**SLOVENSKÁ TECHNICKÁ UNIVERZITA**

**POKROČILÉ INFORMAČNÉ TECHNOLÓGIE**

Technická dokumentácia

Bc. Boris Dobrovič, Bc. Matúš Hutár, Bc. Zdenko Pucovski,

Bc. Matej Kubala, Bc. Martin Supek

Inštalačná príručka

Konfigurácia Raspberry Pi

Inštalácia OS Raspbian

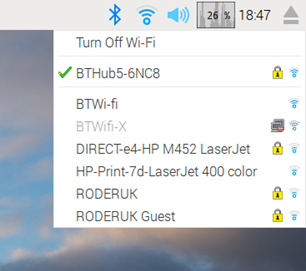
* Je potrebná micro SD karta s kapacitou min. 4GB, ale odporúčaná je aspoň 8GB a viac v závislosti na požadovaných aplikáciách. Formátovanie karty musí byť FAT16 alebo FAT32.
* Zo stránky raspberrypi.org/downloads je potrebné stiahnuť NOOBS (Next Out Of Box Software) súbor typu ZIP, rozbaliť ho a iba jeho obsah nahrať na kartu. Takto pripravenú kartu treba zasunúť do slotu na doske RPi (Raspberry Pi).
* Na SD kartu možno nahrať aj verziu NOOBS Lite (nutnosť v prípade 4GB karty, vtedy sa zvyšok doinštaluje z Internetu)
* Na SD kartu možno nahrať aj obraz (image) OS systému, treba však na to program na nahrávanie obrazov (Etcher, Win32DiskImager,...)
* Po zapnutí RPi pripojením k napájaniu po prvýkrát sa spustí inštalácia operačného systému (OS).
* Treba si zvoliť, aký OS chceme nainštalovať – odporúča sa Raspbian
* Je možné si zvoliť aj jazyk a klávesnicu
* Po úspešnom nainštalovaní OS a potvrdení OK nasleduje reboot.
* Po nabehnutí systému sa odporúča zmeniť používateľovi „pi“ štandardné heslo „raspberry“ na niečo iné.
* Tiež sa odporúča aktualizovať systém spustením terminálu a príkazmi:

**sudo apt-get update**

**sudo apt-get dist-upgrade**

Nastavenie WIFI pripojenia

Najjednoduchším spôsobom ako nastaviť WiFi je pomocou grafického prostredia OS Rasbian. V pravom hornom rohu kliknutím na ikonu WiFi, vybraním správneho SSID a zadaním správneho hesla sa vieme bezdrôtovo pripojiť.



*Obrázok 1 - Pripojenie na WIFI*

Inštalácia a konfigurácia web kamery

Prvým krokom je inštalácia balíka Motion pomocou príkazu:

**sudo apt-get install motion**

Následne je potrebné upraviť nasledovné parametre v konfiguračnom súbore **motion.conf**.

Na editovanie súboru motion.conf je potrebné zadať príkaz:

**sudo nano /etc/motion/motion.conf**

V súbore je potrebné nastaviť/skontrolovať nasledovné parametre:

* daemon musí byť nastavený na ON
* framerate musí byť nastavený na 30
* stream\_maxrate musí byť nastavený na 30
* stream\_port musí byť nastavený na 8081
* stream\_quality musí byť nastavená na 100
* stream\_location musí byť nastavená na OFF
* webcontrol\_localhost musí byť nastavený na OFF
* qualitymusí byť nastavená na 100
* parametre width a height bude napríklad 640 & 480
* parameter post\_capturebude nastavený na5
* output\_pictures bude nastavený na OFF
* ffmpeg\_outputbude nastavený na OFF.

Keďže pracujeme s textovým editorom “nano” tak je potrebné na uloženie zmien je potrebné stlačiť Ctrl+x, potom napísať y a nakoniec stlačiť Enter.

V ďalšom súbore, do ktorého pristúpime nasledovne:

**sudo nano /etc/default/motion**

Pričom je potrebné zmeniť parameter “start\_motion\_daemon” v tomto súbore na “yes”.

Posledným krokom je reštartovanie servisu pomocou príkazu:

**sudo service motion restart**.

Inštalácia Flask-u

Na inštaláciu Flasku je potrebné použiť príkazový riadok, v ktorom sa použijú príkazy:

**sudo apt-get install python-pip**

**pip install flask**

Inštalácia podpory pre websockety:

**sudo pip install eventlet**

**sudo pip install flask-socketio**

Konfigurácia Arduina

Arduino obsluhuje riadenie motorov, serva, snímanie intenzity svetla, zapínanie svetiel, meranie vzdialenosti od objektu a detekciu čiar pomocou dvoch IR senzorov. Pre komunikáciu medzi arduinom a raspberry bola použitá sériová komunikácia. Komunikácia je nastavená na 8 bitov a rýchlosť na 115200.

