

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

ZAVRŠNI RAD br. 4238

**METEOROLOŠKA STANICA TEMELJENA
NA UMREŽENOM UGRADBENOM
RAČUNALU**

Vicko Čagalj

Zagreb, lipanj 2015.

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA
ODBOR ZA ZAVRŠNI RAD MODULA

Zagreb, 13. ožujka 2015.

ZAVRŠNI ZADATAK br. 4238

Pristupnik: **Vicko Čagalj (0036474848)**
Studij: Elektrotehnika i informacijska tehnologija
Modul: Elektroničko i računalno inženjerstvo

Zadatak: **Meteorološka stanica temeljena na umreženom ugradbenom računalu**

Opis zadatka:

U okviru završnog rada potrebno je razviti meteorološku stanicu temeljenu na umreženom ugradbenom računalu. Stanica mora ulazne veličine preuzimati s odgovarajućih senzora korištenjem prirodnih sučelja. Odabrati i nabaviti pogodne senzore s naglaskom na male troškove. Programska potpora treba izmjerene veličine sa senzora pretvoriti u odgovarajuće fizikalne veličine uz eventualnu kalibraciju. Potpora treba osigurati periodično očitavanje svih veličina i pohranu u datoteku uz vremensku oznaku iz sada stvarnog vremena. Istražiti mogućnost realizacije korisničkog međusklopa korištenjem dinamičkog WEB servera za udaljeni pristup. Ugradbeno računalo temeljiti na Raspberry PI platformi s pripadnim razvojnim programskim okruženjem.

Zadatak uručen pristupniku: 13. ožujka 2015.
Rok za predaju rada: 12. lipnja 2015.

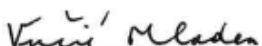
Mentor:

Prof. dr. sc. Davor Petrinović

Djelovoda:

Izv. prof. dr. sc. Dražen Jurišić

Predsjednik odbora za
završni rad modula:


Prof. dr. sc. Mladen Vučić

Sadržaj

1.	Uvod	1
2.	Raspberry Pi	2
	2.1 Povijest	2
	2.2 <i>Hardware</i>	3
	2.3 Operacijski sustav i programiranje	5
	2.4 Primjena	6
3.	Meteorološka stanica	9
	3.1 Opis problema	9
	3.2 Komponente i njihovo spajanje	10
	3.3 Instalacija operacijskog sustava i podešavanje postavki	14
	3.4 Komunikacija putem SSH protokola	15
	3.5 Apache web server	17
	3.6 Prijenos skripti s laptopa na Raspberry Pi	20
	3.7 Testiranje senzora	21
	3.7.1 DHT22	21
	3.7.2 BMP180	23
	3.8 Glavna skripta	25
4.	Zaključak	33
5.	Literatura	34
6.	Sažetak	35
7.	Abstract	36

1. Uvod

U današnje vrijeme gotovo je nemoguće zamisliti život bez tehnologija koje su se utkale u sve pore modernog društva. Velikim razvojem informatičke industrije dolazi do proširenja ponude čime proizvodi postaju dostupni širem spektru ljudi. Pretpostavlja se da danas u svijetu postoji oko 1,5 milijardi računala, a ta bi se brojka do kraja 2016. godine mogla podebljati na više od 2 milijarde. Zadnjih nekoliko godina, tehnologija teži tome da računala budu što manja i jeftinija, a da pritom nastoje pružiti što više mogućnosti. Jedno od takvih je i Raspberry Pi računalo, korišteno u obrazovanju u nadi da će približiti svijet tehnologija mlađim generacijama s idejom da se pokaže kako računala rade i kako ih programirati te da se znanje rada na računalima ne svodi samo na znanje korištenja aplikacija. Njegova pojava izazvala je oduševljenje širom svijeta, kako svojim dimenzijama i mogućnostima, tako i vrlo pristupačnom cijenom.

Cilj ovoga rada je upoznavanje s ovim malim čudom tehnike i izgradnja meteorološke stanice temeljene na njegovom povezivanju sa sensorima.

2. Raspberry Pi

2.1 Povijest

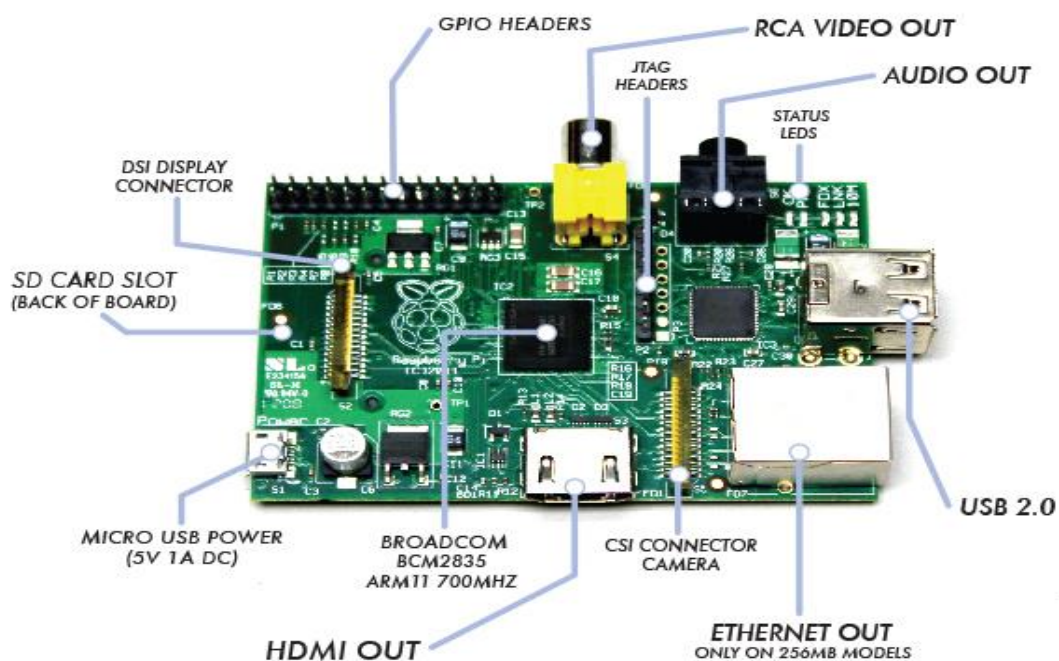
Ideja o malom i pristupačnom računalu javila se 2006. godine, kada su Eben Upton, Robb Mullins, Jack Lang i Alan Mycroft sa Sveučilišta u Cambridgeu postali zabrinuti nivoom predznanja studenata koji su se prijavljivali na računarske predmete. Za razliku od 1990. godine kada je većina kandidata imala solidno predznanje iz programiranja, 2000. godine kandidati su bivali sve lošiji. Zaključili su da se formirala grupa informatičara s vrlo malo praktičnog programerskog znanja potrebnog za ispravno rješavanje konkretnih problema. Oni su imali iskustva u korištenju kućnog računala ili konzole za igru i zabavu. Iskustvo stečeno svakodnevnim korisničkim radom na računalu davalo im je lažnu sliku o znanju iz računarstva. Prethodne generacije učile su programiranje na legendarnim ZX Spectrum, Commodore 64 ili Amiga računalima, koji su osim mogućnosti zabave nudili i dobro okruženje za pisanje prvih programa. Zbog toga su došli na ideju da naprave nešto što je učenicima pristupačno po cijeni, a pruža okruženje spremno za pisanje programa. Od 2006. - 2008. godine dizajnirano je nekoliko verzija od kojih je nastalo računalo Raspberry Pi. Kada su se 2008. godine pojavili procesori koji su se mogli koristiti za mobilne uređaje, pristupačni po cijeni i dovoljno snažni za dobru podršku grafičkom okruženju, postalo je izvjesno da će projekt zaživjeti.

Kada se na tržištu pojavio prvi primjerak računala, svojim izgledom i mogućnostima privukao je veliku pažnju, kako kod programera, tako i kod entuzijasta, stoga postoji mnogo opreme, a na internetu je objavljeno mnoštvo konkretnih objašnjenja i praktičnih uputstava.

2.2 Hardware

Stariji modeli posjeduju Broadcom BCM2835 sustav na integriranom krugu, koji uključuje ARM1176JZF-S 700 MHz procesor, VideoCore IV GPU i originalno je imao 256 MB RAM, ali kasnije dolazi s 512 MB. Nemaju ugrađen hard disk ili SSD (*solid-state drive*), već koriste SD kartice za *bootanje* i pohranu podataka. Model A ima samo jedan USB ulaz i nema Ethernet kontroler, ali se može spojiti na mrežu preko USB Etherneta ili Wi-Fi uređaja. Model B posjeduje dva USB utora i 10/100 Ethernet kontroler. Kao i na ostala računala na Raspberry Pi se mogu spojiti generički USB miševi i tipkovnice. Raspberry Pi ne dolazi s ugrađenim satom u BIOS-u, već operativni sustav koristi vrijeme mrežnog servera ili se mora ručno unositi prilikom paljenja računala. Broadcom SoC (*System on a chip* – Sustav na integriranom krugu) je ekvivalentan mikroprocesorima korištenima u starijim smartphone uređajima. Što se tiče centralnog procesora (takt 700 MHz) on je ekvivalentan procesoru od 300 MHz Pentium II (1997.-1999. godina), a grafički procesor je ekvivalentan Xbox uređaju (2001. godina). Raspberry Pi ne treba dodatni hladnjak ili ikakvo posebno rashlađivanje. Detaljne specifikacije uređaja su prikazane u nastavku, uz naznačene razlike među modelima:

- SoC: Broadcom BCM2835
- CPU: 700 MHz, ARM1176JZF-S jezgra
- GPU: Broadcom VideoCore IV 250 MHz
- Memorija: 512 MB, dijeljeno s GPU (model A ima samo 256 MB)
- USB 2.0 ulazi: 2 (model A ima samo 1)
- Video ulaz: konektor za kameru, odnosno slikovni senzor
- Video izlaz: kompozitni RCA (PAL i NTSC), HDMI (verzija 1.3 i 1.4), podržava Full HD rezoluciju od 1080p
- Audio izlaz: 3.5 mm jack, HDMI
- Pohrana podataka: ulaz za SD/MMC/SDIO kartice (3.3 V napajanje)
- Mreža: 10/100 Mbit/s Ethernet USB (Model A nema ovu mogućnost)
- Dodaci: 8 x GPIO, UART itd.
- Napajanje: 5 V preko MicroUSB-a ili GPIO-a



Slika 2.2.1 Raspberry Pi model B

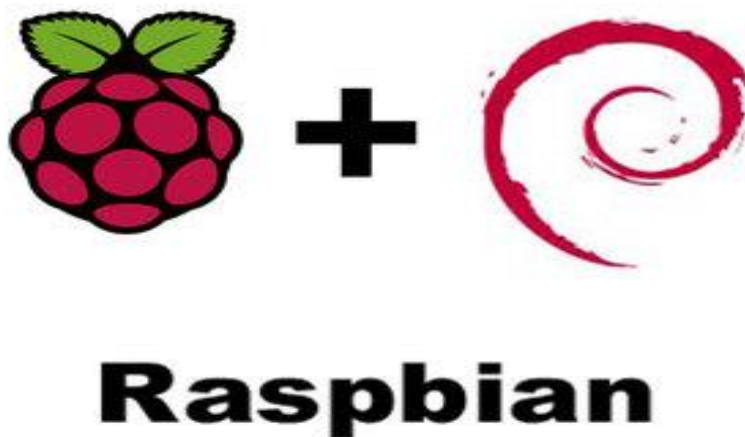
U svom radu koristit ću najnoviji model Raspberry Pi 2. Za razliku od prethodnika odlikuje se ugrađenim procesorom Broadcom BCM2836 s 4 ARM Cortex-A7 jezgre koje rade na taktu od 900 MHz u pratnji 1 GB SDRAM-a što ga čini 6 puta bržim. Izgled jednog takvog modela prikazan je na slici 2.2.2.



