

Šolski Center Celje,  
Srednja šola za kemijo, elektrotehniko in računalništvo  
Pot na Lavo 22, 3000 Celje

## UPRAVLJANJE AVTOMOBILA Z RASPBERRY PI-JEM



Mentor:

Borut Slemenšek, univ. dipl. inž.

Avtorji:

Žan Korez  
Anže Lazar  
Rok Mešiček

Celje, februar 2014

## KAZALO VSEBINE:

1. POVZETEK.....	2
2. UVOD.....	3
2.1 Opis raziskovalnega problema .....	3
2.2 Teze .....	3
2.3 Opis raziskovalnih metod.....	3
3. OSREDNJI DEL NALOGE .....	4
3.1 Predstavitev rezultatov raziskovanja .....	4
3.2 Delovanje .....	10
3.3 Razprava.....	11
4. ZAKLJUČEK.....	12
5. ZAHVALA .....	13
6. VIRI .....	14

## KAZALO SLIK:

Slika 1: PHP koda.....	4
Slika 2: Program v Pythonu. ....	5
Slika 3: Vežje potrebno za delovanje. ....	6
Slika 4: Spletna stran.....	7
Slika 5: Koda spletne strani 1. ....	8
Slika 6: Koda spletne strani 2. ....	9
Slika 7: Zagon pythona v programu Putty.....	9

## **1. POVZETEK**

V raziskovalni nalogi z naslovom Upravljanje avtomobila z Raspberry pi-jem smo raziskovali upravljanje avtomobila preko spletne strani z Raspberry pi-jem. Tak način upravljanja smo tudi izvedli. Praktični rezultat našega dela je upravljanje avtomobila s pomočjo vezja v daljincu; vezje smo uspešno povezali na Raspberry pi.

In this assignement with the title Driving a remote controlled car with raspberrry pi we researched how to do it and then made it by ourselves. The practical result was driving the car with the help of a chipset in a remote; we connected the chispset sucessfully with the raspberrry pi.

## 2. UVOD

### 2.1 Opis raziskovalnega problema

Tema naše seminarske naloge je Upravljanje avtomobila z Raspberry pi-jem. Za to temo smo se odločili, ker se nam je zdela zanimiva, pokriva področje naše stroke in na tržišču izdelka, kot je praktični rezultat našega raziskovanja, še ni. Ravno slednja ugotovitev nas je še bolj spodbudila k delu; menimo, da je naša rešitev tudi tržno zanimiva.

### 2.2 Teze

Na začetku raziskovanja smo postavili naslednji tezi:

- Raspberry pi bomo priključili na avtomobilček in z računalnikom dostopali do njega preko spletne strani;
- Raspberry pi bomo priključili na krmilnik avtomobilčka, napisali program za krmiljenje avtomobilčka in nato preko računalniške tipkovnice upravljali avtomobilček.

### 2.3 Opis raziskovalnih metod

Uporabili smo več raziskovalnih metod. Z metodo spraševanja smo pred začetkom konkretnega raziskovanja prišli do vedenja, kako bi se stvari lotili strokovnjaki na tem področju. Seveda smo to metodo pogosto uporabljali tudi med delom.

Naslednja uporabljena metoda je metoda opazovanja. Z njo smo med delom opazovali odzive avtomobilčka na opravljene korake. Odzivi so nas usmerjali v nadaljnje delo, ki je bilo odvisno od naših želja in dejanskega odziva. V primeru nepravilnega delovanja smo morali program popravljati in ga dopolnjevati z novimi idejami.

Uporabili smo tudi metodo programiranja. Pri delu smo uporabili več programskih jezikov, kot so JavaScript, Python, PHP. Ta metoda dela nam je povzročala kar nekaj težav, saj večine omenjenih programskih jezikov v šoli nismo spoznali, zato smo se morali o njih podučiti sami preko spletnih strani.

