

VHODNO-IZHODNE NAPRAVE

Projekt – Vremenska postaja

Luka Šveigl (63200301)

Vsebina

1. UVOD	3
2. PROJEKT 1.0	4
2.1 UPORABLJENE KOMPONENTE:.....	4
2.2 POVEZOVALNA SHEMA:.....	4
3. PROJEKT 2.0 – TRENUTNA VERZIJA	6
3.1 UPORABLJENE KOMPONENTE:.....	6
3.2 VEZALNA SHEMA:	7
3.3 STREŽNIK:.....	7
3.4 POSTAJA:	8
4. NODE-RED DASHBOARD	9
5. MOŽNE IZBOLJŠAVE.....	11

1. UVOD

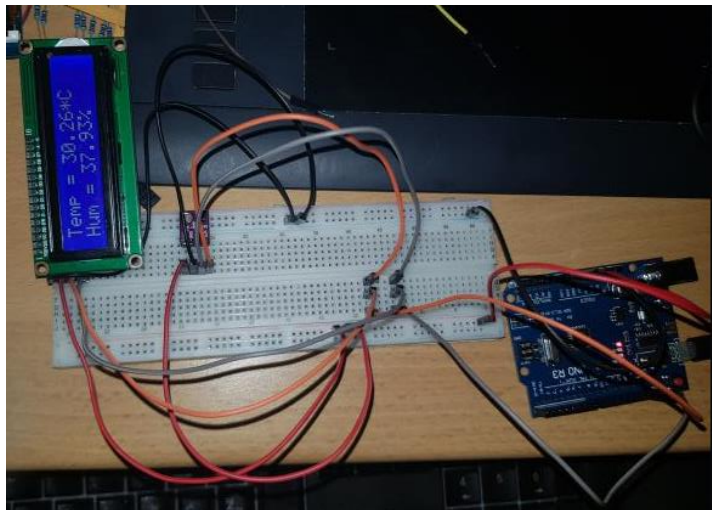
Za projekt pri predmetu Vhodno-izhodne naprave, sem se odločil, da bom naredil preprosto vremensko postajo, ki temelji na Arduino mikrokontrolniku. Najprej sem naredil implementacijo vremenske postaje z le enim Arduino UNO, a sem se odločil, da bi jo rad nadgradil, bolj specifično, želel sem ločiti upravljanje z vremenskim senzorjem in LCD zaslonom. Iz tega torej izhajata dve verziji projekta, ki bosta opisani v prihodnjih poglavjih. Pri obeh implementacijah sem se močno zanašal na komunikacijski protokol I2C.

Vsa izvorna koda, sheme ter detaljne razlage implementacij se nahajajo na naslednjih povezavah:

- Prva verzija: <https://github.com/LukaSveigl/VIN-Project-WeatherStation-deprecated>
- Druga verzija: <https://github.com/LukaSveigl/VIN-Project-WeatherStation-v2>

2. PROJEKT 1.0

Prva iteracija moje vremenske postaje je uporabljala le en Arduino UNO, ki je preko I2C protokola upravljal tako vremenski senzor kot LCD zaslon. Prav tako so podatki, ki so bili prikazani na LCD zaslonu, bili poslani na Node-RED "dashboard".



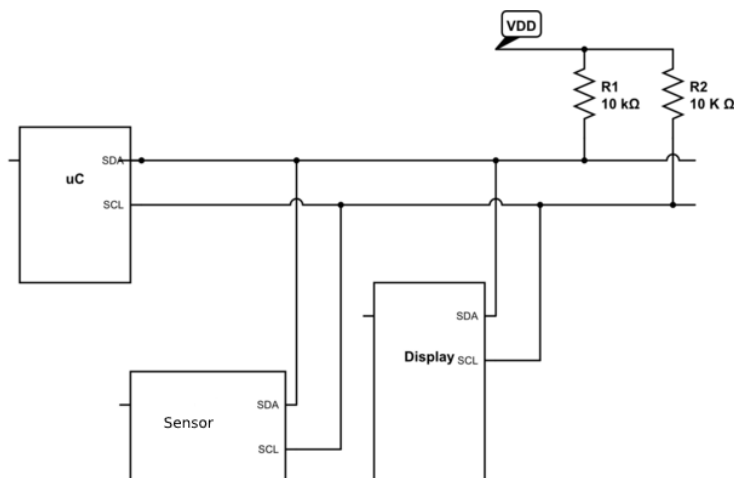
Slika 1: 1. iteracija vremenske postaje (lastni vir)

2.1 UPORABLJENE KOMPONENTE:

Ta iteracija projekta je uporabljala sledeče komponente:

- 1x Arduino UNO R3
- 1x Breadboard
- 1x LCD I2C zaslon
- 1x BME/BMP 280 (I2C naprava)
- 12x povezovalne žice

