

Univerza
v Ljubljani

Fakulteta
*za gradbeništvo
in geodezijo*



Jamova cesta 2
1000 Ljubljana, Slovenija
<http://www3.fgg.uni-lj.si/>

DRUGG – Digitalni repozitorij UL FGG
<http://drugg.fgg.uni-lj.si/>

To je izvorna različica zaključnega dela.

Prosimo, da se pri navajanju sklicujete na bibliografske podatke, kot je navedeno:

Sečnik, M., 2013. Razvoj hidrometričnega sistema na Ljubljani v realnem času. Diplomski naloga. Ljubljana, Univerza v Ljubljani, Fakulteta za gradbeništvo in geodezijo. (mentor Brilly, M. somentor Vidmar, A.): 55 str.

University
of Ljubljana

Faculty of
*Civil and Geodetic
Engineering*



Jamova cesta 2
SI – 1000 Ljubljana, Slovenia
<http://www3.fgg.uni-lj.si/en/>

DRUGG – The Digital Repository
<http://drugg.fgg.uni-lj.si/>

This is original version of final thesis.

When citing, please refer to the publisher's bibliographic information as follows:

Sečnik, M., 2013. Razvoj hidrometričnega sistema na Ljubljani v realnem času. B.Sc. Thesis. Ljubljana, University of Ljubljana, Faculty of civil and geodetic engineering. (supervisor Brilly, M., co-supervisor Vidmar, A.): 55 pp.

Univerza
v Ljubljani

Fakulteta za
*gradbeništvo in
geodezijo*



Jamova 2
1000 Ljubljana, Slovenija
telefon (01) 47 68 500
faks (01) 42 50 681
fgg@fgg.uni-lj.si

UNIVERZITETNI ŠTUDIJ
VODARSTVA IN
KOMUNALNEGA
INŽENIRSTVA

Kandidat:

MATEJ SEČNIK

**RAZVOJ HIDROMETRIČNEGA SISTEMA NA
LJUBLJANICI V REALNEM ČASU**

Diplomska naloga št.: 208/VKI

**DEVELOPMENT OF THE HYDROMETRIC SYSTEM
ON THE LJUBLJANICA RIVER IN REAL TIME**

Graduation thesis No.: 208/VKI

Mentor:

prof. dr. Mitja Brilly

Predsednik komisije:

doc. dr. Dušan Žagar

Somentor:

mag. Andrej Vidmar

Član komisije:

izr. prof. dr. Viktor Grilc

doc. dr. Mojca Šraj

Ljubljana, 09. 09. 2013

Stran za popravke, Errata

Stran z napako

Vrstica z napako

Namesto

Naj bo

IZJAVA O AVTORSTVU

Podpisani Matej Sečnik izjavljam, da sem avtor diplomske naloge z naslovom: »Razvoj hidrometričnega sistema na Ljubljani v realnem času«.

Izjavljam, da je elektronska različica v vsem enaka tiskani različici.

Izjavljam, da dovoljujem objavo elektronske različice v repozitoriju UL FGG.

Ljubljana, september 2013

.....
(podpis)

BIBLIOGRAFSKO – DOKUMENTACIJSKA STRAN IN IZVLEČEK

UDK: 556.04+556.07(497.4)(043.2)
Avtor: Matej Sečnik
Mentor: prof. dr. Mitja Brilly
Somentor: mag. Andrej Vidmar
Naslov: Razvoj hidrometričnega sistema na Ljubljani v realnem času
Tip dokumenta: Diplomaska naloga – univerzitetni študij
Obseg in oprema: 55 str., 2 pregl., 36 sl., 1 pril.
Ključne besede: Ljubljana, Starflow, nadzor v realnem času, Matlab, spletna stran, CMS Joomla

Izvilleček:

Za uspešno upravljanje z vodami je potreben stalen nadzor. Tako za optimalno preventivno ukrepanje, kot tudi pravilno kurativno ravnanje so ključnega pomena podatki, pridobljeni v realnem času. V okviru projekta Ljubljana povezuje je na porečju Ljubljane predvidena postavitvev treh samodejnih vodomernih postaj Kamin, Lipe in Ižanska. Postaje bodo razporejene po reki Ljubljani in reki Ižici. Cilj diplomske naloge je razvoj sistema, s pomočjo katerega prikažemo hidrološke podatke (pretok, hitrost in temperatura) iz vodomernih postaj na spletu. Za zajem hidroloških podatkov na terenu sem uporabil ultrazvočni merilec Starflow. Opisana je komunikacija merilca Starflow z računalnikom preko radijske zveze. Razvili smo programsko orodje (software), s katerim se daljinsko (preko radijske zveze) povežemo z merilcem Starflow in od njega pridobimo hidrološke podatke. Podatke nato obdelamo in izvozimo v podatkovno bazo Microsoft Access. Opisan postopek je popolnoma avtomatiziran. V drugem delu diplomskega dela je opisana izdelava spletne strani, na kateri so prikazani podatki vodomernih postaj v realnem času.

BIBLIOGRAPHIC – DOKUMENTALISTIC INFORMATION AND ABSTRACT

UDC: 556.04+556.07(497.4)(043.2)
Author: Matej Sečnik
Supervisor: Prof. Mitja Brilly, Ph.D.
Cosupervisor: Andrej Vidmar, M.Sc.
Title: Development of the Hydrometric System on the Ljubljana River in Real Time
Document type: Graduation Thesis – University studies
Notes: 55 p., 2 tab., 36 fig., 1 ann.
Key words: Ljubljana river, Starflow, real time monitoring, Matlab, web page, CMS Joomla

Abstract:

Water management requires constant supervision. Real-time data is crucial for the optimal preventive action as well as for the proper curative treatment. In the framework of the project *Ljubljana povezuje*, three automatic gauging stations are planned to be set up along the river basin i.e., Lipe, Kamin and the Ižanska station. They will be sited on the Ljubljana and Ižica rivers. The aim of the thesis is to develop a system which would transfer the hydrological data (flow, speed, temperature) from the stations and display them on the internet. To collect the hydrological data in the field, I used the ultrasonic meter Starflow and then described the radio communication of the meter Starflow with the computer. We have developed a software tool (software) which is remotely linked (via radio communication) to the Starflow meter and collects the hydrological data. Data is then processed and exported into Microsoft Access database. The described procedure is completely automated. The second part of the thesis describes the task of designing a web page where real-time data (collected from gauging stations) is displayed.

ZAHVALA

Za strokovno pomoč in nasvete pri izdelavi diplomske naloge se iskreno zahvaljujem mentorju prof. dr. Mitji Brillyju.

Iskreno se zahvaljujem somentorju mag. Andreju Vidmarju, ki mi je preko številnih pogovorov s svojimi izkušnjami in znanjem na tem področju ogromno pomagal. Za potrebe raziskovanja je brez pomisleka odstopil instrument Starflow.

Zahvalil bi se tudi družini in prijateljem, ki so mi v vseh letih študija stali ob strani in mi nudili pomoč.

KAZALO VSEBINE

1	UVOD	1
2	LJUBLJANICA	2
2.1	Splošna opredelitev reke Ljubljanice.....	2
2.2	Pretok in poplave	3
2.3	Varovanje Ljubljanice.....	4
2.3.1	Natura 2000	4
2.3.2	Krajinski park Ljubljansko barje	5
2.4	Lokacije samodejnih vodomernih postaj na Ljubljanici.....	6
3	ULTRAZVOČNI MERILEC STARFLOW	8
3.1	Sestavni deli.....	8
3.2	Tehnične lastnosti	9
3.3	Področja uporabe.....	9
3.4	Princip delovanja	10
3.5	Izračun pretoka.....	10
3.6	Komunikacija merilca Starflow z računalnikom.....	11
3.6.1	Serijski protokol.....	11
3.6.2	Serijski asinhroni protokol.....	12
3.7	Komunikacija z računalnikom (STARLOG protokol).....	14
3.7.1	Podatkovne zahteve	14
3.7.2	Naslovni prostor	16
3.7.3	Sporočanje napak v delovanju.....	17
3.8	Komunikacija z merilcem Starflow na daljavo.....	18
3.8.1	Radijski modem Telit PowerOne 868MHz.....	18
3.8.2	Alternativni načini komunikacije.....	20
4	PROGRAMSKI JEZIK MATLAB	22
4.1	Komunikacija s serijskim vmesnikom v okolju Matlab	23
4.2	Krmilni stavki	24
4.2.1	While zanka	24
4.2.2	If - else zanka.....	25

4.3	Izvoz podatkov v Microsoft Excel	25
4.4	Izvoz podatkov v podatkovne baze	25
4.5	Funkcija hex2dec	27
5	MICROSOFT ACCESS.....	28
5.1	Kaj so relacijske baze podatkov	28
5.2	Ustvarjanje nove baze podatkov	28
5.3	Vnašanje podatkov v tabele	28
6	IZDELAVA SPLETNEGA MESTA, KI PRIKAZUJE HIDROLOŠKE PODATKE	30
6.1	Spletni strežnik Microsoft Internet Information Services (IIS)	30
6.1.1	Namestitev IIS.....	30
6.1.2	Zagon IIS in nastavitve potrebne za gostovanje spletne strani	32
6.2	Izdelava spletne strani s programom Microsoft Visual Studio	35
6.2.1	Izdelava spletne strani, ki prikazuje podatke iz podatkovne baze Microsoft Access	36
6.2.2	SQL poizvedba.....	39
6.2.3	Vstavljanje preglednic in grafov	41
6.3	Izdelava spletnega mesta s CMS Joomla	44
6.3.1	Namestitev CMS Joomla:	45
6.3.2	Izdelava spletnega mesta, ki prikazuje meritve samodejnih vodomernih postaj ..	46
6.3.3	Končni izgled spletnega mesta.....	48
7	ZAKLJUČEK	51
	VIRI	52

KAZALO PREGLEDNIC

Preglednica 1: Naslovni prostor merilca Starflow	16
Preglednica 2: Mesto in vrsta napake v delovanju merilca Starflow	17

KAZALO SLIK

Slika 1: Diagram poteka	1
Slika 2: Vzdolžni prerez sistema Ljubljane od Prezida do Vrhinke	2
Slika 3: Porečje Ljubljane.....	2
Slika 4: Območja Natura 2000 na Ljubljanskem barju	4
Slika 5: Krajinski park Ljubljansko Barje, kategorizacija območij.....	5
Slika 6: Lokacije samodejnih hidroloških postaj.....	7
Slika 7: Sestavni deli merilca Starflow.....	8
Slika 8: Komunikacijski parametri serijskega protokola.....	12
Slika 9: Grafični prikaz prenosa podatkov preko asinhronskega serijskega protokola	13
Slika 10: Telit Power One 868 MHz terminal	19
Slika 11: Serial to GPRS modem	20
Slika 12: Serial to Wi-Fi pretvornik (levo); Serial to Bluetooth pret.(desno)	21
Slika 13: Serijski vmesnik.....	21
Slika 14: Delovno okolje Matlab.....	22
Slika 15: Sestavni deli tabele v podatkovni bazi Access	29
Slika 16: Nadzorna plošča, kjer namestimo IIS	31
Slika 17: Izbira namestitvenih funkcij IIS	31
Slika 18: Dodajanje pravice branja in pisanja programu IIS	32
Slika 19: Delovno okolje IIS	33
Slika 20: Dodajanje spletne strani v IIS	34
Slika 21: Sprememba .NET Frameworka v verzijo 4.0.....	34
Slika 22: Izbira novega projekta.....	36
Slika 23: Izbira spletnega obrazca	37
Slika 24: Izbira ukaza AccessDataSource	37
Slika 25: Izbira lokacije podatkovne baze.....	38
Slika 26: Možnosti izbire prikaza podatkov iz podatkovne baze	38
Slika 27: Primer SQL poizvedbe.....	40
Slika 28: Končni prikaz izbranih podatkov iz podatkovne baze	41
Slika 29: Izbira preglednice ali grafa v oknu Toolbox	42
Slika 30: Primer prikaza podatkov v preglednici	42
Slika 31: Primer grafičnega prikaza temperature in pretoka	43
Slika 32: Namestitev CMS Joomla.....	45
Slika 33: Skrbniška stran CMS Joomla	46
Slika 34: Satelitska slika območja z označenimi hidrološkimi postajami	48
Slika 35: Grafični prikaz podatkov zadnjih 7 dni.....	49
Slika 36: Tabelarni prikaz podatkov zadnjih 7 dni.....	50

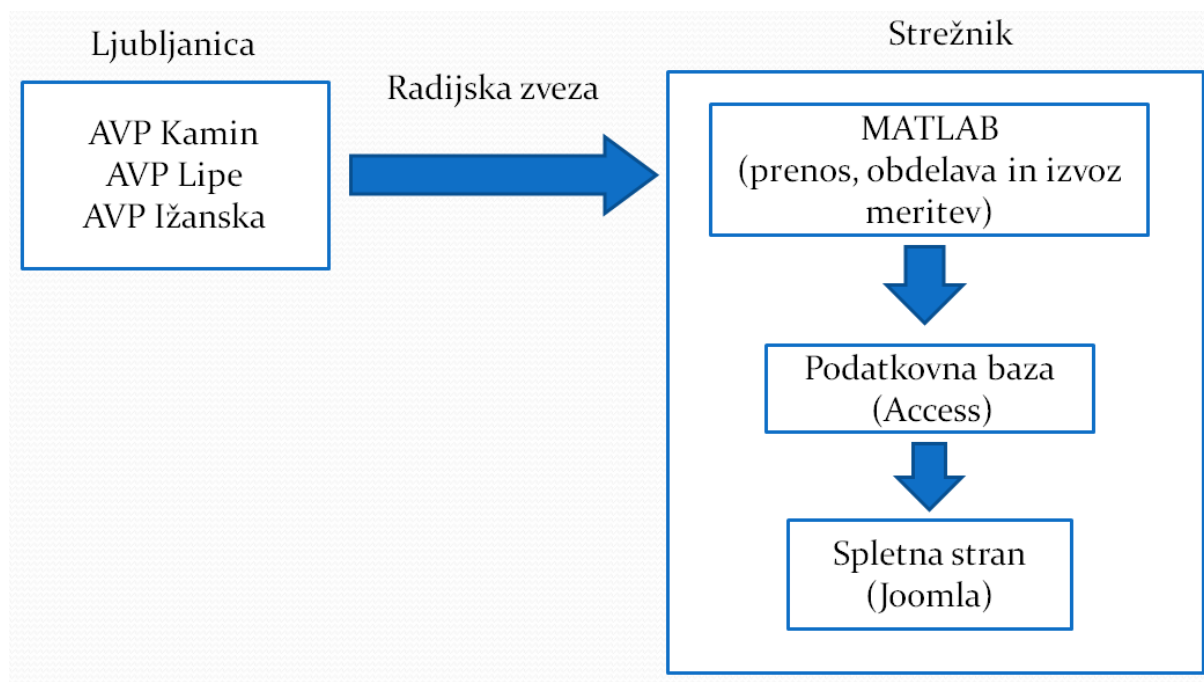
OKRAJŠAVE IN SIMBOLI

ADMP	akustični Dopplerjev merilec pretoka
CMOS	complementary metal–oxide–semiconductor
CMS	content management system (sistem za upravljanje vsebin)
COM	serijski vmesnik
CSS	cascading Style Sheets (prekrivni slogi)
DBM	database manager
GPRS	general packet radio service
GSM	global system for mobile communications
HTML	hyper text markup language (jezik za označevanje nadbisedila)
HTTP	hypertext Transfer Protocol (protokol za prenos spletnih strani)
IIS	internet information services
JDBC	java database connectivity
ODBC	open database connectivity
PC	personal computer (osebni računalnik)
RAM	random-access memory (bralno-pisalni pomnilnik)
SQL	structured query language (strukturirani povpraševalni jezik)
SUBP	sistem za upravljanje baz podatkov
URL	uniform resource locator (enolični krajevnik vira)
WWW	world wide web (svetovni splet)
XAML	extensible application markup language
XML	extensible markup language (razširljiv označevalni jezik)
Hz	hertz
l	liter
m	meter
mm	milimeter
s	sekunda
V	volt
%	odstotek
°C	stopinj Celzija

1 UVOD

V okviru projekta Ljubljana povezuje je na porečju Ljubljanice predvidena postavitvev treh samodejnih vodomernih postaj (AVP Kamin, AVP Lipe in AVP Ižanska). Postaje bodo razporejene po reki Ljubljani in pritokih. Z vodomernimi postajami nadziramo pretok, ki je pomemben za uravnavanje zapornice na Ambroževem trgu in Grubarjevem prekopu. Zapornici omogoča reguliranje vodostajev reke Ljubljanice predvsem pri nizkih vodostajih in v sušnem obdobju. To pa je ključnega pomena za ohranjanje primerne gladine Ljubljanice, ki vpliva na celotni vodni režim.

Vodometne postaje bodo opremljene z ultrazvočnim merilcem Starflow. Hidrološke veličine, ki jih lahko merimo z merilcem Starflow so hitrost, globina in temperatura vode. Vodometne postaje komunicirajo s strežnikom preko radijske zveze. Razvili smo programsko orodje (software), s katerim se daljinsko (preko radijske zveze) povežemo z merilcem Starflow in od njega pridobimo hidrološke podatke. Podatke obdelamo in izvozimo v podatkovno bazo Microsoft Access. V drugem delu diplomskega dela je opisana izdelava spletne strani, na kateri so vidni hidrološki podatki vodomernih postaj v realnem času. Opisan postopek je popolnoma avtomatiziran.

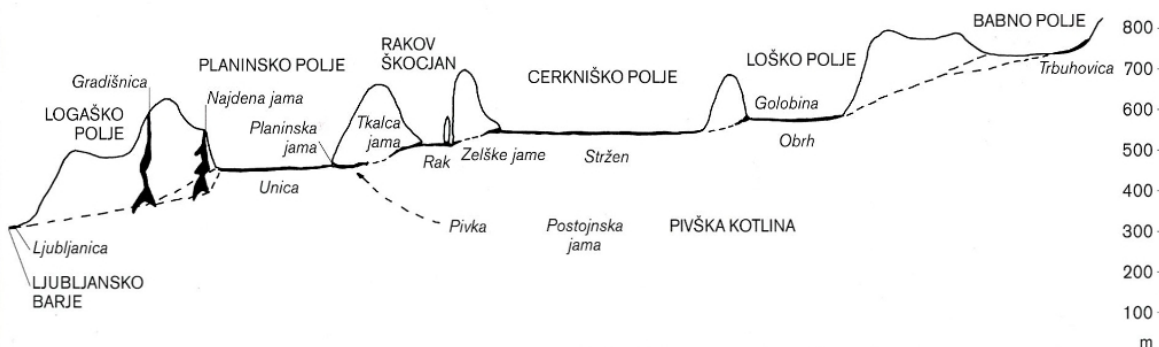


Slika 1: Diagram poteka

2 LJUBLJANICA

2.1 Splošna opredelitev reke Ljubljani

Reka Ljubljani je zelo vodnata, saj se v njej zbira voda s površine 1884 km². V Ljubljani se stekajo vode tako s kraškega površja (60 %) kot nekraškega (40 %). Večina voda s kraškega zaledja priteče podzemno s Planinskega in Cerkniskega polja. Iz nekraškega zaledja pritekajo površinske iz Polhograjskega in Posavskega hribovja (Glinščica, Gradaščica, Horjulščica, Škofeljščica, Iška, Iška, idr.). Omenjeni vodotoki dajejo režimu Ljubljani tudi deloma hudourniški značaj. Od izvirov na Vrhniki do Ljubljane preteče reka 27 km, do izliva v Savo pa 41 km (Dolšina, 2012).



Slika 2: Vzdolžni prezek sistema Ljubljani od Prezida do Vrhnike (vir: Ljubljani...)



Slika 3: Porečje Ljubljani (vir: Ljubljani...)

