

Факултет инжењерских наука Универзитета у Крагујевцу

Ђорђе Р. Мијаиловић

**Развој сензорских система за прикупљање
података и њихов приказ на “web-u”
применом “NodeMCU” платформе
Мастер рад**

Крагујевац, 2019.

Факултет инжењерских наука Универзитета у Крагујевцу



Назив студијског програма: Индустијско инжењерство

Ниво студија: Мастер академске студије

Модул: Пословни информациони системи

Предмет: Индустијски рачунарски системи

Број индекса: 342/2018

Ђорђе Р. Мијаиловић

**Развој сензорских система за прикупљање
података и њихов приказ на “web-u”
применом “NodeMCU” платформе
Мастер рад**

Комисија за преглед и одбрану:

1. др Миладин Стефановић - ментор

2. _____

3. _____

Датум одбране: _____

Оцена: _____

У оквиру овог мастер рада кандидат треба да

Потребно је приказати сензорски систем за прикупљање података и њихов приказ на “web-u” применом “NodeMCU” платформе. Приказати расположиве технологије, извршити критичку анализу. Дефинисати архитектуру софтерског система, приказати валидацију и верификацију софтерског система. Развој и имплементација система који повезује “NodeMCU” платформу и сензорске компоненте са функцијама за приказивање података о факторима окружења како за приказивање података у реалном времену, тако статистичких података о факторима окружења. Рад демонстрира три начина за презентацију података који се прикупљају помоћу “NodeMCU” сензорских чворова.

Препоручена литература:

[1] Rouse Margaret, Embedded system.

[2] Breuer David, Monitoring ambient air quality for health impact assessment.

[3] Гржетић Иван, Суспендоване и распирабилне честице у урбаним срединама.

Крагујевац, 14.05.2019.

Ментор:
др Миладин Стефановић

Садржај:

1. Увод	7
2. Сензорски системи.....	9
2.1. Уграђени системи.....	9
2.2. IOT (internet of things).....	10
2.3. Улога сензора	10
2.4. Сензорске мреже.....	11
3. NodeMcu	12
3.1. Software.....	12
3.2. Hardware	14
3.3. Модели NodeMcu платформе	15
4. Arduino платформа.....	17
4.1. Software.....	17
4.2. Hardware	18
4.3. Модели Arduino платформи.....	20
5. Разлика између NodeMcu и Arduino платформе.....	21
6. Мониторинг фактора окружења	22
7. Имплементација сензорских система за прикупљање података и њихов приказ на web-у применом NodeMcu платформе.....	24
7.1. Хардверска инфраструктура првог модела	25
7.2. Хардверска инфраструктура другог модела.....	27
7.3. Хардверска инфраструктура трећег модела	28
8. Софтверска инфраструктура.....	31
8.1. Модел 1. Директан интерфејс за приказ података на web-у	31
8.2. Модел 2. Десктоп визуелизација помоћу Excel-а	35
8.3. Модел 3. Приказ података на web-у користећи ThingSpeak као проху сервер	38
9. Унапређење система заснованог на сензорским системима и NodeMcu платформи	46
10. Закључак	47
11. Литература	48

Abstract:

This paper presents the NodeMcu platform as well as the sensor systems. In addition to describing the platform and its characteristics, the work contains a practical part in which a system for monitoring the factors collected by the sensor systems is created. The purpose of this system is to show how NodeMcu communicates with sensors and how much of the capabilities provided by this platform. In this paper, the NodeMcu V3 platform model as well as sensors fc-22 and dht-11, breadboard, relay module were used. This system displays data on a web site in both real-time and packet-arithmetic environments, minimum and maximum values, collected by sensors. Data can be accessed at any time and from anywhere. The data displayed are the temperature expressed in celsius and kelvin, as well as in farenhites, air humidity, condensation temperature, accelerated condensation, air pollution covering several types of harmful gases. It also provides the ability to store data in a database that is located on the Internet as well as in a local database, provides the ability to control the platform itself over the Internet, with a computer or telephone subnet. The work itself demonstrates four different application models.

Key words: NodeMcu, sensor systems, fc-22, dht11, breadboard, relay module, embedded systems, IOT

Резиме рада:

У овом раду представљена је NodeMcu платформа као и сензорски системи. Поред описа платформе и њених карактеристика рад садржи практични део у ком је израђен систем за праћење фактора прикупљених сензорским системима. Циљ овог система је да покаже како NodeMcu комуницира са сензорима као и колике могућности пружа ова платформа. У раду је коришћен NodeMcu V3 модел платформе као и сензори fc-22 и dht-11, breadboard, relay modul. Овај систем показује податке на web страници како у реалном времену тако и пакетно-аритметичке средине, минималне и максималне вредности, прикупљене сензорима. Подацима се може приступити било када и са било ког места. Подаци који се приказују су температура изражена у целзијусима и келвинима као и у фаренхајтима, влажност ваздуха, температура кондезације, убрзана кондезација, загађеност ваздуха који обхвата више врста штетних гасова. Такође пружа могућност складиштења података у базу која се налази на интернету као и у локалну базу, пружа могућност контроле саме платформе путем интернета а подсретством компјутера или телефона. Сам рад демонстрира три различита модела примене.

Кључне речи: NodeMcu, сензорски системи, fc-22, dht11, breadboard, relay module, уграђени системи, IOT

Предговор

Захваљујем ментору на помоћи приликом одабира теме мастер рада као и на усмеравању током израде мастер рада. Захваљујем свим професорима који су ме пратили током досадашњег школовања.

1. Увод

Фактори који се налазе у окружењу људи, запослених а који се могу прикупљати сензорским системима уз примену од неких многобројних микроконтролерских платформи, могу у великој мери утицати на здравље и понашање запослених. Када се говори о факторима који се односе на неко одређено окружење као што је једна поросторија или производна хала подразумева се да се говори и амбијенталним факторима или факторима окружења (Jazizadeh et al., 2011). Као резултат досадашњих истраживања развијени су стандарди који дефинишу прихватљиве границе ових фактора, а које се односе на дозвољене концентрације појаве одређених фактора. Сходно томе да фактори из окружења запослених утичу на њихово здравље као и понашање у великој мери се односи и на продуктивност, са тог аспекта а који се односи пре свега на здравље људи као и околине уопште, потребно је контролисати ове факторе и одржати их границама погодним за рад као и живот људи (Sagha et al., 2018). Према наведеном ако су радно окружење и клима са одговарајућим факторима позитивни за запослене они ће бити задовољнији и даће већи максимум, што утиче на продуктивност на послу.

Као један од многих начина праћења амбијенталних фактора а посредством сензорских система као и контроле електричних уређаја на основу прикупљених података а као једно најбољих решења је NodeMcu платформа. Ова платформа пружа могућност конекције бежичним путем на интернет, пружајући на тај начин како прикупљање података и њену визуализацију на web-у тако и контролу електричних уређаја такође користећи интернет. Самим тим NodeMcu платформа представља један од уграђених система који заједно са неки другим електричним системом пружа веће функционалности, обавља брже и боље одређене задатке. Поред самог приказа података како на web-у тако и на lcd дисплеју, као поред контроле електричних уређаја ова платформа пружа могућност употребе различитих типова сензора као и сензорских мрежа. Употреба сензора омогућава прикупљање података о факторима околине и на тај начин пружа бољи увид у окружење у којем запослени раде. У зависности од прикупљених података појединачним сензорима или користећи сензорске системе, могу се применити различите методе за смањење негативног утицаја ових фактора на здравље људи и загађеност животне средине.

Са порастом употребе електричне енергије у облику у којем је данас користимо за свакодневне потребе па и за индустријске потребе, расте и заинтересованост за контролу електричних уређаја. Захваљујући појави све више врста микроконтролерских платформи сама контрола електричних уређаја је олакшана. Упоредо са развојем компјутера развијао се и интернет, па и сам приступ одређеним уређајима. IOT је допринео контроли како компјутерима, платформама, телефонима, па и свим осталим уређајима који се могу повезати на интернет. Често је потребно контролисати уређаје који обављају монотоне послове или јако прецизне а на неки начин им је потребан приступ као и контрола, као добро решење се намећу микроконтролерске платформе које се могу повезати на интернет и на тај начин се може имати приступ са било које локације у реалном времену. Што пружа низ предности као основна је та да запослени или неко ко је задужен за одређени уређај не мора физички бити на тој локацији када она обавља одређену операцију. Преднос се може посматрати и са стране уколико машина обавља неке операције или ради са неким супстанцама које су штетне по здравље људи или запослених они не морају бити у контакту са тим супстанцама већ контролу могу обављати са безбедне дистанце.

Имајући у виду огроман значај и утицај фактора из околине који могу утицати на здравље људи циљ рада је да представи три модела који су базирани на сензорским

системима за прикупљање података повезаних са NodeMcu платформом. Као и представљање података у реалном времену на web-u, приказ графика за вредности прикупљене сензорским системима, чување података у бази на интернету као и у локалној бази. Поред система за праћење фактора из околине у раду су представљени сензорски системи као једна од основних компоненти рада а који представљају системе који се састоје од два или више сензора циљ већег броја сензора као и сензорских система уопште је прикупљање веће количине података из свог окружења. Представљена је NodeMcu платформа која је коришћена за демонстрацију овог система у поглављу 3 су представљене карактеристике ове платформе, насупрот NodeMcu платформе представљена је и Arduino платформа разлог навођења ове платформе је могућност израде овог рада коришћењем и ове платформе у конкретном случају ова платформа би захтевала додатне компоненте, али би свакако пружала још одређених могућности уколико је кориснику потребно проширење система. Такође је представљена суштинска разлика ових платформи као и разлози примене платформе која је коришћена у раду.

Представљени су фактори окружења као и њихове границе које су дозвољене и које не угрожавају здравље људи, ови фактори имају огроман значај како у овом раду тако и значај који се односи на заштиту животне средине. Представљена је *хардверска* као и *софтверска* инфраструктура рада која се заснива на три модела, где се први модел односи на директан интерфејс за приказ података на “web-u” подаци прикупљени сензорским системима се приказују на на “web” страници, други модел рада се заснива на десктоп визуелизацији помоћу “Excel-a” циљ овог модела је приказ података на десктопу уколико се платформа налази на локацији на којој нема приступ интернету такође на овој начин обезбеђује базу података, складишти податке прикупљене сензорским системима у “Excel” пружајући могућност приказа графика као и даље обраде података, трећи модел рада се односи на приказ података на “web-u” користећи “ThingSpeak” као “проху” сервер, идеја трећег модела се заснива на постављању подата у удаљену базу која се налази на интернету а којој можемо приступати било када и са било које локације, такође пружа и могућност обраде података и приказа графика у реалном времену као додатна функција трећег модела је “Virtuino” који обезбеђује паметним телефонима преузимање података из ове базе и прикази истих на телефону на овај начин приказа података омогућен им је приступ у сваком тренутку. Сва три модела се могу користити у индустријским халама као и другим окружењима за прикупљање фактора окружења у циљу заштите здравља људу и уопште животне средине. Циљ рада се заснива на представљању сензорских система за прикупљање података и њихов приказ на “web-u” применом “NodeMCU” платформе.

2. Сензорски системи

Захваљујући брзом развоју технологије, чипова, микрочипова, микроконтролерских платформи а првенствено сензора данас имамо приступ многобројним подацима. Сензорски системи проналазе примену у свим областима инжењерства, роботике, науке, технологије као и у једној од најбитнијих области у медицини. Налазе се свуда око нас, користе се за потребе мерења временских промена, потребе позиционирања, праћења. Да бих добили статус сензорских система и да бих постали системи морају обухватати најмање два сензора а могу поседовати и огроман број. Захваљујући већем броју различитих врста сензора добијамо већи број различитих података прикупљених сензорима које даље можемо обрађивати и користити. Основна улога сензора у оквиру сензорских система је прикупљање података. Основни циљ сензорских система је добијање информација са високом тачношћу из више сензора, мерења сензора користећи напредне технологије. Коришћење вишеструких сензорских података заједно зове се фузија података. Да бих се добили прецизни подаци информатика користи адаптивне методе обраде сигнала, статистичке метода коришћене заједно са математичким моделима.

Једна од најважнијих примена сензорских система је свакако у медицини где се сензори користе за контролу људског здравља, као телесних функција користећи различите врсте биосензора. Користе се за сликање мозга, контролу рада срца, сензори који мере активност мишића, сензори покрета тела. Свакако једна од све чешћих употреба сензорских система је у паметним телефонима, пружајући информације о покрету, праћење оријентације и локације паметног телефона у реалном времену користећи сензоре унутрашњег убрзања, жироскопе, магнетометре. Ако се базирамо на индустрију и све чешћу употребу робота као и роботских руку схватамо да су сензорски системи и овде незаобилазни. Као основна потреба робота је свакако његов положај у простору као информација о времену које је протекло док је обавио одређену операцију, а по некада је потребан и приступ интернету како бих послао податке до базе. Захваљујући низу сензора робот добија све потребне податке. Поред наведених места сензорски системи се данас све чешће користе и у аутомобилској индустрији како за саму производњу аутомобила тако и уграђени у аутомобиле. Данас сваки аутомобил поседује већи број сензора самим тим поседује уграђене сензорске системе. Користе се и у авионима, метеоролошким станицама за праћење промена као и мерења истих.