2.2 POVEZOVALNA SHEMA:



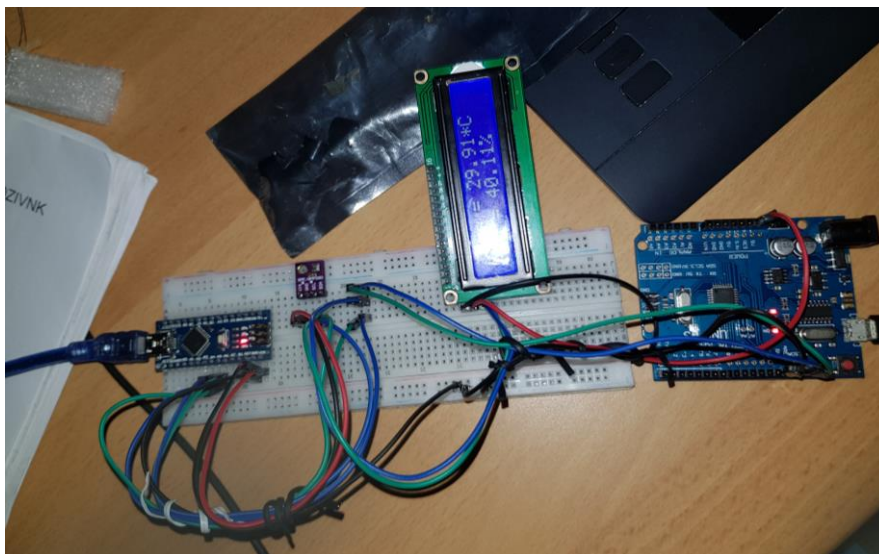
Slika 2: Povezovalna shema 1. iteracije projekta (prilagojen internetni vir)

Kot lahko vidimo na zgornji sliki, je bila prva iteracija kar se tiče vezja precej preprosta. Ker ima vsaka I2C naprava svoj naslov (ki ji ga dodeli proizvajalec, a ga lahko spremenimo z malo spajkanja), lahko naprave na Arduino SDA (podatkovni pin) in SCL (ura) povežemo paralelno. To omogoča nemoteno komunikacijo Arduina z napravami.

Če naslov naprave ni znan, ta iteracija projekta vsebuje tudi razhroščevalno funkcijo `findAddr()` – le ta se nahaja v *weather_station.ino* datoteki – ki pregleda vse možne I2C naslove, in na serijska vrata izpiše naslove naprav, ki so trenutno priključeni.

3. PROJEKT 2.0 – TRENUTNA VERZIJA

Za drugo iteracijo moje vremenske postaje sem se odločil, da bom upravljanje vremenskega senzorja in LCD zaslona ločil (idejo sem dobil iz profesorjevih predlogov na vajah). Tako sem prišel do implementacije server – station, kjer je server Arduino UNO R3, ki upravlja z LCD zaslonom in pošilja podatke na Node-RED dashboard, station pa je šibkejši Arduino NANO, ki upravlja z vremensko postajo in ob zahtevi strežnika podatke posreduje preko “request event” funkcije. Tako sem torej dobil gospodar – suženj arhitekturo, kjer sta napravi povezani z protokolom I2C.



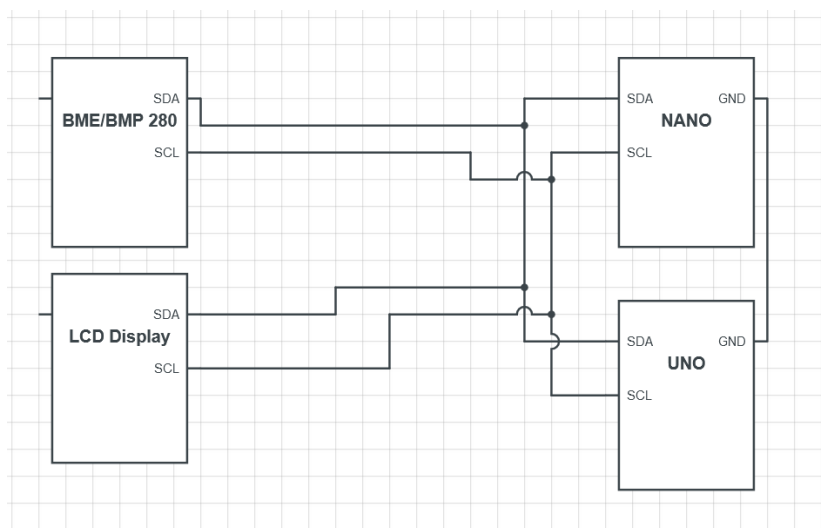
Slika 3: 2. iteracija vremenske postaje (lastni vir)