2.2 Pretok in poplave

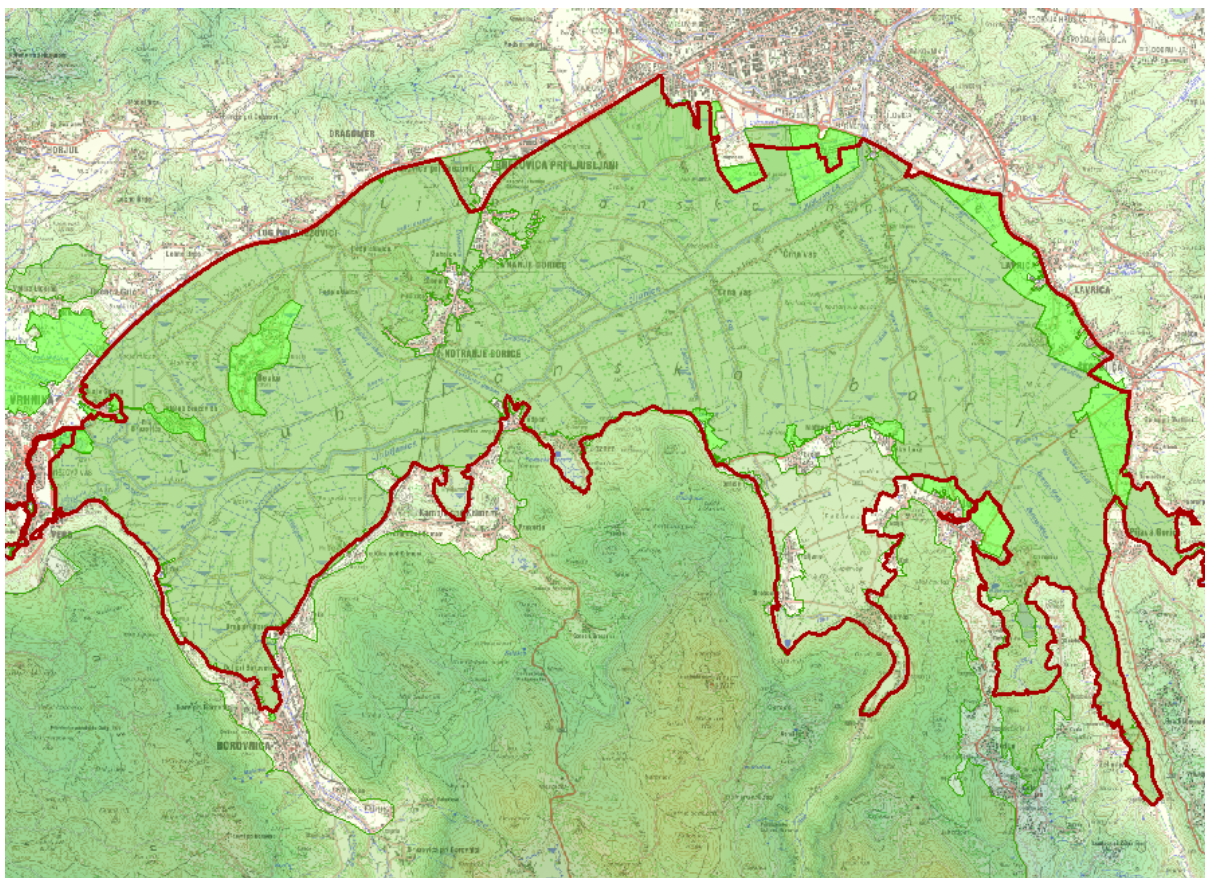
Povprečni pretok reke Ljubljanice je 25 m³/s pri Vrhniki in 55 m³/s na izlivu v Savo pri Mostah. Za reko so značilna dnevna, mesečna in letna nihanja v pretokih. Dnevna nihanja so izrazita na prehodih med suhimi in deževnimi dnevi. Z naraščanjem intenzitete padavin pretok reke hitro naraste (Lebar, 2009). Ena pomembnih značilnosti Ljubljanice in Ljubljanskega barja so poplave. Poplave so pogost pojav in so posledica razlike med dotoki in odtoki z Ljubljanskega barja. Ljubljansko barje deluje kot največji naravni zadrževalnik poplavne vode v Sloveniji. Razlivne površine so pomembne predvsem za stoletne vode. Vzroki poplav so v močnih dotokih nekraških površinskih tekočih voda (Gradaščica, Glinščica, Horjulščica, Škofeljščica, Iška, Išca, idr.), visoki talni vodi na obrobni delih Ljubljanskega barja in v majhni nagnjenosti ravnega površja. Na razdalji 20 kilometrov od Vrhnike do Ljubljane ima reka Ljubljanica manj kot metrski padec in s tako počasnim tokom ne more odvesti vseh voda, ki se stekajo vanjo. Zaradi majhnega padca reke ter razmeroma globoke struge se pri majhnih pretokih kota gladine od zapornic v Ljubljani do Vrhnike dvigne približno le za nekaj centimetrov na kilometer (Lebar, 2009).

Poplave so najpogostejše ob spomladanskem in jesenskem deževju ter pozimi, trajajo pa večinoma dva do pet dni. Ob času brez poplav zavzema Ljubljanica s pritoki približno 3 % površine barja, redne poplave zajamejo osrednje dele Barja, kjer voda prekrije okoli 14 % celotne površine, redke poplave 30 % površine, stoletne vode pa kar 60 % površine Ljubljanskega barja. Globina poplavne vode je največja med Sinjo Gorico in Notranjimi Goricami, v povprečju pa je globina poplavne vode okoli 30 – 70 cm. Najplitvejša je poplavna voda med Podpečjo in Črno vasjo (Perko, 1999). Poplavne površine se zaradi urbanizacije in širitve mesta Ljubljane zmanjšujejo. Problem predstavlja tudi nasipanje poplavnih površin, ter zmanjševanje pretočnosti pritokov Ljubljanice zaradi zaraščanja in zacevljenja. Otok je odvisen od pretočnosti Ljubljanice. Na območju mesta Ljubljane se uravnava z zapornico. Leta 1912 je bila zgrajena zapornica v Grubarjevem prekopu. Leta 1955 je bila na Ambroževem trgu zgrajena še druga zapornica. Od tedaj gladino reke uravnava obe zapornici, s čimer je omogočeno lažje obvladovanje toka, ki je lahko zelo spremenljiv (Lebar, 2009).

2.3 Varovanje Ljubljane

2.3.1 Natura 2000

Natura 2000 je evropsko omrežje varstvenih območij, kjer želimo ohranjati, izboljšati ali vzdrževati živalske in rastlinske vrste ter habitate, ki so redki ali ogroženi zaradi vpliva človeka. Glavni cilj Nature 2000 je ohraniti biotsko raznovrstnost za prihodnje rodove. To najpogosteje pomeni, da je na teh območjih treba vzdrževati ugodno stanje z različnimi aktivnimi ukrepi, ki vzdržujejo visoko biotsko raznovrstnost. Na območju Nature 2000 imajo tako prednost dejavnosti, ki ugodno vplivajo na biotsko raznovrstnost. Ugodno vplivajo bodisi neposredno (ohranjanje habitata, prilagojena kmetijska raba) bodisi posredno (ozaveščanje javnosti preko raziskovalne dejavnosti, komuniciranja). Evropska unija je to omrežje uvedla kot enega od mehanizmov za izvajanje t.i. Direktive o ohranjanju prostoživečih vrst ptic iz leta 1979 in Direktive o ohranjanju naravnih habitatov ter prostoživečih živalskih in rastlinskih vrst iz leta 1992 (Natura 2000...).

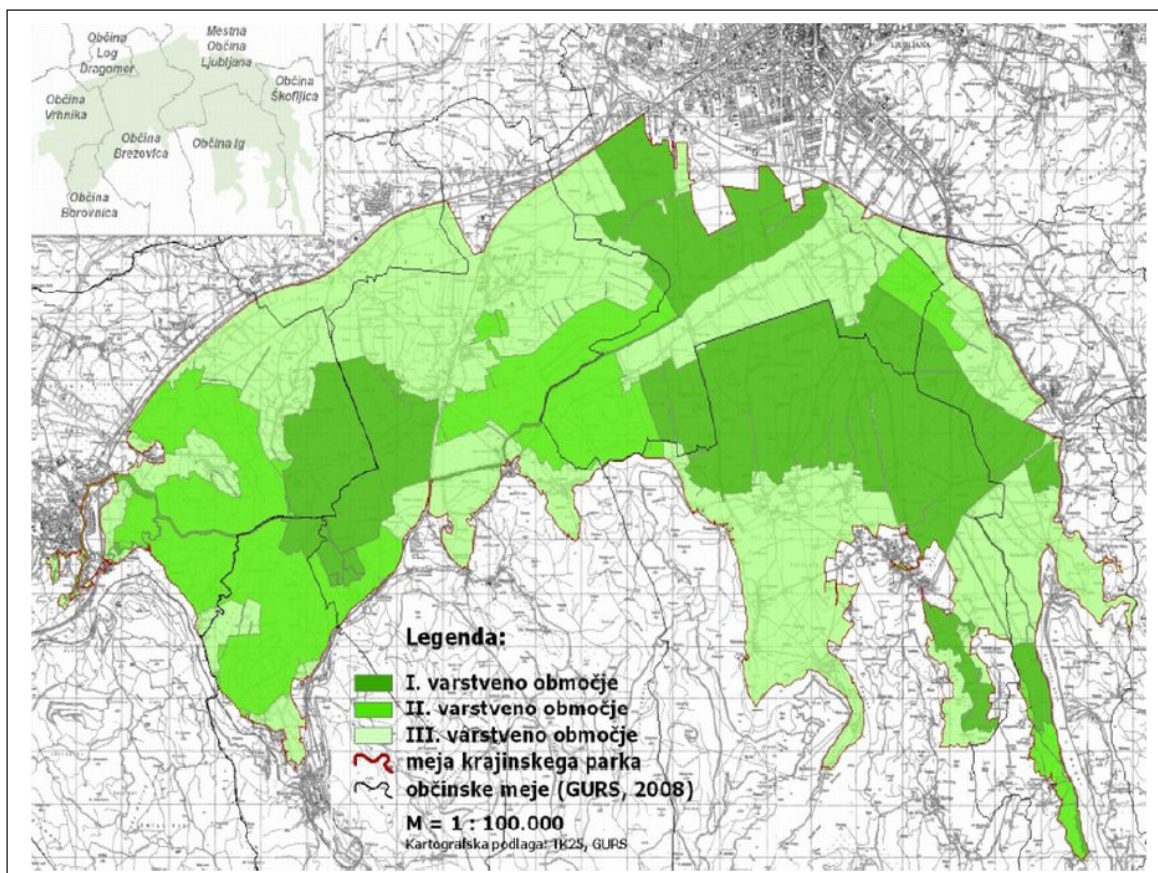


Slika 4: Območja Natura 2000 na Ljubljanskem barju (vir: Ljubljansko...)

2.3.2 Krajinski park Ljubljansko barje

KPLB je bil ustanovljen oktobra leta 2008 s ciljem, da se zavarujejo naravne vrednote, zagotavlja trajnosti razvoj, ohrani biotsko raznovrstnost in ohranja ter krepi krajinsko pestrost na območju. Park obsega 130 kvadratnih kilometrov površine in je razdeljen na tri varstvena območja, za katera so določeni različni varstveni režimi.

Prvo varstveno območje obsega 33 % celotnega območja in je naravovarstveno najpomembnejše. Prednostno je namenjeno uresničevanju varstva in ohranjanju naravnih vrednot, ugodnega stanja rastlinskih in živalskih vrst ter njihovih habitatov. Na prvem varstvenem območju se izvaja kmetijska dejavnost, ki je prilagojena doseganju ugodnega stanja rastlinskih in živalskih vrst in njihovih habitatov. Drugo varstveno območje obsega 19 % in je naravovarstveno pomembno. Namenjeno je uresničevanju varstva in ohranjanju naravnih vrednot ter biotske raznovrstnosti in pestrosti krajine. Na drugem varstvenem območju se opravljata tudi sonaravna kmetijska dejavnost in trajnostna raba prostora na način, da sta za naravo čim manj moteči. Tretje varstveno območje obsega 49 % in je prednostno namenjeno ohranjanju krajinske pestrosti in spodbujanju trajnostnega razvoja. Pozidava prostora je dovoljena v primeru, da je usklajena s cilji in omejitvami krajinskega parka (Uredba..., 2008).



Slika 5: Krajinski park Ljubljansko Barje, kategorizacija območij (vir: Varstvena..., 2008)

2.4 Lokacije samodejnih vodomernih postaj na Ljubljani

Načrtovanje merilnega mesta je kompleksen proces, pri katerem je treba upoštevati veliko dejavnikov. Predvsem pa moramo upoštevati hidravlične razmere na širšem odseku merilnega mesta in robne pogoje, npr. vpliv visokih vod odvodnika na delovanje merilnega objekta. V praksi se lahko pojavljajo številne napake na merilnih objektih (Cerar, in sod., 2000):

- napaka zaradi neustrezne nastavitve merilne opreme,
- napaka neustrezne izvedbe merilnega mesta,
- vpliv dolvodne zajezitve na merilno mesto,
- prevelika nihanja pretokov za vzpostavljeno merilno mesto (izven merilnega območja instrumenta),
- motnje zaradi prisotnosti plavja in usedlin

Sanacija neustreznih merilnih objektov v mnogih primerih zahteva velika finančna vlaganja, ali pa sploh ni mogoča, zato je najbolje že v fazi načrtovanja merilnega mesta izvesti celovito presojo ustreznosti (Cerar, in sod., 2000).

Lokacije samodejnih vodomernih postaj na Ljubljani:

Po projektu je predvidena postavitvev treh samodejnih vodomernih postaj AVP Kamin, AVP Lipe, AVP Ižanska. Meritve vodostaja in temperature izvaja ultrazvočni merilec Starflow.

Merilno mesto 1:

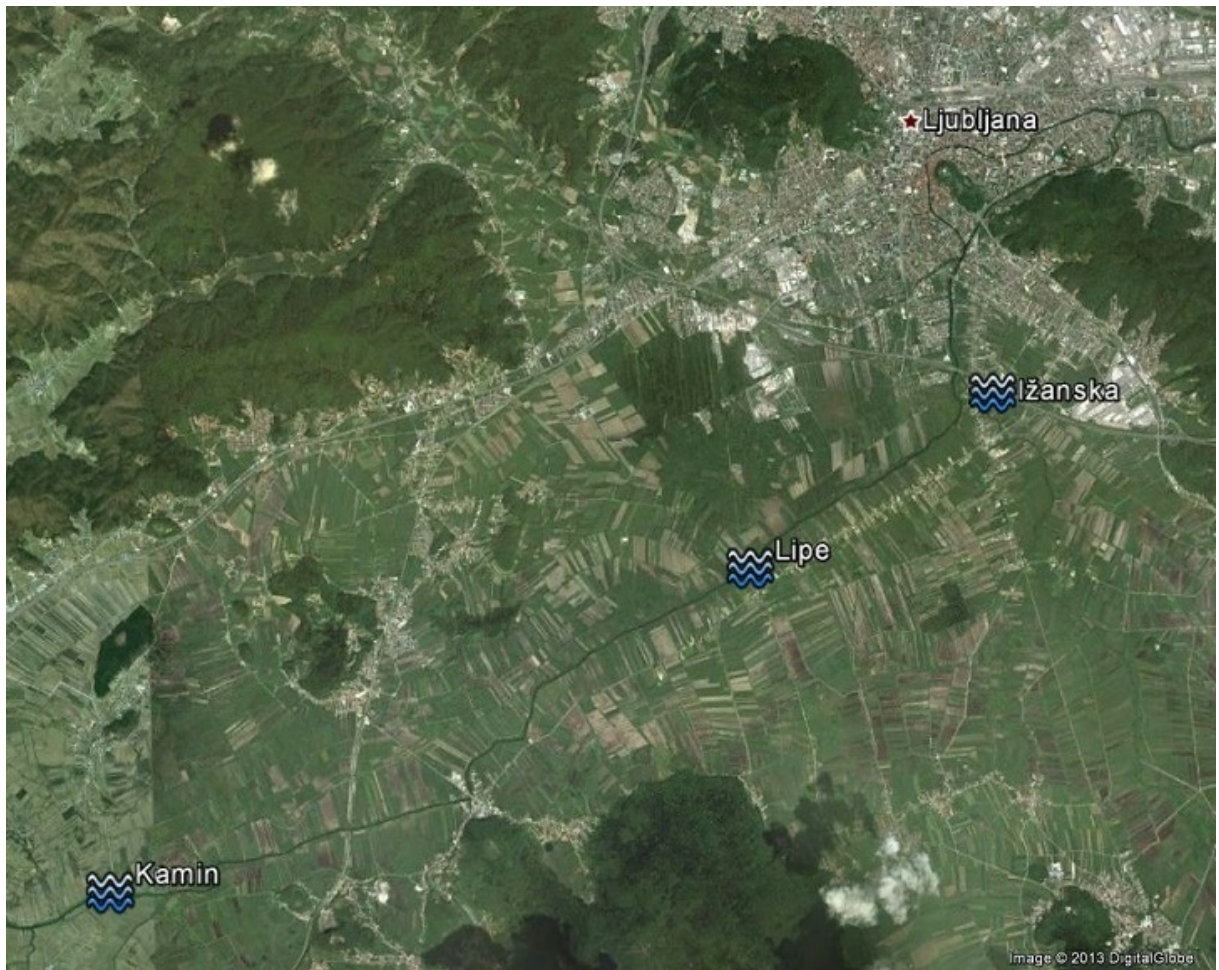
Samodejna vodomerna postaja Kamin; reka Ljubljana

Merilno mesto 2:

Samodejna vodomerna postaja Lipe; reka Ljubljana

Merilno mesto 3:

Samodejna vodomerna postaja Ižanska; reka Ižica

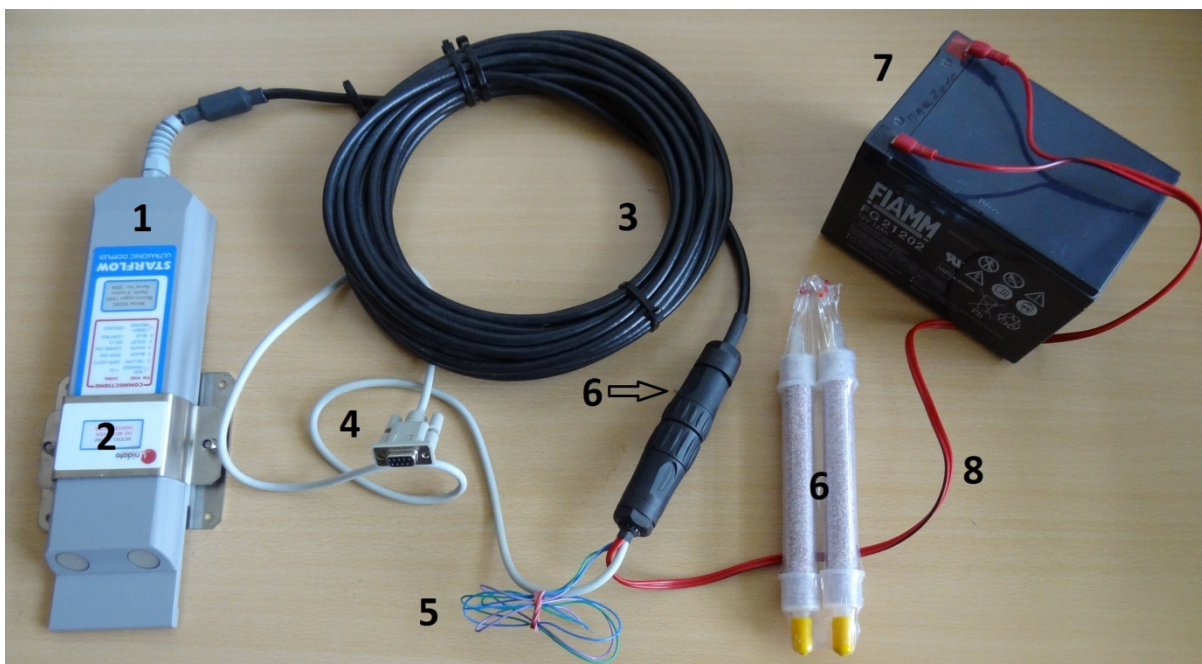


Slika 6: Lokacije samodejnih vodomernih postaj

3 ULTRAZVOČNI MERILEC STARFLOW

Ultrazvočni merilec Starflow je instrument, ki se uporablja za merjenje hitrosti, globine, temperature in pretoka. Z njim lahko merimo v kanalih, ceveh in na majhnih vodotokih, pri tem pa se ni treba ozirati na kakovost vode. Uporablja se lahko v kanalizacijskih ceveh, čistilnih napravah, morski vodi... Pretok lahko meri v obe smeri. Ta lastnost je zelo uporabna pri merjenju pretokov, ki so pod vplivom plimovanja, ali so regulirani z zapornicami oziroma tam, kjer se pojavlja povratni tok. V ceveh in odprtih kanalih lahko izračuna pretok in skupni pretok. Starflow model 6526B je kombinacija ultrazvočnega merilca in elektronike za obdelavo signalov. Deluje na principu Dopplerjevega pojava, torej meri spremembo med oddano in sprejeto frekvenco. Zasnovan je tako, da se namesti na dno vodnega kanala. Energijo za delovanje pridobiva iz 12V akumulatorja, ki je z instrumentom povezan preko priloženega kabla.

3.1 Sestavni deli



Slika 7: Sestavni deli merilca Starflow

- 1... Ultrazvočni merilec Starflow z dnevnikom podatkov (logger).
- 2... Nerjaveči jekleni nosilec
- 3... Povezovalni kabel dolžine 15m
- 4... Serijski vmesnik za povezavo z računalnikom (RS-232)
- 5... Priključek za dodatne instrumente npr. merilec količine padavin
- 6... Priključek cevi in cev s sušilom (silica gel)
- 7... Baterija
- 8... Priključni kabel baterije (6.3mm)

3.2 Tehnične lastnosti

Hitrost

Razpon: 21 mm/s do 4500 mm/s obojestransko

Natančnost: 2 % od izmerjene hitrosti

Ločljivost: 1 mm/s

Globina

Natančnost: ± 0.25 %

Model 6525-21:

Razpon: 0 to 2 m

Ločljivost: 0-1 m: 1 mm; 1-2 m: 2 mm

Model 6525-51:

Razpon: 0 to 5 m

Ločljivost: 0-2.5 m: 2.5 mm; 2.5-5 m: 5mm

Temperatura

Območje merjenja: -17 °C do 60 °C

Natančnost: 0.1 °C

Pretok

Izračun: Trenutni, skupni pretok

Izračun pretoka za: Cev, odprt kanal, naravno strugo

Integriran dnevnik podatkov

Količina spomina: 100 KB, CMOS RAM

Interval zapisovanja: nastavljiv, 5 sekund do 1 teden

Komunikacija: RS-232, 300-38400 bps

3.3 Področja uporabe

- Odtočni kanali
- Nadzor vod iz čistilnih naprav

- Manjši potoki in reke
- Opazovanje padavinskega odtoka / infiltracije
- Namakanje
- Merjenje pretokov rek, ki so pod vplivom plimovanja

3.4 Princip delovanja

Starflow za merjenje pretočnih hitrosti uporablja Dopplerjev pojav. Dopplerjev pojav je fizikalni pojav, kjer zaradi gibanja vira, opazovalca ali obeh, nastane navidezna razlika v valovni dolžini zvoka. Zvočni valovi, ki prihajajo od objektov premikanja oziroma se od teh odbijajo, se razlikujejo od zvočnih valov objektov, ki mirujejo po višini tona. Delci, ki se v vodi premikajo (lebdeči delci, majhni zračni mehurčki), katerih hitrost in smer se s pomočjo Dopplerjevega pojava lahko izračunata. To je tudi razlog, zakaj instrument ne deluje dobro v zelo čisti vodi brez delcev. Za merjenje hitrosti vode, Starflow izkorišča delce, ki se gibljejo z vodo. Če poznamo hitrost zvoka v vodi, lahko izračunamo hitrost odboja in posledično dobimo povprečno hitrost okoliške vode.