Nastavenie senzorov arduina

**Zapojenie arduina**

Senzory:

Svetelný senzor : Pin A0

IR senzor\_1: Pin 7

IR senzor\_2: Pin 5

HC\_SR04\_echo: Pin 12

HC\_SR04\_trigger: Pin 13

Svetlá:

LED\_1: Pin 4

LED\_2: Pin 2

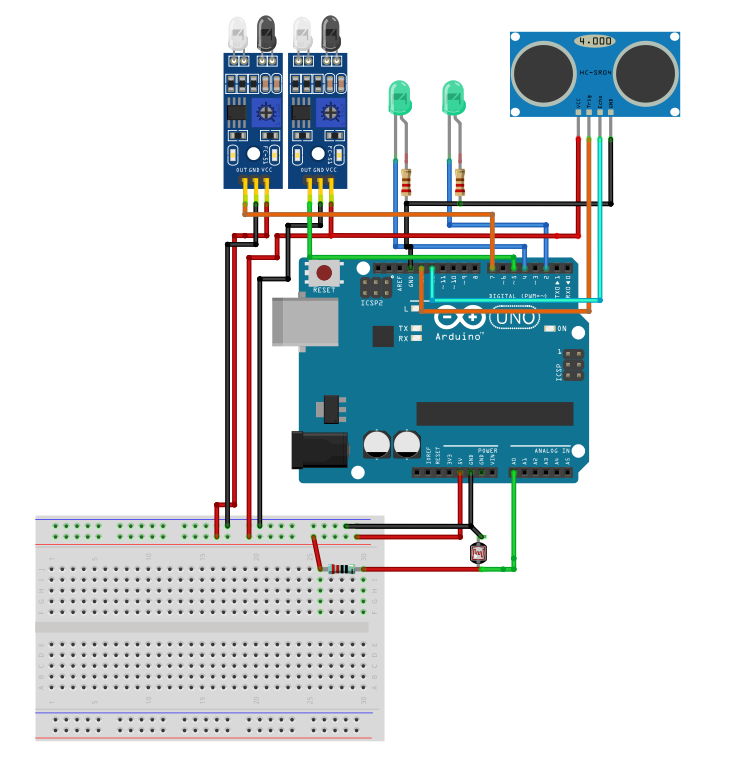
Riadenie:

Signál pre motory z auta : Pin 6

Signál pre motory z arduina: Pin 9

Signál pre servo z auta: Pin 3

Signál pre servo z arduina: Pin 10



Obrázok 2 :Schematické zapojenie arduina

Verziovanie projektu

Náš online repozitár sme vytvorili na online platfrome “Github.com”. Hlavné nastavenie repozitára je “verejný”, pretože repozitár typu “privátny” nedisponoval rovnakými možnosťami. Repozitár je pod licenciou GNU GPL v3. Všetky súbory v tomto repozitári je možne voľne sťahovať. Upravovať tento konkrétny repozitár je umožnené len členom nášho tímu. Náš repozitár sa nachádza na adrese **“**[**www.github.com/dobinoo/ppoit**](http://www.github.com/dobinoo/ppoit)**”** . Repozitár je možné stiahnuť si pomocou grafického rozhranie, ktoré poskytuje webová stránka a to tlačidlom “clone or download”. Je možné použiť príkazový riadok s nainštalovaným verziovacím systémom git. Na stiahnutie pomocou tohto nástroja slúži jednoduchý príkaz:

**git clone https://github.com/dobinoo/ppoit.git**

Potom sa stačí presunúť do adresára “ppoit”. Nachádza sa tu súbor pre arduino, licencia, readme text a priečinok “flask\_default”. Vo vnútri tohto priečinka sa nachádza webová stránka a server pre raspberry pi.