## 3. OSREDNJI DEL NALOGE

### 3.1 Predstavitev rezultatov raziskovanja

Najprej smo se lotili programiranja z Raspberry pi-jem, saj je to obsegalo večji del naše naloge. Pri tem smo naleteli tudi na prve težave. Programiranje je potekalo v programskem jeziku Python, ki ga še nismo uporabljali in ga zato nismo dobro poznali. Da bi se bolje seznanili s tem programskim jezikom, smo se udeležili predavanj, ki so bila organizirana na naši šoli. Zaradi veliko novosti, smo morali znanje nadgraditi še doma, tudi s pomočjo spleta [vir 2]. Končni rezultat dela je napisan program, ki pregleduje tipkovnico. Program reagira na pritisk tipk (v našem primeru naprej, nazaj, levo, desno). Če prepozna katero izmed omenjenih tipk, zapiše v datoteko input.txt, katera je bila.

```
<?php
    $file = 'script/input.txt';

    /*
     * move:
     * - 1: Up
     * - 2: Down
     * - 3: Left
     * - 4: Right
     */

    if(isset($_GET['move'])) {
        $input = $_GET['move'];
        file_put_contents($file, $input);
    }

?>
```

Slika 1: PHP koda.

```

import RPi.GPIO as gpio
from time import sleep

def init(pin):
    gpio.setup(pin, gpio.OUT)

def turnon(pin):
    gpio.output(pin,False)

def turnoff(pin):
    gpio.output(pin,True)

def getStatus():
    f = open('../script/input.txt', 'r')
    return f.read(1)

### Main Program

up = 7
down = 8
left = 11
right = 15

print "script is starting..."

print "Initializing pins..."
init(up)
init(down)
init(left)
init(right)

print "Done."

while True:
    status = getStatus()

    if status == '1':
        print "Going up"
        turnOn(up)

    elif status == '2':
        print "Going Back"
        turnOn(down)

    elif status == '3':
        print "Going Left"
        turnOn(up)
        turnOn(left)

    elif status == '4':
        print "Going Right"
        turnOn(up)
        turnOn(right)

    else:
        turnOff(up)
        turnOff(down)
        turnOff(left)
        turnOff(right)

    sleep(0.1)

```

Slika 2: Program v Pythonu.