Globino vode Starflow meri s hidrostatičnim senzorjem tlaka. Za natančno določitev globine ima instrument tudi senzor atmosferskega tlaka, ki ga ustrezno upošteva pri preračunu globine vode. Instrument meri tudi temperaturo vode. Uporablja jo za izračun hitrosti zvoka v vodi. Hitrost zvoka v vodi je odvisna tudi od temperature. Temperaturo lahko zapišemo v pomnilnik za kasnejšo obdelavo.

3.5 Izračun pretoka

Pretok izračunamo tako, da pomnožimo povprečno hitrost vodnega toka s površino prečnega prereza vodotoka. V instrument lahko vnesemo stacionarne točke za cevi in korita trapezne oblike. Starflow izmeri povprečno hitrost vode in njeno globino ter izračuna pretok.

Za večje vodotoke, kjer težko vnesemo stacionarne točke, Starflow pa zaradi velikosti vodotoka težko izmeri povprečno hitrost vode, pa moramo najprej izdelati pretočno krivuljo. Pretočna krivulja je krivulja, ki prikazuje zvezo med gladino vode v vodotoku in pretokom. Omogočajo transformacijo zabeleženih vodostajev v pretoke. Informacijo o trenutnem pretoku dobimo s pomočjo merjenja vodostaja z vodomeri in odčitku odtoka s pretočnih krivulj. Zveza $Q - H$ mora biti določena in preverjena z neposredno merilno metodo (npr. pretok izračunamo kot produkt izmerjenega prečnega prereza in povprečne hitrosti), kot je npr. na podlagi meritev z akustičnim Dopplerjevim merilnikom pretoka (ADMP) ali s hidrometričnim krilom (Šupek, 2008). Izmerjene vodostaje s pripadajočimi pretoki vnesemo npr. v Excel in s pomočjo linearne ali eksponentne regresije dobimo enačbo premice ali krivulje, ki jo uporabimo pri izračunu pretoka.

3.6 Komunikacija merilca Starflow z računalnikom

3.6.1 Serijski protokol

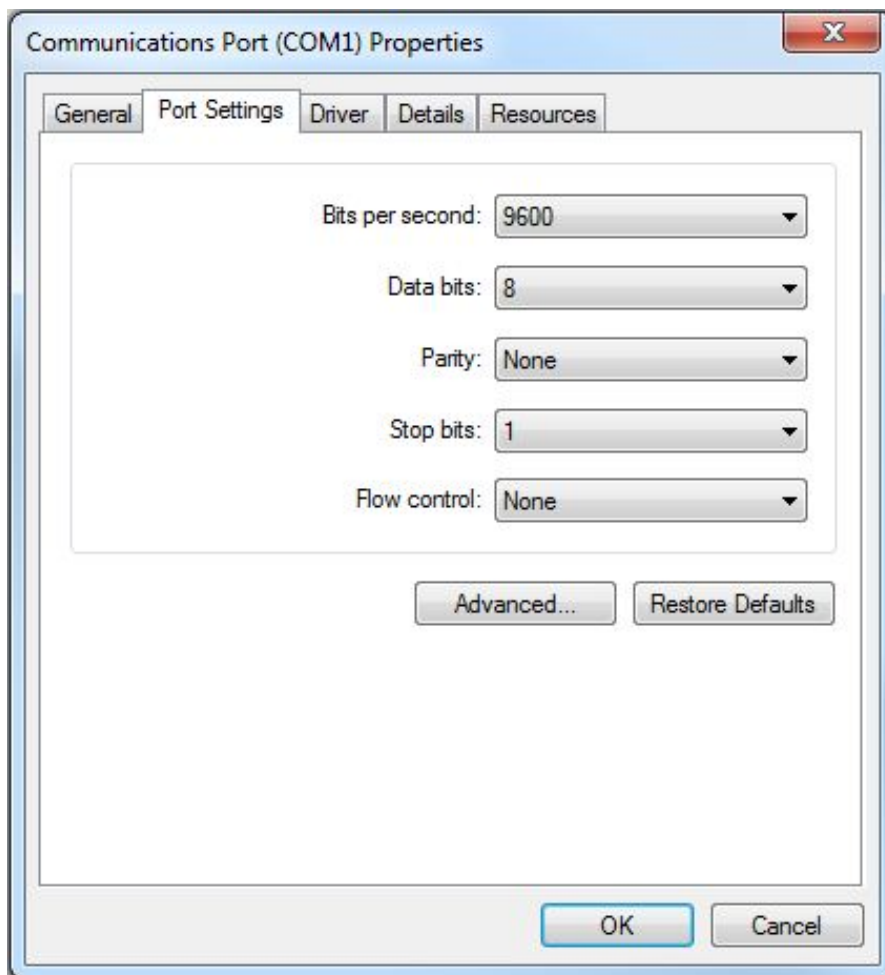
Starflow komunicira z računalnikom preko priloženega kabla, ki ga priključimo na serijski vmesnik v računalniku.

Velika prednost serijskega prenosa podatkov je, da je dobro standardiziran in ga podpira velika večina računalniških sistemov, vključno s PC-ji, delovnimi postajami, mikroročunalniki, nekateri mikrokrmilniki... Omogoča zanesljivo, poceni in preprosto povezovanje tam, kjer niso potrebne velike hitrosti. Serijski vmesnik je elektronsko vezje v računalniku, ki podpira serijski prenos podatkov. Njegova naloga je, da strojno podpira nastavitve predpisanega podatkovnega formata in električne karakteristike (Serijski vmesnik..., 2012).

Vrednosti **osnovnih komunikacijskih parametrov** nastavljam, in sicer:

- takt (hitrost takta: tipične vrednosti 300, 1200, 2400, 4800, 9600 bitov/s in več, do 115 kbitov/s),
- število podatkovnih bitov (5, 6, 7, ali 8),
- prisotnost bita parnosti in vrsta parnosti (brez parnosti, soda, liha),
- število zaključnih bitov (1, 2).

Pred prenašanjem podatkov morata imeti oddajnik in sprejemnik te vrednosti enako nastavljene. Na sliki je prikazana nastavev komunikacijskih parametrov v operacijskem sistemu Windows.



Slika 8: Komunikacijski parametri serijskega protokola

3.6.2 Serijski asinhroni protokol

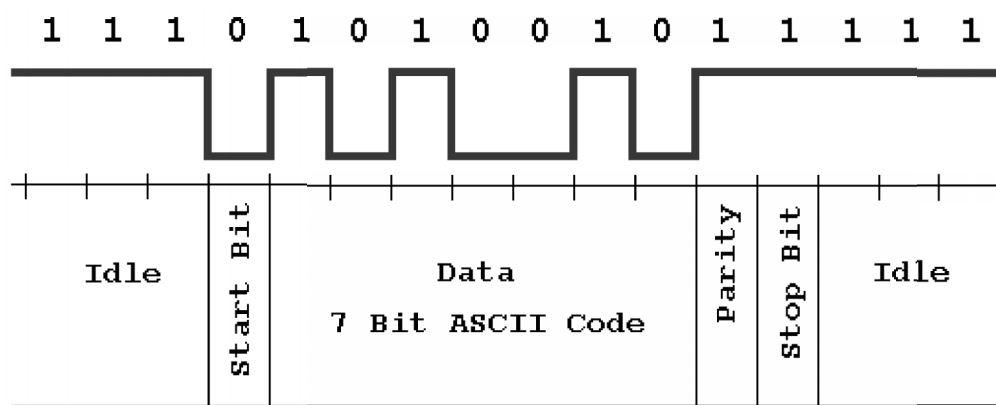
Starflow za komunikacijo uporablja serijski asinhroni protokol:

Podatki, ki prihajajo k sprejemniku, so v obliki samih "enk" in "ničel". Le te bi bile za sprejemnika brez vrednosti, če ne bi obstajal med njim in oddajno napravo nek dogovor ki pove, kako bosta pošiljala podatke. Dogovor mora vsebovati iste nastavitve o:

- hitrosti prenosa (biti/s)
- številu prenesenih bitov informacije (5, 6, 7 ali 8)
- vrsti paritete
- in številu stop bitov (1 ali 2)

Podatki, ki jih pošilja naprava imajo v okviru tega dogovora točno določeno mesto. Ko se prenos začne, oddajnik najprej postavi prenosno linijo na nizko raven. S tem pove sprejemni napravi, da je začela s prenosom. Ta začetni prehod na nizko raven označujemo s start bitom. Trajanje start bita je točno določeno s hitrostjo prenosa podatkov, ki je enaka za vse bite, ki se prenesejo. Prenos se konča s stop bitom. Ta je vedno na visokem logičnem nivoju. Linija po stop bitu ostane na visoki logični ravni, dokler oddajnik ne pošlje zopet start bita (Blažič, 2000).

Format podatkov za asinhroni protokol prikazuje naslednja slika:



Slika 9: Grafični prikaz prenosa podatkov preko asinhronskega serijskega protokola
(vir: softwareforeducation.com...)

Start bit tudi poskrbi, da se sinhronizirata oddajnik in sprejemnik. Sledijo podatki in na koncu še bit stop.

3.7 Komunikacija z računalnikom (STARLOG protokol)

Računalnik povežemo z merilcem Starflow preko 9 pinskega serijskega priključka. To nam omogoča nalaganje in prenašanje shem in podatkov. Sheme in shranjene podatke lahko prenašamo preko priložene programske opreme, vendar nam ta način ne omogoča prenos meritev v realnem času. Zato sem se odločil, da komuniciram z merilcem Starflow z uporabo programskega okolja Matlab. Opis komunikacije preko serijskega vmesnika v Matlabu je razložen v naslednjem poglavju. Celoten program je priložen v prilogi A.

Starflow komunicira z računalnikom asinhronsko, preko standardnega RS-232 serijskega vmesnika v naslednji obliki:

- RS-232C združljivost
- hitrost prenosa podatkov 300 do 76800 bitov/s (9600 bitov/s privzeto)
- 8 podatkovnih bitov, 1 stop / start bit, brez parnosti.

Ko smo povezani, nam Starflow pošlje sporočilo (*), ki mu sledi vračalka (c/r). S tem sporoči, da je pripravljen za komunikacijo. Nato Starflow čaka 1 sekundo na podatkovno zahtevo iz računalnika. Ko pošljemo zahtevo, jo merilec Starflow obdela in nam pošlje podatke, ki smo jih zahtevali. V kolikor do zahteve ne pride, se merilec Starflow izključi do naslednje meritve (običajno 5 sekund).

3.7.1 Podatkovne zahteve

V nadaljevanju so podane podatkovne zahteve (ukazi), ki jih pošljemo preko serijskega vmesnika merilcu Starflow.

Starflow pozna tri osnovne ukaze: *GET*, *PUT* in *KILL*. Podatke pošiljamo in prejemamo v šestnajstinski (heksalni) obliki.

GET ukaz:

Z GET ukazom beremo podatke oziroma meritve, kot so na primer globina, temperatura, hitrost vode, napetost baterije, pretok, kontrolne parametre... Ti podatki so zapisani na točno določenem mestu pomnilnika. V preglednici 1 so podana mesta, kjer so shranjeni podatki.

Ukaz:

GXXXXYY, kjer G predstavlja GET ukaz, XXXX je 16-bitni šestnajstiški naslov mesta, kjer se podatek prebere, YY je število bajtov, ki jih preberemo.

Primer:

Če želimo prebrati podatke o globini vode, pošljemo naslednji ukaz: G00C802.
Ukaz prebere 2 bajta podatkov na naslovu 200 do 202 (0xC8 do 0xCA).

Če želimo v enem zahtevku prebrati podatke o globini vode, temperaturi vode, napetosti baterije in hitrosti vode pošljemo naslednji ukaz: G00C808.
Ukaz prebere 8 bajtov podatkov na naslovu 200 do 208 (0xC8 do 0xD0).

Prejeti podatki so v šestnajstiški obliki.

PUT ukaz:

So zelo podobni GET ukazom s to razliko, da pri get ukazih ne beremo podatke iz merilca, ampak mu podatke pošljemo npr. sprememba ure.

Ukaz:

PXXXXYY, kjer P predstavlja PUT ukaz, XXXX je 16-bitni šestnajstiški naslov mesta, kamor pošljemo podatek, YY je število bajtov, ki jih pošljemo.

KILL ukaz:

S poslanim ukazom K (KILL) omogočimo, merilcu Starflow prehod v način mirovanja do naslednje meritve. Posledica je nižja poraba energije.

Primer:

Ukaz, ki prebere 2 bajta na naslovu 200 (0xC8) (globina vode), nato pa omogočimo merilcu Starflow, da gre v način mirovanja do naslednje meritve, lahko zapišemo: G00C802K.

Oblika podatkov, ki jih dobimo oziroma pošiljamo preko GET in PUT ukazov, je stalni niz šestnajstiških znakov. Niz podatkov se zaključuje z vračalko.

Na koncu prejetih podatkov sta dodana dva bajta, oziroma štiri šestnajstiške znake dolga kontrolna vsota (checksum), s katero preverimo prisotnost napak v prenosu.

Velikost podatkov posameznih meritev je večinoma dva bajta, oziroma štiri šestnajstiška mesta (razen podatka o napaki (1) in pretoku (4)). Podatke moramo pretvoriti iz šestnajstiške v desetiško obliko. To storimo posebej za vsak bajt podatkov. Najprej preberemo zadnji dve šestnajstiški števili in ju pretvorimo v desetiško obliko, nato pa še prvi dve šestnajstiški števili, in ju pretvorimo v desetiško število. Nato obe števili združimo in dobimo končni podatek.

3.7.2 Naslovni prostor

Podani so naslovi, na katerih preberemo podatke o zadnji meritvi.

Podatki so shranjeni v registrih z naslednjim naslovnim prostorom:

Preglednica 1: Naslovni prostor merilca Starflow (Unidata, 2011)

Naslov (dec)	Naslov (hex)	Velikost (byte)	Opis
3	3	1	Napake
160	A0	2	Območje pretočnega prereza
162	A2	4	Trenutni Pretok (l/s)
170	AA	4	Skupni Pretok (l)
200	C8	2	Globina vode (mm)
202	CA	2	Temperatura vode (.01°C)
204	CC	2	Napetost baterije (.01V)
206	CE	2	Hitrost vode (mm/s)
216	D8	2	Dopplerjev premik (Hz)
218	DA	2	Porazdelitev (%)
220	DC	2	Število vzorcev (#)

V preglednici so podani naslovi, kjer so zapisani podatki meritev. Celotna preglednica je dostopna v Starflow priročniku ver. 3.4 na strani 120. Priročnik je dostopen na naslovu:

http://www.unidata.com.au/support/manuals/suite/6526_starflow.pdf

3.7.3 Sporočanje napak v delovanju

Starflow lahko zazna nekatere napake v delovanju. Napake se delijo na opozorila in kritične napake. Kritične napake ustavijo delovanje merilca.

Vrsta napake je zapisana v registru na naslovu 3. Velikost podatka je 1 byte oziroma 8 bitov. Vsak bit predstavlja določeno vrsto napake. V primeru, da je vrednost določenega bita 1 pomeni, da napaka obstaja, v kolikor je vrednost vrednost 0, napake ni.

Mesto in vrsta napak sta podana v spodnji preglednici.

Preglednica 2: Mesto in vrsta napake v delovanju merilca Starflow (Unidata, 2007)

Številka bita	Vrsta Napake	Opozorilo / Kritična napaka
0	Checksum napaka oziroma druga nepravilnost v nastavitveni tabeli ali programski opremi	Kritična napaka
1	Napetost baterije je prenizka za varno delovanje	Kritična napaka
2	Delovanje ustavljeno, neznan napaka	Kritična napaka
5, 4	00 – Baterija OK, 01, 10 – kritična napetost 11- baterija izpraznjena	Opozorilo
6	Napaka v programski shemi (Program exceeded maximum runtime)	Opozorilo
7	Napaka v delovanju števca skeniranja	Opozorilo

Primer:

Starflow pošlje vrednost 00000000, kar pomeni, da ni napak v delovanju.

Starflow pošlje vrednost 00001000, kar pomeni, da je baterija dosegla kritično napetost.

3.8 Komunikacija z merilcem Starflow na daljavo

Pošiljanje in sprejemanje podatkov preko prenosnega medija imenujemo telemetrija. Podatki so lahko meritve, kot so na primer pretok, napetost, hitrost... Mediji za prenos podatkov so lahko različni: optični ali električni kabel, telefonska linija, pogosto pa se odločamo tudi za prenos preko radijskih frekvenc (Rojc, 2005).

Merilec Starflow lahko s strežnikom povežemo na več načinov. Kateri način (vrsto) izberemo, je odvisno od več dejavnikov, kot sta npr. medsebojna razdalja in prisotnost GSM omrežja.

3.8.1 Radijski modem Telit PowerOne 868MHz

Radijski telemetrijski modemi so primerni za razdeljene sisteme, kjer posamezne enote občasno komunicirajo z osrednjim sistemom in med seboj, predvsem pa jih uporabljamo tam, kjer bi bila uporaba drugih vrst modemov neizvedljiva ali predraga. Telemetrijske modeme najpogosteje srečujemo pri vodovodnih sistemih, v naftni industriji, pri daljinskem ogrevanju, v varnostnih sistemih in podobno (Rojc, 2005).

Radijski modem Telit PowerOne 868 MHz omogoča brezžično povezavo naprav, ki uporabljajo RS232/RS485/RS422 komunikacijo. Nadomešča kabelsko povezavo med senzorjem in računalnikom. Model PowerOne 868 MHz deluje na frekvenci 869.400 – 869.650 MHz, za katero ni potrebno pridobiti odločbe o dodelitvi radijskih frekvenc (European Directive 1999/05/EC). Oddajna moč modema je do 500 mW. Vsi deli modula (radijski oddajnik, napajanje, serijski priključki in mikroprocesor) so vgrajeni v močnem aluminijastem industrijskem ohišju s stopnjo zaščite IP65 ali IP67. V modemu je vgrajen radijski oddajnik in sprejemnik, kar omogoča komunikacijo v obe smeri. Modem omogoča povezavo več postaj, pri čemer lahko vsaka postaja deluje tudi kot repetitor.

Tehnični podatki:

- največja hitrost radijske komunikacije: 9.6 Kbit/s
- oddajna moč: od 25 mW do 500 mW
- doseg: 16 km, možna uporaba "repeaterjev" za večji doseg
- komunikacija: RS232/RS485/RS422
- napajanje: 6 - 40V enosmerno
- frekvenca: 869,400 - 869,650 MHz, širina kanala 250 kHz, 10 kanalov
- poraba energije pri 12V: Sprejem: 40mA; Prenos: 210mA; Pripravljenost: 15µA



Slika 10: Telit Power One 868 MHz terminal

Komunikacija modema s strežnikom

Komunikacija modema s strežnikom poteka s pomočjo Hayesovih ukazov oziroma ukazov AT. Ukaz AT izhaja iz angleške besede attention (pozor) in opozarja, da za nizom AT sledi ukaz modemu. Načelo le-te temelji na izmenjevanju podatkovnega (data mode) in ukaznega (command mode) prenosa podatkov. Ukaze AT pošiljamo v ukaznem načinu za inicializacijo začetnih nastavitvev, spreminjanje hitrosti prenosa podatkov... V podatkovnem načinu prenašamo čiste podatke, na primer prenos podatkov o globini, hitrosti, temperaturi vodotoka. (Kranjčič, 2005)

Modem nastavimo preko serijske zveze z AT ukazi. Uporabimo ustrezen program za komunikacijo npr. HyperTerminal. V nadaljevanju je podan primer povezave dveh radijskih modemov:

Najprej pošljemo modemu ukaz "+++". Modem s prejetim ukazom "+++" preklopi v nastavitveni način.

Nato določimo frekvenco delovanja (komunikacijski kanal). Komunikacijski kanal nastavimo v registru S200. Telit radijski modem podpira 10 kanalov v frekvenčnem območju 869,400 - 869,650 MHz. Širina kanala je 250 kHz. Če želimo uporabiti kanal 0, modemu pošljemo naslednji ukaz: "ATS200=0<CR>". AT predstavlja Hayesov ukaz, S200 pa register, kjer nastavimo komunikacijski kanal 0 (S200=0). Ukaz se zaključi z vračalko (<CR>).

Način (vrsto) delovanja nastavimo v registru S220. Radijski modem lahko deluje na več načinov. Najbolj osnovni način je Transparentni način. Način delovanja nastavimo v registru S220.

Transparentni način izberemo s številko 1. Omenjeni način omogoča point-to-point ter point-to-multipoint povezave in ga nastavimo s poslanim ukazom "ATS220=1<CR>".

Izhod iz nastavitvenega načina in preklon v komunikacijski način dosežemo z ukazom "ATO<CR>".

Podobno nastavimo tudi drug modem. Pazljivi moramo biti pri izbiri komunikacijskega kanala. Potrebno je izbrati enak kanal, kot pri prvem modemu, sicer zveza ne bo delovala.

Celoten seznam registrov je dostopen na spletni strani:

<http://www.telit.com/module/infopool/download.php?id=2491> (Telit Functionalities and Operation Modes) (str. 13 - 16).

3.8.2 Alternativni načini komunikacije

GPRS modem

V primeru, da med strežnikom in merilcem Starflow na terenu ni vidne linije, oziroma je razdalja med njima velika, uporabimo serial to GPRS modem, ki ga priključimo na merilec Starflow in strežnik. Pogoji je pokritost območja z GSM signalom. Slabost tega načina povezovanja je, da moramo imeti pri mobilnem operaterju zakupljen prenos podatkov, kar predstavlja dodaten strošek. V praksi največkrat uporabimo ta način komunikacije.