Slika 2.2.2 Raspberry Pi 2

2.3 Operacijski sustav i programiranje

Raspberry Pi ne dolazi s predinstaliranim operativnim sustavom. Operativni sustav se instalira na SD karticu (zapremina može ići do 32 GB i dalje), ali prije toga se mora skinuti *image* sustava koji želimo instalirati. Potreban je program koji će instalirati operativni sustav na SD karticu. To se može napraviti na Windowsima, Linuxu ili Mac OS X-u. Na službenim stranicama postoje programi i upute, ali također i *image* određenih operativnih sustava. Raspbian možemo smatrati glavnim operativnim sustavom. Dolazi od riječi Raspberry Pi i Debian, a riječ je o Debian baziranoj Linux distribuciji optimiziranoj upravo za Raspberry Pi. Spomenuti OS omogućuje skidanje više tisuća softverskih paketa, ima ugrađene mehanizme za laku integraciju i optimiziran je za ARM11 procesor kojeg uređaj koristi. Raspbian i Raspberry Pi su odlična edukacijska platforma.



Slika 2.3.1 Logo Raspbian operacijskog sustava

IDLE, koji dolazi s operacijskim sustavom, je IDE za programski jezik Python. Često se postavlja pitanje zašto je Python izabran kao jezik za upotrebu na ovom mini računalu. Ima ugrađene funkcije kao i ostali alati namijenjeni programiranju, osvjetljavanje sintakse, automatsko uvlačenje, prijelomne točke itd. On je možda i najbolji programski jezik dosta visokih mogućnosti, a razumljiv je i prilagođen apsolutnim početnicima kako bi postepeno upoznali koncepte i metode na kojima su zasnovani moderni programski jezici. S Raspbianom dolazi i programski jezik

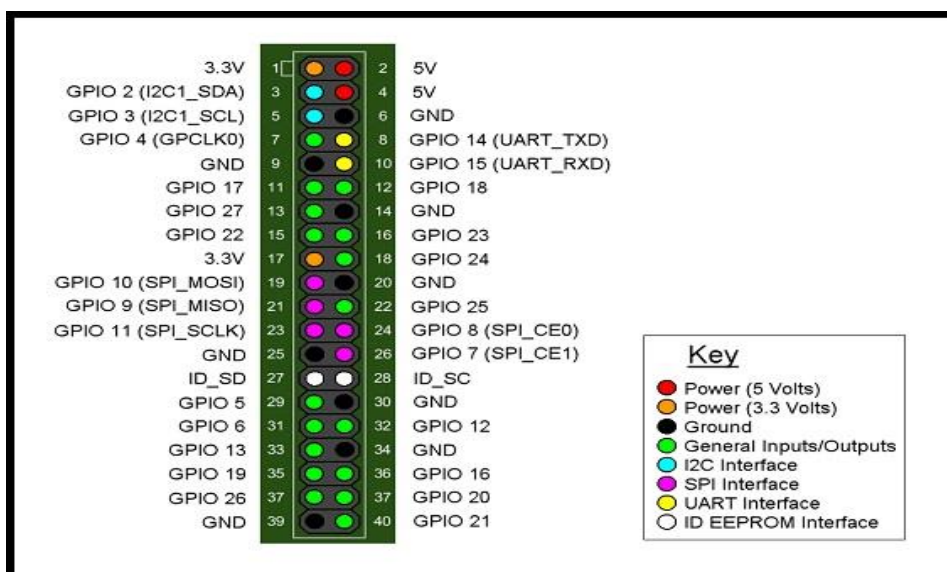
Scratch koji je namijenjen najmlađima koji se prvi put susreću s programiranjem. Služi za izradu interaktivnih priča, glazbe, filmova i sl. Raspbian je optimiziran, ima podršku za ogroman broj aplikacija, ali i neke nedostatke. Podržava dosta malen broj audio/video formata i korisnici se žale kako je dosta teško postaviti Raspberry Pi da radi kao server.

2.4 Primjena

Neke primjene sam već spominjao u ranijem tekstu, a u ovom poglavlju ćemo detaljnije opisati neke od najzanimljivijih u elektronici i robotici.

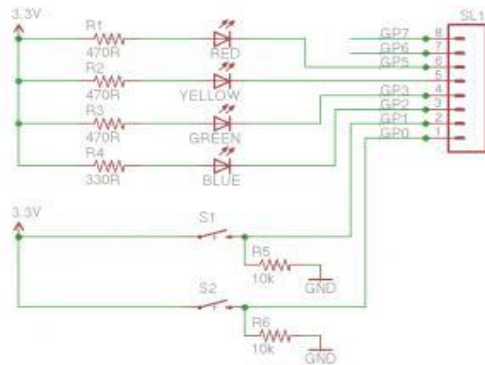
Raspberry Pi je jako efikasan način uvođenja djece i studenata u osnove elektronike. Tako primjenjuju računalo direktno, uče izraditi kontrolne ploče i shvaćaju da se njihov rad na računalu može projicirati u stvarni svijet. Zahvaljujući velikoj bazi podataka temeljenoj na Pythonu, moguće je izraditi mnoštvo jednostavnih projekata. Primjer jednog takvog dan je u nastavku.

Najnoviji model Raspberry Pi ima čak 26 GPIO pinova (prikazani na slici 2.4.1) s promjenjivim funkcijama. Svi su prilagođeni za *INPUT* i *OUTPUT* način rada.



Slika 2.4.1 Raspored pinova na Raspberry Pi 2

Moramo biti oprezni s naponom i jakosti struje koju puštamo, jer su pinovi spojeni direktno na glavni integrirani krug, stoga može doći do oštećenja uređaja. Prvo instaliramo RPi.GPIO koji je dostupan besplatno na Internetu i relativno je jednostavan za rad. Koristimo četiri LED žarulje različitih boja i dvije tipke.



Slika 2.4.2 Električna shema i izgled spoja s 3 LED žarulje i 2 tipke [3]

Nakon toga je dovoljno upisati tri linije koda:

```
$ sudo python
>>> import RPi.GPIO as GPIO
>>> GPIO.setup(18, GPIO.OUT)
>>> GPIO.output(18, False)
```

Rezultat izvršavanja koda prikazan je na slici 2.4.3.

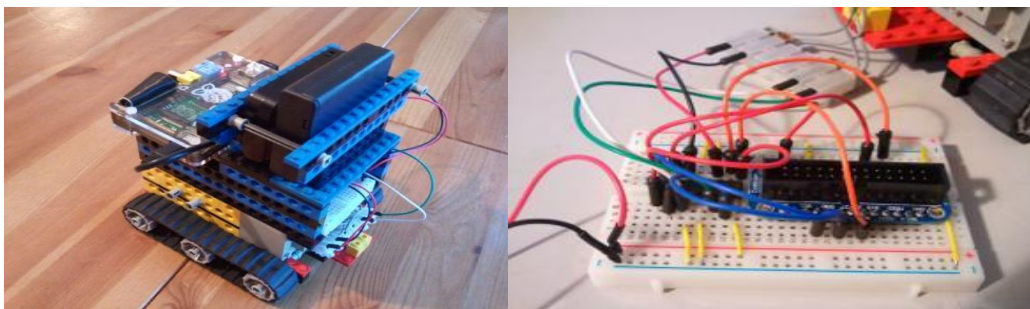


Slika 2.4.3 Svijetli crvena LED dioda [3]

Tema robotike je jako popularna u današnjem svijetu. Djecu i studente učimo konstrukciji strojeva i čemu služe pojedini dijelovi od serva do motora. Nadalje, na

taj način možemo objasniti kako zapravo centralno računalo, u našem slučaju Raspberry Pi, procesira i kontrolira svaki dio robota. Također, za robotiku možemo naći veliku količinu podataka za rad, od baza podataka, uputa itd. Robotika je jako pogodna za edukaciju jer se mora koristiti znanje programiranja, elektronike, konstruiranja te tako objedinjujemo veliki broj područja.

Pokušat ću dočarati samo dio mogućnosti spajanja Raspberry Pi i područja robotike s LEGO robotom. Kućište robota je u potpunosti napravljeno od LEGO kocaka u čijem se središtu nalazi Raspberry Pi računalo. Potrebna nam je i kontrolna kutija za elektroniku. U ovom slučaju moramo upotrijebiti svoje znanje o GPIO pinovima i osnova elektronike. Što se tiče zvuka za to mogu poslužiti bilo koji zvučnici i jedan od besplatnih programa za sintezu govora. Kamera se može koristiti, a službena Raspberry Pi kamera daje dobru kvalitetu slika. Nakon spajanja bi sve trebalo izgledati ovako:



Slika 2.4.4 Spoj robota i Raspberry Pija [4]

Programiranje samog robota je dosta jednostavno budući da postoji veliki broj gotovih modula spremnih za implementaciju. Primjer jedne takve:

```
Import robot  
myRobot = robot.Robot()  
myRobot.forward(2)  
myRobot.right(3)  
myRobot.takePhoto()  
myRobot.say(" Thank you for your photo ")
```

Raspberry Pi se može koristiti i za izradu superračunala, u medicini, pri automatizaciji u kućanstvu, kao BitTorrent Server, Media Server te u mnogim drugim zanimljivim primjenama. Jedna od njih je i tema moga završnog rada – meteorološka stanica ostvarena umrežavanjem Raspberry Pija i senzora.

3. Meteorološka stanica

3.1 Opis problema

Od davnih vremena ljudi su zaokupljeni vremenom i vremenskom prognozom. Nekoć su ljudi promatrali prirodu da saznaju kakvo će biti vrijeme. Gledali su nebo, proučavali ponašanje biljaka i životinja te na temelju toga donosili zaključke. Razvojem znanosti, više se ne moramo oslanjati na prirodu te su dovoljne male meteorološke stanice koje nam daju očitavanja u vidu brzine i smjera vjetra, temperature, vlažnosti i tlaka kako bi mogli predvidjeti vrijeme za neposrednu budućnost. Raspberry Pi je upravo savršen za stvaranje jedne takve jer ne iziskuje mnogo računalne snage, te je u mogućnosti komunicirati s mrežom senzora. Jednostavnim očitavanjem podataka sa svakog senzora u round-robin stilu, moguće je dobiti točnu sliku o vremenu svakom trenutku.

U okviru završnog rada potrebno je razviti meteorološku stanicu temeljenu na umreženom ugradbenom računalu. Podaci se dohvaćaju putem senzora, vrše se periodička očitavanja čiji se rezultati u konačnici prikazuju putem WEB servera.