2.1. Уграђени системи

Представљају системе са јаком интеграцијом хардвера и софтвера унутар неког другог система, намењени обављању специфичних задатака. Базирани су на рачунарима и уграђени су у оквиру другог система коме обезбеђују боље функционалности и перформансе (Електронски факултет, 2005). Уграђени системи су системи специјалних намена, који се углавном примењују на тешко доступним местима. Насупрот рачунарима опште намене уграђени системи обављају један или више унапред дефинисаних задатака од стране рачунара. Узимајући у обзир намену уграђених система као и могућност обављања специфичних задатака, пројектант може да оптимизује ове системе са аспекта перформанси, а да при томе репродукује физичке величине, потрошњу енергије, цену производа. Уграђени системи данас представљају најчешће масовне производе који се састоје од многобројних компонената (Rouse, 2015). Свакако поред већ дефинисаних уграђених

система неопходно је такође описати и IOT (internet of things) јер укључују уграђене системе.

2.2. IOT (internet of things)

У односу на уграђене системе који се базирају на инсталацију хардвера у други систем пружајући му на тај начин боље перформансе као и хардверског програмирања, IOT подразумева и хардверску контролу од стране интернет подсистема. Ова карактеристика која је јако значајна у односу на сензорске системе представља и једну од основних предности као и подразумеваних особина данашњице. Захваљујући брзом и све бржем развоју интернет мрежа, паметних телефона, рачунара, контрола система је олакшана и пружа могућност контроле са било које локације у реалном времену. Ово укључује микроконтролерске платформе, сензоре, бежични интернет као и друге хардверске компоненте (Doukas, 2012).

Са обзиром да се у овом раду описују три модела као и сам начин приказа података на web-у, чување података у бази на интернету, а при томе користећи микроконтролерску платформу као и сензорске ситеме, уграђене системе, контролу од стране рачунара и паметног телефона може се рећи да овај систем припада IOT-у. Први модел овог система који ће бити представљен у поглављу 7.1. припада IOT-у сходно томе да се подацима који добијени прикупљањем са сензорских система може приступити у реалном времену са било које локације као и да је приступ подацима омогућен са телефона као и са компјутера. Трећи модел који ће бити представљен у поглављу 7.3. такође припада IOT-у сходно томе да ThingSpeak припада IOT-у.

2.3. Улога сензора

Једна од најбитнијих компоненти уграђених система као и сензорских система су свакако сензори. Захваљујући развоју технологије и потреба саме примене сензора данас постоје многе врсте сензора и намена им је самим тим веома велика и различита. Једна од основних подела сензора а када посматрамо и микроконтролерске платформе као што је NodeMcu а која ће бити коришћена и представљена у овом раду је на аналогне и дигиталне. Основна разлика је у излазу односно улазу да ли су аналогни сензори или дигитални. Поред основне поделе постоје и мерни претварачи који једну величини претварају у другу, аналогно дигитални конвертери претварају аналогне вредности у дигиталне, дигитално аналогни конвертери претварају дигиталне вредности у аналогне (Гржетић, 2010). Основна улога сензора у уграђеним системима као и сензорским системима је прикупљање информација из околине у којој се налазе, информације се даље од сензора преноси до микроконтролерских платформи где се даље врши обрада и добијају се подаци. У даљем процесу подацима се може манипулисати на нама одговарајући начин а који ће бити дефинисан програмирањем (Мијаиловић et al., 2018). Захваљујући великом броју сензора као и микроконтролерских платформи можемо добити широк спектар података који се може користи на нама одговарајући начин. Сензорима се могу прикупљати подаци као што су: количина загађености ваздуха, температура, влажност ваздуха, приисак, растојање од одређеног предмета, густина течности као и састав и температура итд. Сходно њиховој улози може се рећи да сензори имају огромну улогу у прикупљању информација и самим тим касније у донешењу одлука које су засноване на прикупљеним информацијама а чијом

енкапсулацијом се долази до квалитетних и прецизних података. Налазе своју примену у многим областима неке од њих су аутомобилска индустрија, медицина, роботика, као и за научна истраживање. За обраду прикупљених информација можемо користити неку од микроконтролерских платформи као што је Arduino или NodeMcu постоји више врста наведених платформи неке од њих ће бити представљене у даљем тексту као и наведене суштинске разлике предности и мане. Једна од основних предности наведених платформи а која карактерише све је релативно приступачна цена. За добијање квалитетнијих података као и прецизнијих често се могу комбиновати две или више платформи. Пружају тачне информације које су прикупљене сензорима у реалном времену што значи да се информације преносе у тренутку прикупљања. Примена сензора заједно са неком од микроконтролерских платформи, а које се могу користити у зависности од намене и самог физичког места примене зову се уграђени системи обухватајући микропроцесоре и микроконтролере као и многе друге компоненте (Гржетић, 2010).

2.4. Сензорске мреже

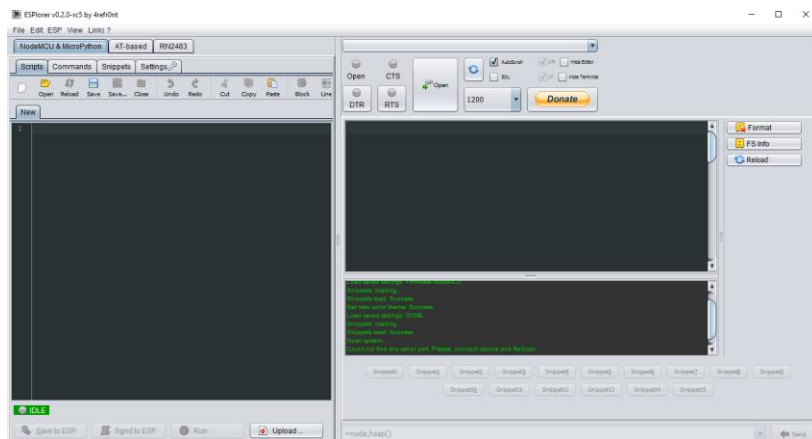
Сензорске меже представљају дистрибуиране системе који се састоје од сензора који повезани прикупљају различите врсте података. Примена сензорских мрежа темељи се на коришћењу малих јефтиних мултифункционалних платформи које прикупљене податке из сензора достављају кориснику. Налазе огромну примену у екологији, здравству, саобраћају и другим областима. Састоје се често од великог броја различитих сензора лоцираних на физички удаљеним и различитим локацијама врло често и неприступачним обухватајући аутономне сензоре за праћење физичких или еколошких услова у којима се налазе. Основна намена сензорских мрежа је сакупљање, обрада и достављање информација некој платформи или крајнем кориснику. За мерење и прикупљање резултата служе одговарајући типови пасивних или активних сензора који су део мултифункционалних платформи, сензорских чворова. Сензорски чворови омогућавају складиштење и обраду прикупљених података [Раденковић и сар., 2017].

3. NodeMcu

NodeMcu је IOT платформа отвореног кода, укључујући *фирмвер* који ради на ESP8266 чипу, као и хардвер који се базира на ESP-12 модулу. Термин “NodeMcu” се односи на *фирмвер* а не на развојно окружење. Пружа могућност програмирања бежичног модула ESP8266 помоћу скриптног језика LUA или Arduino IDE окружења. С’ обзиром да се хардвер ове платформе заснива на хардверу Ардуино платформе са основном разликом која се односи на ESP8266 чип који је интегрисан у овој платформи у наставку текста ће бити представљена и Ардуино платформа као и дефинисана разлика ових платформи. Представља сложени уређај који комбинује неке од Ардуино платформи са могућношћу повезивања на интернет. Поседује већи број GPIO пинова на плочи што му пружа могућност повезивања са другим периферијским уређајима и способност генерисања PWM, I2C и UART сријске комуникације. Интерфејс је подељен на два дела *фирмвер* и хардвер који се некада заснивао на ESP8266 Wi-Fi чипу док је данас то ESP-12 модул (Al Dahound et al., 2018).

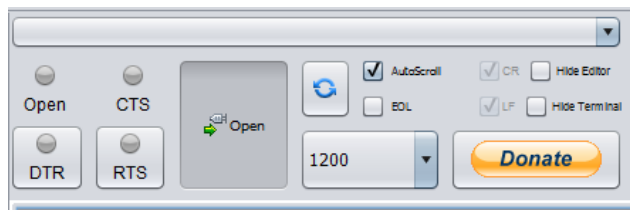
3.1. Software

Software се покреће на компјутеру, у случају коришћења ове платформе постоје две опције прва је коришћење Arduino IDE окружења друга опција представља коришћење скриптног језика LUA. IDE је посебан програм који се користи за писање и слање кода на плочу, окружење ће бити опусана под насловом 4.1. Пре почетка самог програмирања неопходно је да се NodeMcu повеже са компјутером што се остварује USB конекцијом. Након конектовања за програмирање користећи LUA скриптни језик неопходно је инсталирати на компјутеру ESPlorer. ESPlorer се користи као LUA едитор, представљајући софтвер отвореног кода за развој LUA програма. На слици 1. се може видети приказ ESPlorer.



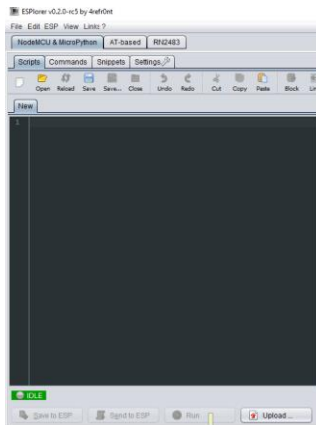
Слика 1. Приказ ESPlorer едитора

Након покретања ESPlorer-а за сам почетак програмирања неопходно је подесити COM порт као и Baud који представља брзину преноса података. Ове опције се налазе у горњем десном углу ESPlorer прозора, приказ подешавања се може видети на слици 1.2.



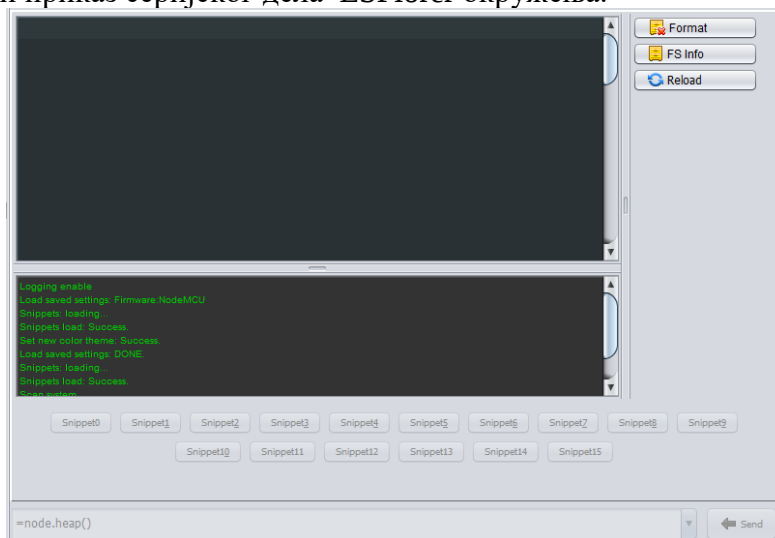
Слика 1.2. Подешавање COM порта и Baud

По завршетку потребних подешавања и инсталација може се прећи на програмирање NodeMcu платформе, код се пише у крајњем левом углу окружења које се може видети на слици 1.3.



Слика 1.3. Приказ окружења намењеног писању кода

Након завршетка писања кода отпремање на NodeMcu платформу се врши кликом на дугме Upload које се такође може видети на слици 1.3. Статус послатог кода као и статус конекције на WiFi бежичну мрежу се може видети у десном делу овог окружења које представља серијску комуникацију платформе и компјутера а која се прекида при дисконектовању-искључењу USB конекције NodeMcu платформе и компјутера. На слици 1.4. се може видети приказ серијског дела ESPlorer окружења.



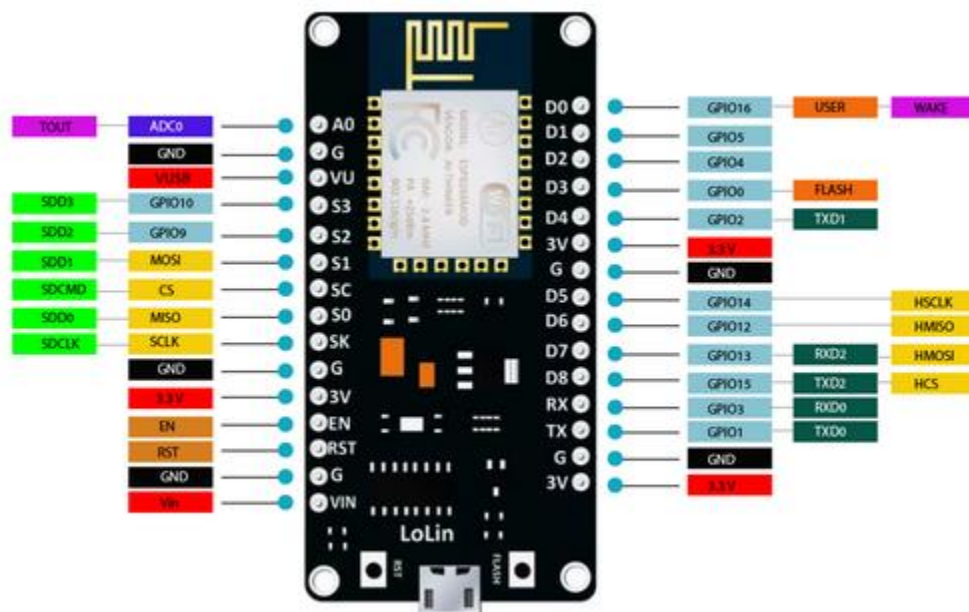
Слика 1.4. Приказ серијског дела ESPlorer-а

По успешном завршетку преноса кода на NodeMcu платформу и након добијања поруке о статусу кода, платформа се може користити независно од компјутера. Код који се учита на платформу ће се извршавати до наредног преноса кода или до завршетка своје

функционалности уколико је тако предвиђено програмом. Када се прекине USB конекција за исправно функционисање NodeMcu платформе неопходно је обезбедити напајање. При укључењу USB конекције исписаће се порука да је NodeMcu платформа дисконектована и да је неопходна поновна конекција за отпремање кода.