3.1 UPORABLJENE KOMPONENTE:

Ta iteracija projekta je uporabljala sledeče komponente:

- 1x Arduino UNO R3
- 1x Arduino NANO
- 1x Breadboard
- 1x LCD I2C zaslon
- 1x BME/BMP 280 (I2C naprava)
- 14x povezovalne žice

3.2 VEZALNA SHEMA:



Slika 4: Povezovalna shema 2. iteracije projekta (lastni vir)

Kot lahko vidimo na zgornji shemi, sta sedaj pina SDA in SCL na Arduino mikrokrmilnikih povezana, prav tako pa sta povezana tudi njuna GND pina. Ostali vhodno-izhodni napravi (senzor in zaslon), pa sta tako kot v prvi iteraciji na SDA/SCL povezavo vezana vzporedno.

3.3 STREŽNIK:

Strežniška naprava (Arduino UNO), v tej implementaciji opravlja 3 dela. Ta naprava zahteva podatke iz suženjske naprave, te podatke prikaže na LCD zaslonu, ter jih izpiše na serijska vrata, iz kjer jih kasneje ekstrahira Node-RED.

Strežnik deluje tako, da iz I2C naprave na naslovu 8 (arbitrarno izbran naslov), zahteva 10 bajtov podatkov. Ko podatki prispejo, jih bere bajt za bajtom, do “zero-terminala” (to je znak ‘\0’) in jih shranjuje v buffer tipa String.

Ko so vsi podatki prebrani, strežnik iz bufferja ekstrahira številčno vrednost in pa specifikator. Specifikator je znak, ki pove, katera metrika je bila trenutno prebrana (to je lahko temperatura, vlaga, zračni pritisk, približna nadmorska višina), kar je potrebno, da to vrednost shrani v pravi buffer. Ko so prebrane vse štiri metrike, strežnik podatke prikaže na LCD zaslonu (2 metrike hkrati, z 2 sekundno zakasnitvijo), jih izpiše na serijska vrata, sprazni bufferje in ponovno začne z zahtevami.

Implementacija: <https://github.com/LukaSveigl/VIN-Project-WeatherStation-v2/blob/main/server/server.ino>

3.4 POSTAJA:

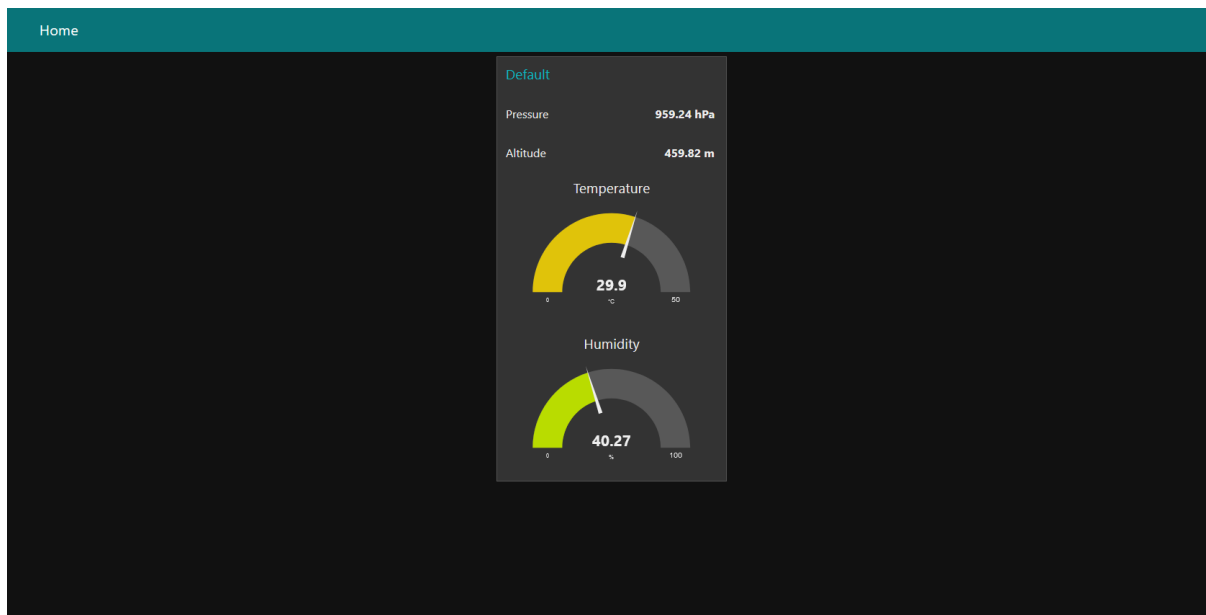
Postaja (Arduino NANO), opravlja le 2 deli. Ta naprava bere podatke iz vremenskega senzorja in jih pripravi na pošiljanje, nato pa čaka na zahtevo gospodarja, da te podatke posreduje. To stori tako, da ob zahtevi sproži funkcijo requestEvent, ki podatke trenutno v bufferju pošlje na strežnik.