Slika 3.1.1 Meteorološka stanica

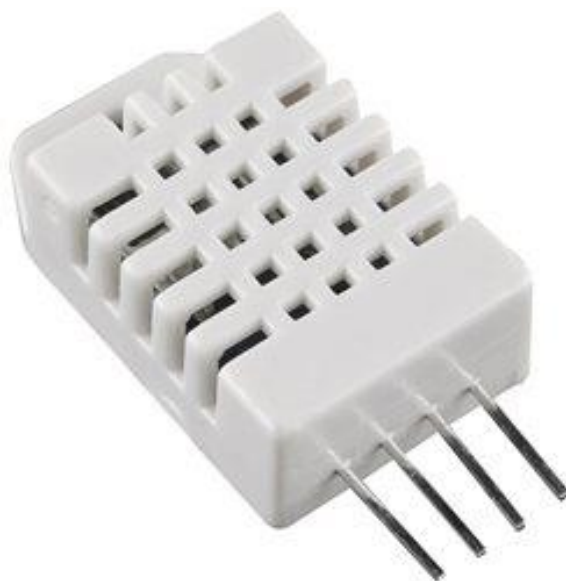
3.2 Komponente i njihovo spajanje

Pri konstrukciji meteorološke stanice potrebne su nam sljedeće komponente:

1. **Raspberry Pi 2**, prethodno opisan

2. **DHT22** – senzor za mjerenje temperature i vlage s preciznijim i točnijim očitanjima od njegovog prethodnika DHT11. Neke od karakteristika:

- Niska cijena
- Napon: **3–5 V**
- Raspon vlage: **0-100 %** s točnošću očitavanja **2-5 %**
- Raspon temperature: **-40 do 80 °C** s točnošću očitavanja **±0.5 °C**
- Veličina: **27 mm x 59 mm x 13.5 mm (1.05" x 2.32" x 0.53")**
- 4 pina
- Frekvencija uzorkovanja: **0.5 Hz** i manja



Slika 3.2.1 DHT22

3. BMP180 – senzor za mjerenje temperature, tlaka i nadmorske visine.

Novija je verzija senzora BMP085. Neke od karakteristika su:

- Niska cijena
- Niska potrošnja energije
- Raspon tlaka: **300-1100 hPa**
- Raspon temperature: **-40 do 80 °C** s točnošću očitavanja **±2 °C**
- Raspon nadmorske visine: **-500 do 9000 m**
- Napon: **3–5 V**
- Koristi I²C 7-bitnu adresu **0x77**



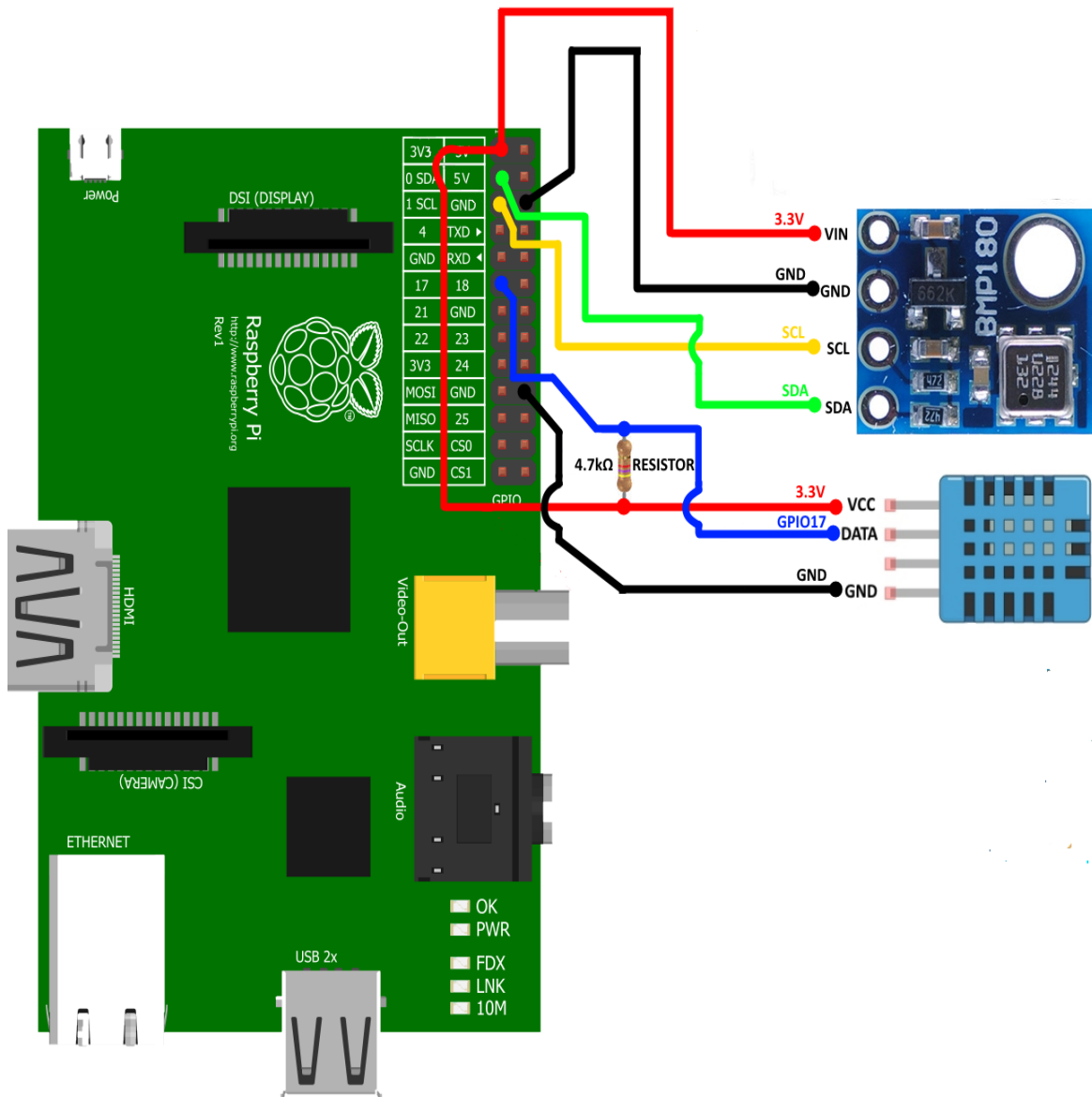
Slika 3.2.2 BMP180

4. Brojne druge komponente kao što su:

- Tipkovnica
- HD monitor
- Miš
- Eksperimentalna pločica
- Otpornik od **4.7 kΩ**
- Kratkospojnici
- WiFi adapter
- SD kartica od **16 GB**

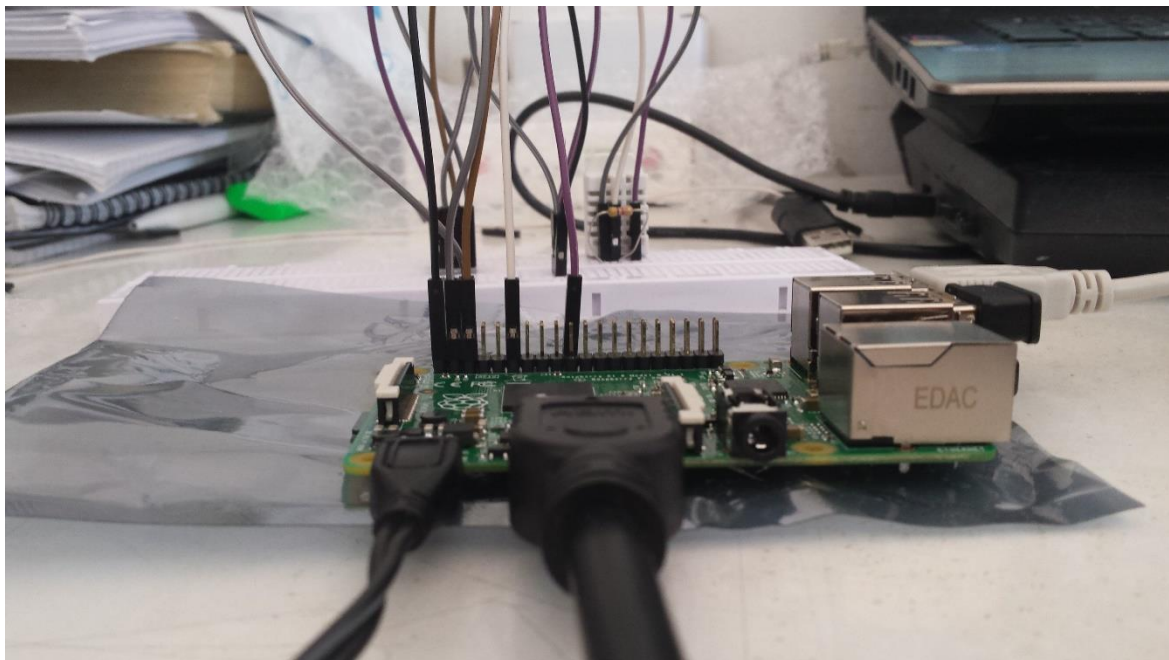
- Adapter za napajanje **5 V/2 A**
- HDMI i ETHERNET kabel

Raspored spajanja senzora DHT22 i BMP180 na Raspberry Pi pinove je prikazan na slici 3.2.3.



Slika 3.2.3 Spajanje senzora s Raspberry Pijom

Nakon što su sve komponente uspješno spojene, izgled meteorološke stanice je sljedeći:



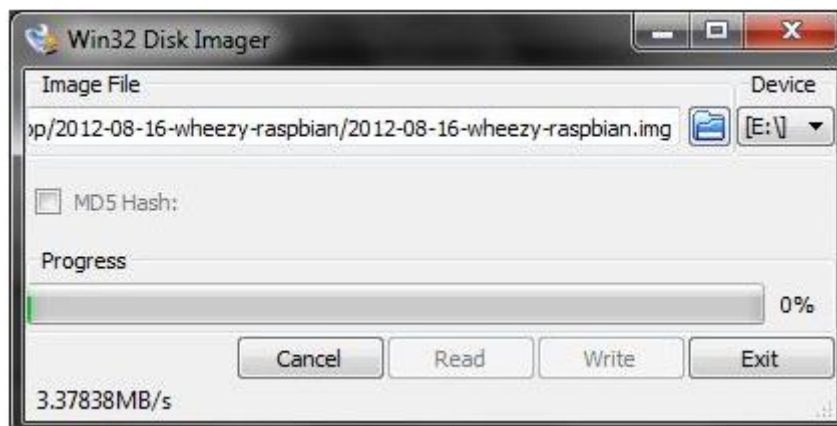
3.2.4 a) Izgled meteorološke stanice



3.2.4 b) Izgled meteorološke stanice

3.3 Instalacija operacijskog sustava i podešavanje postavki

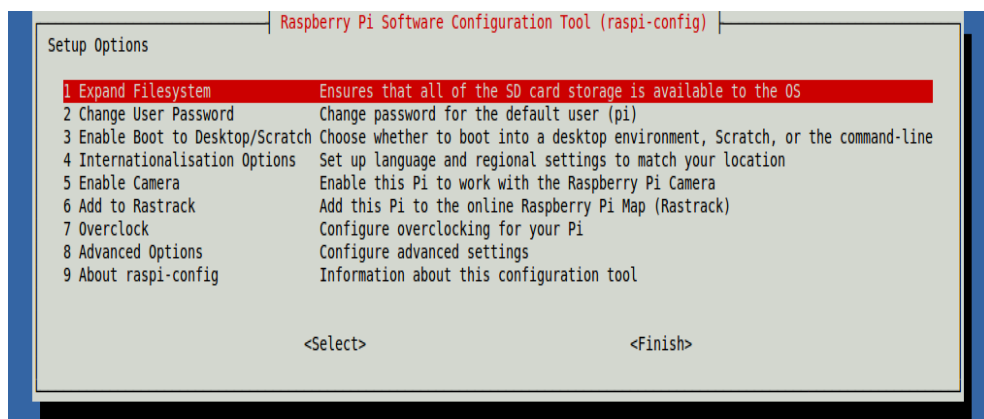
Formatiranje SD kartice izvedeno je putem programa Win32 Disk Imager. Tijek instalacije je prilično intuitivan. Prvo je potrebno s Raspberry Pi službene stranice (<https://raspberrypi.org/downloads>) skinuti *image* Raspbian operacijskog sustava. Nakon toga slijedi umetanje SD kartice, otvaranje programa Win32 te učitavanje *imagea* (izgled prozora je vidljiv na slici 3.3.1).