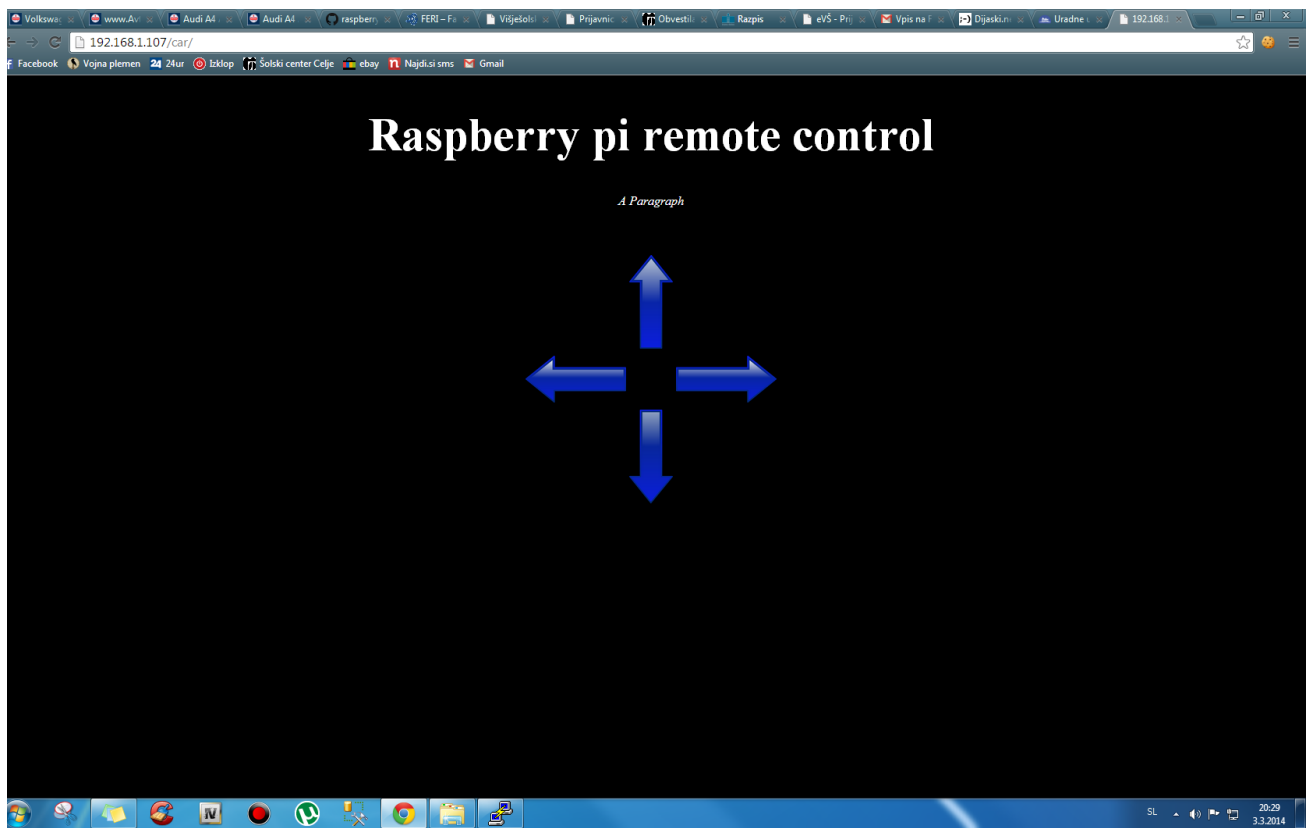
Podatek o pritisnjeni tipki javi Raspberry pi integriranemu vezju, ki smo ga vzeli iz daljinca in priključili nanj, kot je razvidno na spodnji sliki. Ta del nam je predstavljal precej težav, saj tega nismo še nikoli počeli in ker je to delo bolj elektrotehniške narave. Da smo delo pravilno izvedli, smo morali pregledati kar precej spletne literature [vir 1].



Slika 3: Vezje potrebno za delovanje.

Na zgornji sliki je razvidna izvedba napajanja Raspberry pi-ja z baterijo. Začetna ideja projekta je bila takšna, da bi bil Raspberry pi na avtomobilčku. Ta predpostavka se je kasneje izkazala za nepotrebno. Baterija v končni fazi napaja samo vezje daljinca, saj je to edini način napajanja. Na sliki se vidi tudi, da je na Raspberry pi priključen usb z wi-fi-jem. Ta usb služi za povezavo med Raspberry pi-jem in računalnikom, s katerim dostopamo do spletne strani Raspberry pi-ja. Ko smo sestavili »strojni del« našega projekta in ga podprli s Pythonovim programom, smo pristopili k izdelavi spletne strani. To je bilo nekoliko lažje, saj smo imeli za izvedbo tega koraka več predznanja kot pri Pythonu. Spletno stran smo začeli pripravljati v jeziku HTML, nekaj ključnih rešitev pa smo morali izvesti z uporabo JavaScripta [vir 8].

Spletno stran smo oblikovali tako, da smo na sredino postavili tipke, ki se ob pritisku na tipkovnici ustrezno označijo in tako vemo, da smo pritisnili tipko in tudi katero tipko smo pritisnili. Seveda smo jo nekoliko preuredili tudi vizualno. Pri tem so se začele pojavljati težave pri pisanju kode. Na začetku smo uporabljali samo HTML jezik, ki pa ne omogoča dinamičnega prikazovanja slik ob pritisku na tipko. Da smo dosegli želeni učinek, smo uporabili JavaScript, s katerim je spletna stran postajala vse bolj podobna našim zamislim.