Slika 11: Serial to GPRS modem (vir: <http://tutorialelectronic.com/?p=281>)

Bluetooth in Wi-Fi

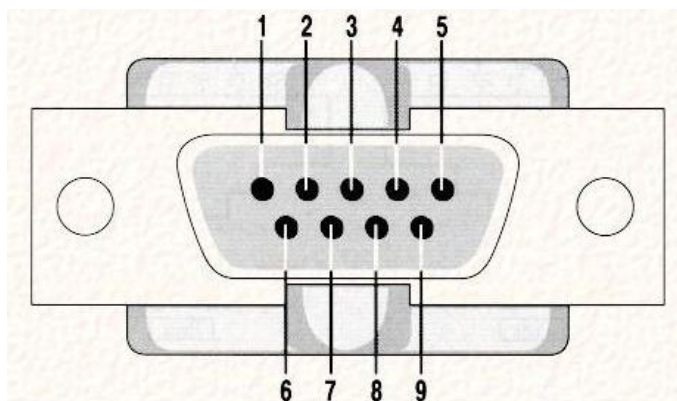
V primeru, da je medsebojna razdalja do ~10 km in obstaja vidna linija med strežnikom in merilcem Starflow, lahko podatke prenašamo preko Wi-Fi ali bluetooth protokola. Potrebujemo le ustrezen serial to Wi-Fi pretvornik, oziroma serial to bluetooth pretvornik. Doseg teh vmesnikov je nekaj kilometrov z ustrežno usmerjeno anteno (okvirni doseg lahko izračunamo s programom, ki je dostopen na naslovu: <http://www.afar.net/rf-link-budget-calculator/>). Pri sledečem načinu povezave je obvezna vidna linija med sprejemno in oddajno anteno, še posebej pri večjih razdaljah.



Slika 12: Serial to Wi-Fi pretvornik (levo) (vir:Made...); Serial to Bluetooth pret.(desno) (vir: Project...)

Povezava neposredno preko kabla

Če imamo merilec Starflow v vodotoku, ki je dokaj blizu strežnika, se lahko nanj povežemo neposredno preko serijskega kabla. Za prenos potrebujemo v najboljšem primeru le tri žice. Žica za sprejem podatkov (označena s številko 2), pošiljanje podatkov (številka 3) in žico za skupno maso (številka 5). Taka linija je lahko zelo dolga. Dolžina linije je odvisna od kakovosti kabla in hitrosti prenosa podatkov.



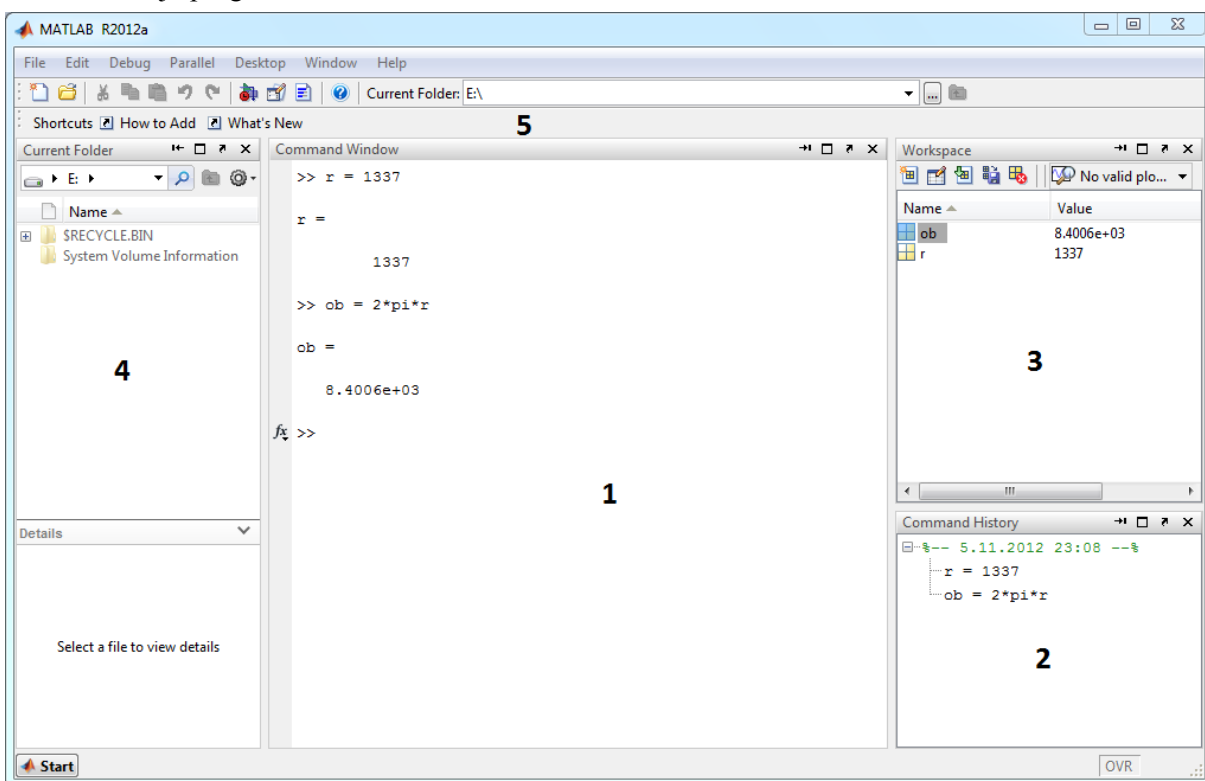
Slika 13: Serijski vmesnik (vir: <http://www.arcelect.com/rs232.htm>)

4 PROGRAMSKI JEZIK MATLAB

Matlab (**M**atrix **l**aboratory – laboratorij za matrike) je programski paket za numerično analizo. Razvija ga podjetje MathWorks. Med drugim je primeren za operacije z matrikami, risanje funkcij, implementacijo algoritmov, analizo slik, analizo in razvoj vodenja sistemov in digitalno obdelavo signalov. Omogoča tudi razvoj uporabniških vmesnikov. Programi, pisani v MATLAB-u, pa lahko uporabljajo tudi knjižnice v C-ju, C++-u, javi in fortranu.

(<http://sl.wikipedia.org/wiki/MATLAB>)

Delovno okolje programa Matlab:



Slika 14: Delovno okolje Matlab

- 1 ... Ukazno okno (command Window)
- 2 ... Okno uporabljenih izrazov (Command History)
- 3 ... Okno za prikaz spremenljivk (Workspace)
- 4 ... Izbira aktualne mape
- 5 ... Prostor za bližnjice (Shortcuts)

Matlab shranjuje programe v programskih datotekah (script files). Ime programa je lahko poljubno, vendar pa mora imeti končnico .m. Program v datoteki izvedemo tako, da napišemo njegovo ime v

ukazno okno in pritisnemo ↵. Datoteke .m ustvarimo in urejamo z lastnim urejevalnikom. Lahko ga uporabimo tudi za odkrivanje napak v programu. Zaženemo ga z ukazom Ctrl+N.

Če pri delu z ukazi naletimo na težave, je zelo uporabna funkcija help. Primer: help xlsxwrite.

V nadaljevanju so predstavljeni ukazi, ki jih potrebujemo, da od merilca Starflow pridobimo podatke o meritvah. Meritve ustrezno obdelamo in izvozimo v relacijsko podatkovno bazo Microsoft Access.

4.1 Komunikacija s serijskim vmesnikom v okolju Matlab

Osnovni princip komunikacije s serijskim vmesnikom v programu Matlab (Komun..., 2012):

1. Ustvarimo serijski objekt z ukazom *serial*. V narekovajih podamo serijska vrata (COM), na katera je merilec priključen.

```
obj = serial('port')
```

2. Z instrumentom se povežemo z ukazom *fopen*. V oklepaju podamo objekt, na katerega se povezujemo.

```
fopen(obj)
```

3. Spremenimo oziroma nastavimo vrednosti osnovnih komunikacijskih parametrov serijskega porta z ukazom *set*, in sicer: hitrost prenosa (300bps – 115kbps), število podatkovnih bitov, prisotnost bita parnosti in vrsta parnosti, število zaključnih bitov. Načeloma lahko ta korak preskočimo, saj ima Matlab že nastavljen privzete vrednosti, ki bodo delovale v večini primerov.

```
set(obj, 'BaudRate', 4800, 'Terminator', CR, ...)
```

4. Pošiljanje podatkov — Podatke napravi pošljemo z ukazom *fprintf*, ki pošlje napravi niz podatkov v tekstovni obliki, oziroma ukazom *fwrite*, ki napravi pošlje podatke v binarni obliki. 'cmd' in A predstavljata podatke, ki jih pošljemo napravi.

```
fprintf(obj, 'cmd')  
fwrite(obj, A)
```

5. Branje podatkov — Podatke preberemo z ukazom *fscanf*, za tekstovno ali *fread*, za binarno obliko. Kater ukaz bomo uporabili, je odvisno od oblike komunikacije, ki jo naprava podpira.

```
out = fscanf(obj)  
out = fread(obj)
```

6. Prekinitev povezave in izbris podatkov iz medpomnilnika — Ko ne potrebujemo več aktivne povezave z napravo, ki je povezana preko serijskega vmesnika, jo lahko zapremo z ukazom *fclose*. Z ukazom *delete* in *clear*, odstranimo objekt iz spomina oziroma iz delovnega okolja Matlab.

```
fclose(obj)
delete(obj)
clear obj
```

Primer Matlab kode za komunikacijo z merilcem Starflow:

```
s = serial('COM1');
s.Terminator = 'CR';
set(s, 'BaudRate', 9600);
fopen(s);
fprintf(s, 'G00D808')
out = fscanf(s);
fclose(s)
delete(s)
```

4.2 Krmilni stavki

Pri komunikaciji z merilcem Starflow, lahko pride do napak. Lahko se merilec na klice ne odziva, ali pa se povezava med prenosom podatkov prekine. Zato moramo v programsko kodo dodati krmilne stavke, ki del programa ponavljajo, če je neki pogoj izpolnjen (ni odziva, prekinjen prenos podatkov).

4.2.1 While zanka

Če pri komunikaciji z merilcem Starflow pride do napak, lahko dobimo nepopolne podatke. V tem primeru, lahko z *while* zanko ponavljamo zahtevo, dokler ne dobimo popolnih podatkov.

Zanka *while* označuje del programa, ki se izvaja, dokler je pogoj za izvajanje zanke izpolnjen. Program najprej preveri pogoj (expression), ki ga napišemo v izraz. Če pogoj ni izpolnjen (če ima vrednost *false*), se ne zgodi ničesar, sicer pa se izvede *telo zanke*. To je stavek (statement), ki sledi pogoju. Potem se spet preveri pravilnost pogoja. Če je še vedno izpolnjen, se telo zanke izvede še enkrat. To se ponavlja, dokler je pogoj izpolnjen. Ko pogoj ni več izpolnjen, se zanka konča.

(Zaveršnik, 2012)

Primer sintakse:

```
while pogoj
    stavek
end
```

4.2.2 If - else zanka

If – else zanko lahko uporabimo za prikaz povratnih informacij, med izvajanjem programa.

Primer sintakse:

```
if pogoj
    stavki1
elseif pogoj2
    stavki2
else
    stavki3
end
```

V primeru, če je izpolnjen *pogoj1* se torej izvedejo *stavki*. Sicer se preveri *pogoj2* in če je izpolnjen ta, se izvedejo *stavki2*. Če nista izpolnjena ne *pogoj1* ne *pogoj2* se izvedejo *stavki3*. Pri pisanju pogojev uporabljamo logične operatorje <, >=, >=, ~=, ==.

Pogojni stavki se uporabljajo za prikaz različnega izhoda v odvisnosti od postavljenih pogojev. (Pogojni..., 2012)

4.3 Izvoz podatkov v Microsoft Excel

Po obdelavi podatkov, ki smo jih dobili iz merilca Starflow, jih izvozimo v podatkovno bazo ali program za nadaljnjo obdelavo.

Matlab zapisuje podatke v Microsoft Excel z uporabo ukaza *xlswrite*.

Primer sintakse:

```
xlswrite(filename,A)
xlswrite(filename,A,sheet)
xlswrite(filename,A,xlRange)
xlswrite(filename,A,sheet,xlRange)
```

Opis:

filename = ime Excelovega dokumenta (npr. dokument.xls)

A = podatki, ki so shranjeni v Matlab oknu z imenom A (npr. A= '20.38')

Sheet = podatki se zapišejo v določen delovni list v Excelu (npr. Sheet2)

xlRange = območje celic, v katerega zapišemo podatke (npr. C2:D5)

4.4 Izvoz podatkov v podatkovne baze

Podatke v Matlabu lahko izvozimo v podatkovne baze s pomočjo ODBC (Open Database Connectivity) in JDBC (Java Database Connectivity) gonilnika. S pomočjo ODBC in JDBC gonilnika imamo možnost izvoza podatkov v podatkovne baze, kot so na primer: Oracle, My SQL, Microsoft SQL Server, Microsoft Access.

Ukaz za povezavo z bazo prek JDBC in ODBC gonilnika v Matlabu:

```
conn = database(instance, username, password, driver, databaseurl)
```

Opis:

instance = Vir podatkov za ODBC in JDBC povezavo.

username = Uporabniško ime, ki je potrebno za dostop do baze. Če je dostop mogoč brez uporabniškega imena, zapišemo "".

password = Geslo, ki je potrebno za dostop do baze. Če je dostop mogoč brez gesla, zapišemo "".

driver = ime JDBC gonilnika. Primer za Microsoft Access:

```
'sun.jdbc.odbc.JdbcOdbcDriver'
```

databaseurl = Je sestavljen iz JDBC in ODBC gonilnika ter URL-ja (enolični krajevnik vira) povezave baze podatkov. JDBC gonilnik in URL povezava sta za vsako vrsto podatkovne baze različna. Primer za Microsoft Access:

```
['jdbc:odbc:Driver={Microsoft Access Driver (*.mdb)}; DBQ='  
'povezava_do_baze'.mdb]
```

Seznam JDBC, ODBC gonilnikov in URL povezav za različne podatkovne baze, je dostopen na naslovu: http://www.mathworks.com/help/database/ug/database.html#inputarg_databaseurl

Po vzpostavitvi povezave s podatkovno bazo, podatke izvozimo z ukazom *fastinsert*.

```
fastinsert(conn, 'tablename', colnames, exdata)
```

Opis: *fastinsert* izvozi zapise iz spremenljivke *exdata*. Zapise izvozimo v podatkovno bazo, ki vsebuje tabelo z imenom '*tablename*', preko povezave *conn*. V spremenljivki *colnames*, so podana imena stolpcev v tabeli, kamor se podatki zapišejo.

Primer kode, s katero vzpostavimo povezavo s podatkovno bazo Microsoft Access in izvozimo podatke o meritvah:

```
dbpath = 'C:\rootWWW\SFData.mdb';  
connurl = ['jdbc:odbc:Driver={Microsoft Access Driver (*.mdb)}; DBQ='  
dbpath];  
conn = database('', '', '', 'sun.jdbc.odbc.JdbcOdbcDriver', connurl)  
colnames={'ImeSF', 'SN', 'Datum', 'Ura', 'Globina', 'Hitrost',  
'Temperatura', 'Baterija', 'Napaka'};  
exdata={'Kamin', '3884', Datum, Ura, GLOBINA, HITROST, TEMPERATURA,  
BATERIJA, NAPAKA};  
fastinsert(conn, 'SF_Podatki', colnames, exdata)
```

4.5 Funkcija hex2dec

Podatke iz merilca Starflow dobimo v šestnajstiški obliki, ki jih spremenimo v desetiško. To storimo s funkcijo *hex2dec*, ki pretvori šestnajstiška števila v desetiška.

Primer sintakse:

```
d = hex2dec('šestnajstiška_vrednost')
```

Program za komunikacijo z merilcem Starflow, s katerim od merilca pridobimo podatke, jih obdelamo in izvozimo v podatkovno bazo Microsoft Access je priložen v prilogi A.

5 MICROSOFT ACCESS

Microsoft Access je sistem za upravljanje baz podatkov (SUBP) in je vključen v paket Microsoft Office. Je preprost za uporabo ter hkrati dovolj zmogljiv, da z njim izdelamo preproste relacijske baze podatkov. Poleg Microsoft Accessa poznamo tudi druge relacijske SUBP, ki so zmogljivejši. Taki SUBP so npr. Oracle, Microsoft SQL Server, DB2 ter Informix, med odprtokodnimi SUBP pa je npr. poznan MySQL.

Microsoft Access 2007, 2010 datoteke imajo končnico .accdb. Če pa želimo ustvariti podatkovno bazo, ki je združljiva s prejšnjimi različicami Microsoft Accessa (2003), pa uporabimo končnico .mdb.

5.1 Kaj so relacijske baze podatkov

- podatki so vidni kot zbirka imenovanih tabel
- tabele so sestavljene iz stolpcev in vrstic
- vrstica je horizontalnih del tabele, ki predstavlja zbirko vrednosti, ki pripadajo določeni vrednosti ključa
- vse vrstice v tabeli imajo enako strukturo, vrstni red vrstic pa ni pomemben
- vrstice v neki tabeli so enolično določene s pomočjo vrednosti ključa
- stolpec je vertikalni del tabele in predstavlja zbirko vrednosti istega tipa, vsak stolpec pa ima svoje ime
- kjer se križata stolpec in vrstica, je specifični podatek, ki se imenuje vrednost
(www.squidoo.com/sql-osnove)

5.2 Ustvarjanje nove baze podatkov

V bazi podatkov so podatki shranjeni v tabelah. Še preden se lotimo ustvarjanja posameznih tabel, moramo narediti novo prazno bazo podatkov (ustvariti moramo novo Accessovo datoteko) v katero bomo te tabele shranjevali. Novo bazo podatkov ustvarimo tako, da izberemo prazno bazo podatkov (Blank database). V pogovornem oknu vpišemo ime nove baze podatkov in izberemo mesto, kjer bo baza shranjena. Naslednji korak je priprava tabel. Tabele so osnovni gradnik baze podatkov, saj so v njih shranjeni podatki. Za razliko od ostalih programov iz zbirke Microsoft Office je potrebno v Accessu bazo najprej shraniti, preden lahko začnemo z njo delati.

5.3 Vnašanje podatkov v tabele

Ko imamo tabelo pripravljeno, lahko začnemo podatkovno bazo uporabljati. Tabele so sestavljene iz stolpcev, kjer vnesemo kategorijo podatkov (na spodnji sliki označeno s številko 1) in vrstic, kjer so zapisani podatki (na spodnji sliki označeno s številko 2). V stolpcih je zapisana kategorija podatkov,

na primer: Datum, Napetost baterije, Globina... V vrsticah pa nabor polj, ki so povezane z entiteto (na primer Datum, Napetost baterije, Globina...).

Zaporedna št.	ImeSF	SN	Datum	Ura	Pretok	Globina	Hitrost	Temperatura	Baterija
40224	Kamin	3884	12.08.2013	19.30.03	0.00	0.006	0.00	27.08	12.28
40223	Kamin	3884	12.08.2013	19.15.03	0.00	0.006	0.00	27.21	12.28
40222	Kamin	3884	12.08.2013	19.00.03	0.00	0.006	0.00	27.34	12.28
40221	Kamin	3884	12.08.2013	18.45.03	0.00	0.006	0.00	27.28	12.28
40220	Kamin	3884	12.08.2013	18.30.03	0.00	0.006	0.00	27.15	12.28
40219	Kamin	3884	12.08.2013	18.15.03	0.00	0.006	0.00	27.08	12.28
40218	Kamin	3884	12.08.2013	18.00.03	0.00	0.006	0.00	27.08	12.28
40217	Kamin	3884	12.08.2013	17.45.03	0.00	0.006	0.00	27.08	12.28
40216	Kamin	3884	12.08.2013	17.30.03	0.00	0.006	0.00	27.08	12.28
40215	Kamin	3884	12.08.2013	17.15.03	0.00	0.006	0.00	27.01	12.28

Slika 15: Sestavni deli tabele v podatkovni bazi Access

V našem primeru Matlab avtomatično izvozi podatke ob vsaki meritvi v Microsoft Access. Narediti moramo le tabelo in poimenovati kategorije podatkov, ki jih uvažamo.

Za kasnejši prikaz podatkov na spletu pa moramo poznati SQL poizvedbe, s katerimi iz množice podatkov, ki jih hranimo v bazi, pridobimo informacije. Pri tem razumemo informacijo kot podatek, ki ima za uporabnika neko vrednost, na primer prikaz pretoka v nekem časovnem obdobju. Uporaba SQL poizvedb je razložena v poglavju Microsoft Visual Studio.

6 IZDELAVA SPLETNEGA MESTA, KI PRIKAZUJE HIDROLOŠKE PODATKE

Če želimo izdelati spletno mesto, ki je vidno na spletu, potrebujemo spletni strežnik. V tem poglavju je najprej prikazan postopek namestitve spletnega strežnika Internet Information Services (IIS). Nato je prikazana izdelava spletne strani. Spletna stran prikazuje podatke samodejne vodomerne postaje v grafični in tabelarični obliki. Spletna stran je vključena v spletno mesto. Izdelava spletnega mesta s CMS Joomla je prikazana na koncu poglavja.

V naslednjem poglavju je prikazana namestitev spletnega strežnika Internet Information Services (IIS), ki omogoča gostovanje spletnih strani.

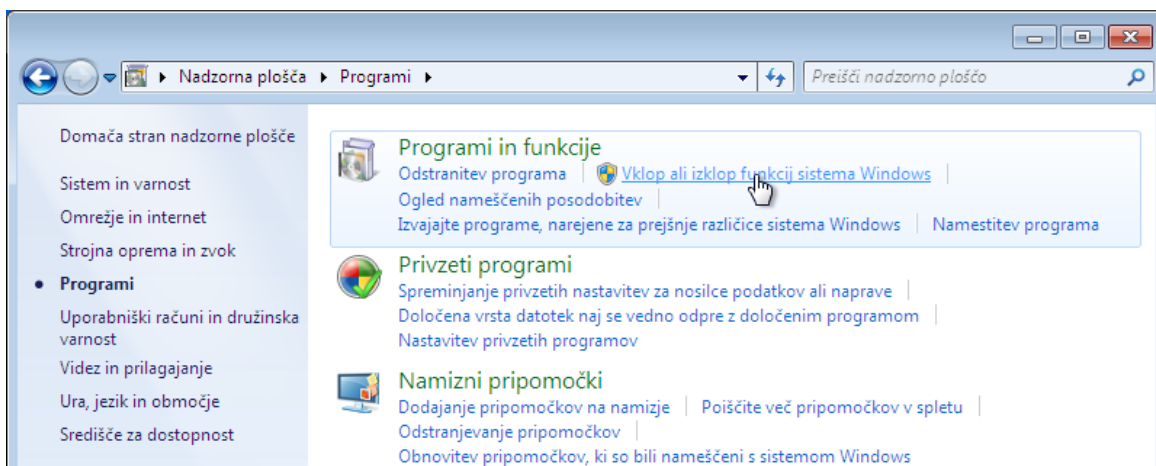
6.1 Spletni strežnik Microsoft Internet Information Services (IIS)

Microsoft Internet Information Services (IIS) 7.0 je spletni strežnik. Je varen in preprost za uporabo. IIS 7.0 omogoča različne načine upravljanja spletnega strežnika, spletnih mest in spletnih programov. Je drugi najpopularnejši spletni strežnik na svetu. Ponuja paleto novih zmožnosti v primerjavi s prejšnjimi različicami, kot so na primer: boljše diagnosticiranje in odpravljanje napak, boljše upravljanje z orodji, večja varnost... Podpira gostovanje poljubnih framework knjižnic med drugim tudi: ASP.NET, ASP, PHP, PERL, Ruby. Ob tem pa je stroškovno učinkovit, saj omogoča gostovanje več tisoč strani na enem strežniku IIS.