Slika 3.3.1 Formatiranje SD kartice

Nakon što je zapisivanje na SD karticu završeno, Raspberry Pi je spreman za prvo *bootanje*.

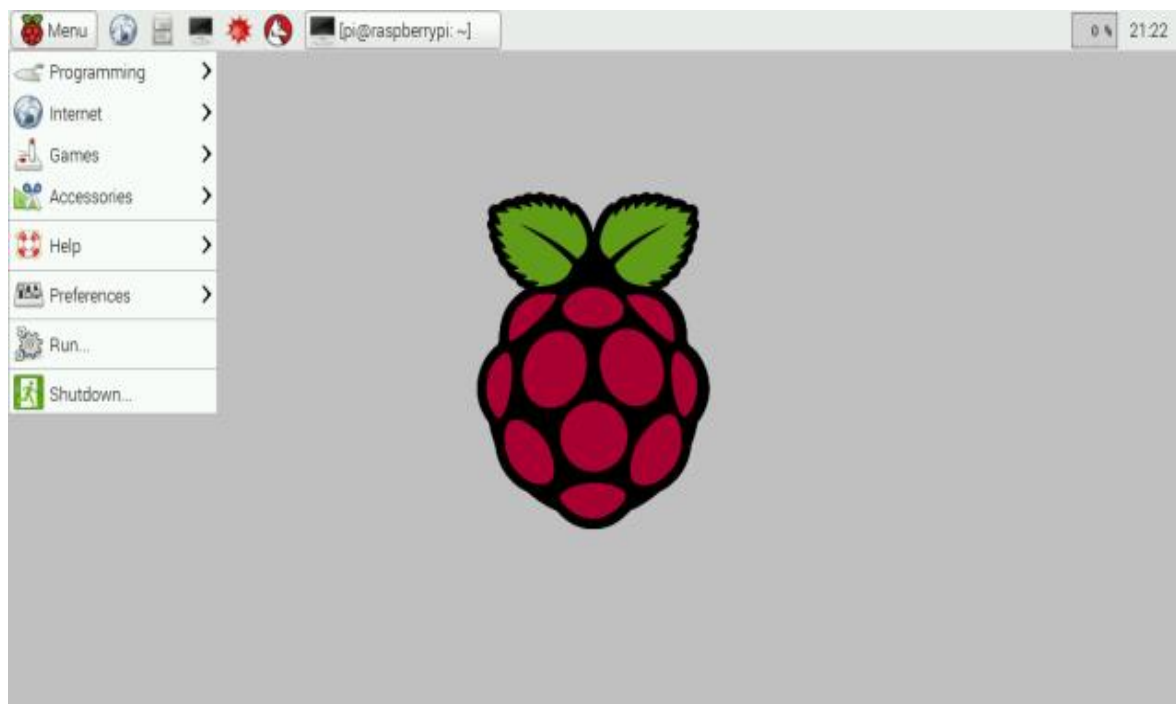
Prilikom prvog pokretanja računala s osnovnim Raspbian operacijskim sustavom, automatski se prikazuje ekran na kojem se nalazi popis parametara koje je moguće prilagoditi individualnim potrebama korisnika.



Slika 3.3.2 Podešavanje postavki

Prva stvar koju bi trebali napraviti jest proširiti *root* particiju na dostupnu veličinu SD kartice. Tako se za korištenje dobije puni kapacitet kartice, ako nemamo dodatni vanjski disk. Od ostalih bitnijih stvari tu su podešavanje jezika i regionalnih postavki, vremenske zone, raspored tipki na tipkovnici te postavke imena računala (*hostname*) i ssh servisa. Po završetku sustav treba ponovno pokrenuti i spremni smo za rad. Ako ih niste promijenili na konfiguracijskom prozoru, podaci za prvu prijavu na sustavu su "pi" za korisničko ime i "raspberry" kao lozinka. Konfiguracijski prozor možemo uvijek pozvati naredbom ***sudo raspi-config*** u LX terminalu.

Naredbom ***startx*** pokrećemo grafičko sučelje te možemo krenuti s upoznavanjem Raspbian operacijskog sustava.



Slika 3.3.3 Grafičko sučelje Raspbian operacijskog sustava

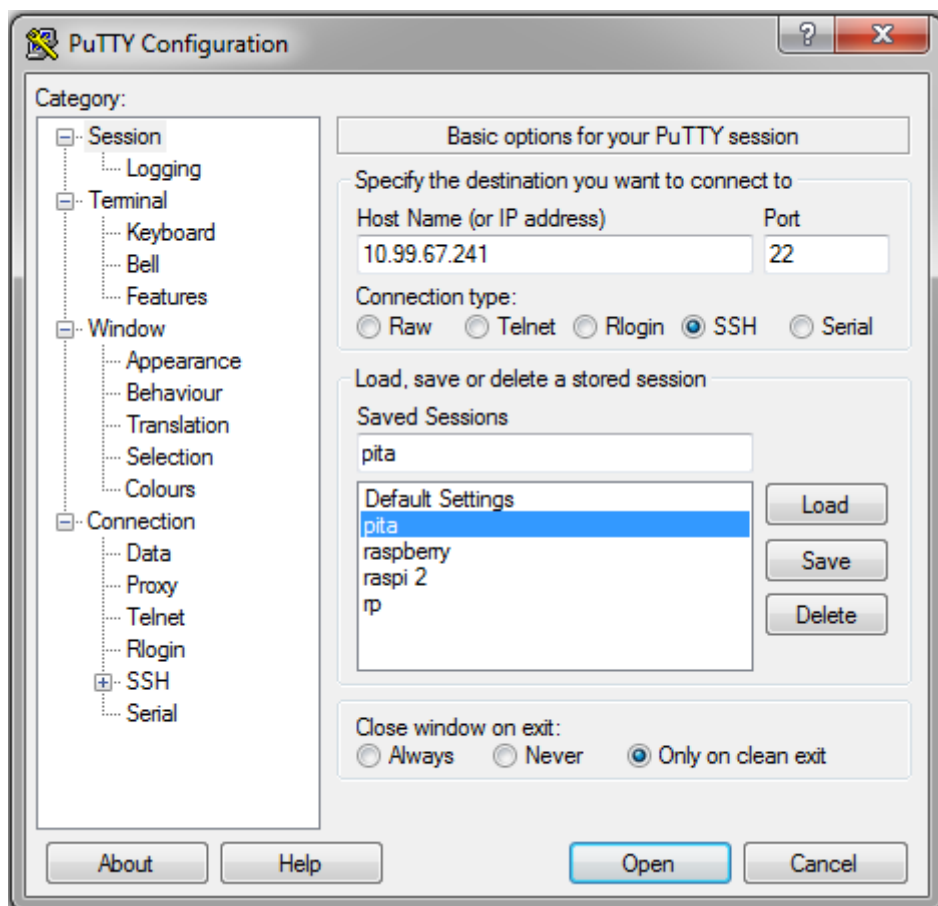
3.4 Komunikacija putem SSH protokola

SSH je mrežni protokol koji omogućuje da se između dva računala uspostavi sigurni (kriptirani) kanal. Prisluškivanjem toka podataka kroz takav kanal, ne mogu se doznati stvarni podaci već samo kriptirani. SSH se obično koristi za prijavu s jednog računala na drugo putem tekstualnog sučelja čime se može ostvariti i daljinska

administracija sustava. No SSH omogućuje i tuneliranje, tj. preusmjeravanje prometa s nekog drugog TCP-porta kroz sigurni kanal, a i kompresiju podataka koji se prenose takvim kanalom (izvor: wiki.open.hr).

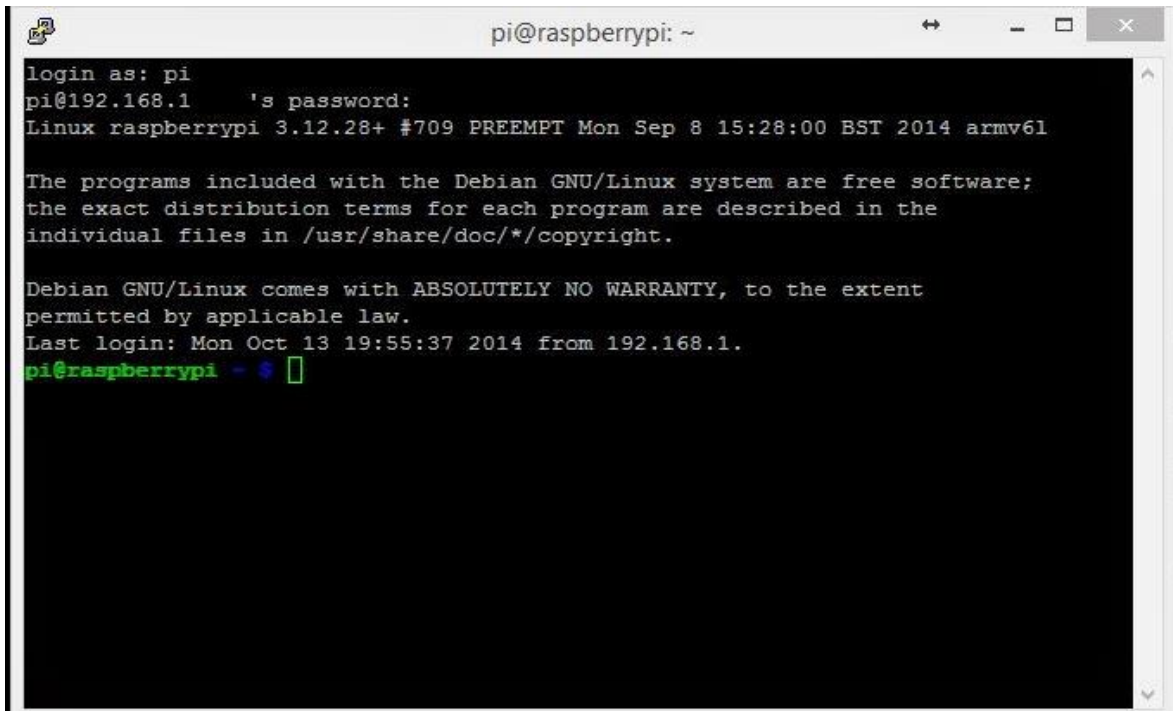
Koristit ćemo SSH kako bi povezoao svoj laptop s Raspberry Pi računalom. Razlog tome je mogućnost slanja potrebnih skripti s laptopa na Raspberry i lakša implementacija programskog dijela zadatka. Pri povezivanju mi je potreban ETHERNET kabel kako bi saznao IP adresu dodijeljenu Raspberryju prilikom spajanja na modem i koja je potrebna pri uspostavljanju komunikacije. U *command promptu* potrebno je upisati ***ping raspberry*** koja rezultira ispisivanjem IP adrese.

Nakon što sam saznao adresu svog Raspberryja, za sljedeći korak mi je potreban program Putty SSH Tool. Izgled njegovog prozora prikazan je na slici 3.4.4.