Postaja deluje tako, da ob zahtevi gospodarja pošlje trenutno prebrane podatke. Ti podatki so ena izmed metrik, in pa specifikator. Do podatkov naprava pride tako, da konstantno cikla čez različne metode za branje senzorja, kar pomeni, da naprimer prebere podatke za temperaturo, jih drži v bufferju za 1 sekundo, nato pa preklopi na branje pritiska. Če v tisti sekundi strežnik zahteva podatke, mu jih pošlje, drugače pa nadaljuje z izvajanjem.

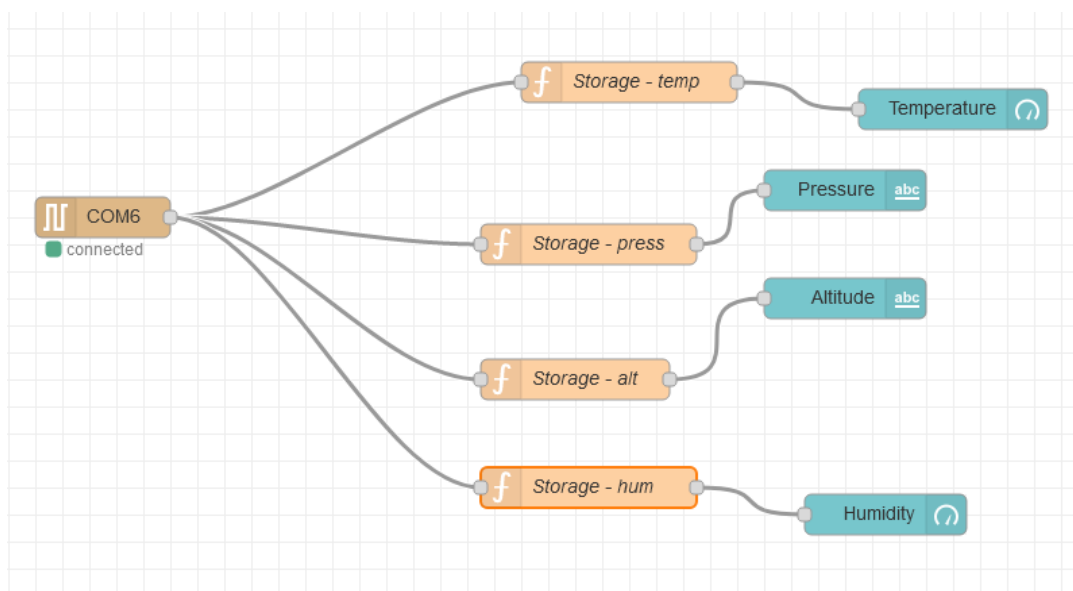
Implementacija: <https://github.com/LukaSveigl/VIN-Project-WeatherStation-v2/blob/main/station/station.ino>

4. NODE-RED DASHBOARD

Obe iteraciji uporabljata tudi Node-RED implementacijo dashboarda (to je spletna stran, ki prikazuje podatke, prebrane z strežnika). Za to tehnologijo sem se odločil, ker je izjemno preprosta za uporabo in postavitev, prav tako pa omogoča hitro interaktivno spreminjanje izgleda. Dashboard trenutno teče na localhost naslovu, bi pa ga bilo možno narediti javnega, torej gostiti ga na nekem strežniku.



Slika 6: Dashboard za projekt (lastni vir)



Slika 7: Izgled implementacije v Node-RED razvojnem okolju

Vse kar prikazani dashboard počne, je da sprejme podatke z serijskega vrat COM6, in jih pošlje v funkcije "storage". Kar te funkcije počnejo je le to, da pogledajo, ali so dobile prave podatke, in če so, ekstrahirajo številčno vrednost in jo pošljejo na prikazovalnike.

Implementacija: <https://github.com/LukaSveigl/VIN-Project-WeatherStation-v2/blob/main/dashboard/flows.json>

5. MOŽNE IZBOLJŠAVE

Na splošno sem z novo iteracijo projekta dokaj zadovoljen, a se da trenutno verzijo projekta seveda še močno izboljšati. Komunikacijo med strežnikom in postajo bi lahko implementiral preko WiFi ali GSM povezave, prav tako pa bi lahko Arduino NANO napajal z preprosto baterijo (trenutno potrebuje povezavo na računalnik). Poleg tega pa bi seveda postaji lahko dodal zmožnosti merjenja mnogih drugih metrik.