Používateľská príručka

Spustenie systému

Po pripojení Raspberry Pi na napájanie, čiže do power-banky, sa spustí operačný systém Raspbian. Na spustenie systému sa potrebujeme pripojiť sa na bežiaci operačný systém na našom RPi. To môžeme vykonať viacerými spôsobmi. Jednou z nich je vzdialeným prístupom napríklad pomocou realVNC softvéru. Ten však treba predom nainštalovať a povoliť vzdialený prístup na RPi. Ďalšou z možností je pripojiť sa na raspberry pomocou ssh. Nevýhodou je, že pri oboch možnostiach potrebujeme vždy poznať IP adresu nášho RPi. Keďže sme nechceli nastavovať statickú IP adresu lebo je možné, že s náš projekt budeme prezentovať v inej miestnosti a tým pádom sa pripojiť na inú WiFi sieť, využili sme tretiu možnosť a to priamy prístup na RPi cez monitor a klávesnicu.

Spustenie systému je veľmi jednoduché. Otvoríme terminál a ako prvé spustíme motion service na streamovanie obrazu z webkamery príkazom

**sudo motion**

Po vykonaní celého úvodného nastavenia balíka motion, môžeme nájsť online prenos z web kamery na IP adrese nášho Rasberry Pi a dopísaním portu 8081. (Príklad: 192.168.0.54:8081)

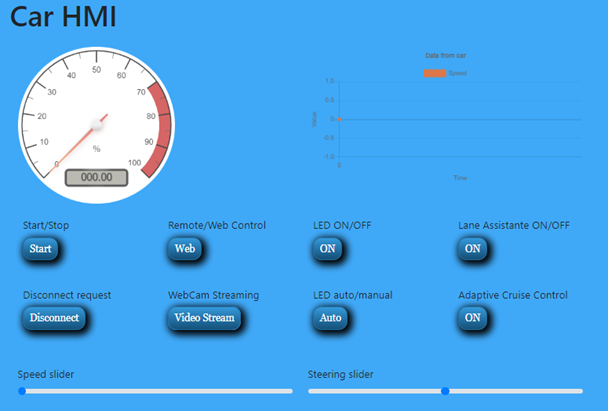
Ďalšou úlohou je spustiť webový server, teda súbor “app.py”. Vykonáme to tak, že sa v príkazovom riadku navigujeme do priečinka, kde máme priečinok so všetkými súbormi potrebnými pre správny beh serveru. Následne v tomto priečinku spustíme súbor “app.py” týmto príkazom:

**sudo python app.py**

Web server je teraz spustený a je možné sa na neho pripojiť pomocou webového prehliadača.

Ovládací panel vo webovom prehliadači

Po spustení systému je možné sa pripojiť na webový server bežiaci na RPi z akéhokoľvek webového prehliadača na zariadení, ktoré je pripojené na rovnakú WiFi sieť ako naše auto. Do poľa pre URL adresu napíšeme IP adresu nášho RPi a stlačením tlačidla “Enter” sa pripojíme naň pripojíme.

Obrázok 3 - Ovládací panel vo webovom prehliadači

Ovládací panel pre auto je veľmi intuitívny. Vo vrchnej časti obrazovky sa nachádzajú dva zobrazovacie prvky. Oba ukazujú percentuálny výkon vozidla, teda na koľko percent zo svojho maxima momentálne vozidlo ide. Vľavo je toto zobrazenie vo formáte klasického budíka ako poznáme z bežných áut a vpravo vo formáte grafu, pričom v tomto prípade sa na x-ovej osi nachádza čas v sekundách, ktorý určuje kedy sa hodnota rýchlosti zmenila a na y-ovej osi je už spomínaný percentuálny podiel výkonu.

Nasleduje niekoľko tlačidiel, ktorých funkcie sú nasledovné:

Start/Stop – inicializácia komunikácie a všetkých senzorov/ zastavenie komunikácie

Disconnect – ukončenie spojenia so serverom

Remote/Web Control – ovládanie pomocou diaľkového ovládača/ ovládacieho panela

Video Stream – je odkaz na port 8081, kde môžeme sledovať streamovanie z webkamery

LED ON/OFF – zapne/vypne predné „svetlomety“

LED auto/manual – zapne/vypne inteligentné osvetlenie

Lane Assist ON/OFF – zapne/vypne asistenta v jazdnom pruhu

Adapt Cruise Control – zapne/vypne adaptívny tempomat

V spodnej časti ovládacieho panela sa nachádzajú dva bežce, pomocou ktorých ovládame rýchlosť a natočenie kolies auta.

Vývojárska príručka

Serverová časť

Ako server nám slúži mikropočítač Raspberry Pi. Je dostatočne výkonný na pokrytie všetkých potrebných súčastí serverovej časti tohto projektu.