Slika 3.4.4 Prozor Putty SSH Toola

Prije pokretanja Putty terminala potrebno je još pod postavkama potvrditi mogućnost X11 *forwardinga*. Ako su sve postavke dobro namještene te ako je prepoznata komunikacija između dva medija, otvara se Putty terminal prikazan na slici 3.4.5.



```
pi@raspberrypi: ~
login as: pi
pi@192.168.1      's password:
Linux raspberrypi 3.12.28+ #709 PREEMPT Mon Sep 8 15:28:00 BST 2014 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 Oct 13 19:55:37 2014 from 192.168.1.
pi@raspberrypi ~$
```

Slika 3.4.5 Putty terminal

Jedini problem se javio prilikom spajanja na internet koje vjerojatno nije bilo moguće zbog modema u domu. Shodno tome, dijelove koji su obuhvaćali korištenje interneta sam morao realizirati spajanjem WiFi adaptera i tipkovnice na USB 2.0 ulaze te HD monitora putem HDMI kabela koristeći mobitel kao modem. Jedan od tih je i postavljanje vlastitog web servera.

3.5 Apache web server

Ideja je da server služi za pohranu i ispis podataka. Osim toga, korisnik će biti u mogućnosti vidjeti podatke na dnevnoj, tjednoj, mjesečnoj i godišnjoj bazi u okviru grafova koji će se iscrtavati.

Apache je WEB server aplikacija koja je igrala jednu od ključnih uloga u početnom rastu interneta. Razvoj je započeo 1995. godine i ubrzo preuzeo dominantnu ulogu WEB servera. Od 2009. godine nadalje, Apache služi kao WEB server više od 100 milijuna internet stranica. Apache se razvija i održava putem otvorene zajednice programera pod pokroviteljstvom Apache Software Foundationa. Najčešće se koristi

na UNIX sustavima, premda je proširen toliko da ga danas možemo vidjeti na Unix-u, FreeBSD, Linux, Solaris, Novell NetWare, OS X, Microsoft Windows itd. Od lipnja 2013. procjenjuje se da Apache poslužuje 54,2% aktivnih web stranica i da spada u 53,3% najboljih servera u svim domenama.

Prije postavljanja servera, važno je da je operacijski sustav Raspbian ažuriran što se postiže unošenjem naredbi:

sudo apt-get update
sudo apt-get upgrade

Instalacija traje nekoliko minuta, nakon čega je potrebno izvršiti ponovno pokretanje sustava naredbom:

sudo reboot

Nakon što je komunikacija uspostavljena, potrebno je unijeti naredbu:

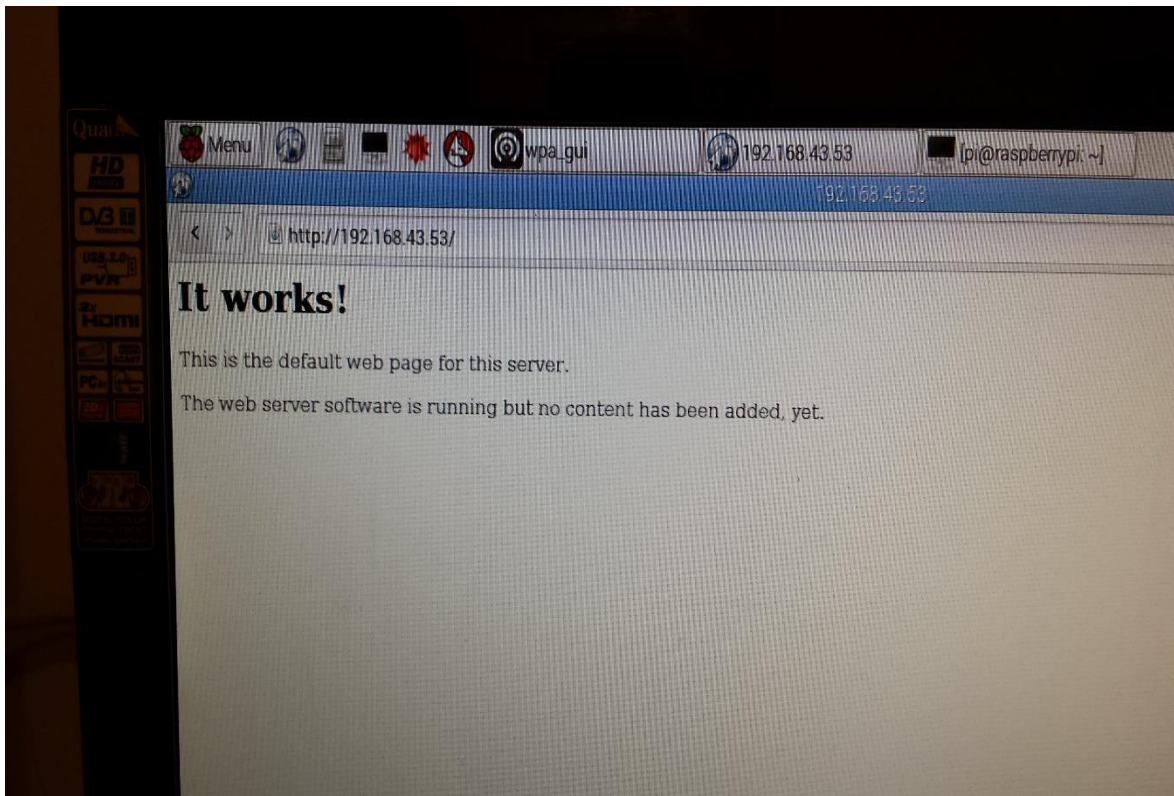
sudo apt-get install apache2 php5 libapache2-mod-php5

Izvršavanjem iste, instaliramo Apache, PHP5 i povežemo Apache i PHP. PHP će nam biti potreban kako bismo mogli napraviti stranicu preko koje ćemo pozivati skripte. Razlog zašto smo uzeli PHP umjesto HTML leži u tome što je PHP bolji za pozivanje skripti putem terminala.

Potrebno je ispitati uspješnost instalacije otvaranjem web preglednika i upisivanjem IP adrese dodijeljene Raspberry Piju:

http://192.168.43.53/

Sljedeća slika nam potvrđuje kako je web server uspješno podignut:



Slika 3.5.1 Podizanje web servera

Za skladištenje podataka koristit ćemo RRDtool. Ovakve baze podataka čuvaju najsvježije informacije u svom izvornom obliku, dok se one starije brišu i zamjenjuju svojim osvježanim ekvivalentom. Što su podaci stariji, stupanj osvježavanja je veći. Kada podatak sasvim zastari, on biva potpuno izgubljen. Osvježavanje je osmišljeno tako da jednom kreiran RRD datoteka ima konstantnu veličinu, neovisno o broju ažuriranja.

RRDtool instaliramo naredbom:

```
sudo apt-get install rrdtool
```

U ovom koraku možemo stvoriti rrd datoteku u koju ćemo spremati podatke. Potrebno je napraviti sljedeće:

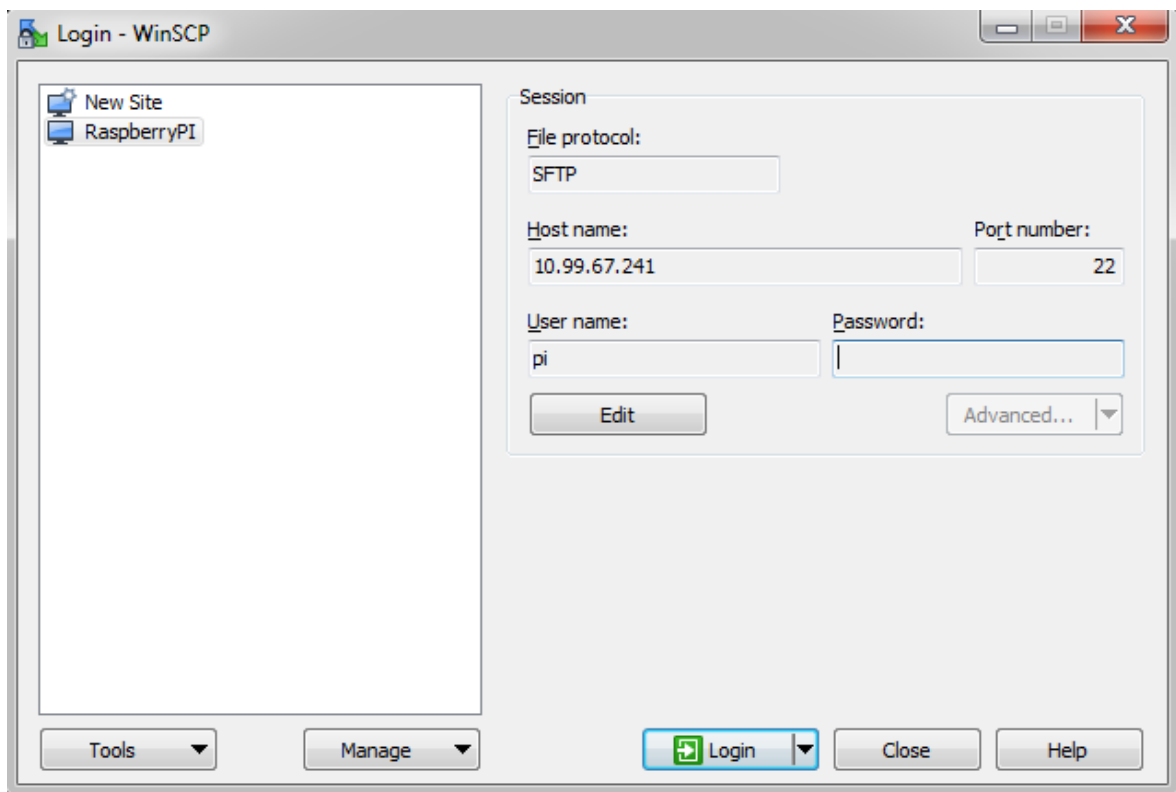
```
cd/home/pi/project/WeatherStation/  
rrdtool create weatherRRD.rrd
```

Uz to, moramo instalirati *python smb* i *python dev* biblioteke. Prva će služiti za komunikaciju sa senzorima koji koriste I²C sučelje (BMP180), dok druga služi za proširivanje mogućnosti programiranja u Pythonu:

sudo apt-get install git build-essential python-dev python-smbus

3.6 Prijenos skripti s laptopa na Raspberry Pi

Izrada skripti (o samom sadržaju skripti bit će govora nešto kasnije, u sklopu njihove implementacije) uvelike je olakšana mogućnošću njihovog prijenosa s laptopa na Raspberry Pi. Za to nam je potreban WinSCP alat. Izgled njegovog prozora prikazan je na slici 3.6.1. Pri prijenosu nam je potrebna već poznata IP adresa, te korisničko ime i lozinka koje koristimo prilikom pokretanja Raspberryja.