Slika 4: Spletna stran.



```

<!DOCTYPE html>
<html>
<head>
<link rel="stylesheet" type="stylesheet/css" href="style.css">
<script src="http://ajax.googleapis.com/ajax/libs/jquery/1.10.2/jquery.min.js"></script>
<script>
$(document).ready(function() {
    $(".control").mousedown(function() {
        move($(this).attr("move"))
    })
    $(document).mouseup(function() {
        stop()
    });
});
keypressed = false
$(document).keydown(function(e) {
    if(!keypressed) {
        if (e.keyCode == 37) {
            keypressed = true
            move("left")
            return false;
        } else if(e.keyCode == 38) {
            keypressed = true
            move("up")
            return false;
        } else if(e.keyCode == 39) {
            keypressed = true
            move("right")
            return false;
        } else if(e.keyCode == 40) {
            keypressed = true
            move("down")
            return false;
        }
    }
});
$(document).keyup(function(e) {
    if (e.keyCode == 37 || e.keyCode == 38 || e.keyCode == 39 || e.keyCode == 40) {
        keypressed = false
        stop()
        return false
    }
});
function move(direction) {
    console.log("moving "+direction)
    $("."+direction).css("background-color", "green")

    moveit = 0
    if(direction == "up") {
        moveit = 1
    } else if(direction == "down") {
        moveit = 2
    } else if(direction == "left") {
        moveit = 3
    } else if(direction == "right") {
        moveit = 4
    }
    update(moveit)
}
}

```

Slika 5: Koda spletne strani 1.

```

function stop() {
    $(".control").css("background-color", "black")
    console.log("stopped")
    update(0)
}

function update(moveit) {
    $.ajax({
        url: "update.php?move="+moveit,
        context: document.body
    });
}
</script>
</head>
<body bgcolor="black">

<div class="header"><center><h1>Raspberry pi remote control</h1></center></div>

<center><p id="demo" style="color: white">A Paragraph</p></center>
<br>
<br>

<div class="body">
<center><table border="0">
<tr>
<td></td>
<td>
<center>

</center>
</td>
</tr>
<tr>
<td><center></center>
</td>
<td><center></center></td>
</tr>
<tr>
<td><center></center></td>
</tr>
</table></center>
</div>
</body>
</html>

```

Slika 6: Koda spletne strani 2.

Nato smo morali postaviti le še strežnik, na katerem bi bila prej opisana spletna stran. Ta strežnik je integriran v Raspberry pi, kamor pa je potrebno namestiti še spletno stran. Preko wi-fi komunikacije mora biti omogočen dostop računalnika do prej omenjenega strežnika in spletne strani.

Na že integriran Apache strežnik smo naložili spletno stran. Nato smo se z računalnikom s programom Putty povezali z Raspberry pi-jem. Spletne strani nam ni bilo potrebno zagnati, saj se zažene, ko zaženemo Raspberry pi. S programom Putty se povežemo z Raspberry pi-jem preko IP naslova. Če želimo na računalniku prikazati spletno stran, v brskalnik napišemo ta IP naslov in spletna stran se prikaže. Preko programa Putty z vpisom kode, ki je prikazana na spodnji sliki zaženemo program v Pythonu [vir 3].

```

pi@RaspberryPi10: /var/www/car/python
login as: pi
pi@192.168.1.107's password:
Linux RaspberryPi10 3.6.11+ #538 PREEMPT Fri Aug 30 20:42:08 BST 2013 armv6l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon Mar  3 20:29:22 2014 from 192.168.1.102
pi@RaspberryPi10 ~ $ cd /var/www/car/python
pi@RaspberryPi10 /var/www/car/python $ sudo python script.py
Script is starting

```

Slika 7: Zagon pythona v programu Putty.