3.2. Hardware

NodeMcu V3 платформа заснован је на ESP8266 који представља WiFi микрочип који укључује TCP/IP стек, поседујући на себи и микроконтролер што га чини јако сличним Arduino платформи. Представљен је од стране произвођача Espressif Systems са седиштем у кини. Првобитне старије верзије NodeMcu платформе су се заснивале на ESP8266 чипу док се новије верзије заснивају на ESP-12 модулу. На самом модулу се налази микро USB порт који се користи за конекцију са компјутером и користи се за напајање и програмирање такође се може користити за програмирање са паметног телефона. Поседује на себи и статусне LED лампице које трепере дајући тренутни статус модула ако ради исправно када је повезан са рачунаром (<https://www.theengineeringprojects.com/2018/10/introduction-to-nodemcu-v3.html>, 2019). На слици 1.5. представљен је NodeMcu V3 платформа са представљеним пиновима.



Слика 1.5. NodeMcu V3 платформа са представљеним пиновима

Извор: <https://www.theengineeringprojects.com/2018/10/introduction-to-nodemcu-v3.html>

Као што се може видети са слике 1.5. NodeMcu поседује 16 дигиталних пинова, док на себи има само један аналогни пин.

Поседује 5 пинова 3V3 који се могу користити за напајање сензора и других модула са којима се повезује ова платформа. Основни проблем је тај да не може подржати већи број потрошача-модула, с обзиром да не подржава излаже веће од 800mA. Што некада може представљати проблем. Наравно овај проблем се може решити додатним напајањем где у том случају модули не би користили напајање са NodeMcu платформе већ то екстерно напајање, са платформом би били повезани само преко својих аналогних-дигиталних

пинова преко којих би се вршио пренос података. Ово је јако битан податак с' обзиром да већа потрошња напона од дозвољене би уништило платформу сходно томе препоручује се примена екстерног напајања. Свакако још један недостатак је што NodeMcu платформа не поседује пин који подржава 5v што многи модули захтевају, такође у овом случају је неопходна примена екстерног напајања.

Vin пин је регулатор напона који долази са могућношћу да подржи до 800mA. Може поднети између 7 и 12v. Користећи овај пин не могу се напајати модули који раде на 3,3v, из разлога што овај пин не може генерисати напон те јачине.

Gnd пин, је пин негативног напона подједнако значајан као и пин од 3.3v, NodeMcu поседује 5 Gnd пинова. Уколико се користи нека врста екстерног напајања тада ће се користити и Gnd пин тог напајања .

USB to UART конвертер се додаје на модул који помаже у конвертовању USB података у UART податке који служе са серијску комуникацију.

Поседује два дугмета Flash и Rst који служе за брисање кода који се већ налазе на платформи. Карактеристике меморије поседује флеш меморију од 4MB као и SRAM 80Mhz.

3.3. Модели NodeMcu платформе

Неки од модела NodeMcu платформе ће бити представљене у табели 1.1. Поред платформи се могу видети и њихове карактеристике. Као што се из табеле може видети ове платформе се разликују како физички у визуелном смислу тако и у јачини чипова као и по броју пинова. У наведеној табели се могу видети и платформе Wemos D1 Mini и Wemos D1 R2 које иначе не припадају NodeMcu платформи али су наведене у табели из разлога што користе ESP8266 Module који користи и NodeMcu платформа. Разлика ових платформи је и у одређеним пиновима док су остале карактеристике исте.

Табела 1.1. Модели NodeMcu платформе

NodeMcu V0.9	
	• GPIO Pins – 11 • ESP8266 Module - ESP12 (AI-Thinker) • Serial Chip - CH340G • 4MB flash memory • SRAM: 64 KB
NodeMcu V1.0	
	• GPIO Pins – 11 • ESP8266 Module - ESP12 (AI-Thinker) • Serial Chip - CP2102/CH340G • 4MB flash memory • SRAM: 64 KB

Wemos D1 Mini



- GPIO Pins – 11
- ESP8266 Module - ESP12E (Wemos)
- Serial Chip - CH340G
- Flash Memory: 4 MB
- SRAM: 64 KB

Wemos D1 R2



- GPIO Pins – 11
- ESP8266 Module - ESP12E (Wemos)
- Serial Chip - CH340G
- Flash Memory: 4 MB
- SRAM: 64 KB

NodeMcu V3



- GPIO Pins – 11
- ESP8266 Module - ESP12E (Wemos)
- Serial Chip - CH340G
- Flash Memory: 4 MB
- SRAM: 64 KB

4. Arduino платформа

Разлог навођена Arduino платформе у овом раду је што су NodeMcu и Arduino платформа врло сличне тачније и NodeMcu као и Arduino платформа припадају микроконтролерским платформама. Самим тим овај пројекат могао би се реализовати и коришћеном Arduino платформе сходно томе биће представљена у наставку текста са својим карактеристикама као и суштинске разлике ове две платформе такође да бих се програмирала NodeMcu платформа може се програмирати користећи окружење у коме се програмира Arduino платформа. Развој Arduino платформе се везује за институт за интерактивни дизајн Design Institute Ivrea IDII једноставан алат за брзу израду пројеката, намењен студентима без великог предзнања у електроници и програмирању. Један од главних циљева настанка Arduino платформе био је да олакша процес рада са технологијом и електроником. Са Arduino платформом се могу израдити како једноставни тако и доста захтевни пројекти. Arduino је физичко-рачунарска платформа отвореног кода. Састоји се од једноставног хардверског дизајна Arduino плоче са Amdel AVR процесором и пратећим улазно-излазним елементима, на себи поседује микроконтролер и software дела.

Arduino хардверске компоненте се програмирају користећи програмски језик заснован на Wiring језику кога чине синтакса и библиотеке. У основи овај језик је веома сличан програмском језику C++ али има неких измена. Arduino плоче имају могућност читавања улазних вредности као што су светлост, удаљеност предмета, прикупљање многих вредности помоћу сензора као што је квалитет ваздуха температура, SMS поруке или поруке које се могу примити са интернета да бих примао поруке са интернета Arduino мора бити повезан на интернет и Arduino их претвара у излаз чиме може да покреће мотор, укључује светло, покретање спољних догађаја. Arduino плоча се састоји од 8-битни Amdel AVR микроконтролер са компонентама које омогућавају програмирање и повезивање са другом електроником. Arduino има стандардизован распоред конектора који омогућава лако повезивање са додатним модулима. Ове додатне модуле, производе разни произвођачи широм света. Званичне Arduino плоче углавном користе megaAvr серију чипова, Atmega8, Atmega 168, Atmega 1280, Atmega 2560 (Sandhu, 2016). Arduino је састављен од два главна дела: Arduino плоче који представља део хардвера и IDE који представља део софтвера који се покреће на компјутеру. IDE се користи за креирање малог програма који ће бити отпремљен на Arduino плочу тај мали програм ће говорити шта ће Arduino радити [Massimo, 2014].

4.1. Software

Software се покреће на компјутеру IDE је посебан програм који се користи за писање и слање кода на плочу, састављену од компајлера и bootloader-а на плочи. За сам пренос кода на Arduino платформу неопходна је USB конекција Arduino платформе и компјутера. Након успешно пренесеног кода USB конекција може бити прекинута а за сам Arduino се може користити друга врста напајања. Резултат пренетог кода се може видети на компјутеру док је Arduino конектован USB везом са компјутером или на други начин уколико је у пројекат укључен и LCD module, или приказ на Web-у. Док је Arduino конектован USB везом са компјутером може се вршити тестирање програма и видети резултат. Након покретања IDE-а потребно је изабрати тип Arduino платформе или NodeMcu платформе уколико се она програмира у овом окружењу, као и порт на који је повезана платформа. Уколико су сви

параметри исправно подешени биће могућ преност кода на платформу у супротном биће исписана порука о грешци и код неће бити пренет. Све ово је јако битно с' обзиром да све наведено важи и за NodeMcu платформу.

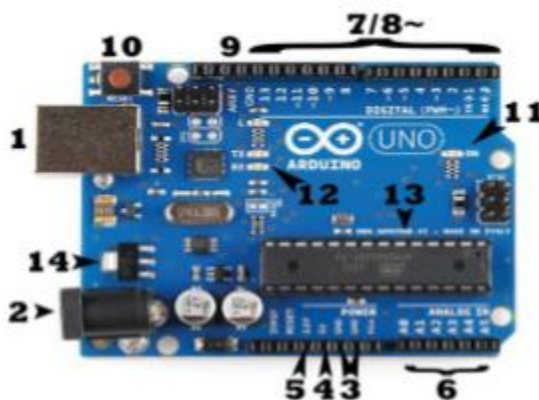


Слика 1.6. IDE Environment

На слици 1.6 се може видети приказ IDE окружења у коме се могу програмирати наведене платформе.

4.2. Hardware

Hardware део је значајан како бих показао сличност Arduino платформе и NodeMcu платформе као и евентуалне разлике. Постоји много врста Arduino плоча које се могу користити у различите сврхе. У зависности од модела плоче могу изгледати другачије, али већина Arduino плоча се састоји од ових заједничких компоненти које су представљене на слици 1.7:



Слика 1.7. Arduino модел uno

Свака Arduino плоча мора се прикључити на напајање. Arduino Uno се може прикључити на напајање са USB кабла који долази са рачунара или из зидног напајања. На слици 1.7 USB конекција је означена са (1), а прикључница из зида је означена (2). USB конекција такође служи како бих се пренео код са рачунара на Arduino. Јачина напона која је

предвиђена да се користи на већини плоча је од 6v до 12v. Напон већи од 20v може уништити Arduino.

Пинови (5V, 3.3V, GND, Analog, Digital, PWM, AREF). Пинови су места на плочи на којима се врши конекција жица са конструкцијом склопа. Обично се на Arduino налазе црна пластична заглавља која омогућавају једноставно прикључивање жица. Има неколико различитих типова пинова, од којих је сваки означен на плочи и има различиту намену.

GND (3): Постоји неколико GND пинова на плочи у зависности од модела различит је и број пинова, сваки од њих се може користити за прикључење кола.

5V (4) и 3.3V (5): 5V пин се користи за напајање од 5V, пин 3.3V се користи за напајање од 3.3V, већина компоненти које се повезују на Arduino користи напајање од 5V и 3.3V за покретање.

Аналогни пинови (6): област пинова означена са 'Analog In' (A0 до A5 на UNO плочи) су 'Analog In пинови'. Ови пинови могу прочитати сигнале од аналогних сензора (као што су сензори температуре) и претворити их у дигиталне вредности које можемо читати (Massimo, 2014).

Дигитални пинови (7): Преко пута аналогних пинова су дигитални пинови (од 0 до 13 на UNO плочи). Ови пинови се могу користити за дигиталне улазе (као пример је дугме које је притиснуто или не) и дигиталне излазе (попут напајања LED).

PWM (8): тилде tilde (~) поред неких дигиталних пинова (3, 5, 6, 9, 10 и 11 на UNO плочи). Ови пинови делују као нормални дигитални пинови, али се такође могу користити за нешто што се зове Pulse-Width Modulation (PWM). Поред тога ови пинови су способни да симулирају аналогни излаз, као што је бледећи LED улаз и излаз постепено смањивање јачине светла и постепено појачавање.

AREF (9): Користи за постављање спољашњег референтног напона (између 0 и 5 волти) као горње границе за аналогне улазне пинове.

Reset Button (10) дугме за ресетовање. Притискањем привремено ће се прикључак за ресет поставити на земљу и поново покренути било који код који се налази на Arduino плочи. Ово може бити веома корисно ако се ваш код не понавља и ако желите да уклоните код са плоче.

Power LED Indicator (11). Испод и десно од речи "UNO" на плочи, налази се LED поред речи "ON". Ова LED лампица треба да светли када прикључите Arduino у извор напајања. Уколико се ова LED лампица не укључи постоје проблеми са колом.

TX RX LEDs (12). Ове LED диоде пружају визуелне индикације кад год Arduino прима или преноси податке. TX LED ће светлити када Arduino преноси податке, док RX LED ће светлити када Arduino прима податке.

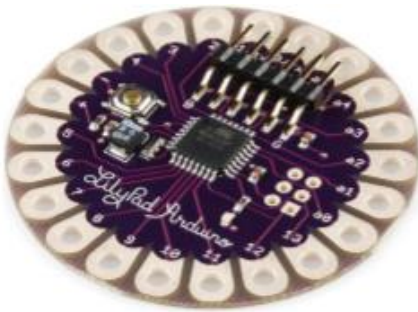
Main IC (13). Main IC се разликује од типа плоче, али обично га производи ATMEЛ компанија. Ово је веома важно зато што се мора знати IC тип пре него што се учита програм. Ове информације се могу наћи у писаној форми на горњој страни IC.

Voltage Regulator (14). Регулатор напона контролише количину напона који је прикључен на плочу. Регулатор напона има границу од 20 волти коју не треба прелазити, већи напон може оштетити Arduino или га уништити.

4.3. Модели Arduino платформи

Неки од модела Arduino плоча које се данас највише користе ће бити представљени у табели 1.2. У табели се поред платформи могу видети и њихове карактеристике. Као што се може видети из табели 1.2 платформе се разликују како у визуелном смислу тако и по снази њихових чипова, броју пинова. Све Arduino платформе поседују компоненте које су описане код модела UNO који сам користио како бих описао компоненте. Поред модела описаних у табели 1.2 данас се користе и следећи модели: LilyPad SimpleSnap, LilyPad USB, Micro, Pro, Pro Mini, Due, Zero и многи други.