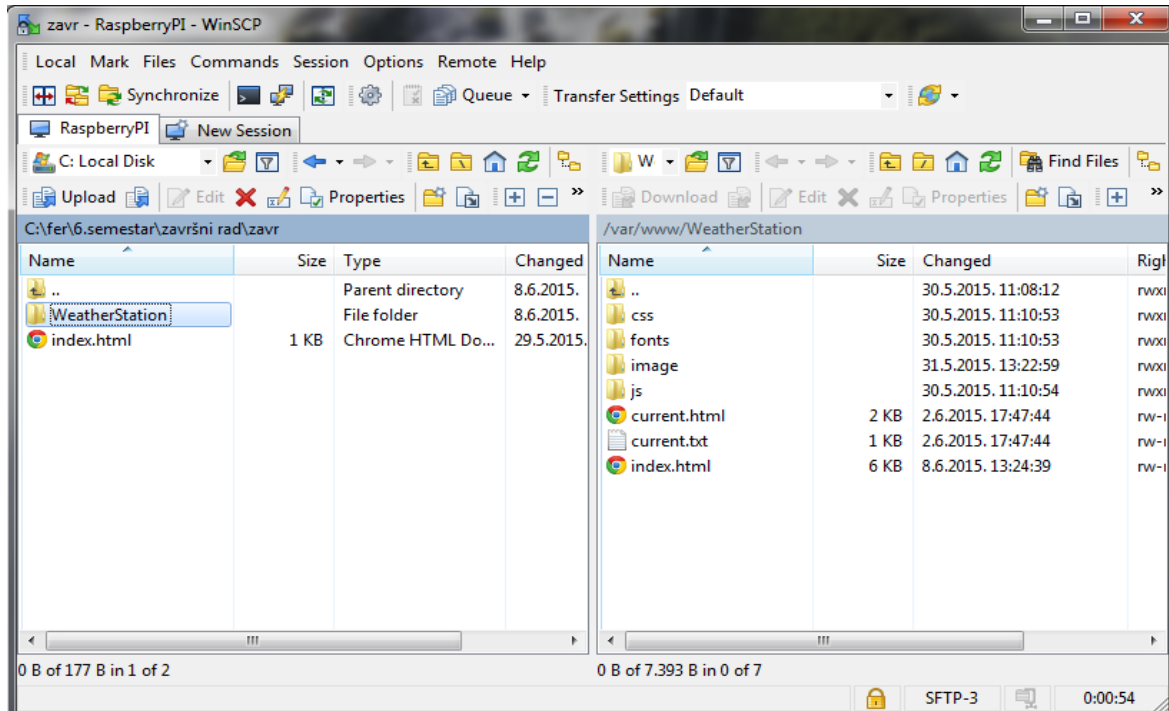
Slika 3.6.1 WinSCP prozor

Prije slanja podataka, potrebno je stvoriti direktorije u kojeg ćemo prebaciti skripte. Jedan će nam služiti kao *home* direktorij, a drugi za dio povezan s web serverom:

`sudo mkdir -p /home/pi/project/WeatherStation`

`sudo mkdir -p /var/www/WeatherStation`

Sada imamo direktorije spremne za prijenos podataka. Izgled WinSCP prozora prikazan je na slici 3.6.2.



Slika 3.6.2 Prebacivanje datoteka s laptopa na Raspberry

3.7 Testiranje senzora

U ovom koraku, potrebno je ispitati funkcionira li spoj senzora s Raspberryjem te daju li skripte referentna očitavanja. Na internetu se može naći mnoštvo gotovih biblioteka koje je samo potrebno instalirati za provjeru ispravnosti rada senzora.

3.7.1 DHT22

Prilikom testiranja DHT22 senzora koristit ćemo Adafruit biblioteku. Potrebno je napraviti sljedeće:

cd/home/pi/project/WeatherStation/library/Adafruit_Python_DHT

Ovom naredbom se pozicioniramo unutar biblioteke koja sadrži skripte za testiranje senzora DHT22.

Naredbom:

sudo python setup.py install

Instaliramo sve module povezane s modelom Raspberry Pi računala koje koristimo.

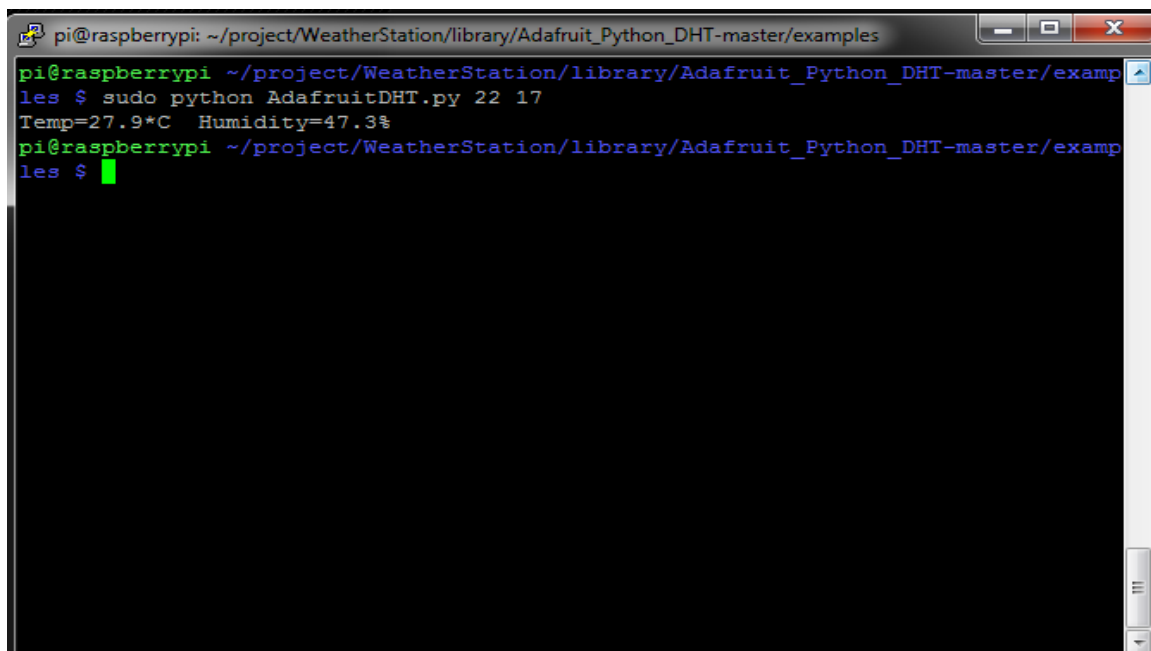
Slijedi pozicioniranje u direktorij u kojem se nalaze primjeri za testiranje:

cd/home/pi/project/WeatherStation/library/Adafruit_Python_DHT/examples

Prilikom pokretanja skripte, potrebno je navesti model senzora te Raspberry Pi GPIO pin koji smo spojili s istim. Primjer:

sudo python AdafruitDHT.py 22 17

Rezultat izvođenja je prikazan na slici 3.7.1.1.



```
pi@raspberrypi: ~/project/WeatherStation/library/Adafruit_Python_DHT-master/examples
les $ sudo python AdafruitDHT.py 22 17
Temp=27.9*C Humidity=47.3%
pi@raspberrypi ~/project/WeatherStation/library/Adafruit_Python_DHT-master/exam
les $
```

Slika 3.7.1.1 Očitavanje temperature i vlage sa senzora DHT22


Usporedbom s referentnim veličinama (primjerice one s meteoroloških stranica) možemo ustanoviti da senzor radi te daje valjane rezultate.

3.7.2 BMP180

Testiranje BMP180 senzora će biti nešto složenije zbog korištenja I²C sabirnice. Raspberry Pi ima dva pina, 3 i 5, koja su već konfigurirana kao dio I²C protokola: SDA (podaci) i SCL (takt), čime se može komunicirati s I²C uređajem. Prvo ćemo omogućiti njen rad sljedećim nizom radnji:

sudo nano /etc/modules

Otvora se prozor u kojem ćemo dodati dva potrebna modula (slika 3.7.2.1).



```
GNU nano 2.2.6 File: /etc/modules
# /etc/modules: kernel modules to load at boot time.
#
# This file contains the names of kernel modules that should be loaded
# at boot time, one per line. Lines beginning with "#" are ignored.
# Parameters can be specified after the module name.

snd-bcm2835
i2c-bcm2708
i2c-dev

[ Read 10 lines ]
^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell
```

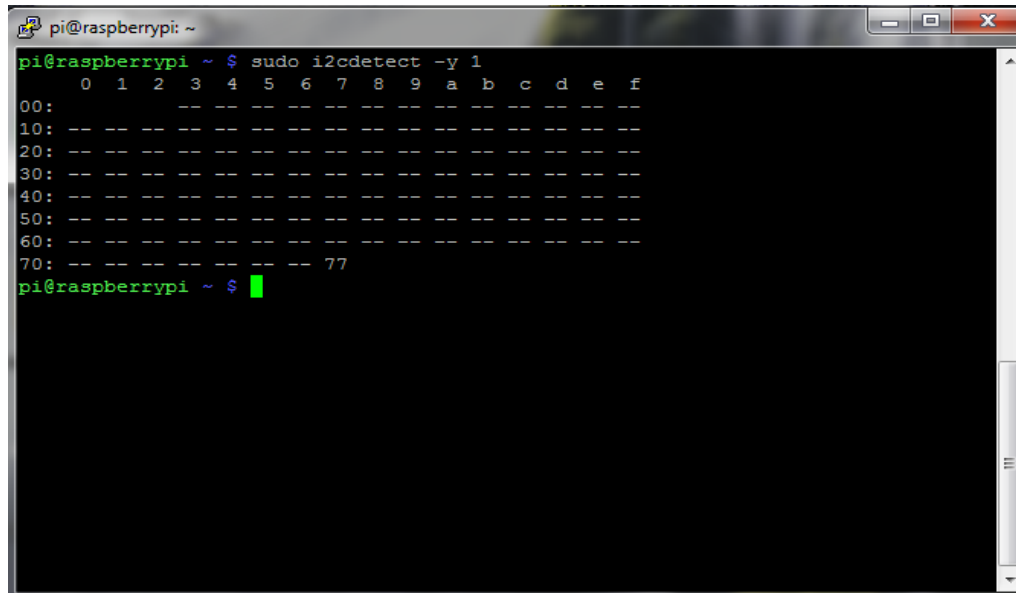
Slika 3.7.2.1 Dodavanje modula potrebnih za rad I²C sabirnice

Osim ručnog namještanja, module je moguće podesiti korištenjem ***raspi-config***.