6.1.1 Namestitev IIS

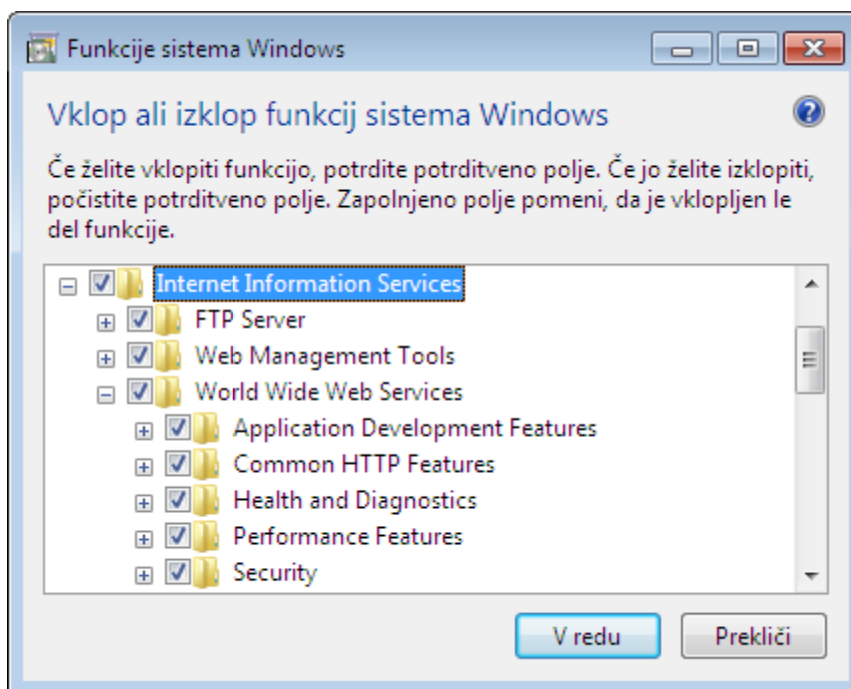
Preden lahko gostujemo spletno stran, do katere lahko dostopamo preko spleta, moramo namestiti IIS. Postopek je enostaven in je opisan za windows 7, postopek je podoben tudi v drugih Windows operacijskih sistemih, XP in novejši.

IIS lahko namestimo v: Start->Nadzorna plošča->Programi-> Vklon ali izklop funkcij sistema Windows.



Slika 16: Nadzorna plošča, kjer namestimo IIS

V oknu označimo Internet Information Services in World Wide Web Services, ter potrdimo izbiro.

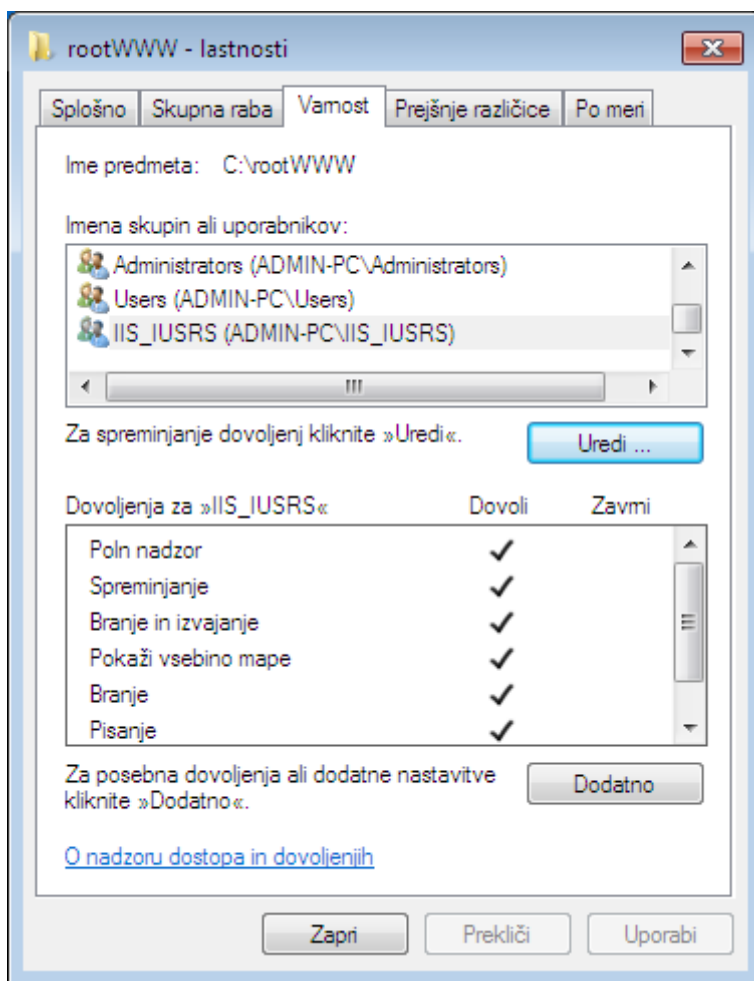


Slika 17: Izbira namestitvenih funkcij IIS

Nameščanje lahko traja nekaj minut. Ko je namestitev končana, smo uspešno namestili IIS.

6.1.2 Zagon IIS in nastavitve potrebne za gostovanje spletne strani

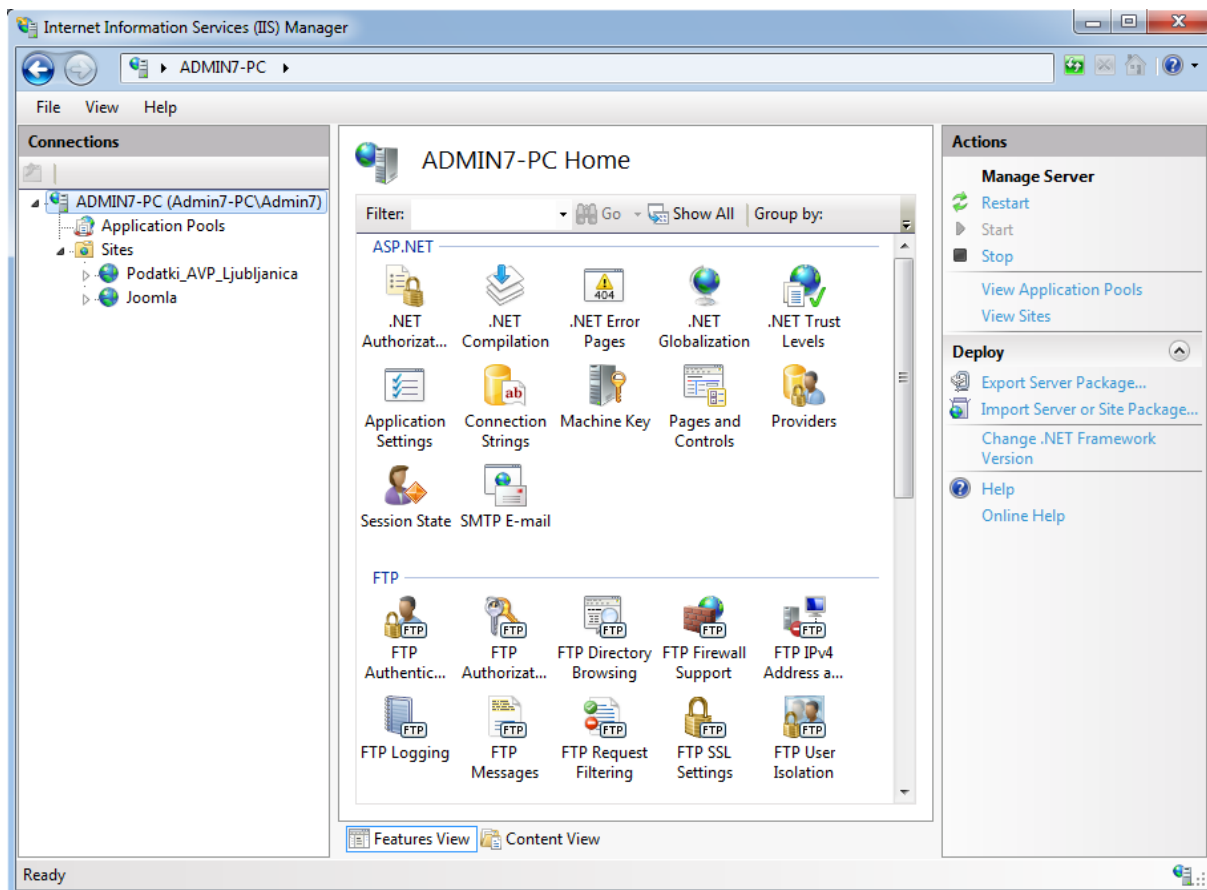
Najprej izberemo mesto na računalniku, kjer bodo shranjene datoteke spletne strani, v našem primeru je to C:\rootWWW. Programu IIS moramo podeliti ustrezne pravice branja in pisanja v sledeči mapi. To storimo tako, da kliknemo na mapo, ki vsebuje spletno stran in izberemo možnost 'lastnosti'. V zavihku varnost dodamo v okno 'Imena skupin ali uporabnikov' novega uporabnika *IIS_IUSRS*, in mu omogočimo popoln nadzor v mapi.



Slika 18: Dodajanje pravice branja in pisanja programu IIS

Nato zaženemo program IIS.

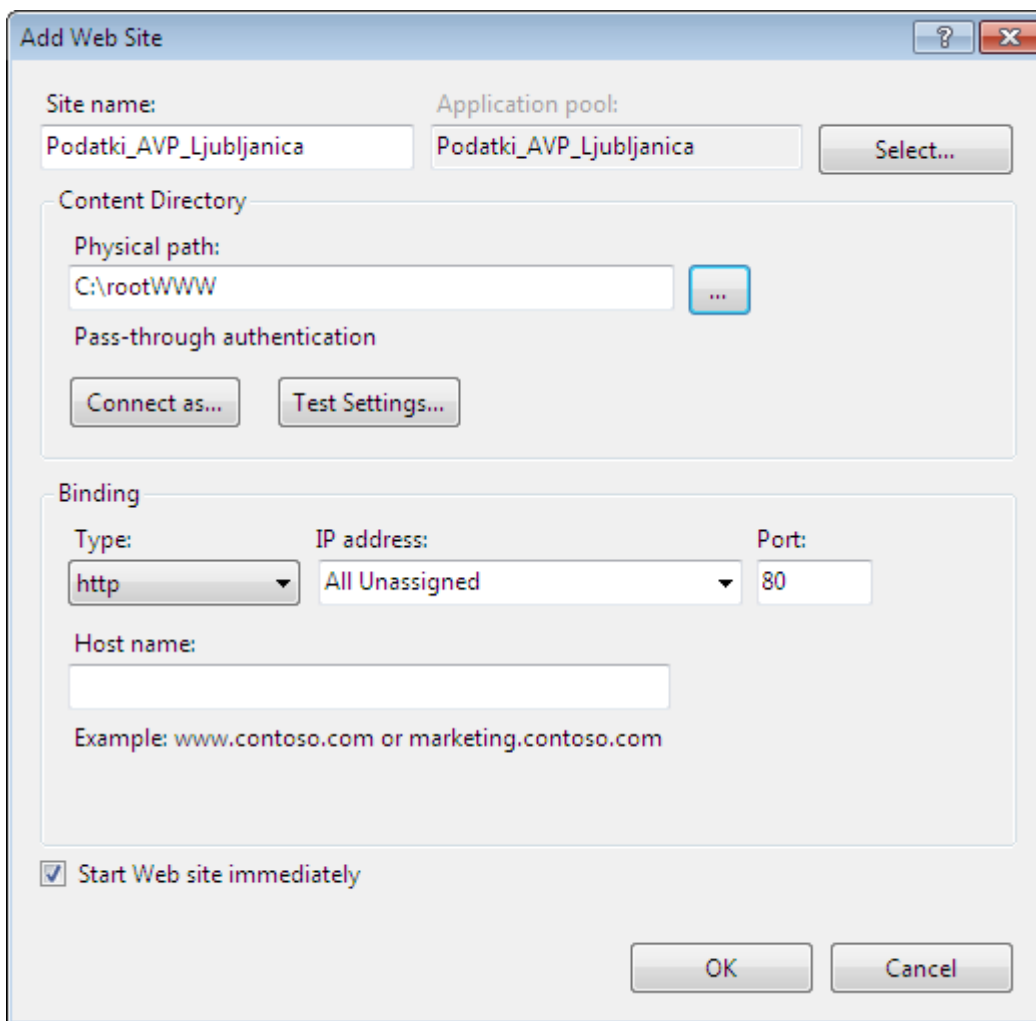
IIS Manager najdemo v: Start-> Nadzorna plošča-> Skrbniška orodja -> IIS Manager, lahko ga pa zaženemo tudi v komandnem (cmd) oknu z ukazom *inetmgr*.



Slika 19: Delovno okolje IIS

V IIS Managerju najprej dodamo novo spletno stran (Add Web Site).

V okno *Site name* vnesemo ime spletne strani in določimo direktorij, v katerem je spletna stran shranjena (Physical Path).



Slika 20: Dodajanje spletne strani v IIS

Na koncu moramo še spremeniti verzijo .NET Frameworka v Applications Pools. Naša spletna stran je zgrajena v ASP.NET v4.0. Privzeto pa je v IIS nastavljena verzija v2.0, ki jo spremenimo v verzijo v4.0.

Name	Status	.NET Frame...	Managed Pipeli...	Identity	Appli...
Podatki_AVP_Ljubljana	Started	v4.0	Integrated	ApplicationPoolIdentity	1

Slika 21: Sprememba .NET Frameworka v verzijo 4.0

V primeru napak, ki jih prikaže spletni brskalnik (Error 500, 403.7, 500.19...), jih enostavno diagnosticiramo s pomočjo spletnega iskalnika.

6.2 Izdelava spletne strani s programom Microsoft Visual Studio

Spletna stran (ang. Web page) je dokument z besedilom. Spletno stran prikažemo z brskalnikom. Na spletni strani so lahko različne vsebine: besedilo, slike, povezave, zvočni in video posnetki, programi. Spletno stran opišemo z jezikom HTML. Za izdelavo spletnih strani so potrebni posebni programi. Npr. Front Page, Microsoft Visual Studio, Microsoft Office SharePoint Designer ipd. Načeloma je možno spletno stran izdelati tudi v beležnici, vendar je s priročnimi programi lažje, saj imajo prednastavljene različne ukaze in pripomočke, ki nam olajšajo delo.

(http://sl.wikipedia.org/wiki/Spletna_stran)

Visual Studio je integrirano razvojno orodje. Uporablja se za razvoj in izdelavo spletnih aplikacij ASP, XML, spletnih storitev, namiznih in mobilnih aplikacij. Programski jeziki Visual Basic .NET, Visual C++ , Visual C# in Visual J# uporabljajo skupno integrirano razvojno okolje (IDE), kar nam pomaga pri izdelavi aplikacij in storitev. Visual Studio nudi podporo tudi drugim programskim jezikom, kot sta na primer Python in Ruby, ki ju lahko ločeno namestimo.

Visual Studio uporablja enotni izbor knjižnic .NET Framework, kar omogoča hitrejše in enostavnejše razvijanje programske kode, poleg tega pa podpira več kot 20 programskih jezikov (Radinović, 2006).

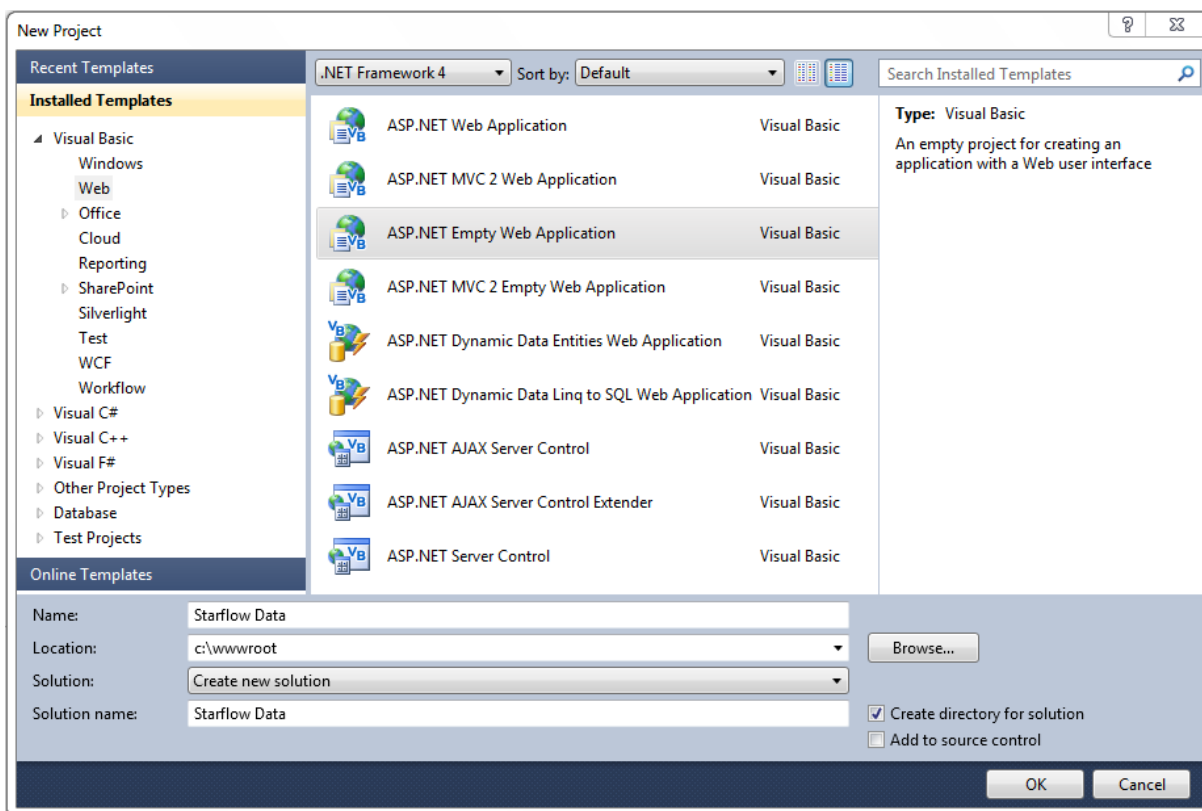
Za uporabo programa Microsoft Visual Studio sem se odločil zaradi dobre podpore pri izdelavi grafov in tabel, ki zajemajo podatke iz podatkovne baze Microsoft Access.

Programsko okolje Visual Studio 2010 je študentom in raziskovalnemu osebju dostopno brezplačno na spletni strani Microsoft Developer Network Academic Alliance (MSDNAA).

6.2.1 Izdelava spletne strani, ki prikazuje podatke iz podatkovne baze Microsoft Access

V nadaljevanju je prikazana izdelava preproste spletne strani v programu Microsoft Visual Studio 2010, ki vključuje prikaz meritev iz podatkovne baze Microsoft Access. Meritve prikažemo v grafični in tabelarni obliki. Spletna stran je narejena v knjižnici ASP.NET, z uporabo programskega jezika Visual Basic .NET.

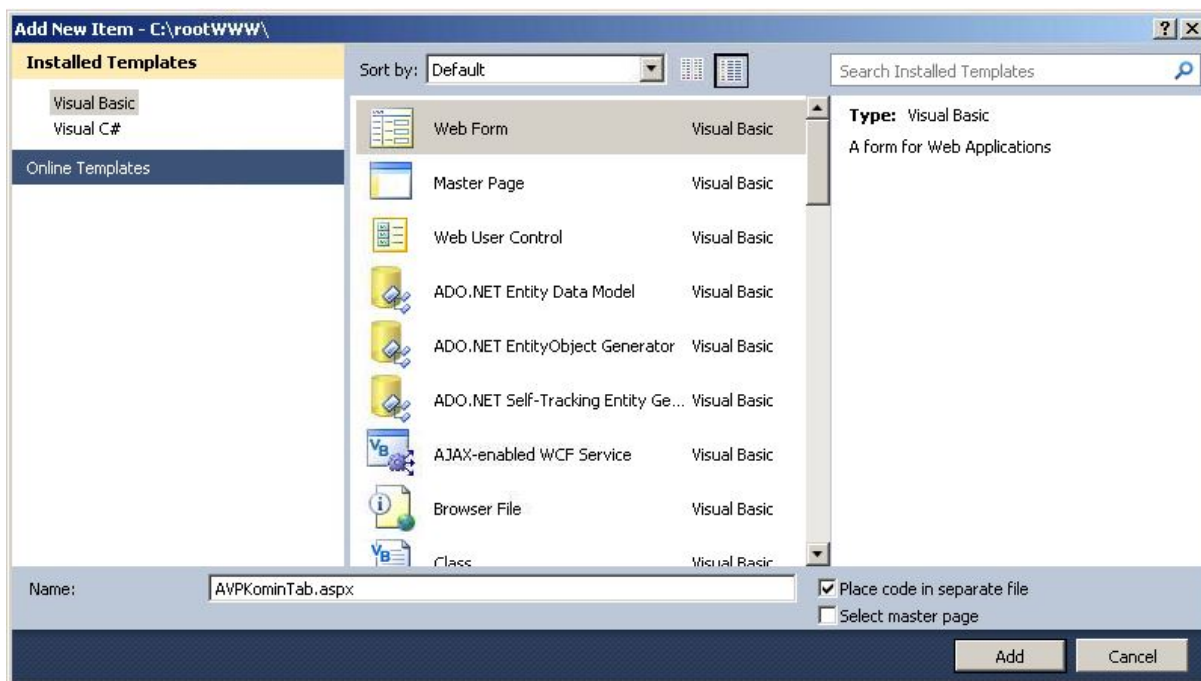
Najprej ustvarimo nov Projekt. Projekt upravlja z izvorno kodo, podatkovnimi povezavami in referencami. To storimo tako, da v levem zgornjem kotu izberemo možnost: File->New-> Project (Ctrl+Shift+N). Odpre se nam pogovorno okno, kjer izberemo vrsto aplikacije, ime aplikacije, lokacijo in programski jezik. Za izdelavo spletne strani izberemo ASP.NET Empty Web Application.