Таблица 1.2. Arduino модели

	Arduino Mega 2560 • Atmega 2560 чип • 54/15 Digital IO/PWM • 16/0 Analog In/Out • 256 kB flash меморије • Радни напон 5V • Напајање 7-12 V • Улазни напон 5-20 V
	LilyPad Arduino • Atmega 168 чип или ATmega328V • Радни напон 2.7-5.5 V • 14 Digital IO • 6 PWM • 6/0 Analog In/Out • 16 kB flash memory
	Arduino Nano • ATmega168 или ATmega328P чип • Радни напон 5 V • Улазни напон 7-9 V • 8/0 Analog In/Out • 14/6 Digital IO/PWM • 16 kB flash memory за чип ATmega168, 32 kB flash memory за чип ATmega328P
	Arduino Due • ATSAM3X8E чип • Радни напон 3.3 V • Улазни напон 7-12 V • 12/2 Analog In/Out • 54/12 Digital IO/PWM • 512 kB flash memory

5. Разлика између NodeMcu и Arduino платформе

Ове две платформе поседују доста заједничких особина, али свакако и разлика. Разлог навођења обе платформе је што је у пројекту могућа примена било које од ове две. Насловима 3.1, 3.2 и 4.1, 4.2 су обухваћене суштинске карактеристике ових платформи како софтверских тако и хардверских. Из поменутог описа се може видети да и NodeMcu и Arduino платформа припадају микроконтролерским платформама сходно томе да на себи поседују микроконтролере. Обе платформе поседују аналогне и дигиталне улазе-излазе с' тим што је у поменутом случају Arduino платформа поседује много већи број ових улаза-излаза, што је јако значајно за било који пројекат. Захваљујући аналогним и дигиталним улазима-излазима се повезују разни модули и пожељно је да их има што више, наравно у зависности од модела платформе сходно томе да постоји више свака поседује одређене карактеристике. У овом делу текста ће се поредити Arduino уно модел платформе који је описан у насловима 4.1, 4.2 као и NodeMcu V3 платформа. Такође једна од основних и битних карактеристика је излазни напон NodeMcu платформа поседује 5 излаза од по 3.3V док Arduino поседује 3 излаза од по 3.3V и 3 3 излаза од по 5V, у наведеном случају Arduino платформа има предност сходно томе да велики број модула захтева да платформа има излазе од 5V, такође подржава више напона од NodeMcu платформа омогућавајући за неке мање пројекте да раде без екстерног напајања. Предност NodeMcu платформе у односу на Arduino је та што на себи поседује модул за интернет док је Arduino потребан додатни модул. Што пре свега повећава број коришћених компоненти као и саму цену пројекта. Али Arduino такође обезбеђује и повезивање на Ethernet модул ово је значајно уколико је потреба LAN конекција. Када је реч о димензијама NodeMcu је у предности, док када су остале карактеристике као што су снага микроконтролера, меморија парира Arduino уно моделу. Arduino је у малој предности када узмемо у обзир да постоји много већи број платформи него што поседује NodeMcu самим тим пружајући шири спектар карактеристика и сходно томе се бира одговарајућа платформа за одређени пројекат, док NodeMcu на свим својим платформама нуди готово исте или сличне карактеристике са основном разликом у чипу за WiFi конекцију. У табели 1.3 ће бити представљена разлика ове две платформе.

Табела 1.3. Разлике NodeMcu V3 и Arduino уно платформе

Карактеристике	NodeMcu V3	Arduino uno
Програмски језици	C/C++/LUA	C/C++
Брзина процесора	16 MHz	16 MHz
Hardware design	Open source	Open source
Real time	In real time	In real time
Analog to Digital	Да	Да
Integrisan WiFi чип	Да	Не
Analog пин	1	6
Digital пин	11	13
Flash Memory	4 MB	4 MB

6. Мониторинг фактора окружења

Када је реч о факторима окружења подразумевају се фактори који су прикупљени сензорским системима. Такође се често у литератури наводе као Амбијентални фактори окружења, подразумевајући да се односе на неко одређено окружење као што је једна фирма или канцеларија. У суштини и овај систем који је описан овим радом ће прикупљати факторе из одређеног окружења сходно томе могу се дефинисати као амбијентални фактори. Амбијентални фактори представљају факторе из одређеног окружења који негативно утичу на здравље људи који раде или који живе у том окружењу (Noah и et al., 2012). Ти фактори могу бити опасне хемикалије у ваздуху, бука, вибрација, електрична и магнетна поља, високе и ниске температуре као и влажност ваздуха, повећан ниво штетних честица у ваздуху, прашина, испарења, разне врсте гасова. У нашој околини могу се појавити и супстанце које имају следећи утицај на људе: Токсични утицај (честице, прашина, влакна, течност, гасови), супстанце које узрокују хроничне ефекте (као што је астма), Супстанце које су иритантне или су алергени. Ако су људи дужи временски период изложени високим температурама или ниским то може имати негативан утицај на здравље. Сви ови фактори имају дозвољене границе које када се прекораче могу имати негативан утицај на здравље људи, биљака, животињског света и уопште животне околине.

Утицај амбијенталних фактора је веома велики и јако битан, данас постоје многобројни закони и стандарди којима се прописује за сваки од фактора дозвољене границе као и оне које нису и које су штетне. Прекорачене границе могу довести до одређених ризика који имају репродуктивне, тератогене, мутагене и ендокрине факторе. Данас се користе разне врсте сензора који могу прикупљати информације из околине у којој се налазе и могу нам дати прецизне податке колика је количина загађености ваздуха у нашој животној средини. Неки од највећих загађивача данашњице су аутомобили као и многобројне фабрике које нажалост не поштују законе. Атмосферске честице или РМ су микроскопска чврста или течна материја суспендована у земљиној атмосфери. Извори честица могу бити природни или антропогени. Подтипови честица су грубе честице пречника између 2,5 и 10 микрометара (μm) (PM10), fine честице пречника од 2,5 μm или мање PM2.5, ултраfine честице и чађи. Честице које се јављају природно најчешће долазе од вулкана, шумских пожара, травнатих пожара. Људске активности које доводе до повећања честица су сагоревање фосилних горива, електране, индустријски процеси [International labour office Geneva, 2001].

PM10 честице пречника између 2,5 и 10мм механички су произведене од стране разбијања још већих чврстих честица. Порекло грубих суспендованих честица или PM10 је двојако, урбано и рурално основни извори су: моторна возила, пећи за сагоревање дрвета, прашина са градилишта, прашина са одлагалишта и депонија, прашина из пољопривредних региона, пожари, индустријска постројења која обухватају(термоелектране, постојења попут ливница метала, цементаре), прашина коју подижу јаки ветрови . PM10 представља смешу која обухвата дим, чађ, прашина соли, киселине, метали. Мање честице могу настати у процесу сагоревања фосилних горива као што је уље и бензин. Штетне честице попут пепела и чађи, секундарне честице кроз атмосферске реакције сумпор оксида и азот оксида који се ослобађају као гасови.

PM2.5 пречника 2.5мм или мање представљају респираторне честице које доводе до појаве разних врста плућних болести. Уносе се у организам дисајним путевима самим тим брзо стижу до плућа. PM2.5 обухватају fine и ултраfine суспендоване честице.

Обухватају аеросоли, дим, загушљива испарења, пепео. Настају у хемијским реакцијама које се одвијају у атмосфери или настају сагоревањем горива у моторним возилима, термоелектранама, индустријским постројењима, при сагоревању дрвета или приликом сагоревања појединих пољопривредних отпадних материјала на њивама. Састав PM2.5 честица се може видети у табели 1.4. Док PM10 представљају најопасније честице које се могу наћи у ваздуху оне приликом удисања нападају респираторни систем, утичу на његову отпорност и депонују се на најдубљим деловима плућа. PM10 доводе до појаве или ослабевају астму, бронхитис и друга обољења плућа, самим тим смањују отпорност организма. Ове честице доводе до оштећења вида, стварајући ефекат измаглице (Breuer, 2014).

Табела 1.4. Састав PM2.5 честица и њихов проценат

Категорија	Проценти
Чађ или органски угљеник из процеса сагоревања	30%-50%
Сулфати	30%-40%
Нитрати	10%-20%
Прашина са тла	3%-10%

Дозвољене границе:

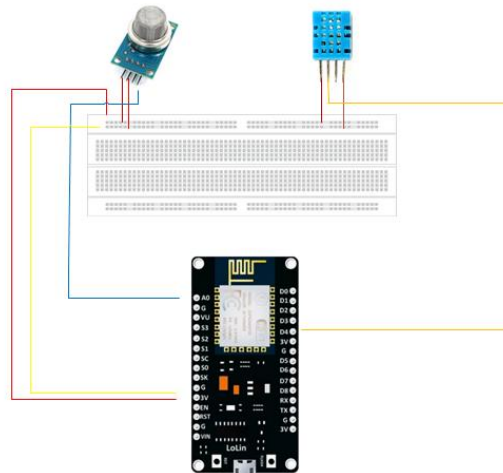
За људски организам најбоља је влажност ваздуха од 50 процената при температури од 18-20°C. Законски дозвољене граничне дневне вредности концентрације појединих полутаната (ГВИ) у ваздуху: Олово (Pb) 250/100 µgm/2, Цинк (Zn) 400/200 µgm/2, Чађ (дим) 50 µgm/3, Сумпордиоксид (SO₂) 150 µgm/3, Азотдиоксид (NO₂) 150 µgm/3, Угљенмоноксид (CO) 10 µgm/3 (Гржетић, 2010).

7. Имплементација сензорских система за прикупљање података и њихов приказ на web-у применом NodeMcu платформе

За систем су коришћене следеће компоненте:

- NodeMcu платформа,
- сензор Fc-22,
- сензор dht11,
- relay modul,
- breadboard,
- lcd display.

У наставку текста ће бити описане све наведене компоненте појединачно, биће приказане комплетне хардверске инфраструктуре за сваки модел. Такође за сваки модел од три која су обрађена овим радом ће бити представљена и софтверска решења, као и представљене могућности проширења истих. Биће представљени проблеми сваког модела као и могућа унапређена, као и опис зашто је одређени модел урађен на баш тај начин. На слици 1.8 се може видети приказ шеме споја NodeMCU платформе са сензорима.

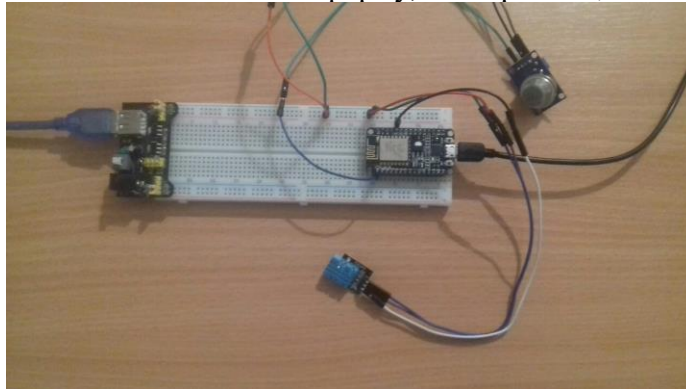


Слика 1.8. Шема споја NodeMCU платформе са сензорима

Приказ шеме на слици 1.8 представља начин конектовања сензора са NodeMCU платформом. На шеми се могу видети пинови сензора и NodeMCU платформе који конектовани а који представљени детаљније у делу рада који описује коришћене компоненте, такође на шеми се може видети и breadboard који се користи за лакшу конекцију као и пружања могућности повезивања екстреног напајања. Шема која је приказана на слици 1.8 се користи у сва три модела рада.

7.1. Хардверска инфраструктура првог модела

Хардверском инфраструктуром ће бити обухваћена сва три модела овог рада и свака компонента ће бити описана. На слици 1.9. се може видети хардверска инфраструктура која је примењена у првом моделу а односи се на директан интерфејс за приказ података на web-у као и у друга два модела а који се односе на приказ података на web-у користећи ThingSpeak као проху сервер и десктоп визуелизација помоћу Excel-а. Ова инфраструктура обухвата следеће компоненте: NodeMcu платформу, сензор dht11, сензор fc-22, breadboard.



Слика 1.9. Приказ хардверске инфраструктуре коришћене у сва три модела

На слици 1.9 се такође може видети модул за екстерно напајање који је конектован са леве стране breadboard-а, и служи за напајање сензора. Као што се може видети са слике 1.9 све наведене компоненте су повезане чинећи један систем. На слици 2 се може видети сензор dht11.



Слика 2. Приказ DHT11 сензора са означеним пиновима

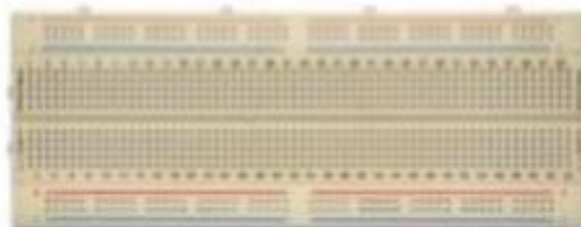
Сензор DHT11 представља сензор температуре и влажности ваздуха, у сва четири модела овај сензор је коришћен за прикупљање података о наведеним факторима из одређеног окружења. Користи се за прикупљање података у реалном времену, прикупљени подаци се могу користити одмах. Припада групи дигиталних сензора, дужина преноса сигнала је двадесет метара, што му даје идеалне карактеристике за имплементацију на теже доступним местима, свакако још једна од предности је мала величина као и ниска цена овог сензора. Један од недостатака је неподношење превисоких температура па се не може користити у свим гранама индустрије. Пин под редним бројем 1 користи се за напајање сензора од 3-5V, у овом моделу ће се користити екстерно напајање што може бити батерија или пуњач телефона. Пин 2 се користи за слање информација платформи где се касније

врши обрада на основу које се добијају подаци прикупљени овим сензором, пин 3 је празан пин и непотребно га је користити у овом моделу, пин 4 је GND пин или пин негативног напона такође ће се користити са екстерног напајања (Мијаиловић и сар., 2018). На слици 2.1 се може видети сензор fc-22 са означеним пиновима.