Nadalje, potrebno je provjeriti je li senzor uspješno spojen s Raspberry Pijom:

sudo i2cdetect -y 1

Izvršavanjem naredbe, u mogućnosti smo vidjeti skup registara od 00 do 70. Senzor BMP180 trebao bi se nalaziti na adresi 0x77 što potvrđuje slika 3.7.2.2.



```
pi@raspberrypi: ~  
pi@raspberrypi ~ $ sudo i2cdetect -y 1  
 0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f  
00: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --  
10: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --  
20: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --  
30: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --  
40: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --  
50: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --  
60: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --  
70: -- -- -- -- -- -- -- -- 77  
pi@raspberrypi ~ $
```

Slika 3.7.2.2 Adresa senzora BMP180

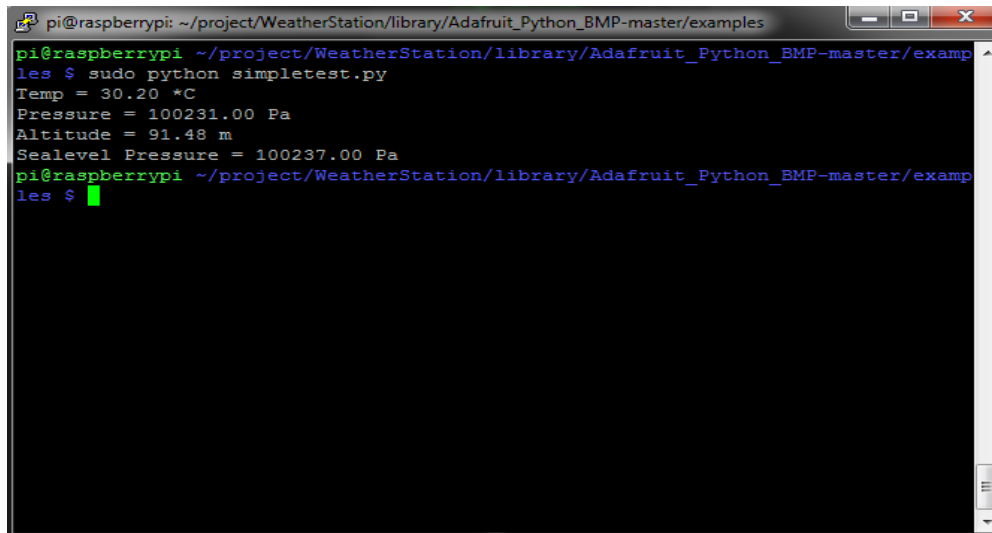
Slijedi instaliranje biblioteka potrebnih za testiranje. Analogno testiranju DHT22 radimo sljedeće:

```
cd/home/pi/project/WeatherStation/library/Adafruit_Python_BMP-master  
sudo python setup.py install
```

Pozicioniravši se u željeni direktorij, možemo pokrenuti naredbu:

```
sudo python simpletest.py
```

Ako izvođenje skripte prođe bez pogreški, dobit ćemo rezultate prikazane na slici 3.7.2.3.



```
pi@raspberrypi: ~/project/WeatherStation/library/Adafruit_Python_BMP-master/examples
les $ sudo python simpletest.py
Temp = 30.20 *C
Pressure = 100231.00 Pa
Altitude = 91.48 m
Sealevel Pressure = 100237.00 Pa
pi@raspberrypi ~/project/WeatherStation/library/Adafruit_Python_BMP-master/examp
les $
```

Slika 3.7.2.3 Očitavanje veličina sa senzora BMP180

Nakon što su oba senzora uspješno testirana, možemo se okrenuti pisanju glavne skripte.

3.8 Glavna skripta

Slijedi najteži dio zadatka: izrada skripte koja će omogućiti očitavanje veličina sa senzora i njihovo upisivanje u bazu podataka te naposljetku na web server. Pritom će biti potrebno povezati znanja programiranja u Pythonu te pisanja na web server. Pisanje koda u C-u bilo bi mukotržno za ovakav tip problema, stoga je Python idealan za takvo što.

U prvom dijelu koda uključujemo sve module koje želimo koristiti:

```
#!/usr/bin/python
import commands
import os
import sys
sys.path.append("/home/pi/project/WeatherStation/library/Adafruit_BMP085")
import re
import datetime
import subprocess
import time
import smbus
from Adafruit_BMP085 import BMP085
```

Slijedi pisanje metoda kojima ćemo čitati senzore:

```
def readDHT22():  
  
    humRead = -100.0  
  
    reading = 0  
  
    while (humRead == -100.0 and reading < 10):  
  
        #try to read 10 times because of errors  
  
        cmdStat, cmdOut = commands.getstatusoutput("sudo  
python/home/pi/project/WeatherStation/library/Adafruit_Python_DHT/examples/AdafruitDHT.py  
22 17")  
  
        dbgprint("DHT22 output: " + str(cmdOut))  
  
        #Parse output  
  
        if cmdStat == 0:  
  
            matchObj = re.search( r'(.*) Humidity=(.*)%', cmdOut, re.M|re.I)  
  
            if matchObj:  
  
                humRead = float(matchObj.group(2))  
  
                dbgprint("Humidity: " + str(humRead))  
  
            else:  
  
                dbgprint( "No match during DHT22 output parse!" )  
  
        else:  
  
            dbgprint( "Error reading DHT22 sensor" )  
  
        reading += 1  
  
    return humRead
```

Kao što je prije navedeno, DHT22 senzor ćemo koristiti za mjerenje vlažnosti. Veličine ćemo pokušati čitati 10 puta jer može doći do neželjenih grešaka. Ako dođe do istih, proces čitanja ponavljamo dok ne očitamo valjanu vrijednost koju na koncu šaljemo na ispis.

Pri očitavanju sa senzora BMP180, iskoristit ćemo instalirani modul Adafruit_BMP085. Postupak je jednostavan: inicijaliziramo adresu senzora te

pomoću ugrađenih metoda vršimo očitavanje temperature, tlaka i nadmorske visine. Metoda je prikazana u nastavku:

```
def readBMP085():  
    bmp = BMP085(0x77)  
    tempRead = bmp.readTemperature()  
    dbgprint("Temperature: " + str(tempRead))  
    pressRead = bmp.readPressure()/100.0  
    dbgprint("Pressure: " + str(pressRead))  
    altitRead = bmp.readAltitude()  
    dbgprint("Altitude: " + str(altitRead))  
    return tempRead, pressRead, altitRead
```

Na redu je pisanje na web stranicu čime ćemo dobiti mogućnost brzog pregledavanja očitanih temperatura u stvarnom vremenu. Poziv funkcije je sljedeći:

```
def writeHTML(ti, hu, pa, al):
```

Potrebno je namjestiti da stranica prikazuje stvarno vrijeme:

```
vNow = datetime.datetime.now().strftime( "%Y-%m-%d %H:%M:%S" )
```

Dio koda koji se odnosi na ispis jedne od veličina na web stranicu:

```
html.write('<div class="col-md-2">\n')  
html.write('<div class="panel panel-danger">\n')  
html.write(' <div class="panel-heading">tlak</div>\n')  
html.write('<div class="panel-body" style="text-align: center">\n')  
html.write('<h1>'+str(round(pa,1))+<small> hPa</small></h1>\n')  
html.write('</div>\n')  
html.write('</div>\n')  
html.write('</div>\n')
```

Osim pisanja na web stranicu, veličine ćemo spremati u txt datoteku:

```
curr = open("/var/www/WeatherStation/current.txt", "w")  
  
triba popraviti  
  
curr.close()
```

Dakle, imamo sve metode koje ćemo koristiti za očitavanje veličina te njihovo spremanje na web stranicu, odnosno txt datoteku. Predstoji nam napraviti *main* dio skripte koji će pozivati metode, pisati na web server, spremati podatke u rrd bazu.

Primjer takvog dijela:

```
humid = readDHT22()  
  
temp, press, altit = readBMP085()  
  
#write to RRD archive  
  
cmd= '/usr/bin/rrdtool update /home/pi/project/WeatherStation/weatherRRD.rrd -t temp:press:humid:altit'  
  
cmd = cmd +str(temp)+' '+str(press)+' '+str(humid)+' '+str(altit)  
  
cmdStat, cmdOut = commands.getstatusoutput(cmd)  
  
dbgprint( "RRD write result: " + str(cmdStat))  
  
writeHTML(temp, humid, press, altit)  
  
dbgprint ( "End!" )
```

Glavna skripta sada je spremna za izvođenje. Prvo ćemo se pozicionirati u direktorij u kojem se nalazi:

```
cd /home/pi/project/WeatherStation/
```

Sljedećom naredbom pokrećemo izvođenje skripte. Ekstenzija `-d` nam služi za *debuggiranje*, odnosno ispravljanje mogućih pogrešaka uz uvid očitanih veličina:

```
sudo python readsensors.py -d
```

Rezultat izvršavanja skripte prikazan je na slici 3.8.1.


```

login as: pi
pi@10.99.67.241's password:
Linux raspberrypi 3.18.11-v7+ #781 SMP PREEMPT Tue Apr 21 18:07:59 BST 2015 armv
7l

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: Tue Jun  2 17:26:03 2015 from 10.4.255.116
pi@raspberrypi ~ $ cd /home/pi/project/WeatherStation/
pi@raspberrypi ~/project/WeatherStation $ sudo python readsensors.py -d
2015-06-02 17:47:44.517782 - DHT22 output: Temp=25.0*C Humidity=51.4%
2015-06-02 17:47:44.520375 - Humidity: 51.4
2015-06-02 17:47:44.538051 - Interior Temperature: 25.4
2015-06-02 17:47:44.554718 - Pressure: 1003.88
2015-06-02 17:47:44.571547 - Altitude: 78.3053591358
2015-06-02 17:47:44.598430 - RRD write result: 256
2015-06-02 17:47:44.601076 - End!
pi@raspberrypi ~/project/WeatherStation $ █

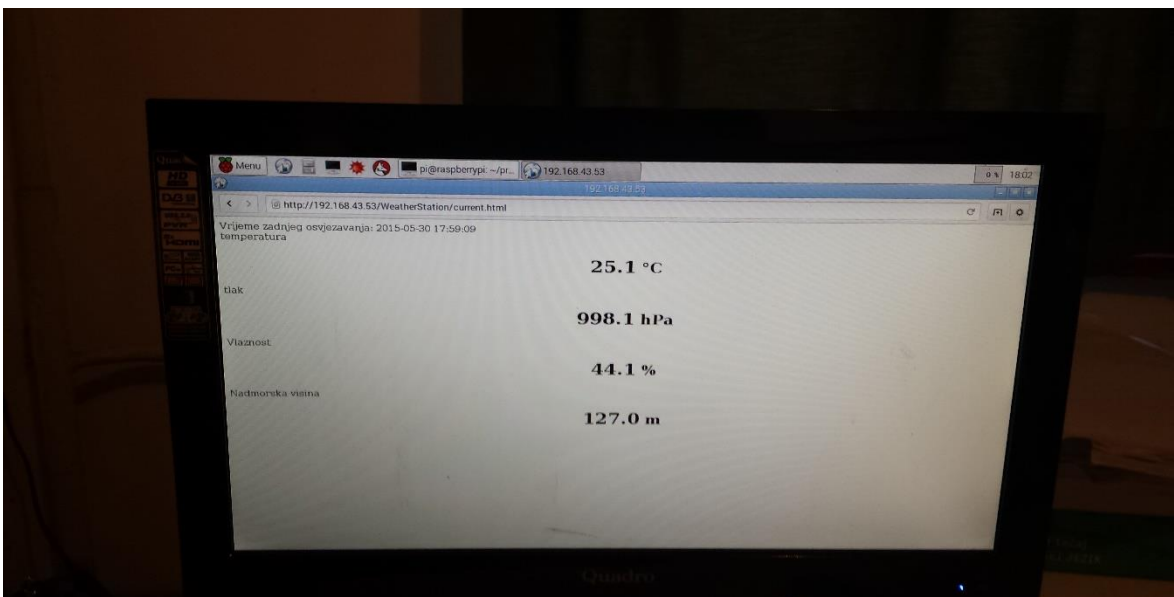
```

Slika 3.8.1 Rezultat očitavanja podataka sa senzora

Osim ispisa na terminalu, podatke o očitanim veličinama možemo vidjeti na web stranici. U web preglednik je potrebno upisati sljedeće:

<http://192.168.43.53/WeatherStation/current.html>

gdje **192.168.43.53** predstavlja IP adresu Raspberry Pija. Dobiven je prikaz izmjerenih veličina u stvarnom vremenu (slika 3.8.2).



Slika 3.8.2 Prikaz rezultata na web stranici

Uvjerili smo se da senzori rade te sada možemo pristupiti izradi konačne verzije web servera koju ćemo temeljiti na JavaScriptu. Primjer jednog takvog dijela koda koji postavlja vrijeme i datum u stvarnom vremenu:

```
<script type="text/javascript">

    function display_c(){

    var refresh=1000; // Refresh rate in milli seconds

    mytime=setTimeout('display_ct()',refresh)}

    function display_ct() {

    var strcount

    var x = new Date()

    document.getElementById('ct').innerHTML = x;

    tt=display_c();}

</script>
```

Ostatak je sadržan u datoteci ***index.html***.