Slika 22: Izbira novega projekta

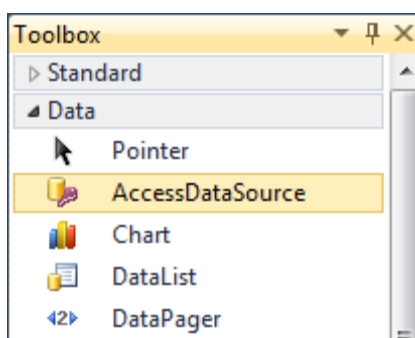
V projekt nato dodamo spletni obrazec (Web Form), to je datoteka s končnico .aspx, v katero dodamo grafe ali tabele s podatki. Spletni obrazec dodamo z ukazom Add New Item (Ctrl+Shift+A). Odpre se nam okno, v katerem izberemo Web Form. V spodnjem delu okna pa poimenujemo spletno stran.

AVPKaminTab.aspx je spletna stran, v katero dodamo tabele in grafe s podatki.



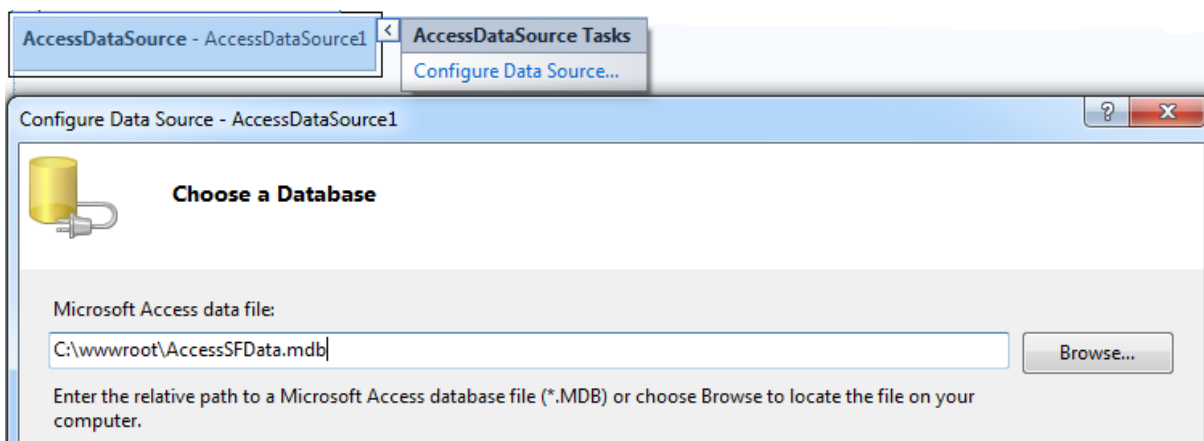
Slika 23: Izbira spletnega obrazca

Preden lahko dodamo grafe in preglednice s podatki, moramo podati lokacijo podatkovne baze (MS Access), iz katere pridobimo podatke. To storimo z ukazom `AccessDataSource`. Ukaz najdemo na desnem robu v oknu Toolbox v zavihku Data.



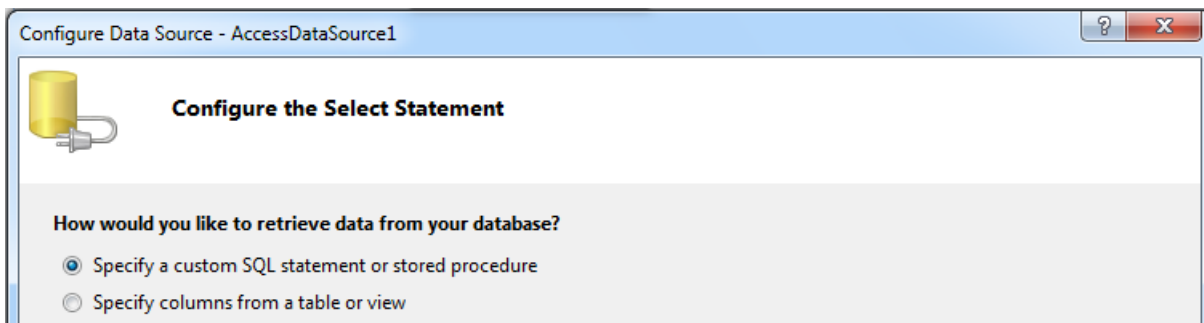
Slika 24: Izbira ukaza `AccessDataSource`

Odpre se pogovorno okno, kjer izberemo lokacijo Microsoft Access datoteke.



Slika 25: Izbira lokacije podatkovne baze

V naslednjem koraku izberemo, kateri podatki iz podatkovne baze, bodo prikazani na spletni strani. Imamo možnost, kjer izberemo kateri stolpci (temperatura, datum, pretok...) iz MS Access baze bodo prikazani, lahko pa sami sestavimo SQL poizvedbo. Lastna SQL poizvedba nam omogoča več svobode pri izbiri podatkov, prikazanih na spletni stran.



Slika 26: Možnosti izbire prikaza podatkov iz podatkovne baze

6.2.2 SQL poizvedba

SQL ali strukturirani povpraševalni jezik za delo s podatkovnimi bazami (angl. *Structured Query Language*) je najbolj razširjen in standardiziran povpraševalni jezik za delo z relacijskimi bazami podatkov. S pomočjo poizvedb iz množice podatkov, ki jih hranimo v bazi podatkov, pridobivamo informacije. Pri tem razumemo informacijo kot podatek, ki ima za uporabnika neko vrednost.

6.2.2.1 Uporaba SQL poizvedb

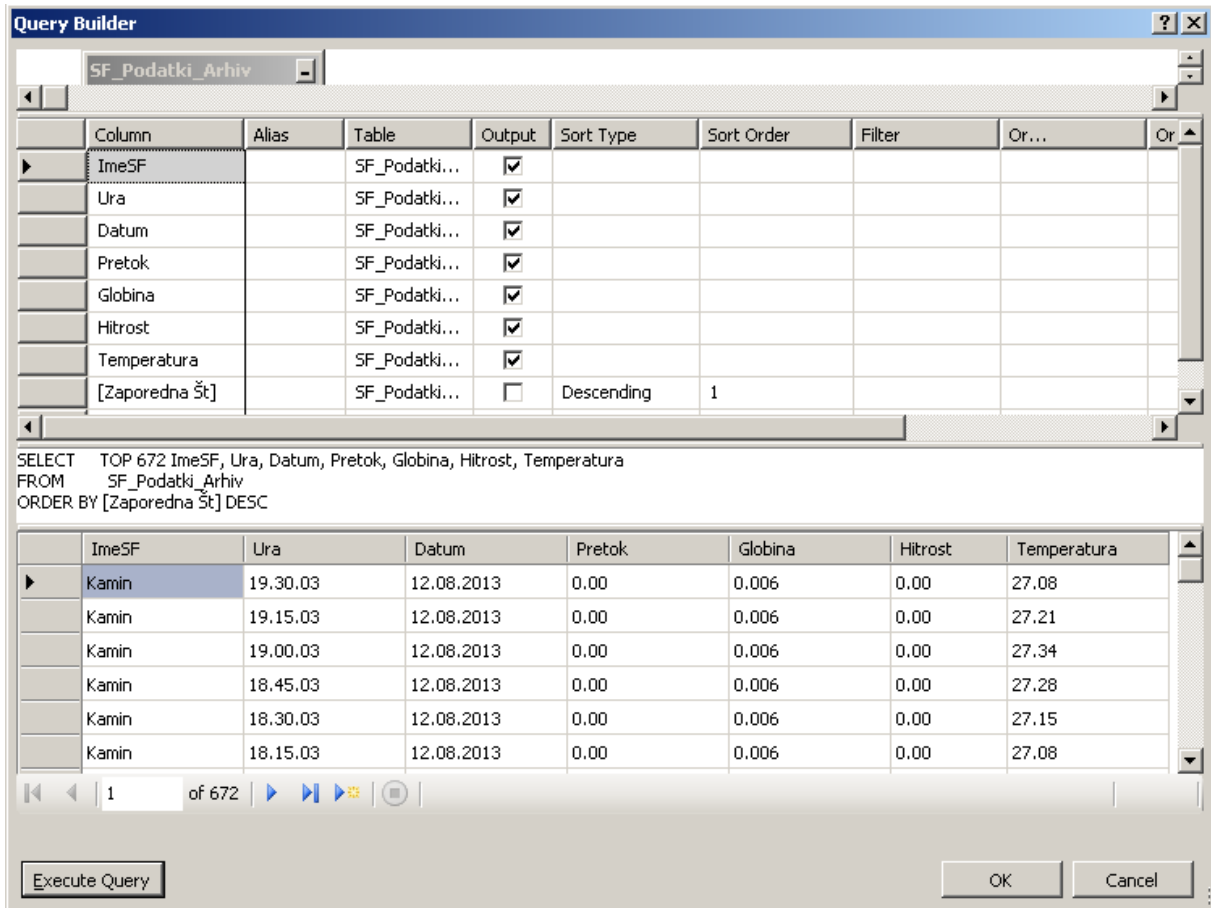
- končni uporabniki s pomočjo SQL jezika navajajo, katere podatke zahtevajo
- SQL ukazi se izvršujejo s pomočjo programa za upravljanje baze podatkov (Database Manager)
- DBM predstavi uporabniku rezultat v obliki tabele, ki se imenuje ciljna tabela
- vse zahteve, izražene s SQL jezikom se lahko nanašajo na več vrstic hkrati (<http://www.squidoo.com/sql-osnove>)

Najbolj pogosta operacija v SQL-u je poizvedba, ki se izvrši s SELECT stavkom. SELECT stavek vrne podatke iz ene ali več tabel. Standardni SELECT stavek nima nobenega vpliva na podatke v podatkovni bazi in jih ne spreminja. Optimizacija SQL stavkov lahko znatno pohitri izvajanje poizvedb.

Struktura SELECT stavka: (<http://sl.wikipedia.org/wiki/SQL>)

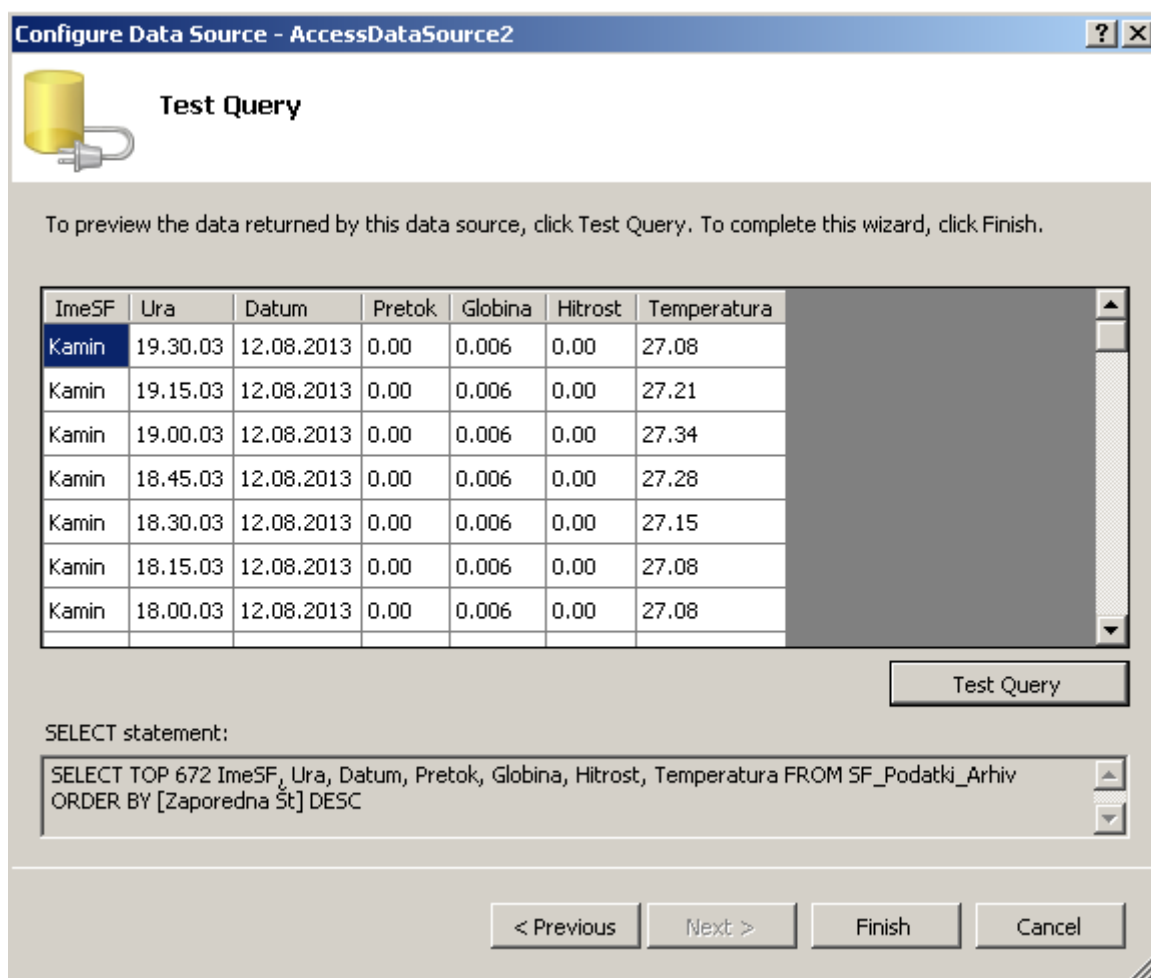
- SELECT stavku lahko sledi znak *, ki pomeni, da se bodo izpisala vsa polja iz tabele oziroma tabel v FROM stavku. Namesto zvezdice lahko napišemo imena polj.
- FROM stavku sledijo imena tabel, iz katerih bomo vzeli podatke.
- WHERE stavek vsebuje pogoje, ki jih bo upoštevala poizvedba.
- GROUP BY projecira vrstice z istimi vrednostmi v manjšo množico vrstic.
- HAVING stavek filtrira vrstice.
- ORDER BY stavek sortira rezultat poizvedbe. Stavku ORDER BY lahko sledi eno ali več polj. Sortiranje je lahko naraščajoče (ASC) ali padajoče (DESC).
- TOP stavku sledi število vrstic, ki jih poizvedba prikaže.

Primer SQL poizvedbe, ki prikaže meritve zadnjih 7 dni:
(zadnjih 672 meritev, meritve se izvajajo v intervalu 15 minut):



Slika 27: Primer SQL poizvedbe

S stavkom SELECT izberemo kategorije podatkov, katere želimo prikazati v tabeli in njihovo zaporedje. V našem primeru so to: ime avp, ura, datum, pretok, globina, hitrost in temperatura. Te podatke bomo kasneje prikazali v tabeli ali grafu. Podatke vzamemo iz tabele z imenom SF_Podatki_Arhiv. Podatke razvrstimo v obratnem vrstnem redu s stavkom ORDER BY, ki mu sledi ime stolpca po katerem sortiramo, ter vrsta sortiranja. To nam omogoča, da s stavkom TOP 672 preberemo zadnjih 672 podatkov, saj se najnovejši podatki v Access bazo zapisujejo na konec tabele. Pravilnost SQL poizvedbe preverimo z ukazom Execute Query.



Slika 28: Končni prikaz izbranih podatkov iz podatkovne baze

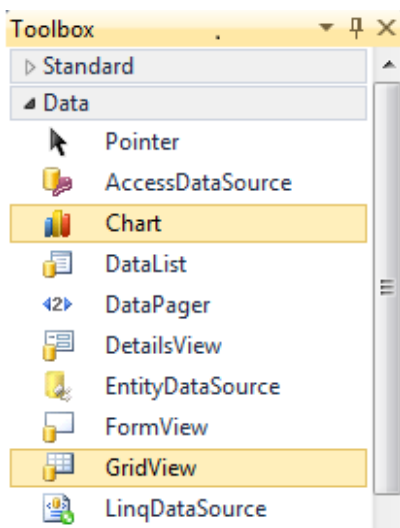
Izbrane podatke nato prikažemo v preglednicah, grafih...

6.2.3 Vstavljanje preglednic in grafov

Preglednica je podatkovni tip, ki ga uporabimo, kadar želimo skupaj (pod istim imenom) hraniti ali prikazati več podatkov enakega tipa (Zaveršnik, Tabela...).

Graf je prikaz številskih podatkov s sliko. Med najbolj običajnimi grafikoni so tortni, palični in črtni grafikon, vsak ima svoje prednosti in slabosti, cilj vseh pa je vizualizacija podatkov, ki si jih drugače težko predstavljamo (About.com...).

Preglednico in graf vnesemo s pomočjo GridView in Chart v zavihku data, ki se nahaja v oknu Toolbox. Toolbox lahko odpremo tudi s kombinacijo tipk Ctrl+Atl+X.



Slika 29: Izbira preglednice ali grafa v oknu Toolbox

Pri preglednicah in grafih moramo izbrati vir podatkov (Choose Data Source), ki smo jih predhodno uredili.

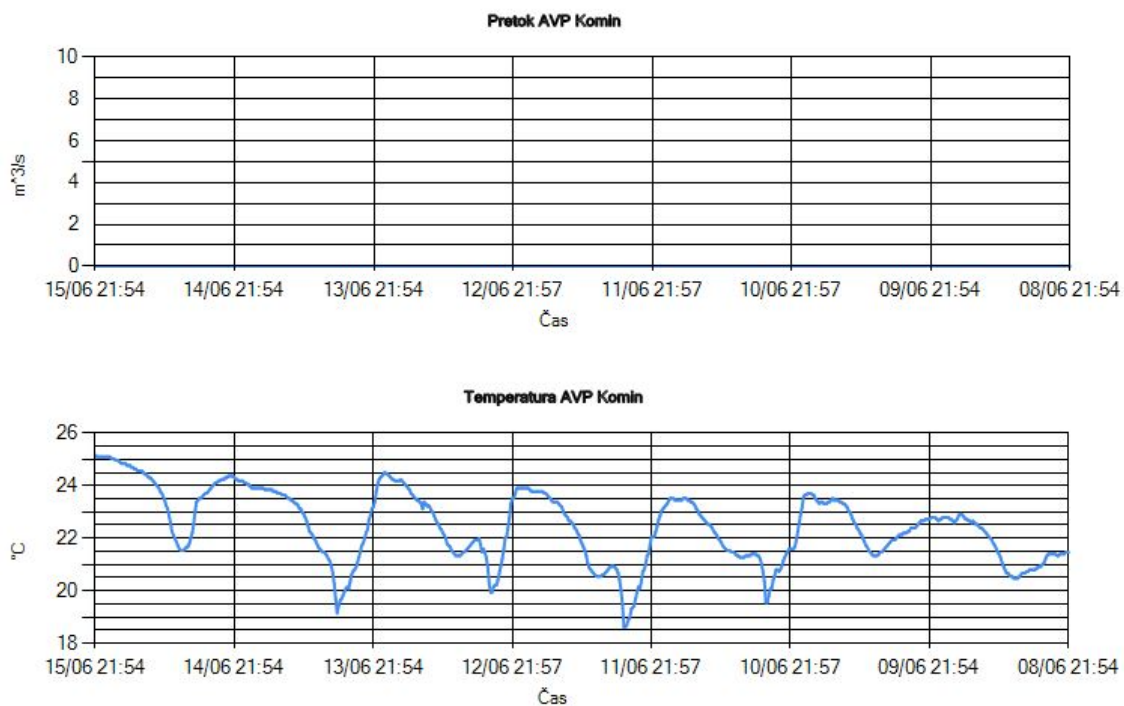
Preglednice in grafi ponujajo bogato paletu nastavitvev. Nastavljamo lahko na primer: videz, barve, vrsto pisave, velikost, sortiranje, položaj na spletni strani...

Pravilnost izpisa spletne strani preverimo s funkcijo Debug. Debug zaženemo s tipko F5.

Ime AVP	Datum	Ura	Pretok [m ³ /s]	Globina [m]	Hitrost [m/s]	Temperatura [°C]
Kamin	12.08.2013	19.30.03	0.00	0.006	0.00	27.08
Kamin	12.08.2013	19.15.03	0.00	0.006	0.00	27.21
Kamin	12.08.2013	19.00.03	0.00	0.006	0.00	27.34
Kamin	12.08.2013	18.45.03	0.00	0.006	0.00	27.28
Kamin	12.08.2013	18.30.03	0.00	0.006	0.00	27.15
Kamin	12.08.2013	18.15.03	0.00	0.006	0.00	27.08
Kamin	12.08.2013	18.00.03	0.00	0.006	0.00	27.08
Kamin	12.08.2013	17.45.03	0.00	0.006	0.00	27.08
Kamin	12.08.2013	17.30.03	0.00	0.006	0.00	27.08
Kamin	12.08.2013	17.15.03	0.00	0.006	0.00	27.01
Kamin	12.08.2013	17.00.03	0.00	0.006	0.00	26.95
Kamin	12.08.2013	16.45.03	0.00	0.006	0.00	26.88
Kamin	12.08.2013	16.30.03	0.00	0.006	0.00	26.82
Kamin	12.08.2013	16.15.03	0.00	0.006	0.00	26.75
Kamin	12.08.2013	16.00.03	0.00	0.006	0.00	26.68

1 2 3 4 5 6 7 8 9 10 ...