Слика 2.1. Приказ fc-22 сензора са означеним пиновима

Сензор fc-22 се користи за детекцију штетних гасова у ваздуху, представља један од многих сензора овог типа а који припадају групу индустријских сензора, самим тим примену проналазе у индустријским халама за праћење загађености ваздуха. Осетљив је на гасове: пару, гас, амонијак, сумпор, бензин, дим, алкохол и друге штетне гасове. Пин под редним бројем 1 се користи за напајање од 5V у сва четири модела овај сензор ће користити екстерно напајање. Пин 2 је GND пин или пин негативног напона такође ће се користити са екстерног напајања, пин 3 је дигитални пин D0, пин 4 је аналогни пин A0. Примена овог сензора се односи на прикупљање података о факторима унутар неке индустријске хале. Сакупљајући ове податке можемо доћи до података којих фактора у окружењу има превише наког чега можемо применити неку методу за смањење негативног утицаја одређених фактора по здравље запослених и уопште животне околине ван индустријске зоне. До појаве ових штетних фактора може доћи како кваром одеђене индустријске машине тако и непажњом послодаваца и запослених, такође могуће су и природне катастрофе које могу изазвати повећану количину одређених штетних гасова. Прикупљене информације овим сензором се шаљу до NodeMcu платформе где се врши њихова обрада на основу које добијамо податке о штетним гасовима у нашој околини, а којима се може манипулисати на различите начине, како чување у бази тако и директан приказ на неки од могућих начина (Мијаиловић и сар., 2018). На слици 2.2 се може видети breadboard.



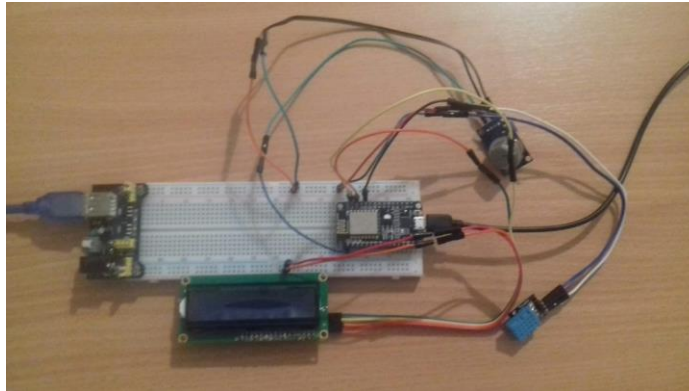
Слика 2.2. Breadboard

Breadboard се користи за брзо повезивање компоненти са NodeMcu платформом као и другим платформама. Предност коришћења breadboard-а поред брзине повезивања компонената је свакако брзина и искључења непотребних, све се ради без оштећења и може се радити бесконачно пута. Жице које се користе за повезивање NodeMcu платформе и других компонената се једноставно утисну у заглавља breadboard-а чиме се остварује

контакт и преноси се напон или подаци у зависности од потреба. Још једна добра карактеристика breadboard-а је свакако повећање броја заглавља у односу на број који је тренутно расположив на платформи и на тај начин се пружа могућност повезивања већег броја компонената. Нарочито када се ради о пиновима позитивног и негативног напона, с' обзиром да ових пинова има мало на платформи овим начином се повећавају њене могућности.

7.2. Хардверска инфраструктура другог модела

На слици 2.3 се може видети приказ хардверске инфраструктуре која представља једну од опција коју корисник може изабрати уколико му је потребан такав приказ података, док је за демонстрацију овог модела коришћена инфратруктура са слике 1.9. други модел је заснован на бази у Excel-у.



Слика 2.3. Приказ хардверске инфраструктуре проширеног другог модела

Као што се може видети са слике 2.3 проширени модел би садржао LCD модулу, на коме би се вршио приказ података добијених са сензорских система. За разлику од првог модела у овом моделу се за приказ података користи се база података заснована на Excel-у, овај модел је намењен за примену на локацијама на којима нема приступа бежичном интернету. За комуникацију са Excel-ом се користи локална конекција путем USB кабла.

Унапређење другог модела са LCD модулом

Као додатна могућност да се подаци приказују у реалном времену може се упоредо коористити LCD модулу пружајући кориснику опцију да бира да ли му је потребна база података или само приказ података као вид информације или обавештења о подацима прикупљеним сензорским системима у реалном времену. Поред LCD модула као опције коју корисник бира у другом моделу су коришћени и сензори DHT11, FC-22, breadboard ове компоненте су већ описане сходно томе у наставку текста ће бити представљен LCD модулу. На слици 2.4 се може видети приказ LCD модула.

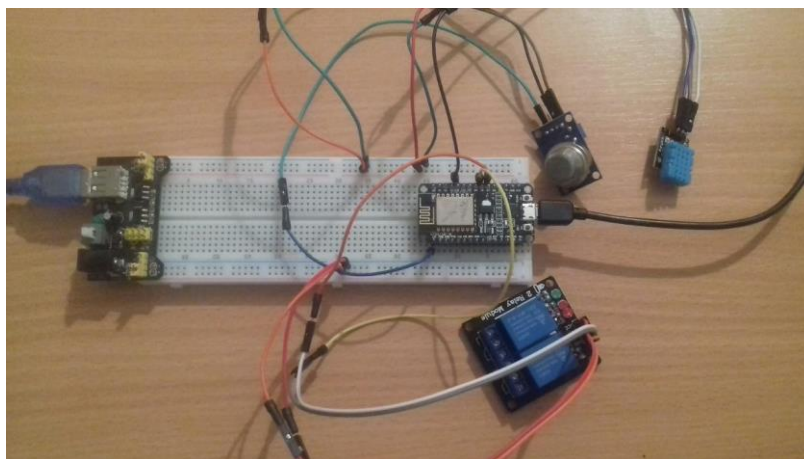


Слика 2.4. LCD модул са означеним пиновима

LCD модул који је приказан на слици 2.4 је 16X2 карактера што значи да је дужина речи која се може приказати у једном реду 16 карактера такође значи да овај LCD модул може исписати текст у 2 реда. Поред LCD модула се налази I2C серијски интерфејс модул који се користи за лакшу конекцију LCD модула са неком од платформи са којом се конектује, тачније за мањи број жица које је потребно конектовати са платформом. Један део пинова се конектује са LCD модулом док други део служи за конекцију са платформом. Наравно LCD модул је могуће конектовати без коришћења I2C серијског интерфејс модула али је тада потребно конектовати 16 пинова са платформом и на тај начин би се заузео велики број потребних пинова за неке друге модуле, уколико се користи I2C серијски интерфејс модул потребно је конектовати 4 пина од којих се 2 користе за напајање а који се не морају конектовати на платформу, што пружа огромну предност с' обзиром да се број потребних пинова смањује 4 пута. Пин под редним бројем 1 се користи за напајање I2C серијског интерфејс модула негативним напоном и може се користити екстерно напајање, пин 2 се користи за напајање позитивним напоном од 5VDC, пин 3 и 4 се користе за пренос података који ће бити исписани на LCD модулу.

7.3. Хардверска инфраструктура трећег модела

На слици 2.5. ће бити приказана хардверска инфраструктура која обухвата проширење модела 3 која представља једну од могућности коју корисник може изабрати у зависности од потреба начина контроле електричних уређаја. Биће демонстриран трећи модел који се заснива на ThingSpeak-у као прокси серверу док приказ слике 2.5. представља додатну могућност овог модела што пружају и ThingSpeak као и Virtuino. Док ће трећи модел користити хардверску инфраструктуру са слике 1.9.



Слика 2.5. Приказ хардверске инфраструктуре као додатне могућности трећег модела

Сходно томе да би ова проширена могућност коју корисник може изабрати у трећем моделу зависно од његових потреба обухватала Relay модул који се користио за контролу електричних уређаја, он ће бити представљен у наставку текста. С' да је у трећем моделу могућ приказ на ThingSpeak као и Virtuino платформи обе платформе нуде могућност контроле електричних уређаја. Према наведеном а сходно потребама ове опција на слици 2.6 ће бити представљен Relay модул.



Слика 2.6. Relay модул са означеним пиновима

Relay модул ће бити у овом моделу коришћен за контролу електричних уређаја. Главна сврха relay модула је да укључи и искључи електричну енергију, на тај начин омогућава два стања у коме се може наћи неки електрични уређај. На слици 2.6. је приказан relay модула који обухвата два relay-а наравно на тај начин могуће је контролисати два уређаја, постоје и модули који обухватају већи број relay модула, док је најмањи број relay-а на модулу један. Пин означен на слици VCC представља пин позитивног напона 5VDC, може се користити са екстерног напајања, GND пин представља пин негативног напона који се такође може користити са екстерног напајања, IN1 и IN2 представљају дигиталне пинове којима relay модул комуницира са платформом. Као и у случају сензора постоји више типова relay модула, у зависности од пројекта и његових захтева користиће се тип relay модула који најбоље одговара потребним карактеристикама. Процес укључивања и искључивања који обавља relay модул се може извршавати бесконачно као и у зависности од програма и задатих параметара. У овом моделу његова функција ће зависити од других параметара који ће бити прикупљени сензорским системима. У зависности од излазне снаге овог модула се могу контролисати електрични уређаји различитих снага као што су електромотори, грејачи, светло као и многи други електрични уређаји наизменичне и једносмерне струје.

Сходно томе да микроконтролерске платформе не могу конетролирати наизменичну струју и напон од 220VAC relay модул представља најбоље решење, с' обзиром да се са микроконтролерским платформама може комуницирати путем интернета као и управљати њима буквално можемо укључити или искључити одређени уређај са било које локације.

Карактеристике relay модула:

- Тип: дигитални модул,
- Снага струје: 10A(NO) 5A(NC),
- Максимални укупни напон: 150VAC/24VDC,
- Дигитални интерфејс,
- Контролни сигнал: TTL ниво,
- Дозвољено оптерећење: 8A 150V AC (NO) / 10A 24V DC (NO), 5A 250V AC (NO / NC) / 5A 24V DC (NO / NC),
- Максимално оптерећење: AC1200VA, DC240V (NO), AC 625VA, DC120V (NC),
- Време деловања: 10мс

Поред пинова користе за комуникацију са микроконтролерским платформама на слици су означени контакти који се користе за конекцију са уређајима који ће бити контролирани. Контакт означен COM представља заједничку везу или долазни напон, позитивни напон, док NC представља нормално затворену везу а NO нормално отворену везу. У зависности од тога да ли је уређај прикључен на NC или NO везу, релеј ће моћи да контролише уређаје различите снаге. NO је отворен контакт све док микроконтролер не каже другачије. За разлику од NO контакта NC је увек укључен или увек у контакту са извором најања све док микроконтролер не каже супротно. Према томе, ова два контакта су у различитим стањима на почетку пре програмирања микроконтролера.

8. Софтверска инфраструктура

У раду неће бити описани споредни делови кода, већ они главни. Са обзиром да се Arduino програмира у Wiring језику кога чине синтакса и библиотеке, који је у основи веома сличан програмском језику C++, сви кодови су писани у њему. Главни проблем је недостатак меморије на Arduino уно моделу, што значи да није могуће сместити на њега програм који заузима много простора. За захтевнија решења је потребно користити други модел Arduino. Ја сам користио три модела која ће бити описана у наставку текста.

8.1. Модел 1. Директан интерфејс за приказ података на web-у

Циљ првог модела је да прочитане податке са сензора страници којој можемо приступати са било ког компјутера или телефона и видети тренутне податке, очитане у реалном времену. Подаци су прикупљени сензорским системима, обрађени од стране NodeMcu платформе. Пре свега је неопходно повезати ову платформу на бежични интернет након чега је могуће слање података, тако у коду коришћене библиотеке за правилно прикупљање података dht сензором као и библиотеке за конекцију. За само слање података неопходно је отворити сервер као и дефинисање ssid и password на који се конектује NodeMcu платформа, дефинисан је пин на који је конектован dht сензором као и тип сензора. Приказ кода се може видети на слици 2.7.

```
11  const char* ssid      = "Mijailovic";
12  const char* password = "Mijailovic";
13
14  #define DHTPIN D4      // what pin we're connected to
15  #define DHTTYPE DHT11 // DHT 11
16  DHT dht(DHTPIN, DHTTYPE);
17
18  // Set web server port number to 80
19  WiFiServer server(80);
```

Слика 2.7. Дефинисање ssid-а и password-а

Отварамо порт и започињемо комуникацију. Захваљујући библиотеци која се користи приликом WiFi конекције није потребно задавање IP, subnet, gateway адресе већ се конекција врши на основу унетих ssid-а и password-а од којих се преузимају адресе. У сериал монитору се врши испис IP адресе коју је потребно унети у web прегледач како бих се приступило подацима. На слици 2.8. се може видети приказ кода.

```

27 void setup() {
28     Serial.begin(115200);
29
30     // Connect to Wi-Fi network with SSID and password
31     Serial.print("Connecting to ");
32     Serial.println(ssid);
33     WiFi.begin(ssid, password);
34     while (WiFi.status() != WL_CONNECTED) {
35         delay(500);
36         Serial.print(".");
37     }
38     // Print local IP address and start web server
39     Serial.println("");
40     Serial.println("WiFi connected.");
41     Serial.println("IP address: ");
42     Serial.println(WiFi.localIP());
43     server.begin();
44 }

```

Слика 2.8. Функција Void setup

Функција void loop се извршава све док је клијент повезан са сервером. Клијент започиње слушање сервера. Подаци се приказују у HTML облику, па клијет мора да посматра податке користећи web browser. Приказ кода се може видети на сликама 2.9 и 3.

```

46 void loop(){
47     WiFiClient client = server.available(); // Listen for incoming clients
48
49     if (client) { // If a new client connects,
50         Serial.println("New Client."); // print a message out in the serial port
51         String currentLine = ""; // make a String to hold incoming data from the client
52         while (client.connected()) { // loop while the client's connected
53             if (client.available()) { // if there's bytes to read from the client,
54                 char c = client.read(); // read a byte, then
55                 Serial.write(c); // print it out the serial monitor
56                 header += c;

```

Слика 2.9. Функција void loop

Након започетог слушања сервера клијент може видети податке који ће бити представљени у наставку описа кода.

```

57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77

```

```

if (c == '\n') { // if the byte is a newline character
// if the current line is blank, you got two newline characters in a row.
// that's the end of the client HTTP request, so send a response:
if (currentLine.length() == 0) {
// HTTP headers always start with a response code (e.g. HTTP/1.1 200 OK)
// and a content-type so the client knows what's coming, then a blank line:
client.println("HTTP/1.1 200 OK");
client.println("Content-type:text/html");
client.println("Connection: close");
client.println();

// Display the HTML web page
client.println("<!DOCTYPE HTML>");
client.println("<html>");
client.println("<meta http-equiv=\"refresh\" content=\"5\">");

// prikazi vrednosti
client.println("<h1>");
client.println("<body style=background:#80BFFF>");
client.print("<center>Trenutne vrednosti</center>");
client.println("<br />");

```

Слика 3. Пролазак кроз if услов

Код који је представљен сликом 3 омогућава дефинисање наслова HTML странице, додељивање боје позадине странице.