## 3.2 Delovanje

Preko spleta komuniciramo z Raspberry pi-jem tako, da ob kliku na tipke za premikanje sprožamo JavaScript metode, ki se sprožijo ob pritisku in spustu posamezne tipke. Te metode preko AJAX-a (sprožanje klicev na PHP v ozadju) povedo PHP-ju, kaj se na spletni strani dogaja.

Te metode so keydown in keyup za vnos preko tipkovnice in mouseDown, mouseUp za klik z miško po sliki.

Torej, če pritisnemo 'gor' na tipkovnici, se preko AJAX-a v PHP pošlje koda '1', kar pomeni 'Naprej' in PHP to zapiše v posebno datoteko 'input.txt', ki služi za komunikacijo med PHP in programom v Pythonu.

Program v Pythonu je napisan tako, da 10x v sekundi prebere, kaj je shranjeno v datoteki input.txt. Glede na njeno vsebino daje signale v GPIO priključke, ki premikajo avtomobilček preko daljinca.

Če je v 'input.txt' zapisana 1, bo Python aktiviral priključke, ki krmilijo in poganjajo avtomobilček naprej. Če je zapisana 2, se bo avto premikal nazaj in tako dalje.

Ko na spletni strani tipko spustimo, se preko AJAX-a v PHP pošlje koda 0, kar pomeni stop. PHP prenese podatek v datoteko input.txt in Python preko GPIO deaktivira priključke za premikanje avta. Slednji se zato ustavi.

### 3.3 Razprava

Upravljanje deluje kot smo si zamislili, vendar pa do rešitve nismo prišli tako preprosto, kot smo si zamislili na začetku. Naleteli smo na kar veliko težav, ki smo jih morali postopno reševati; načrt dela smo morali sproti nekoliko prilagajati.

Na prvo težavo smo naleteli, ko smo hoteli Raspberry pi priključiti na avtomobilček, nato pa smo ugotovili, da to ni izvedljivo. Raspberry pi smo nato priključili na krmilnik avtomobilčka in težava je bila odpravljena.

Naslednja težava je bila z napajanjem Raspberry pi-ja. Mislili smo, da mora biti pritrjen na avtomobilčku, zato bi ga morali napajati brezžično. Ker nismo vedeli, kaj narediti in ker bi za to morali odšteti kar nekaj denarja, smo poiskali drugo rešitev; prišli smo do spoznanja, da Raspberry pi ni potrebno priključiti na avtomobilček, saj je lahko kjerkoli v dometu wi-fi-ja. Zato je lahko priključen na računalnik preko USB kabla.

Naleteli smo tudi na težavo s krmiljenjem, ki pa je nismo znali odpraviti. Težava je v tem, da ko pritisnemo tipko v levo oz. desno, kolesa ne dobijo dovolj energije, da bi se popolnoma zasukala, zato se zasučejo samo do polovice in tako avto ne zavija pravilno. Ta težava je nastala zato, ker smo v programu zapisali, da v primeru, da je pritisnjena tipka levo oz. desno se ob tem izvede še funkcija za naprej. Ker se izvedeta dve dejanji hkrati, se energija razporedi in tako zavijanje ne deluje tako, kot bi moralo.

Veliko breme nam je predstavljala tudi povezava krmilnika z Raspberry pi-jem, saj je bil ta del naloge bolj mehanske kot pa programerske narave, vendar smo si preko spleta ogledali veliko video posnetkov in forumov ter ugotovili, da težava le ni tako velika. Raspberry pi in krmilnik smo povezali z žičkami, ki so bile na Raspberry pi in krmilnik spojene s spajkanjem, ki smo se ga naučili v 1. letniku.

Prvo tezo smo zaradi težav ovrgli, drugo pa potrdili. Naš pogled na rešitev je bil pred začetkom nekoliko drugačen. Do rešitve smo namreč prišli drugače, kot smo sprva načrtovali. Glede na postavljene teze je projekt vseeno uspel, saj je končni rezultat, upravljanje z avtomobilčkom preko spletne strani, realiziran.