Programová súčasť serverovej časti je súbor app.py. Ten má na starosti zaznamenávanie, ukladanie a preposielanie vstupných údajov z ovládacieho panelu. Program je napísaný v programovacom jazyku Python s využitím niekoľkých modulov/ knižníc ako napríklad flask, socketio či serial alebo csv.

K najdôležitejším častiam kódu patrí napríklad kódovanie údajov do reťazca, ktorý následne vieme posielať cez sériovú linku.

Rozhodli sme sa, že údaje budeme posielať ako jeden reťazec, ktorý obsahuje 13 znakov, v ktorom každá pozícia bude predstavovať určitý údaj. Ich reprezentácia je zväčša binárna, teda 0 alebo 1, podľa toho, v akom stave je daná premenná. Nachádzajú sa tu aj dve viacciferné hodnoty, ktoré prislúchajú nastaveniu rýchlosti a natočeniu kolies. Tieto hodnoty sú vždy trojciferné, pričom východzia pozícia je pre rýchlosť 0 a pre natočenie kolies 50. Ide teda o percentuálnu reprezentáciu od 0 do 100. Ukončenie reťazca je definované znakom „R“. Ukončenie je potrebné pre programovú časť v Arduine, aby sme vedeli oddeliť jednotlivé reťazce údajov.

Ďalšou zaujímavou časťou je beh programu na pozadí, teda nie pri konkrétnom výskyte nejakej udalosti. V tejto časti je definované naplnenie reťazca na poslanie po sériovej linke, zápis údajov do csv súboru a tiež samotné poslanie reťazca po linke. Tieto úkony sa vykonajú každých 100 ms.

Zmena jednotlivých častí údajov je vykonaná v samostatných funkciách pri výskyte určitej udalosti, napríklad pri stlačení tlačidla. Podobným spôsobom funguje aj nadviazanie spojenia, respektíve inicializácia systému, ako aj zrušenie spojenia.

Celý zdrojový kód serverového programu je uložený v repozitári na githube.

Streamovanie webkamery beží na servery samostatne, mimo programu app.py. Návod na inštaláciu a spustenie streamovania webkamery nájdete v inštalačnej a používateľskej príručke.

Klientska časť

Klientska časť sa skladá z troch súborov a to HTML, CSS a JavaScript súboru. HTML súbor definuje štruktúru a samotný obsah celej klientskej časti webovej stránky. CSS súbor slúži na definovanie celého dizajnu ovládacieho panelu. A JavaScript súbor dodáva celému panelu funkčnosť a interaktivitu. Pričom najdôležitejšia časť je práve funkčnosť celého systému a teda JavaScript súbor. Tu sú definované všetky akcie, ktoré sa majú vykonať pri výskyte nejakej udalosti. Sú to prevažne zmeny stavov jednotlivých tlačidiel a bežcov a tiež zmena hodnôt a obnovenie grafických prvkov meniacich sa v čase, čiže tachometra a grafu.

Arduino časť

Pre riadenie a zber informácií zo senzorov sa použilo Arduino uno. Arduino je však ovládané pomocou raspberry, ktoré mu posiela správu, v ktorej sú zakódované inštrukcie pre inicializáciu auta, zapnutie a vypnutie svetiel, ovládanie auta cez web alebo ovládač, zapnutie asistentov jazdy a hodnoty pre motory a servo riadenie. Správa sa ukladá do poľa pokiaľ nepríde znak ukončenia správy “R”. Následne sa správa spracuje a program začne vyhodnocovať jednotlivé bity.

Automatické svetlá:

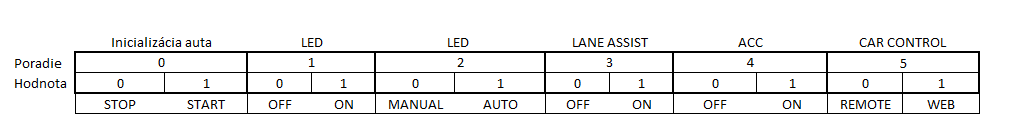
Automatické svietenie je zabezpečené 2 zelenýmí diódami, ktoré sa zapínajú ak je zvolený mód pre automatické svietenie a je nameraná hodnota zo svetelného senzora menšia ako 850.

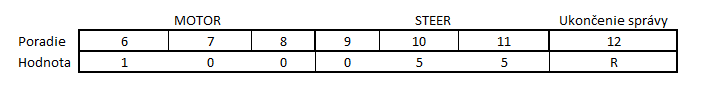
Lane assist:

Pre lane assist boli zvolené 2 IR senzory, ktoré sú umiestnené v prednej časti auta na oboch stranách. Takto bolo docielené, že dokážeme detegovať čiary pri vybočení auta. Pri detegovaní čiary na jednej strane program vie vyhodnotiť, že auto práve vychádza z pruhu a vie natočiť kolesá do správneho smeru tak aby sa udržalo medzi čiarami.