Дефинисање променљивих-вредности које ће се прикупљати сензорима. Неопходно како би било могуће приступити тим подацима, као и сензорима. Дефинисање пина за сензор fc-22. Приказ кода се може видети на слици 3.1.

```

79      float h = dht.readHumidity();
80      float t = dht.readTemperature();
81
82      int val;
83      val = analogRead(0);
84

```

Слика 3.1. Дефинисање променљивих

Овај део кода служи за приказ температуре, влажности ваздуха, температуре у келвинима, температуре кондезације, убрзана кондезација, загађености ваздуха. Променљива val представља вредност загађености ваздуха. Ако је та вредност већа од 350 обавестиће клијента о повећаној количини загађености ваздуха, обавештење у виду поруке која се исписује на web страници. Приказ кода се може видети на сликама 3.2, 3.3 и 3.4.

```

85      client.print("Temperatura (C): ");
86      client.println(t, 1);
87      client.println("<br />");
88
89      client.print("Temperatura (F): ");
90      client.println(Fahrenheit(t), 1);
91      client.println("<br />");
92
93      client.print("Vlaznost vazduha (%): ");
94      client.println(h, 0);
95      client.println("<br />");
96
97      client.print("Temperatura (K): ");
98      client.println(Kelvin(t), 1);
99      client.println("<br />");
100
101      client.print("Temperatura kondezacije (C): ");
102      client.println(dewPoint(t, h));
103      client.println("<br />");
104
105      client.print("Ubrzana Kondezacija (C): ");
106      client.println(dewPointFast(t, h));
107      client.println("<br />");
108      client.print("Zagadjenost vazduha (PPM): ");
109      if(val > 350){
110          client.println("<br />");
111          client.println("<p style = color:red>Povecana kolicina zagadjenosti vazduha</p>");
112      }

```

Слика 3.2. Испис вредности

Вредности добијене сензорским системима се исписују на начин представљен сликом 3.2. На основу добијене температуре помоћу сензора dht11 иста је претворена у келвине, фаренхајте.

```

116 // The HTTP response ends with another blank line
117 client.println();
118 // Break out of the while loop
119 break;
120 } else { // if you got a newline, then clear currentLine
121     currentLine = "";
122 }
123 } else if (c != '\r') { // if you got anything else but a carriage return character,
124     currentLine += c;    // add it to the end of the currentLine
125 }
126 }
127 }
128 // Clear the header variable
129 header = "";
130 // Close the connection
131 delay(1);
132 client.stop();
133 Serial.println("Client disconnected.");
134 Serial.println("");
135 }
136 }

```

Слика 3.3. Испис вредности у наредној линији

Приказ кода на слици 3.3. омогућава након исписа текста и вредности добијених сензорским системом прелазак у наредни ред и испис наредних вредности као и обавештење кориснику уколико је дошло до прекида конекције.

```

140 //Celsius to Fahrenheit
141 double Fahrenheit(double celsius)
142 {
143     return 1.8 * celsius + 32;
144 }
145
146 //Celsius to Kelvin
147 double Kelvin(double celsius)
148 {
149     return celsius + 273.15;
150 }

```

Слика 3.4. Претварање вредности

Након добијених података у целзијусима исти се претварају на начин дефинисан на слици 3.4. у келвине и фаренхајте.

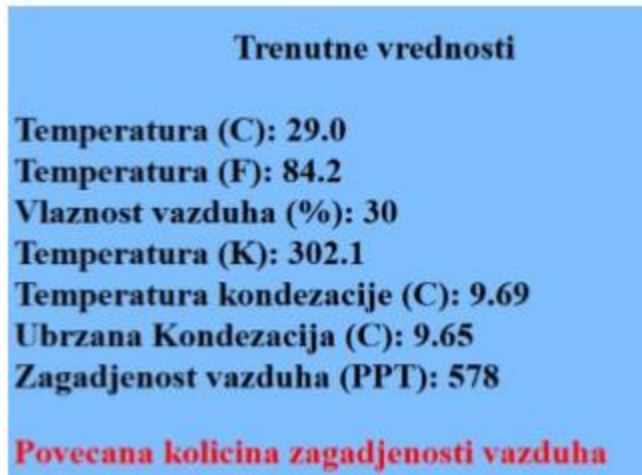


Trenutne vrednosti

- Temperatura (C): 29.0
- Temperatura (F): 84.2
- Vlaznost vazduha (%): 30
- Temperatura (K): 302.1
- Temperatura kondenzacije (C): 9.69
- Ubrzana Kondenzacija (C): 9.65
- Zagadjenost vazduha (PPT): 329

Слика 3.5. Приказ података на интернету

На слици 3.5 се може видети приказ података на “web” страници, време освежавања података је јесна секунда.

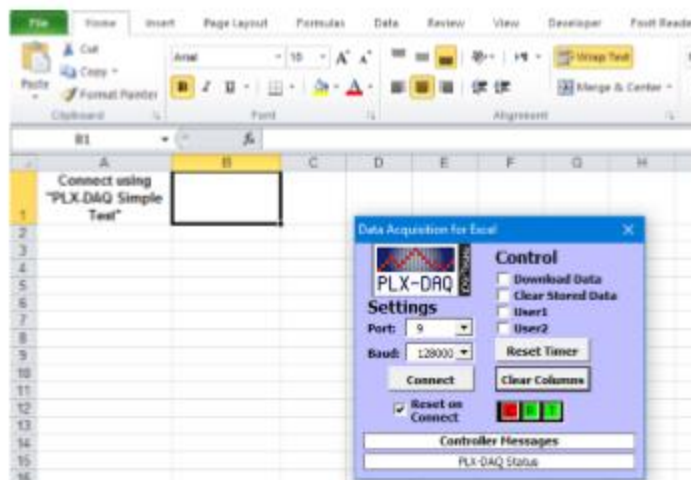


3.6. Приказ података на интернету са поруком

Циљ првог модела је поред приказа података на интернету и обавештење корисника о прекорачењу дозвољених граница истих. Ово обавештење се реализује у облику поруке која се појављује на истој страници. Приказ поруке о повећаној количини загађености ваздуха се може видети на слици 3.6.

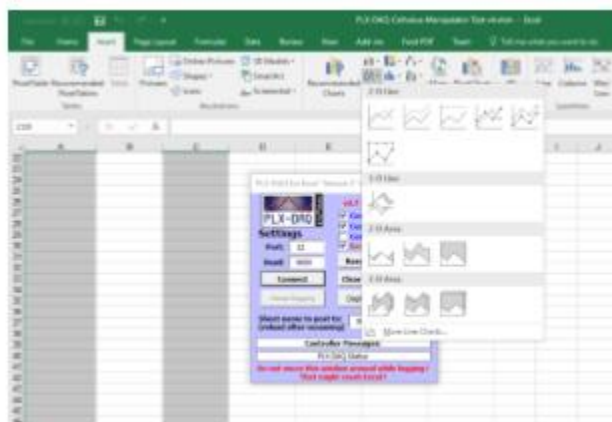
8.2. Модел 2. Десктоп визуелизација помоћу Excel-а

Идеја другог модела се заснива на томе да ако корисник система нема приступ интернету на одређеној локацији где се налази платформа а са које се прикупљају подаци, те податке може чувати у локалној бази и касније их користити и манипулисати њима на жељени начин. Решење ове локалне базе сам пронашао у PLX-DAQ, PLX-DAQ је једноставан додатак за Microsoft Excel. Овај додатак омогућава NodeMcu платформи да прикупљене податке са разних врста сензора уписује у колону Excel-а, даље се ти подаци могу користити за дијаграме или се једноставно могу чувати. На овај начин могу све врсте микроконтролера да комуницирају са Excel-ом и уписивати податке. За овај модел неопходна је USB конекција са компјутером како би се подаци преносили. Уједно ова карактеристика је и мана овог модела а и предност, мана се огледа у томе да је неопходна конекција са компјутером што значи да се платформа мора налазити у близини неког компјутера, предност се огледа у могућности смештања огромне количине података с' обзиром да компјутери који се данас користе поседују доста меморијског простора. С' друге стране посматрано није нужно да то буде десктоп компјутер може бити лаптоп који заузима мање физичког простора или Raspberry PI који представља мини рачунар. Физички је врло сличан Arduino платформи с' тим што на себи поседује микропроцесор а не микроконтролер што га чини компјутером, поседује сопствени оперативни систем. Пре саме употребе PLX-DAQ потребно га је преузети са интернета и инсталирати. Након тога неопходно је покренути PLX-DAQ, прозор који се добије након покретања, може се видети на слици 3.7.



Слика 3.7. Приказ прозора PLX-DAQ

Када се покрене PLX-DAQ битно је да се подесе Port као и Baud, користимо исти порт који смо користили у коду када смо програмирали NodeMcu платформу, битно је да буду оба поља попуњена исправно како би могао да се повеже NodeMcu платформом и пренеси податке. Након исправно попуњених поља потребно је кликнути на Connect. Clear Columns нам служи ако хоћемо да обришемо вредности из тренутно попуњених колона и да почнемо поново да их попуњавамо. Након конектовања и попуњених колона можемо одабрати неки од графика. То радимо тако што означимо колону или колоне на основу које хоћемо да нацртамо график и из менија Insert можемо изабрати график. Можемо истовремено учитати више истих или различитих графика за исте или различите колоне, а можемо и истовремено користити више колона и приказати комплекснији график. Начин на који се то ради је представљен на слици 3.8.



Слика 3.8. Приказ графика

Објашњење кода коришћеног у овом моделу, неопход је коришћење библиотека за сензоре FC-22 као и DHT11. Дефинисање пинова за сензор DHT11 као и дефинисање типа сензора. С' обзиром да се у овом моделу може применити LCD модул који је представљен у инфраструктури, а да не представља суштинско решење опис кода за LCD неће бити узет

у обзир и неће бити представљен. Код који је коришћен у другом моделу ће бити описан у наставку текста. На слици 3.9 се може видети описани код.

```
10 #define DHTPIN D4      // what pin we're connected to
11 #define DHTTYPE DHT11  // DHT 11
12 DHT dht(DHTPIN, DHTTYPE);
```

Слика 3.9. Дефинисање пинова

На слици 3.9 је представљено дефинисање пинова за сензор dht11 као и типа сензора. У функцији void setup постављамо вредност Baud на порту 9600 њега користимо у PLX-DAQ за конектовање. CLEARDATA омогућава касније брисање података из колоне. LABEL представља текст који ће бити исписан у првим колонама у PLX-DAQ. Приказ кода се може видети на слици 4.

```
16 void setup() {
17
18     Serial.begin(9600);
19     Serial.println("CLEARDATA");
20     Serial.println("LABEL,Vreme,Temperatura,Vlaznost,Zagadjenost");
21 }
```

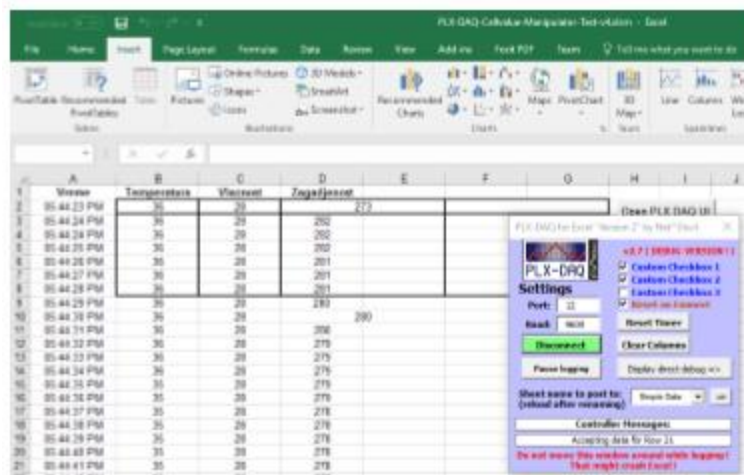
Слика 4. Функција void setup

У функцији void loop исчитавамо податке са сензора и исписујемо их, поред вредности са сензора биће исписано и време. NodeMcu ће податке читати сваке секунде и уписивати у PLXDAQ то ће се понављати до тренутка када ми то желимо. Податке можемо користити, обрађивати или једноставно чувати. Исчитавање се врши у реалном времену а сачуване податке можемо погледати увек и користити. Приказ кода се може видети на слици 4.1.

```
24 void loop()
25 {
26     int h = dht.readHumidity();
27     int t = dht.readTemperature();
28     int val;
29     val = analogRead(0);
30     Serial.print("DATA,TIME");
31     Serial.print(",");
32     //Serial.print("Temperature = ");
33     Serial.print(t);
34     //Serial.print("Humidity = ");
35     Serial.print(",");
36     Serial.print(h);
37     //Serial.print("senzor = ");
38     Serial.print(",");
39     Serial.println(val,DEC);
40     delay(1000);
```

Слика 4.1. Функцији void loop

Приказ добијен приликом покретања PLX-DAQ и NodeMcu платформе се може видети на слици 4.2.