#### **4. ZAKLJUČEK**

Izdelek nam je uspelo uspešno realizirati in tako opraviti projektno nalogo. Preko tipkovnice lahko upravljamo avtomobilček. Tezo, ki smo jo na začetku postavili, lahko delno potrdimo, saj ima končni izdelek nekaj sprememb glede na zamislili na začetku. Raspberry pi smo priključili na krmilnik in ne neposredno na avtomobilček, tako da avtomobilček ni fizično priklopljen. Raspberry pi se napaja preko računalnika in ne brezžično, tako kot smo mislili na začetku. Dodelati bi morali še zunanost izdelka - Raspberry pi in elektroniko daljinskega upravljanja z baterijskim napajanjem bi morali vgraditi v ustrezno ohišje, da bi bil končni izdelek bolj estetski.

## **5. ZAHVALA**

Zahvaljujemo se profesorjem, ki so nam pomagali z odgovori na naša vprašanja in s tem pripomogli k reševanju problemov, s katerimi smo se srečali pri izdelavi. Zahvaljujemo se tudi našemu prijatelju, ki nam je pomagal pri delu, ki ne zadeva naše stroke.

## 6. VIRI

1. Hobbytronics: Raspberry pi pini. [Citirano 24.2.2014; 14:55]. Dostopno na spletnem naslovu: <http://www.hobbytronics.co.uk/image/data/tutorial/Raspberry-pi/gpio-pinout.jpg>
2. Optimal: Spletna stran za pomoč pri programskem jeziku Python. [Citirano 22.2.2014; 16:12]. Dostopno na spletnem naslovu: <http://optimal.si/>
3. Putty. [Citirano 27.2.2014; 16:23]. Dostopno na spletnem naslovu: <http://www.cae.titech.edu/help/remote-access/windows-putty>
4. Raspberry pi: Raspberry pi model. [Citirano 24.2.2014; 14:10]. Dostopno na spletnem naslovu: <http://www.Raspberrypi.org/faqs>
5. Slika puščice navzgor. [Citirano 20.2.2014; 12:30]. Dostopno na spletnem naslovu: [http://4vector.com/i/free-vector-arrowupblueclipart\\_116730\\_Arrowupblue\\_clip\\_art\\_hight.png](http://4vector.com/i/free-vector-arrowupblueclipart_116730_Arrowupblue_clip_art_hight.png)
6. Slika puščice navzdol. [Citirano 20.2.2014; 12:35]. Dostopno na spletnem naslovu: [http://4vector.com/i/free-vectorarrowdownblueclipart\\_116746\\_Arrowdownblue\\_clip\\_art\\_hight.png](http://4vector.com/i/free-vectorarrowdownblueclipart_116746_Arrowdownblue_clip_art_hight.png)
7. Slika puščice levo. [Citirano 20.2.2014; 12:41]. Dostopno na spletnem naslovu: [http://4vector.com/i/free-vector-arrow-left-blue-clip-art\\_116741\\_Arrowleftblue\\_clip\\_art\\_hight.png](http://4vector.com/i/free-vector-arrow-left-blue-clip-art_116741_Arrowleftblue_clip_art_hight.png)
8. Slika puščice desno. [Citirano 20.2.2014; 12:48]. Dostopno na spletnem naslovu: <http://www.downloadclipart.net/large/514-arrow-right-blue-design.png>
9. W3Schools: Spletna stran za pomoč pri programskih jezikih. [Citirano 21.2.2014; 13:19]. Dostopno na spletnem naslovu: <http://www.w3schools.com/>
10. Youtube: Raspberry pi avtomobilček. [Citirano 22.2.2014; 14:10]. Dostopno na spletnem naslovu: <http://www.youtube.com/watch?v=JSP6VKiU7F4>