Slika 30: Primer prikaza podatkov v preglednici



Slika 31: Primer grafičnega prikaza temperature in pretoka

V naslednjem poglavju je prikazana namestitev in izdelava spletnega mesta z uporabo CMS Joomla.

6.3 Izdelava spletnega mesta s CMS Joomla

Joomla je odprtokodni sistem za urejanje spletnih vsebin (Content Management Systems), ki omogoča vzdrževanje in urejanje vsebine spletnih strani brez poznavanja programskih jezikov. Urednik spletne strani tako lahko samostojno spreminja besedila, slike in druge elemente spletne strani (preko spletnega vmesnika) brez pomoči podjetja ali osebe, ki je stran izdelalo (Sistem..., 2013).

Poznamo komercialne in odprtokodne sisteme za upravljanje spletnih vsebin. V zadnjih letih so drage sisteme v veliki meri izpodrinile odprtokodne rešitve sistemov za upravljanje vsebin. Eden od najbolj razširjenih odprtokodnih CMS-jov je Joomla, ki ponuja številne razširitve. Na voljo imamo nekaj tisoč razširitev, ki prinašajo številne dodatne funkcionalnosti. Pri teh razširitvah največjo težavo predstavlja izbira prave razširitve, ki mora biti funkcionalna in preprosta za uporabo (Šalamun, 2012). Z Joomla je možno izdelati zelo napredno in s pomočjo razširitev uporabniku prijazno, sodobno spletno mesto.

Sistem Joomla je sestavljen iz osnove (ospredje in skrbništvo), komponent, modulov, vtičnikov in predlog ter jezikovnih paketov. Gre za modularni sistem, kar pomeni, da z dodajanjem razširitev (komponente, moduli, vtičniki) dodajamo nove funkcionalnosti. Zato se Joomla uporablja za različne namene. Na spletišču lahko med drugim omogočimo (slojoomla.si..., 2013):

- predstavitev spletnih večpredstavnih gradiv,
- predstavitvene strani,
- galerijo slik,
- spletni forum in klepetalnico,
- spletne dnevnik (bloge),
- skladišča različnih vsebin,
- spletne koledarje,
- spletne trgovine,
- objavljanje aktualnih novic in spletnih anket

Končno funkcionalnost spletnim stranem dajejo komponente, moduli, vtičniki in predloge.

Pri izbiri razširitev moramo biti pazljivi, in izbrati zaupanja vredne (plačljive, dobro ocenjene...) saj nekatere lahko vsebujejo škodljivo programsko kodo.

Komponente:

Komponente so najbolj bistven del Joomla. Komponente so tisto, kar v bistvu vidimo kot osrednji del internetne strani. Sistem Joomla omogoča tudi dodatno nameščanje komponent drugih izdelovalcev. Joomla je oblikovana tako, da nalaga in izvaja natanko eno komponento za vsako podstran. Gre za največje, kompleksne, zapletene in najzmogljivejše razširitve, ki lahko vključujejo tudi module in vtičnike.

Moduli:

Za poživitev in funkcionalnost spletnih strani poskrbijo številni moduli, ki so manjši funkcionalni bloki. Moduli so prikazani v stranskih vrsticah na levi in desni strani spletne strani. To so manjši bloki vsebin, ki so lahko prikazani na vseh položajih, ki jih podpira predloga. Med module sodijo npr. prijavní modul, modul za iskanje, reklamni modul, modul z meniji... Stran lahko vsebuje poljubno število modulov.

Vtičniki:

Številne popestritve izvajajo tudi vtičniki, ki niso nikjer prikazani, saj svojo nalogo opravljajo v ozadju. Pogosto se v obliki preproste sintakse {xxxx|xxxx|...} vstavljajo v vsebine. Vtičnike najlaže opišemo, če si jih predstavljamo kot dele kode, ki delujejo na celotni spletni strani, skozi komponente in module.

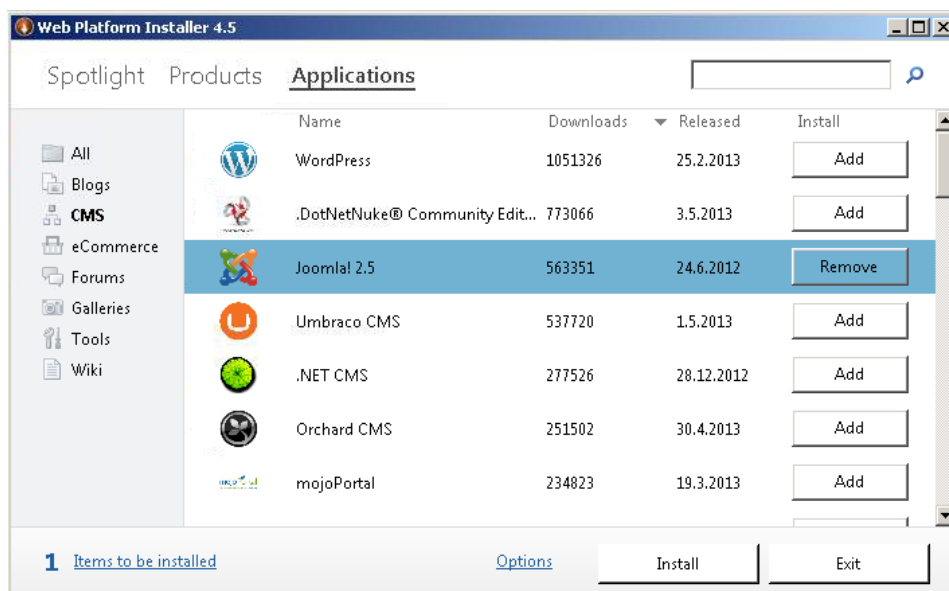
Predloge:

Predloga je vrsta razširitve, ki določa grafično podobo, obliko, položaje na strani, stolpce, menije, barvo in velikost pisave. V njej so tudi določena mesta za prikaz modulov in komponent.

6.3.1 Namestitev CMS Joomla:

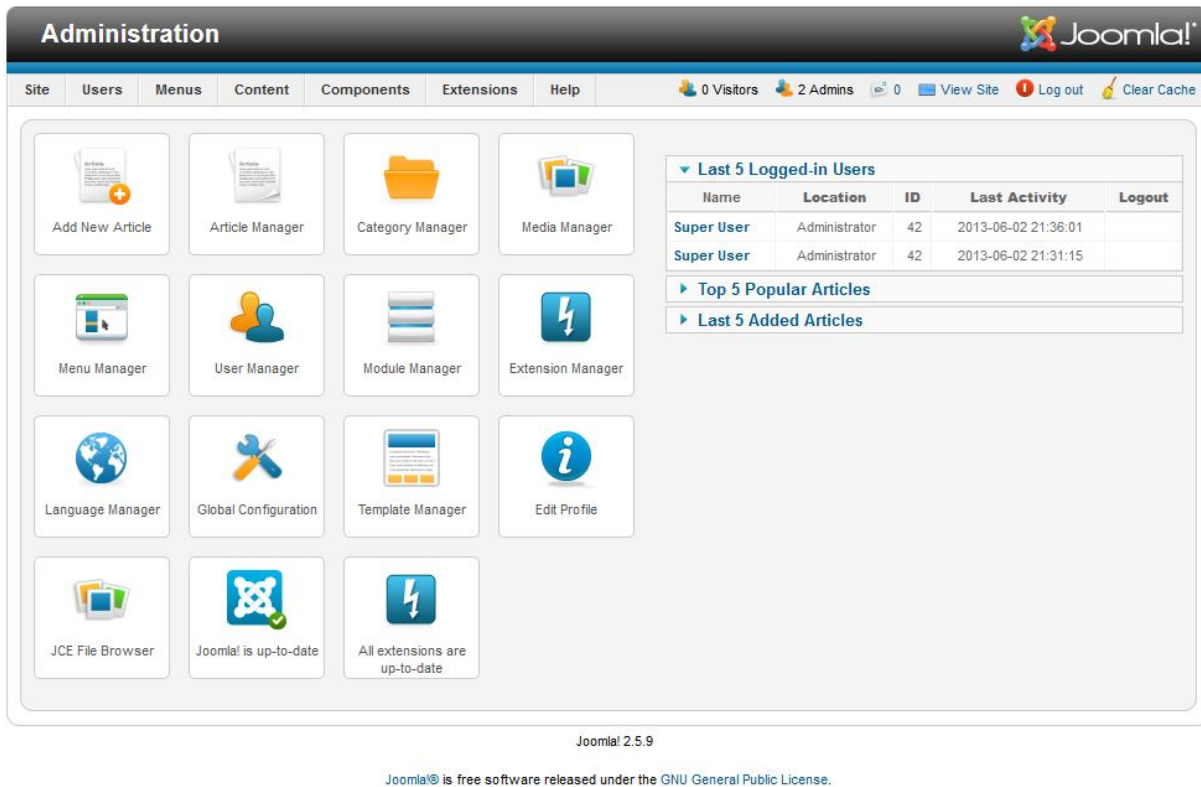
Prikazana je namestitev Joomla na lasten strežnik. V kolikor spletno stran gostujemo pri ponudniku gostovanja se postopek namestitve razlikuje.

CMS Joomla najlaže namestimo s pomočjo programa Microsoft Web Platform Installer, ki je dosegljiv na naslovu: <http://www.microsoft.com/web/downloads/platform.aspx>.



Slika 32: Namestitev CMS Joomla

Po namestitvi se v CMS Joomla prijavimo na naslovu »<http://localhost/applicationname/administrator>«. Applicationname je ime naše aplikacije, ki smo ga vnesli pri namestitvi.



Slika 33: Skrbniška stran CMS Joomla

6.3.2 Izdelava spletnega mesta, ki prikazuje meritve samodejnih vodomernih postaj

Izdelali bomo spletno mesto, na katerem so prikazane meritve samodejnih vodomernih postaj na Ljubljani. Podatki bodo na strani prikazani v grafični in tabelarni obliki. Na spletnem mestu si lahko ogledamo podatke o pretoku, gladini in temperaturi samodejnih vodomernih postaj Kamin, Lipe in Ižanska. Med vodomernimi postajami izbiramo z uporabo vertikalnega menija.

Najprej izberemo ustrezno predlogo za spletno mesto. Predloga določa grafično podobo, obliko, položaje na strani, stolpce, menije, barvo in velikost pisave. Katero predlogo bomo izbrali je tako odvisno od namena spletne strani.

Za izdelavo spletnega mesta smo izbrali podlogo Black & White, ki je dosegljiva brezplačno na naslovu: <http://www.icetheme.com/blog/joomla-templates/free-joomla-template-black-white-for-download>. Zaradi velike prilagodljivosti in dobre podpore v brskalnikih smo za vertikalni meni uporabili modul ARI Ext Menu v.2.2.0, ki je dostopen na naslovu: <http://extensions.joomla.org/extensions/structure-a-navigation/menu-systems/drop-a-tab->

[menus/11710](#). Horizontalni meni smo izdelali z uporabo modula IceMegaMenu, ki je vključen v podlogo Black & White, lahko pa ga prenesemo tudi na naslovu:

<http://extensions.joomla.org/extensions/structure-a-navigation/menu-systems/drop-a-tab-menus/16468>

Zavihek hidrološki podatki predstavlja osnovni del spletnega mesta. V njem se nahaja satelitska slika, kjer so označene vodomerne postaje. Podatke si ogledamo s klikom na vodomerno postajo.

Pri izdelavi smo uporabil naslednjo html kodo:

```
<html>
<body>

<map name="Ljubmap">
<area title="AVP Kamin" shape="circle" coords="62,529,15" href="index.php?
option=com_wrapper&view=wrapper&Itemid=112/>
<area title="AVP Lipe" shape="circle" coords="444,335,15" href="index.php?
option=com_wrapper&view=wrapper&Itemid=113/>
<area title="AVP Ižanska" shape="circle" coords="590,230,15" href="index.php?
option=com_wrapper&view=wrapper&Itemid=114/>
</map>
</body>
</html>
```

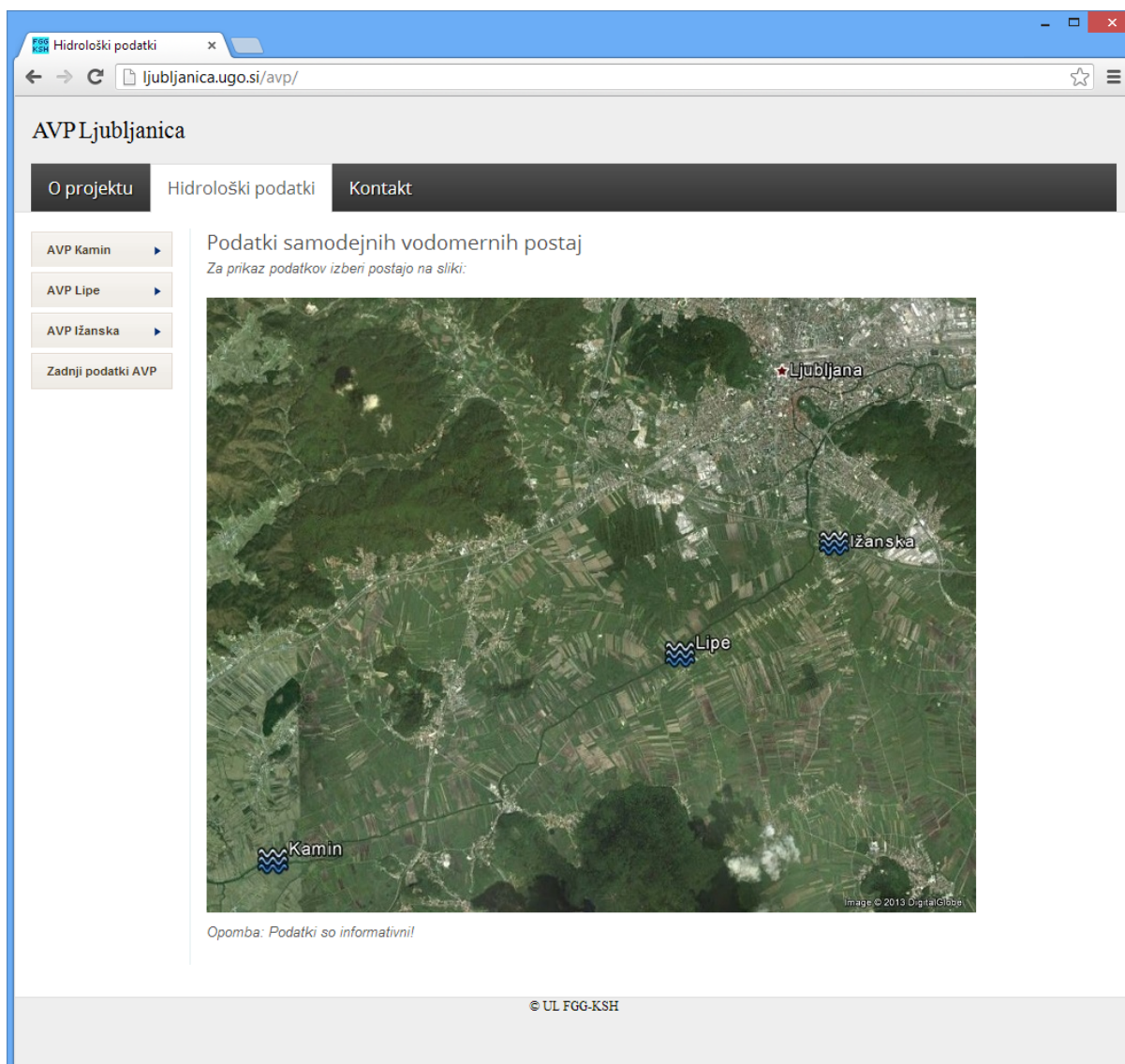
Razlaga HTML kode:

Sliko smo v dokument vnesli z ukazom . Slika je široka 723px in visoka 578px.

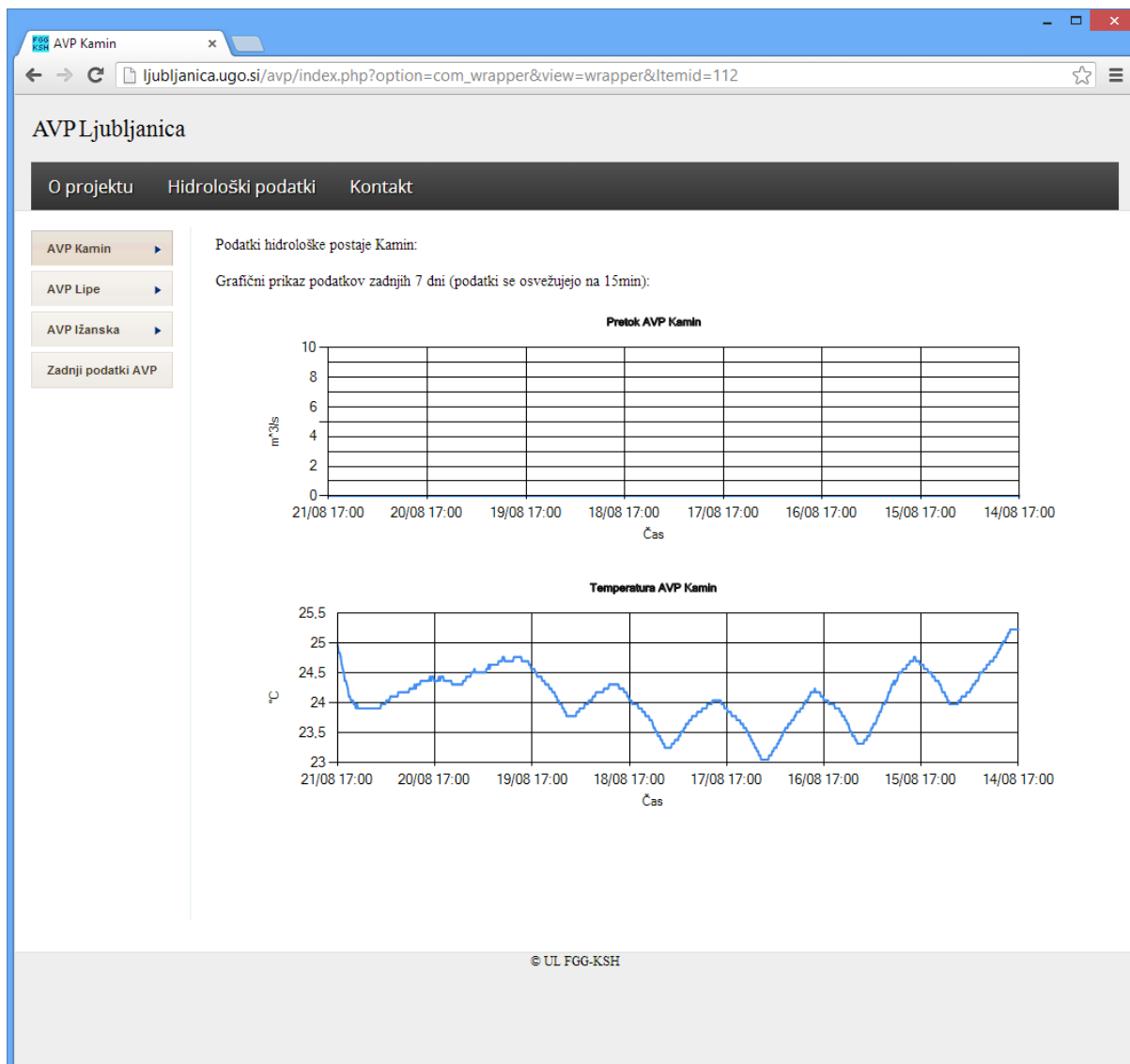
Značka <usemap> določa območja, ki služijo kot povezave.

Shape="circle" coords="62,529,15", »62,529,15« določa obliko kroga (shape – oblika; circle – krog; coords – koordinate). Kot koordinate moramo določiti dve točki, ki določita center kroga x in y (62 – odmik od levega roba slike; 529 – odmik od zgornjega roba slike) in pa polmer kroga (polmer = 15). Href nam pove URL dokumenta, ki ga želimo povezati. V našem primeru podamo povezavo do meritev izbrane vodomerne postaje. Postopek ponovimo še za drugi vodomerni postaji Lipe in Ižanska.

6.3.3 Končni izgled spletnega mesta



Slika 34: Satelitska slika območja z označenimi vodomernimi postajami



Slika 35: Grafični prikaz podatkov zadnjih 7 dni

AVP Ljubljana

O projektu Hidrološki podatki Kontakt

AVP Kamin

AVP Lipe

AVP Ižanska

Zadnji podatki AVP

Podatki hidrološke postaje Kamin:

Tabelarni prikaz podatkov zadnjih 7 dni (podatki se osvežujejo na 15min):

Ime AVP	Datum	Ura	Pretok [m ³ /s]	Globina [m]	Hitrost [m/s]	Temperatura [°C]
Kamin	12.08.2013	19.30.03	0.00	0.006	0.00	27.08
Kamin	12.08.2013	19.15.03	0.00	0.006	0.00	27.21
Kamin	12.08.2013	19.00.03	0.00	0.006	0.00	27.34
Kamin	12.08.2013	18.45.03	0.00	0.006	0.00	27.28
Kamin	12.08.2013	18.30.03	0.00	0.006	0.00	27.15
Kamin	12.08.2013	18.15.03	0.00	0.006	0.00	27.08
Kamin	12.08.2013	18.00.03	0.00	0.006	0.00	27.08
Kamin	12.08.2013	17.45.03	0.00	0.006	0.00	27.08
Kamin	12.08.2013	17.30.03	0.00	0.006	0.00	27.08
Kamin	12.08.2013	17.15.03	0.00	0.006	0.00	27.01
Kamin	12.08.2013	17.00.03	0.00	0.006	0.00	26.95
Kamin	12.08.2013	16.45.03	0.00	0.006	0.00	26.88
Kamin	12.08.2013	16.30.03	0.00	0.006	0.00	26.82
Kamin	12.08.2013	16.15.03	0.00	0.006	0.00	26.75
Kamin	12.08.2013	16.00.03	0.00	0.006	0.00	26.68

1 2 3 4 5 6 7 8 9 10 ...