Слика 4.2 Приказане вредности у PLX-DAQ

Као што се може видети са слике 4.2. вредности се уписују у ћелије унутар Excel документа, поред вредности се могу приказати графици. За сам приказ графика неопходно је селектовати колону или колоне уколико желимо добити сложенији график, након чега се из insert менија бира жељени изглед графика. График се исцртава у реалном времену, паралелно са вредностима које се уписују у колоне сходно томе могуће је пратити и вредности које се уписују у колоне као и само исцртавање графика. Унете вредности се касније могу сачувати и користити за даљу обраду и манипулацију истих. Приказ графика се може видети на слици 4.3.



Слика 4.3. Графици

За демонстацију графика који су приказани на слици 4.3. су коришћене вредности које се односе на температуру, влажност ваздуха као и загађеност ваздуха. Сам избор модела графика зависи од корисника.

8.3. Модел 3. Приказ података на web-у користећи ThingSpeak као proxy сервер

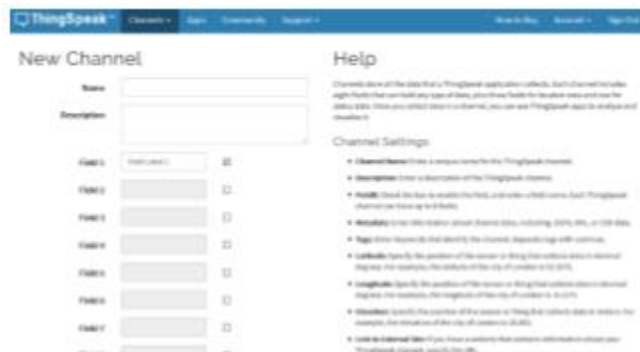
Идеја модела 3 је постављање података у базу којој можемо приступити било када са било које локације. За ово решење је потребан приступ интернету. Користио сам ThingSpeak како бих чувао податке и на основу њих креирао графике. ThingSpeak је „IoT“ аналитичка платформа која се користи за чување и преузимање података преко HTTP протокола или преко локалне мреже, омогућава сакупљање, визуелизацију и анализирање

података у реалном времену. Пружа тренутну визуализацију података које уређаји постављају на интернет. Подацима се може приступати преко компјутера као и преко телефона са Android оперативним системом. Да бих смо чували податке потребно је направити налог на ThingSpeak-у. На слици 4.4. може се видети креирање налога.



Слика 4.4. ThingSpeak налог

Након креирања налога који је представљен сликом 4.4. потребно је креирати channel, потребан нам је channel id као и api key како бих смо могли да прослеђујемо податке. Креирање channel-а се може видети на слици 4.5.



Слика 4.5. Креирање Channel

На једном налогу могуће је креирати више Channel-а поступак је исти као и на слици 4.5. Објашњење коришћеног кода на NodeMcu се може видети на сликама 4.6 и 4.7.

```

15  const char* ssid      = "Mijailovic";
16  const char* password = "Mijailovic";
17
18  #define DHTPIN D4      // what pin we're connected to
19  #define DHTTYPE DHT11 // DHT 11
20  DHT dht(DHTPIN, DHTTYPE);
21
22  byte mac[] = {
23      0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED };
24  WiFiClient client;
25
26  char thingSpeakAddress[] = "api.thingspeak.com";
27  unsigned long mySensorsChannelNumber = 782073;
28  const char * myWriteAPIKey_sensors = "BR00P2JE2J5I38ZE";
29  const int updateThingSpeakInterval = 16 * 1000;
30
31  long lastWriteTime=0;
32  long lastReadTime=0;
33  float responseValue=0;

```

Слика 4.6. Дефинисање ssid-a и password-a

Као што се може видети на слици 4.6. потребно је дефинисати ssid као и password на који ће се ова платформа конектовати приликом коришћења бежичног интернета, постављамо клијента и уносимо свој број канала mySensorsChannelNumber као и myWriteAPIKey_sensors које смо добили када смо креирали нови канал. Дефинишу се пинови потребни за сензор DHT11 као и тип сензора.

```

37  void setup() {
38      Serial.begin(9600);
39      //WiFi.begin(mac, ip);
40      ThingSpeak.begin(client);
41      //dht11.begin();
42      pinMode(4, OUTPUT);
43      Serial.println("Setup completed");
44      delay(1000);
45  }

```

Слика 4.7 Функција void setup

У оквиру функције void setup а која је приказана на слици 4.7. започиње ThingSpeak започиње слушање клијента, поступак се понавља након једне секунде. Функција void loop у којој исчитавамо вредности са сензора, и на основу њих креирамо график на ThingSpeak-у. Време слања података на основу којих ће се исцртавати графици је једна секунда. Креирана су три графика за температуру, влажност и загађеност ваздуха. Приказ кода коришћеног за креирање графика на ThingSpeak-у се може видети на сликама 4.8., 4.9. и 5.

```

48 void loop() {
49     int val;
50     val = analogRead(0);
51     Serial.print(val, DEC);
52     ThingSpeak.setField(4, analogRead(0));
53     delay(1000);

```

Слика 4.8. Функција void loop

У оквиру кода приказаног на слици 4.8. дефинише се пин аналогног сензора и исписује се вредност добије са овим сензором време исписа је једна секунда.

```

56 if (millis()-lastWriteTime>1000) { // every 1 secs.
57     Serial.println("Read sensors' values...");
58     float h = dht.readHumidity();
59     float t = dht.readTemperature();
60     if (isnan(t) || isnan(h)) {
61         Serial.println("Failed to read from DHT");
62     }

```

Слика 4.9. Унос вредности у график

Кодом обухваћеним сликом 4.9 се врши дефинисање променљивих сензора dht11.

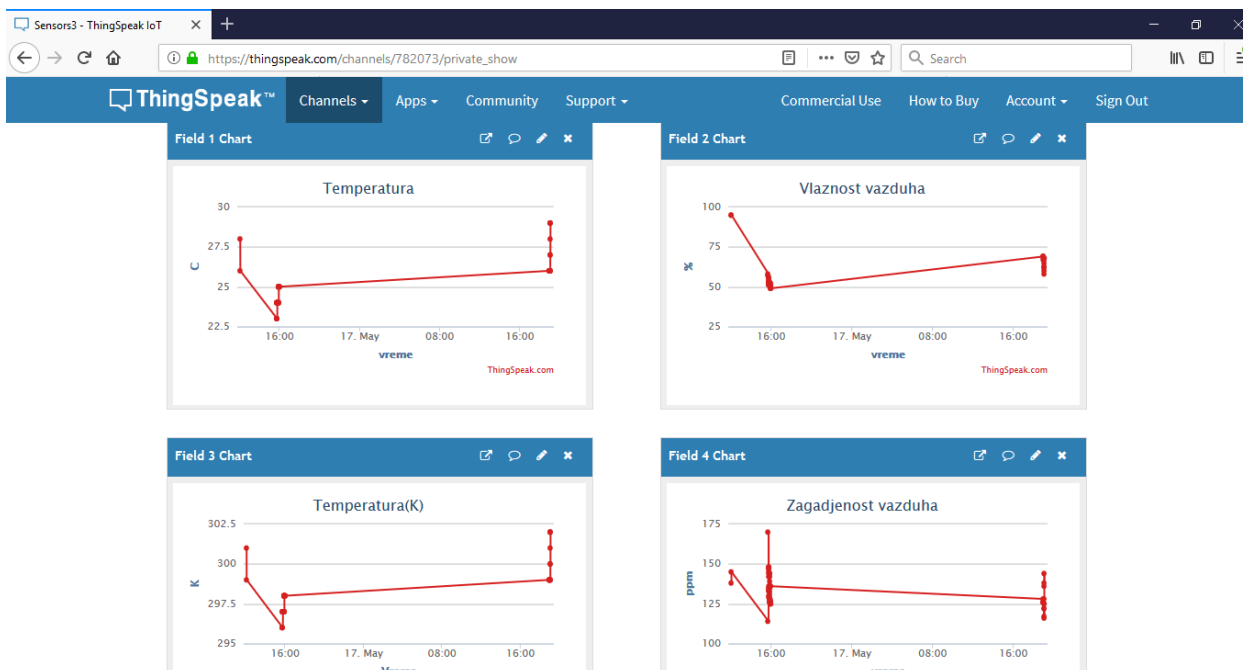
```

63 else {
64     Serial.println("Temp="+String(t)+" *C");
65     Serial.println("Humidity="+String(h)+" %");
66     ThingSpeak.setField(1,t);
67     ThingSpeak.setField(2,h);
68     ThingSpeak.setField(3,int(Kelvin(t))); // This is the value of commands channel
69     ThingSpeak.setField(6,int(dewPoint(t,h)));
70     ThingSpeak.setField(7,int(dewPointFast(t,h)));
71     ThingSpeak.setField(5,int(Fahrenheit(t)));
72     ThingSpeak.writeFields(mySensorsChannelNumber, myWriteAPIKey_sensors);
73     lastWriteTime=millis(); // store last write time
74 }
75 }
76

```

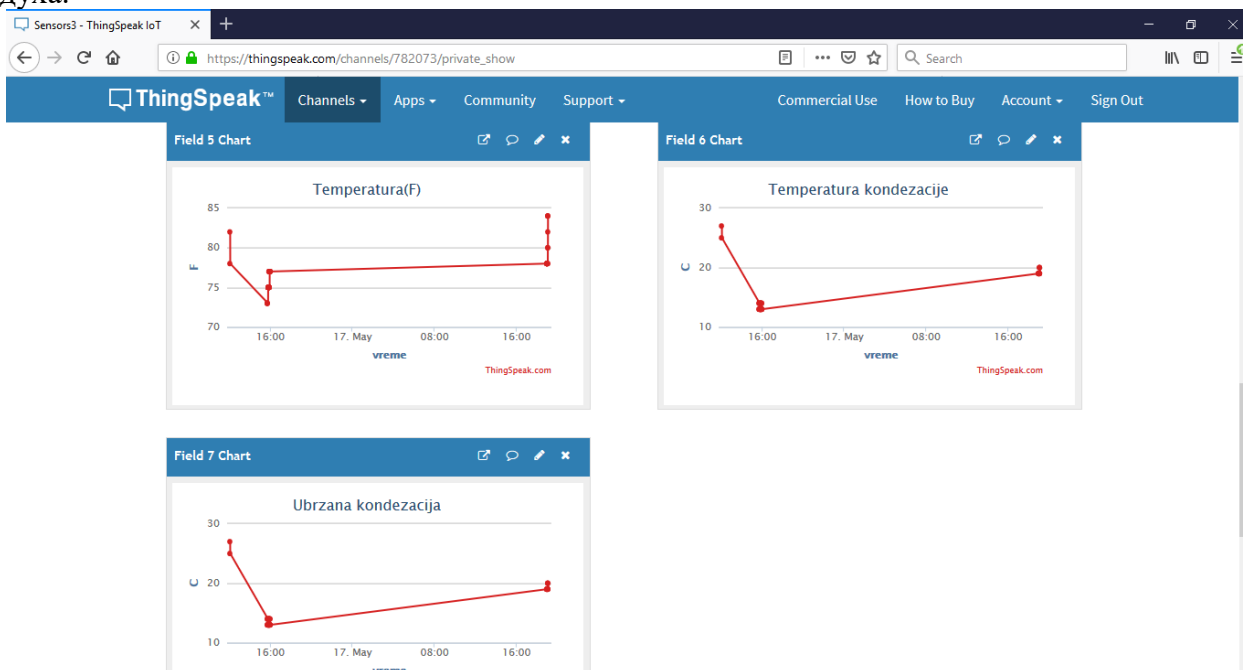
Слика 5. Унос вредности у график

Слика 5. представља унос вредности у графике, дефинише се у који график ће бити унета одређена вредност. На слици 5.1 и 5.2. се могу видети графици добијени на основу прочитаних вредности са сензора, графици се исцртавају у реалном времену, свака вредност се уноси једну секунду након предходне уписане.



Слика 5.1. Графици приказани на ThingSpeak-у

На слици 5.1. се могу видети графици приказани на ThingSpeak-у а који се односе на температуру у целзијусима, влажност ваздуха, температура у келвинима, загађеност ваздуха.



Слика 5.2. Графици приказани на ThingSpeak-у

На слици 5.2. се могу видети графици приказани на ThingSpeak-у а који се односе на температуру у фарењајтима, температура кондензације, убрзана кондензација.