Potrebna nam je još jedna skripta koja će nam omogućiti iscrtavanje grafova na web serveru. Podaci o veličinama se preuzimaju iz ***weatherRRD.rrd*** datoteke koja nam služi kao baza podataka. Slike koje nastanu spremaju se u direktorij *images* te će iste biti uključene prilikom realizacije web servera. Pokretanje skripte izvršava se naredbom:

```
sudo bash /home/pi/project/WeatherStation/rrdGraphs.sh
```

Rezultat izvođenja prikazan je na slici 3.8.3.

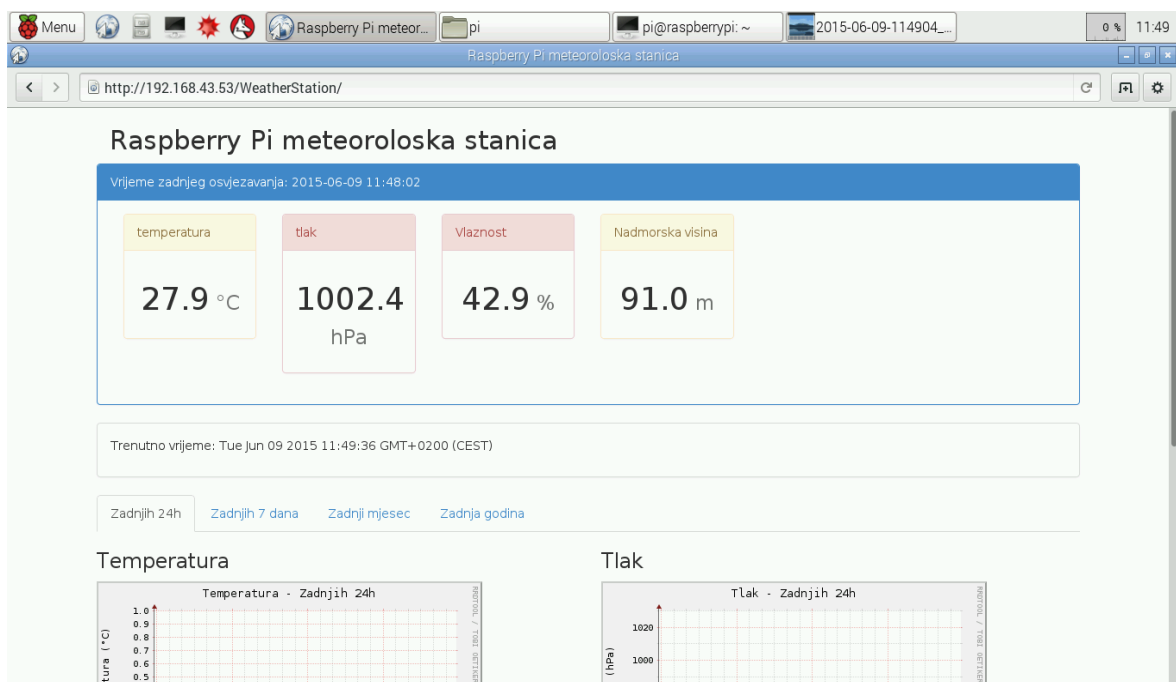
```
pi@raspberrypi: ~
pi@raspberrypi ~ $ sudo bash /home/pi/project/WeatherStation/rrdGrafovi.sh
447x262
447x262
447x262
447x262
447x262
447x262
447x262
447x262
447x262
447x248
447x248
447x262
447x262
447x262
447x262
447x262
pi@raspberrypi ~ $ █
```

Slika 3.8.3 Iscrtavanje grafova

Sada samo trebamo provjeriti jesu li se grafovi iscrtali na web serveru:

<http://192.168.43.53/WeatherStation/>

Slika 3.8.4 otkriva nam konačni izgled web servera:



Slika 3.8.4 Izgled web servera

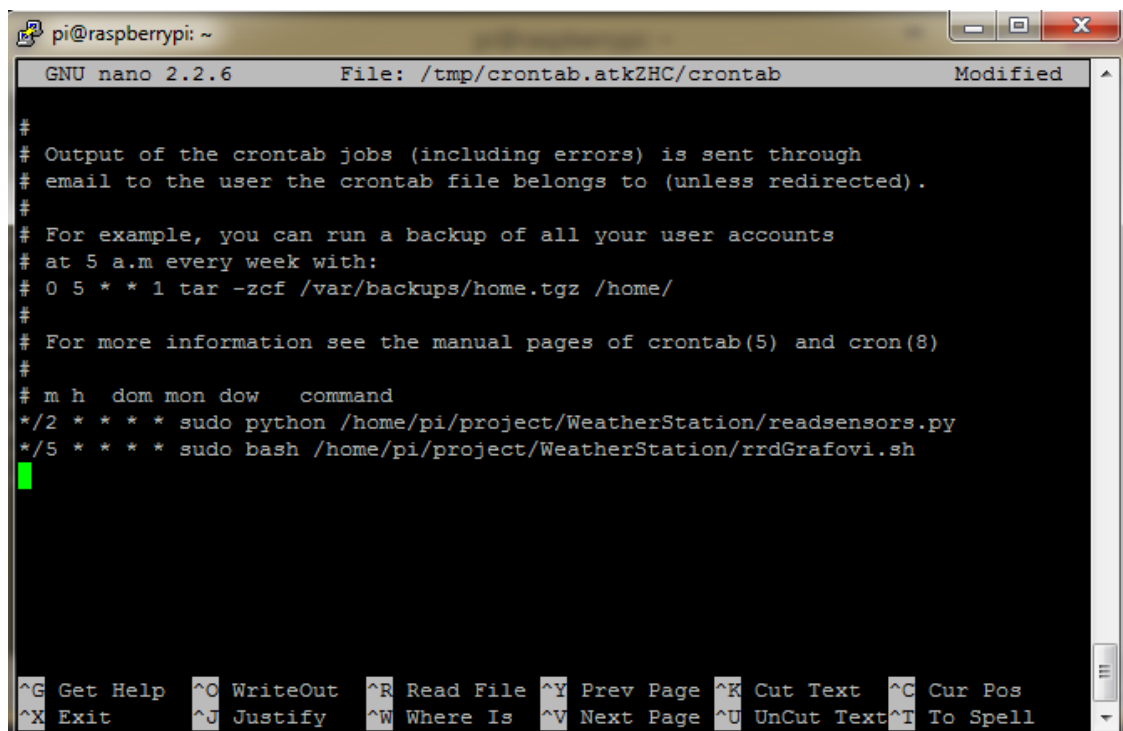
Preostalo nam je namjestiti vremenski interval unutar kojeg će se pokretati skripte kako bi u svakom trenutku imali prikaz najsvježijih podataka na web serveru.

Napravit ćemo sljedeće:

```
sudo su  
cd /home/pi  
crontab -e
```

Otvara se prozor (slika 3.8.5) u kojem je potrebno upisati:

```
*/2 * * * * sudo python /home/pi/project/WeatherStation/readensors.py  
*/5 * * * * sudo bash /home/pi/project/WeatherStation/rrdGrafovi.sh
```



```
pi@raspberrypi: ~  
GNU nano 2.2.6 File: /tmp/crontab.atk2HC/crontab Modified  
#  
# Output of the crontab jobs (including errors) is sent through  
# email to the user the crontab file belongs to (unless redirected).  
#  
# For example, you can run a backup of all your user accounts  
# at 5 a.m every week with:  
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/  
#  
# For more information see the manual pages of crontab(5) and cron(8)  
#  
# m h dom mon dow command  
*/2 * * * * sudo python /home/pi/project/WeatherStation/readensors.py  
*/5 * * * * sudo bash /home/pi/project/WeatherStation/rrdGrafovi.sh  
^G Get Help ^C WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos  
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell
```

Slika 3.8.5 Podešavanje periodičnog izvođenja skripti

Namjestili smo očitavanje sa senzora svake dvije minute, dok će se grafovi iscrtavati svakih 5 minuta. Ostaje nam spremiti izmjene, izvršiti *reboot* i provjeriti ispravnost rada meteorološke stanice.

4. Zaključak

Raspberry Pi je mini računalo koje oduševljava mnoge širom svijeta. Praktično je za izradu mnogih projekata, među ostalima i meteorološke stanice koja je tema mog završnog rada. Rad na ovom projektu mi je pomogao da proširim znanja u programiranju, upoznam neke nove koncepte te poboljšam inženjerski način razmišljanja. Također, realizirana meteorološka stanica je primjenjiva u sobi studentskog doma te sam sada u mogućnosti u svakom trenutku saznati kakvi su vremenski uvjeti. Nadam se da ću je u budućnosti uspjeti proširiti, jer za nešto više i bolje ipak treba malo više novca, a u ovom projektu je bio naglasak na malim troškovima. S veseljem gledam ka novim projektima s Raspberry Pijem jer se uistinu pokazao malo računalo velikih mogućnosti te ga preporučujem svima koji žele usavršiti znanje programiranja.

5. Literatura

- [1] <http://www.raspberrypi.org/>
- [2] http://en.wikipedia.org/wiki/Raspberry_Pi
- [3] <http://openmicros.org/index.php/articles/94-ciseco-product-documentation/raspberry-pi/217-getting-started-with-raspberry-pi-gpio-and-python>
- [4] <http://www.aoakley.com/articles/2013-09-19-raspberry-pi-lego-robot-part1.php>
- [5] Wolfram Donat: Learn Raspberry Pi Programming with Python, 2014.
- [6] Kevin Partner: Ultimate Guide to Raspberry Pi, 2014.

6. Sažetak

Naslov

Meteorološka stanica temeljena na umreženom ugradbenom računalu

Sažetak

Raspberry Pi je cijenom pristupačno računalo veličine kreditne kartice, bazirano na Linux operacijskom sustavu. Ljudima svih dobnih skupina omogućuje istraživanje svijeta računala te učenje programskih jezika kao što su Python i Scratch. U okviru završnog rada bilo je potrebno razviti meteorološku stanicu temeljenu na povezivanju Raspberry Pija sa sensorima s naglaskom na male troškove. Stvoren je dinamički web server na kojem se, uz vremensku oznaku iz sata stvarnog vremena, vrše periodična očitavanja i pohrane veličina. Razvijeno je praćenje promjena istih kroz grafove koji se iscrtavaju na dnevnoj, tjednoj, mjesečnoj i godišnjoj bazi. Postoji mogućnost proširivanja meteorološke stanice dodavanjem drugih senzora.

Ključne riječi

Raspberry Pi, senzori, meteorološka stanica, Python, web server

7. Abstract

Title

Meteorological Station Based on Networked Embedded System

Abstract

The Raspberry Pi is a low cost, credit-card sized computer, based on Linux operating system. It enables people of all ages to explore computing and to learn how to program in languages like Python and Scratch. As the part of the assignment, it was required to develop a meteorological station based on connecting Raspberry Pi with low cost sensors. A dynamic web server was created, along with a timestamp from the real-time clock, to perform periodic readings and to storage informations. It is possible to track changes through graphs being plotted on a daily, weekly, monthly and annual basis. There is a room to improve meteorological station by adding other sensors.

Key Words

Raspberry Pi, sensors, meteorological station, Python, web server