© UL FGG-KSH

Slika 36: Tabelarni prikaz podatkov zadnjih 7 dni

7 ZAKLJUČEK

V diplomski nalogi je opisan postopek, s katerim prikažemo meritve vodomernih postaj na Ljubljani na spletu v realnem času. Opisan je merilec Starflow, princip delovanja, komunikacija z računalnikom (STARLOG protokol) in možnosti povezave z njim na daljavo. Razložena je komunikacija z računalnikom preko serijskega vmesnika. V programskem okolju Matlab so razloženi ukazi, s katerimi od merilca Starflow pridobimo podatke o meritvah. Meritve nato obdelamo in izvozimo v podatkovno bazo Microsoft Access. V nadaljevanju je opisano programsko okolje Visual Studio, kjer je na primeru prikazan način izdelave spletne strani, ki vsebuje meritve. V zadnjem poglavju je opisan CMS Joomla, s katerim naredimo spletno mesto, ki vsebuje spletne strani z meritvami. Do spletne strani dostopamo po medmrežju.

Novejši načini nadzora voda strmiijo k neprekinjenemu zbiranju podatkov z instrumenti na terenu. Do teh podatkov lahko dostopamo s prenosom podatkov na kraju merjenja, ali pa na daljavo. Dostop do podatkov na daljavo v realnem času ima več prednosti v primerjavi s tradicionalnim načinom merjenja. Ta način lahko zmanjša prostor za človeške napake in zamude. Zmanjšajo se tudi stroški zbiranja podatkov, znatno se poveča količina in kakovost podatkov. Daljinsko pridobljeni podatki v realnem času zagotavljajo pomembno zgodnje opozarjanje ter zaznavanje sprememb in trendov na podlagi katerih sprejemamo odločitve (O'Flynn et al., 2010).

Zaznavanje v realnem času v vodotokih omogoča potencialno močno analitično orodje za zagotavljanje varnosti pred poplavami. Poplave postajajo eden glavnih problemov v večini držav. Poplava je naravni pojav, ki nastane zaradi izredno močnih padavin ali naglega taljenja snega ali medsebojnega skupnega delovanja. Kljub temu, da smo sposobni dokaj dobro napovedati količino dežja, ki bo padla na neko območje, oziroma spremljati pot nevihtnih oblakov s sateliti, je potreba po podatkih, ki so dostopni v realnem času velika (Sunkpho in Ootamakorn, 2011). Podatki, kot so na primer pretok, količina padavin, gladina vode so bistveni, da lahko izdamo opozorila in s tem zmanjšamo škodo, ki jo lahko povzroči visoka voda.

Rečni režim Ljubljane uravnavamo z zapornicama na Ambroževem trgu in Grubarjevem prekopu. Na podlagi podatkov, pridobljenih iz samodejnih vodomernih postaj lahko bolj natančno uravnavamo zapornici in tako izboljšamo rečni režim. Lahko bi razvili programsko opremo, s katero bi na podlagi podatkov iz samodejnih vodomernih postaj na Ljubljani, samodejno uravnavali zapornici na Ambroževem trgu in Grubarjevem prekopu.

VIRI

About.com. Definicija grafa. 2012.

http://spreadsheets.about.com/od/c/g/chart_def.htm (Pridobljeno 5. 11. 2012.)

Blažič, L. 2000. Uporaba in programiranje serijskih vrat PC, Nova Gorica.

<http://ro.zrsss.si/projekti/elektrotehnika/serprog/> (Pridobljeno 24. 10. 2012.)

Cerar, U., Banovec, P., Steinman, F. 2000. Hidravlična ustreznost merilnih mest za merjenje pretokov, Acta hydrotechnica 18/28, Ljubljana.

<ftp://ksh.fgg.uni-lj.si/acta/a28-uc.pdf> (Pridobljeno 12. 6. 2013.)

Dolšina, P. 2012. Razširjenost in zastopanost tujerodnih invazivnih rastlinskih vrst v obrežnem pasu reke Ljubljanice. Ljubljana, Univerza v Ljubljani, Biotehnična fakulteta: 31-35.

http://www.digitalna-knjiznica.bf.uni-lj.si/dn_dolsina_primoz.pdf (Pridobljeno 5. 6. 2013.)

Frye, C. 2007. Microsoft Office Access 2007 hitro in jasno [prevod Iolar]. Ljubljana, Pasadena: 19-84.

Functionx.com. Uvod v Microsoft Access. 2012.

<http://www.functionx.com/access/Lesson01.htm> (Pridobljeno 22. 10. 2012.)

Glasilo projekta LIFE10NAT/SI/142: Obnovitev koridorja Ljubljanice in izboljšanje rečnega vodnega režima, Letnik 1, Številka 1, Ljubljana, Univerza v Ljubljani, Fakulteta za gradbeništvo in geodezijo.

2013. http://ksh.fgg.uni-lj.si/ljubljanicaconnects/Data/Bulletin_1.pdf (Pridobljeno 20. 5. 2013.)

Graf, H., 2012. Joomla! 2.5 Beginner`s Guide!. Cocote, Francija: 241-268.

Koler, B., Urbančič, T., Vidmar, A., Globevnik, L. 2012 Analiza višin poplavne vode v Ljubljani in na Ljubljanskem barju. Ljubljana. Geodetski vestnik 56/4: 15str.

http://geodetski-vestnik.com/56/4/gv56-4_846-860.pdf (Pridobljeno 27. 6. 2013.)

Komunikacija preko serijske zveze v Matlabu. Matlab spletni priročnik. 2012.

http://www.mathworks.com/help/matlab/matlab_external/getting-started-with-serial-i-o.html

(Pridobljeno 26. 10. 2012.)

Kranjčič, D. 2005. Programsko pošiljanje sporočil SMS, Revija Monitor, Julij 2004, Ljubljana.

<http://www.monitor.si/clanek/programsko-posiljanje-sporocil-sms/121451/?xURL=301>

(Pridobljeno 9. 6. 2013.)

Ljubljana povezuje, HBV model za porečje Ljubljanice. 2013. Ljubljana, Univerza v Ljubljani, Fakulteta za gradbeništvo in geodezijo.

<http://ksh.fgg.uni-lj.si/LjubljanaConnects/Data/Hydro/Hydrologic%20HBV%20model%20of%20Ljubljana%20River%20Basin.pdf> (Pridobljeno 27. 6. 2013.)

Ljubljansko barje je območje Natura 2000. 2013.

<http://www.ljubljanskobarje.si/ljubljansko-barje/ljubljansko-barje-je-obmocje-natura-2000>
(Pridobljeno 3. 6. 2013.)

Made in China. WiFi Module Serial (RS232) to WiFi (802.11 b/g). 2013.

<http://www.made-in-china.com/showroom/usr2012/product-detailzqMmjuNAhKRt/China-WiFi-Module-Serial-RS232-to-WiFi-802-11-b-g-USR-WiFi232-mod-II-.html> (Pridobljeno 9. 6. 2013.)

Mayo, J. 2010. Microsoft Visual Studio 2010: A Beginner's Guide. McGraw-Hill Osborne Media: 1-34, 181-214.

Matlab. Wikipedia, prosta enciklopedija. 2012.

<http://sl.wikipedia.org/wiki/MATLAB> (Pridobljeno 24. 10. 2012.)

Matlab spletni priročnik. Ukaz hex2dec. 2012.

<http://www.mathworks.com/help/matlab/ref/hex2dec.html> (Pridobljeno 27. 10. 2012.)

Matlab spletni priročnik. Ukaz xlswrite. 2012.

<http://www.mathworks.com/help/matlab/ref/xlswrite.html> (Pridobljeno 27. 10. 2012.)

Matlab spletni priročnik. Ukaz database. 2012.

<http://www.mathworks.com/help/database/ug/database.html> (Pridobljeno 28. 10. 2012.)

Natura 2000. Kaj je Natura 2000. 2013.

<http://www.natura2000.gov.si/index.php?id=44> (Pridobljeno 3. 6. 2013.)

O'Flynn, B., Regan, F., Lawlor, A., Wallace, J., Torres, J., O'Mathuna, C. 2010. Experiences and recommendations in deploying a real-time, water quality monitoring system, Measurement Science and Technology: 1-10.

Perko, D., Orožen Adamič, M. 1999. Slovenija pokrajine in ljudje. Ljubljana, Mladinska knjiga: 382 – 384.

Petrišič, J. 2012. Uvod v MATLAB: za inženirje. Ljubljana, Univerza v Ljubljani, Fakulteta za strojništvo: 231 str.

Project PDA. Bluetooth serial port. 2012.

<http://www.projectpda.com/knowledge-base/b/bluetooth-serial-port.html> (Pridobljeno 27. 10. 2012.)

Pogojni stavki. If-else zanka. 2012.

http://www.ro.feri.unimb.si/predmeti/navodila/Matlab_nav/programiranje/pogojni_stavki1.htm

(Pridobljeno 29. 10. 2012.)

Radinović, S. 2006. Razvoj uporabniške rešitve za podporo pedagoškemu delu na osnovni šoli. Diplomsko delo. Kranj, Univerza v Mariboru, Fakulteta za organizacijske vede: 12-13.

Rojc, B. 2005. Primer uporabe telemetrijskih modemov, Ljubljana, Svet elektronike.

svet-el.si/revija/login.php?Action=READ&ID=540&state=0 (Pridobljeno 23. 5. 2013.)

Sunkpho, J., Ootamakorn, C. 2011. Real-time flood monitoring and warning system, Thailand, Songklanakarinn Journal of Science and Technology: 1, 2.

Sečnik, M. 2012. Prikaz meritev ultrazvočnega merilca Starflow na spletu v realnem času. Seminarska naloga, Ljubljana, Univerza v Ljubljani, Fakulteta za gradbeništvo in geodezijo.

Serijski vmesnik. Univerza v Ljubljani, Fakulteta za računalništvo in informatiko. 2012.

http://colos1.fri.uni-lj.si/eri/rac_sistemi_omrezja/html/RSO-OKOLJE/Serijski_vmesnik.html

(Pridobljeno 24. 10. 2012.)

Sistem za upravljanje vsebin. Wikipedia, prosta enciklopedija. 2013.

http://sl.wikipedia.org/wiki/Sistem_za_upravljanje_vsebin (Pridobljeno 27. 5. 2013.)

Slo Joomla. Kaj je Slojoomla CMS. 2013.

http://www.slojoomla.si/index.php?option=com_content&view=article&id=148&Itemid=114

(Pridobljeno 29. 5. 2013.)

Software for education. Slika serijskega protokola. 2012.

<http://www.softwareforeducation.com/wikileki/images/a/a0/Serial-Start-Stop-R.gif>

(Pridobljeno 24. 10. 2012.)

Spletna stran. Wikipedia, prosta enciklopedija. 2012.

http://sl.wikipedia.org/wiki/Spletna_stran (Pridobljeno 12. 11. 2012.)

SQL. Wikipedia, prosta enciklopedija. 2012.

<http://sl.wikipedia.org/wiki/SQL> (Pridobljeno 30. 10. 2012.)

Squidoo. SQL Osnove. 2012.

<http://www.squidoo.com/sql-osnove> (Pridobljeno 30. 10. 2012.)

Šalamun, Z. 2012. Analiza obstoječih razširitev v sistemu Joomla, Diplomsko delo, Maribor, Univerza v Mariboru, Fakulteta za elektrotehniko, računalništvo in informatiko.

<http://dkum.uni-mb.si/IzpisGradiva.php?id=37872> (Pridobljeno 27. 5. 2013.)

Šupek, M. 2008. Ekstrapolacija pretočne krivulje vodomerne postaje s hidravličnim modelom HEC-RAS. Diplomaska naloga, Ljubljana, Univerza v Ljubljani, Fakulteta za gradbeništvo in geodezijo.

http://drugg.fgg.uni-lj.si/263/1/GRU_3001_Supek.pdf (Pridobljeno 27. 5. 2013.)

Telit Application Note. Radijski modem Power One 868MHz. 2013.

<http://www.telit.com/module/infopool/download.php?id=2491> (Pridobljeno 28. 5. 2013.)

Telit User Guide. Radijski modem Power One 868MHz. 2013.

<http://www.telit.com/module/infopool/download.php?id=1937> (Pridobljeno 28. 5. 2013.)

Unidata, 2007. Unidata Manual - 6526 Starflow User Manual Issue 2.0: 130 str.

Unidata, 2011. Unidata Manual - 6526 Starflow User Manual Issue 3.4.doc: 136 str.

http://www.unidata.com.au/support/manuals/suite/6526_starflow.pdf (Pridobljeno 17. 10. 2012.)

Unidata. Ultrasonic Doppler Instrument. 2012.

<http://www.unidata.com.au/products/starflow/6526/index.html> (Pridobljeno 22. 10. 2012.)

Uprava RS za zaščito in reševanje, 2005. Načrt zaščite in reševanja ob poplavah: 4-7.

<http://www.sos112.si/slo/tdocs/poplava.pdf> (Pridobljeno 20. 11. 2012.)

Uredba o Krajinskem parku Ljubljansko barje, 2008. Uradni list Republike Slovenije

<http://www.uradni-list.si/1/content?id=89429> (Pridobljeno 3. 6. 2013.)

Varstvena območja krajinskega parka Ljubljansko barje. 2008. Ljubljana. MOP.

Zaversnik, M. Tabela. Univerza v Ljubljani, Fakulteta za računalništvo in informatiko. 2012.

<http://zaversnik.fmf.uni-lj.si/Gradiva/Java/sklicniTipi/tabele/> (Pridobljeno 5. 11. 2012.)

Zaversnik, M. Zanka while. Univerza v Ljubljani, Fakulteta za računalništvo in informatiko. 2012.

<http://zaversnik.fmf.uni-lj.si/Gradiva/Java/zanke/while/> (Pridobljeno 29. 10. 2012.)

PRILOGA

PRILOGA A: MATLAB program za komunikacijo z merilcem Starflow

PRILOGA A: MATLAB PROGRAM ZA KOMUNIKACIJO Z MERILCEM STARFLOW

```
while 'Zanka' == 'Zanka'  
%Odziv SF (*)  
s = serial('COM2');  
s.Terminator = 'CR';  
s.TimeOut = 20;  
fopen(s);  
Odg = fscanf(s);  
fclose(s)  
delete(s)  
if size(Odg) == [0,0]  
    Odg = ('0');  
else  
end  
Odg1Char = Odg(1,1);  
KontrOdg = (*);  
if Odg1Char == KontrOdg  
    disp('ODZIV OK')  
else  
    disp('NI ODZIVA')  
end  
clear Odg
```

%Zanka v primeru nepravilnega odziva

```
while Odg1Char ~= KontrOdg  
s = serial('COM2');  
s.Terminator = 'CR';  
s.TimeOut = 20;  
fopen(s);  
Odg = fscanf(s);  
if size(Odg) == [0,0]  
    Odg = ('0');  
else  
end  
Odg1Char = Odg(1,1);  
KontrOdg = (*);  
fclose(s)  
delete(s)  
if Odg1Char == KontrOdg  
    disp('ODZIV OK')  
else  
    disp('NI ODZIVA')
```

```
end
```

```
pause(5)
```

```
end
```

```
clear
```

```
%Podatki iz SF
```

```
s = serial('COM2');
```

```
s.Terminator = 'CR';
```

```
fopen(s);
```

```
Prompt2 = fscanf(s);
```

```
fprintf(s,'G00C808')
```

```
MERITVE = fscanf(s)
```

```
fprintf(s,'G00D808')
```

```
KONTROLA = fscanf(s)
```

```
fprintf(s,'G000301')
```

```
NAPAKA = fscanf(s)
```

```
fprintf(s,'K')
```

```
fclose(s)
```

```
delete(s)
```

```
Check = length (MERITVE)+length (KONTROLA)+length (NAPAKA);
```

```
CheckStr = sprintf('%d',Check);
```

```
KontrOdg = ('49');
```

```
if CheckStr == KontrOdg
```

```
    disp('KOLIČINA PODATKOV OK')
```

```
else
```

```
    disp('NAPAČNA KOLIČINA PODATKOV')
```

```
end
```

```
%Zanka v primeru napačnih podatkov
```

```
while CheckStr ~= KontrOdg
```

```
s = serial('COM2');
```

```
s.Terminator = 'CR';
```

```
fopen(s);
```

```
Prompt2 = fscanf(s);
```

```
fprintf(s,'G00C808')
```

```
MERITVE = fscanf(s)
```

```
fprintf(s,'G00D808')
```

```
KONTROLA = fscanf(s)
```

```
fprintf(s,'G000301')
```

```
NAPAKA = fscanf(s)
```

```
fprintf(s,'K')
```

```
fclose(s)
```

```
delete(s)
```

```
Check = length (MERITVE)+length (KONTROLA)+length (NAPAKA);
```

```
CheckStr = sprintf('%d',Check);
KontrOdg = ('49');
if CheckStr == KontrOdg
    disp('KOLIČINA PODATKOV OK')
else
    disp('NAPAČNA KOLIČINA PODATKOV')
end
pause(5)
end
```

%Obdelava podatkov

%MERITVE%

%GLOBINA

```
GLOB1byteHEX = MERITVE(1,1:2); %preberem prvi byte podatkov v zapisu MERITVE
GLOB2byteHEX = MERITVE(1,3:4); %preberem drugi byte podatkov v zapisu MERITVE
GLOBHEXzdruz = strcat(GLOB2byteHEX, GLOB1byteHEX); %združim GLOB2byteDEC, GLOB1byteDEC
GLOBINAdec = hex2dec(GLOBHEXzdruz); %pretvorim iz HEX v DEC obliko
GLOBINA2 = GLOBINAdec/1000; %pretvorim iz mm v m
GLOBINA = sprintf('%0.3f',GLOBINA2) %omejim izpis podatka na tri decimalna mesta
```

%TEMPERATURA

```
TEMP1byteHEX = MERITVE(1,5:6);
TEMP2byteHEX = MERITVE(1,7:8);
TEMPHEXzdruz = strcat(TEMP2byteHEX, TEMP1byteHEX);
TEMPERATURAdec = hex2dec(TEMPHEXzdruz);
TEMPERATURA2 = TEMPERATURAdec/100;
TEMPERATURA = sprintf('%0.2f',TEMPERATURA2)
```

%BATERIJA

```
BAT1byteHEX = MERITVE(1,9:10);
BAT2byteHEX = MERITVE(1,11:12);
BATHEXzdruz = strcat(BAT2byteHEX, BAT1byteHEX);
BATERIJAdec = hex2dec(BATHEXzdruz);
BATERIJA2 = BATERIJAdec/100;
BATERIJA = sprintf('%0.2f',BATERIJA2)
```

%HITROST

```
HITR1byteHEX = MERITVE(1,13:14);
HITR2byteHEX = MERITVE(1,15:16);
HITRHEXzdruz = strcat(HITR2byteHEX, HITR1byteHEX);
HITROSTdec = hex2dec(HITRHEXzdruz);
```



```
HITROST2 = HITROSTdec/100;  
HITROST = sprintf("%.2f",HITROST2)
```

%KONTROLA%

%ECHO

```
ECHO1byteHEX = KONTROLA(1,1:2);  
ECHO2byteHEX = KONTROLA(1,3:4);  
ECHOHEXzdruz = strcat(ECHO2byteHEX, ECHO1byteHEX);  
ECHO = hex2dec(ECHOHEXzdruz);
```

%QUART

```
QUART1byteHEX = KONTROLA(1,5:6);  
QUART2byteHEX = KONTROLA(1,7:8);  
QUARTHEXzdruz = strcat(QUART2byteHEX, QUART1byteHEX);  
QUART = hex2dec(QUARTHEXzdruz);
```

%SAMPL

```
SAMPL1byteHEX = KONTROLA(1,9:10);  
SAMPL2byteHEX = KONTROLA(1,11:12);  
SAMPLHEXzdruz = strcat(SAMPL2byteHEX, SAMPL1byteHEX);  
SAMPL = hex2dec(SAMPLHEXzdruz);
```

%SIGNAL

```
SIGNAL1byteHEX = KONTROLA(1,13:14);  
SIGNAL2byteHEX = KONTROLA(1,15:16);  
SIGNALHEXzdruz = strcat(SIGNAL2byteHEX, SIGNAL1byteHEX);  
SIGNAL = hex2dec(SIGNALHEXzdruz);
```

%NAPAKA%

%NAPAKA

```
NAPAKAHEX = NAPAKA(1,1:2);  
NAPAKA =hex2dec(NAPAKAHEX);
```

```
%IZRAČUN PRETOKA: Q-H KRIVULJA (vstavi k in n)
```

```
%k =
```

```
%x = GLOBINA
```

```
%n =
```

```
PRETOKIzracun = 0; %(k*x)+n;
```

```
PRETOK = sprintf('%0.2f',PRETOKIzracun);
```

```
%URA%
```

```
URA = (datestr(now,'HH.MM.SS'));
```

```
%DATUM%
```

```
DATUM = (datestr(now,'dd.mm.yyyy'));
```

```
%DATUM&URA%
```

```
DATUMURA = (datestr(now,'dd/mm HH:MM'));
```

```
%Izvoz podatkov v Microsoft Access
```

```
dbpath = 'C:\rootWWW\AccessSFData.mdb';
```

```
conurl = ['jdbc:odbc:Driver={Microsoft Access Driver (*.mdb)}; DBQ=' dbpath];
```

```
con = database('AccessSFData.mdb','','sun.jdbc.odbc.JdbcOdbcDriver', conurl)
```

```
colnames={'ImeSF', 'SN', 'Datum', 'Ura', 'DatumInUra', 'Pretok', 'Globina', 'Hitrost', 'Temperatura', 'Baterija',  
'Echo', 'Quart', 'Sample', 'Signal', 'Error'};
```

```
values={'Kamin', '3884', DATUM, URA, DATUMURA, PRETOK, GLOBINA, HITROST, TEMPERATURA,  
BATERIJA, ECHO, QUART, SAMPL, SIGNAL, NAPAKA};
```

```
fastinsert(con, 'SF_Podatki_Arhiv',colnames, values )
```

```
disp('Program zaključen')
```

```
pause (899.25)
```

```
end
```