Добра страна модела 3 је што постоји Virtuino платформа коју можемо инасталирати на телефонима. Са њом приступамо ThingSpeak-у и преузимамо податке, на основу тих података креирамо графике на својим телефонима и буквално можемо да приступимо подацима где год да се налазимо. За сам приступ подацима и за повезивање са ThingSpeak-ом је потребна интернет конекција. Добра страна ове платформе је поред тога што можемо видети тренутне податке можемо и бирати време и датум из базе и приказати график за тај датум. То нам омогућава база података у овом случају то је ThingSpeak. Поред могућности да видимо податке, помоћу ове платформе можемо контролисати своју NodeMcu платформу, а можемо га контролисати преко интернета, смс поруке, bluetooth, поред ових могућности ова платформа нуди и многе друге. Пре свега је потребно инсталирати на телефон. Након инсталирања када покренемо Virtuino потребно је креирати сервер коме ће приступати и преузимати податке у овом случају приликом креирања морамо подесити Channel ID као и Read Key те податке смо добили када смо креирали ThingSpeak Channel. Подешавања се могу видети на слици 5.3.

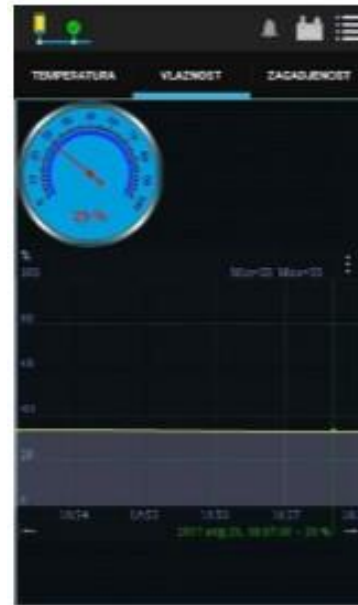


Слика 5.3. Повезивање на сервер

После повезивања на сервер а које је приказано на слици 5.3. потребно је изабрати график, и подесити са ког графика преузима податке када користи ThingSpeak уколико има више графика. Такође можемо и додати приказ података на коме ћемо приказивати тренутне вредности, постоји више различитих начина да прикажемо ове вредности па можемо сами изабрати један од понуђених. Такође можемо постављати услове ако неке вредности прекораче одређену границу да нас Virtuino обавести о тој промени, то такође можемо урадити на више начина. Приказ графика са вредностима се може видети на сликама: 5.4., 5.5. и 5.6.

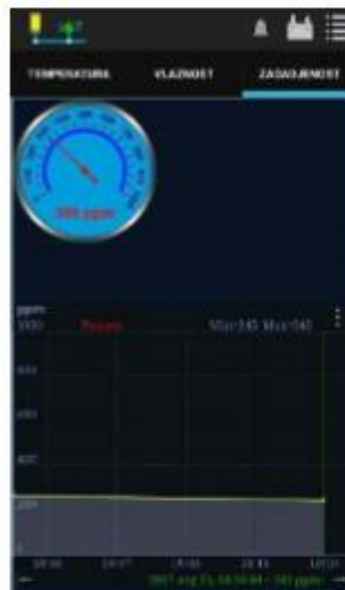


Слика 5.4. График температуре



Слика 5.5 График влажности

Приказ графика са вредностима који се може видети на сликама: 5.4. представља график температуре и 5.5. представља график влажности ваздуха и слика.



Слика 5.6. График загађености ваздуха

На слици 5.6. се може висети приказ графика загађености ваздуха добијен на основу података преузетих са ThingSpeak-a а који се односе на сензор fc-22.

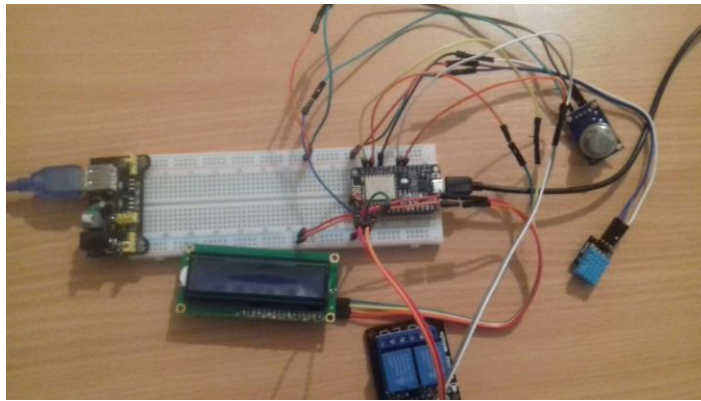
Као што је већ наведено у моделу 3 постоји Virtuino платформа коју можемо користити за контролу микроконтролерских платформи самим тим осталих уређаја и модула који се налазе под употребом NodeMcu платформе. За само управљање коришћењем ове платформе није потребно-нужно креирање базе на ThingSpeak-у и коришћење истог као сервера, могуће је креирати сервер коме ће мо приступати директно са Virtuino платформом и на тај

начин контролисати уређај. Значај овог модела се огледа у тренутној визуелизацији података на интернету коришћењем Virtuino платформе, управљање уређајима са било које локације. Предност коришћења ове платформе огледа се у могућности контроле коришћењем мобилних телефона са Android оперативним системом, што буквално обезбеђује контролу било када са било које локације будући да се мобилни телефон налази код корисника увек. За овај модел неопходна је конекција на интернет. Такође пружа могућност креирања графика на основу података преузетих са сензора, приказ истих на више начина као и креирање локалне базе која се чува у меморији телефона. Као што је већ речено у моделу 3 пре свега је неопходна инсталација Virtuino платформе, након чега се конектује на сервер са кога ће преузимати податке и приступати NodeMcu платформи.

Након повезивања на сервер добијамо могућност приступа NodeMcu платформи, што нам омогућава приказ података са сервера. За сам приказ неопходно је изабрати неки од понуђених начина. Изабрати пин на коме је конектован сензор-и. Потребно је додати дугме путем кога ће се вршити контрола електричних уређаја. Након додавања дугмета неопходно је изабрати пин на коме је повезан Relay модул-и. Кликом на додато дугме укључујемо или искључујемо одређени уређај који је конектован на Relay модул било да се ради о уређењу наизменичног или једносмерног напона. Пружајући на овај начин правремену реакцију и у жељеном тренутку.

9. Унапређење система заснованог на сензорским системима и NodeMcu платформи

С' обзиром да је у моделима наведена могућност проширења истих са одређеним функционалностима у наставку текста ће бити представљена хардверска инфраструктура која би обухватала сва већ поменуте могућности. Овај модел обухватао би NodeMcu платформу, сензор DHT11, сензор FC-22, breadboard, LCD modul, relay modul. Циљ овог модела би се заснивао на контроли електричних уређаја као што је већ описано у хардверској инфраструктури трећег модела као и приказ података на LCD modul-у што је описано у хардверској инфраструктури другог модела. Дакле у зависности од места уградње као и његове примене одређени систем ће обухватати потребне компоненте. Сходно томе да ова хардверска инфраструктура није примењена кроз три модела на којима се заснива рад овде ће бити представљена као избор корисника према томе неће бити описан код овог решења.



Слика 5. Могућност унапређења система

Такође сам систем се може проширивати одређеним сензорима што зависи од потреба пројекта. Разлог заснивања рада на три модела је поред различитог места и начина примене и недостатак меморије на NodeMcu платформи тако да није могуће објединити сва три модела у један, могућност постоји додавањем microSD картице за њу је неопходан модул који додатно повећава број кокришћених компоненти као и цену пројекта. Такође за додатне могућности које су описане у раду потребно је више меморије сходно томе неопходно је додавање модула за microSD картицу.

10. Закључак

У раду је представљена NodeMcu платформа као и Arduino платформа, наведена је суштинска разлика. Примењена су три модела за представљање сензорских система заснованог на NodeMcu платформи, кроз три модела која су представљена у овом раду објашњени су начини како се подаци могу сакупљати из нашег окружења и како се могу чувати како бих се касније примењивали. Кроз први модел представљен је како начин прикупљања података тако и приказ на web-у самим тим и приступ подацима са било које локације. Други модел представља прикупљања података на местима где није могућ приступ интернету што је нарочито значајно уколико се врши нека врста истраживања, поред бројева који представљају вредности података можемо добити и графике исцртане у реалном времену. Трећи модел пружа могућност коришћења проху сервер-а, самим тим чување података у бази којој можемо приступити било када и са било које локације за овај модел потребан је приступ интернету такође овај модел пружа могућност прегледа графика. Као додатна могућност трећег модела је virtuino платформа која обезбеђује приступ телефонима са android системима. Кориснику је на овај начин користећи virtuino омогућен приступ било када и било где, за овај модел је потребан приступ интернету, поред наведене предности овај модел обезбеђује приступ подацима из базе и пружа могућност одабира временског периода за који ће подаци бити приказани на основу означеног опсега исцртавају се графици. Подаци прикупљени на тај начин се могу користити било где. Основни недостатак система који је представљен у овом раду је проблем са мањом количином меморије на коришћеном NodeMcu моделу, због тог проблема није могуће написати програм који би обухватао сва три модела него је то морало бити показано на овај начин који је коришћен у раду. Добра страна пројекта је свакако цена компоненти која је ниска, и са малим средствима можемо направити сопствени систем и пратити факторе. Пројекат се може проширивати са другим врстама сензора, за шта је свакако потребан већи капацитет меморије, што се може обезбедити применом модула за microsd картицу или одабиром друге платформе као и комбинација две или више платформи. Фактори из одређеног окружења који су прикупљени сензорским системима имају значајан утицај на здравље људи и уопште животне средине. Системе које можемо сами креирати и користити нам могу омогућити како бољи увид тако и правовремене информације које можемо искористити како бих смо смањили утицај ових фактора и заштитили здравље људи. Са развојем уграђених система и сензорских система као и микроконтролера и технологије уопште имамо могућност да ове физички мале системе можемо уградити у многе друге системе што нам пружа веће могућности бољи увид брже деловање самим тим и здравији живот људи. Наизглед мала улога коју ови системи могу пружити али ипак веома значајна. Примена коју имају микроконтролери и платформе као што су NodeMcu, Arduino па и Rosberry PI је јако велика и могу се користити за неке друге системе наравно уз коришћење других компоненти, сензора. Развој технологије ће свакако снижити цене постојећим платформама и омогућити већи приступ новим са још бољим перформансама и још већим могућностима.

11. Литература

1. Al Dahoud, Ali & Fezari, Mohamed, (2018). NodeMCU V3 For Fast IoT Application Development. [Интернет] Доступно на: https://www.researchgate.net/publication/328265730_NodeMCU_V3_For_Fast_IoT_Application_Development [Пристапљено: 15.05.2019].
2. Breuer David, (2014), Monitoring ambient air quality for health impact assessment, WHO, Copenhagen.
3. Гржетић Иван, (2010), Суспендоване и распирабилне честице у урбаним срединама, Хемијски факултет, Београд. [Интернет] Доступно на: [https://www.chem.bg.ac.rs/~grzetic/predavanja/Osnovi%20hemije%20atmosfere%20i%20zagadjivaci%20vazduha/SU SPENDOVSANE%20I%20RESPIRABILNE%20CESTICE%20U%20URBANIM%20SR EDINAMA.pdf](https://www.chem.bg.ac.rs/~grzetic/predavanja/Osnovi%20hemije%20atmosfere%20i%20zagadjivaci%20vazduha/SU%20SPENDOVSANE%20I%20RESPIRABILNE%20CESTICE%20U%20URBANIM%20SR%20EDINAMA.pdf) [Пристапљено: 12.05.2019].
4. Doukas Charalampos, (2012), Building Internet of Things with the ARDUINO, CreateSpace Independent Publishing Platform.
5. Електронски факултет, Ниш, (2005), [Интернет] Доступно на: <http://es.elfak.ni.ac.rs/es/Materijal/Pog.1-Uvod-definicija-ES.pdf> [Пристапљено: 13.06.2018].
6. International labour office Geneva, Ambient factors in the workplace, (2001). [Интернет] Доступно на: http://www.ilo.org/wcmsp5/groups/public/---ed_protect/---protrav/--safework/documents/normativeinstrument/wcms_107729.pdf [Пристапљено: 13.05.2019].
7. Jazizadeh Faroukh, Kavulya Geoffrey, Klein Laura, Becerik-Gerber Burcin, (2011), Continuous sensing of occupant perception of indoor ambient factors, In Computing in Civil Engineering, pp. 161-168.
8. Massimo Banzi, Michael Shiloh, (2014), Getting Started with Arduino, Maker Media, United States of America.
9. Мијаиловић Ђорђе, Карабашевић Дарјан, Станујкић Драгиша, (2018), Развој система за праћење амбијенталних фактора применом “arduino” платформе, Трендови у пословању, пп. 37-46.
10. Noah Yusuf, Steve Metiboba, (2012), Work environment and job attitude among employees in a Nigerian work organization, Journal of Sustainable Society, 1(2), pp. 36-43.
11. Раденковић Божић, Деспотовић-Зракић Маријана, Богдановић Зорица, Бараћ Душан, Лабус Александра, Бајовић Живко, Факултет организационих наука, Београд, 2017. [Интернет] Доступно на: <http://www.elab.rs/udzbenik-internet-inteligentnihuredaja/senzorske-mreze/> [Пристапљено: 18.05.2019].

12. Rouse Margaret, (2015), Embedded system, [Интернет] Доступно на:
<http://internetofthingsagenda.techtarget.com/definition/embedded-system>
[Пристапљено: 14.04.2019].
13. Sandhu Robin, (2016), What is Arduino, [Интернет]Доступно на:
<https://www.lifewire.com/what-is-arduino-2495652> [Пристапљено: 15.05.2019].
14. Sagha Zadeh Rana, Shepley Mardele, Owora Artur Hamie, Dannenbaum Marta, Waggener, L. T., Chung, S. S. E., (2018), The Importance of Specific Workplace Environment Characteristics for Maximum Health and Performance, Journal of occupational and environmental medicine, 60(5), pp. e245-e252.

Интернет садржај:

15. <https://www.theengineeringprojects.com/2018/10/introduction-to-nodemcu-v3.html>
[Пристапљено: 13.05.2019].