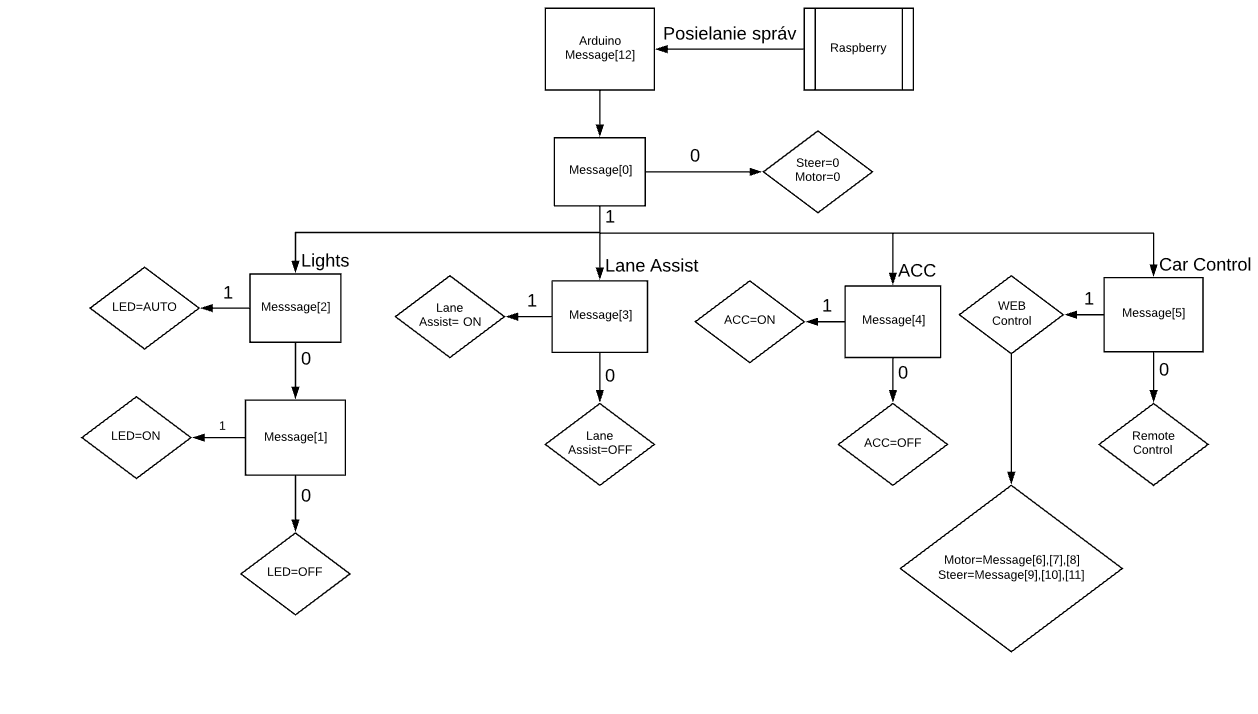
ACC:

Pre adaptívny tempomat bol zvolený ultrazvukový senzor, ktorý meria čas medzi vyslaným signálom a detegovaným signálom. Na základe tohto času sa vypočíta vzdialenosť od prekážky. Táto vzdialenosť je ďalej použitá na riadenie rýchlosti auta. Nastaví sa určitá vzdialenosť, ktorú má auto dodržiavať a následne sa už len riadia motory tak aby požadovaná hodnota sa zhodovala s nameranou.





Obrázok 4:Zadefinovanie správ



Obrázok 5: Vývojový diagram

Riadenie:

V prichádzajúcej správe sa nachádzajú aj údaje pre motory a servo. Tieto hodnoty sú zapísané do 3 bitov. Tieto hodnoty získavame pomocou skladania jednotlivých bitov do jedného trojciferného čísla *viď obrázok 6*. Výsledná hodnota je percentuálne číslo od 0 po 100, ktoré sa ešte ďalej prepočítava na rozsah 1100 až 1700 pre servo a 1500 až 1700 pre motory.



Obrázok 6: Skladanie percentuálnej hodnoty pre motory a servo

Keďže auto vie dosahovať veľké rýchlosti a zatiaľ s ním jazdíme iba v uzavretých priestoroch je jeho výkon softvérovo obmedzený.