenie od klienta

Podobne ako na obrázku 7 sme tento kód na obrázku 8 doplnili o pár riadkov, ktoré posielajú cez sériovú linku do arduina písmeno „N“.

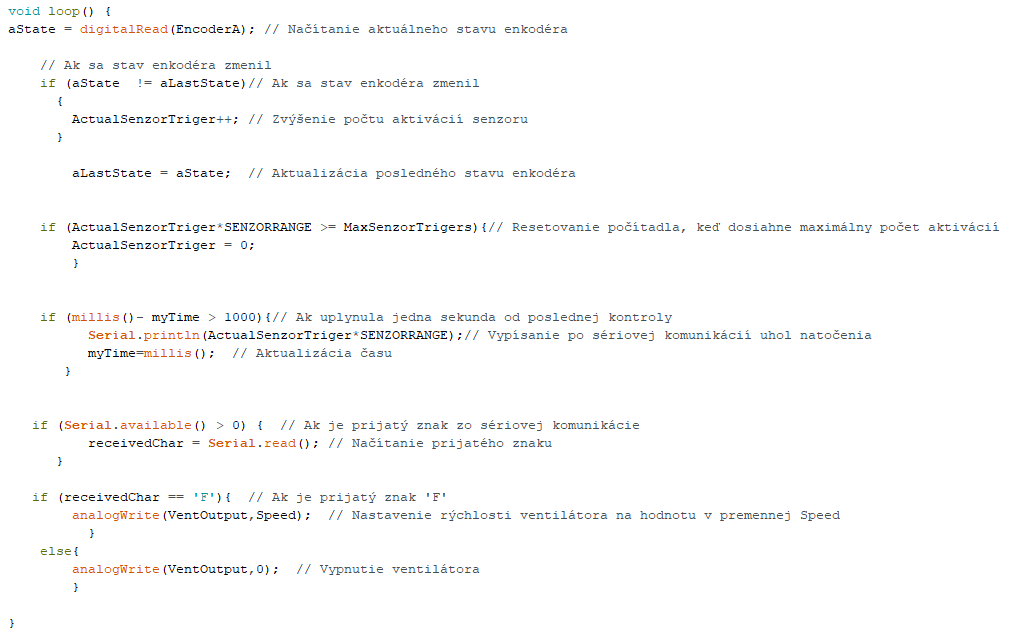
# Vysvetlenie Arduino kódu

Ďalej si vysvetlíme dva obrázky, na ktorých je kód pre arduino.



Obr.9 ½ arduino kódu

Na obrázku vidíme prvú polovicu celého arduino kódu. V kóde primárne inicializujeme premenné, importujeme knižnice a robíme setup pre vstupno výstupné periférie.



Obr.10 2/2 polovica arduino kódu

Druhá polovica kódu obsahuje práve hlavnú slučku, v ktorej načítavame zmenu stavu enkodéra a tým sledujeme aj uhol natočenia enkodéra. Ďalej v slučke ošetrujeme to, že po prekročení uhlu 360stupňov sa uhol zresetuje na nulu. Je to z toho dôvodu, že ručička na stránke je do hodnoty 360 stupňov. V závere kódu je naprogramované primanie správ

# Užívateľský manuál

1. Zapojíme arduino do portu na pravej strane počítača.
2. Otvoríme VirtualBox.
3. Otvoríme terminál
4. V termináli sa presunieme do priečinka, kde sa nachádzajú súbory servera a klienta.
5. Otvoríme internetový prehliadač.
6. Do terminálu napíšeme „sudo python3 Zadanie.py“.
7. V internetovom prehliadači do adresára napíšeme „localhost“.
8. Teraz by sme mali mať spustený server a mali by sme byť aj naňho pripojený. Môžeme ovládať výpis hodnôt otáčaním enkodéra, a podobne môžeme ovládať aj výpis hodnôt na strane klienta.
9. Odpojením od servera uskutočníme stlačením klávesnice CTRL+c.
10. Odpojenie od servera neuzavrie sériovú komunikáciu.

# Záver

Klientsku časť sme riešili podľa cvičenia z ôsmeho týždňa. Aplikovali sme iba malé úpravy a to v načítavaní druhej premennej do grafu. V rámci realizácie zadania sme sa stretli s viacerými technickými výzvami, ktoré si vyžiadali prispôsobenie pôvodného plánu. Najväčšou komplikáciou bolo riešenie synchronizácie sériovej komunikácie medzi arduinom a serverom. Týmto zadanie beriem, ako úspešne spracované.

Úložisko zdrojového kódu a dokuemntácie je na drese: [LINK](https://github.com/MarekWhy/POIT)