

University for Business and Technology in Kosovo

UBT Knowledge Center

Theses and Dissertations

Student Work

Spring 6-2021

LIDHJA SERIKE E ARDUINOVE

Nebi Qerkini

University for Business and Technology - UBT

Follow this and additional works at: <https://knowledgecenter.ubt-uni.net/etd>



Part of the [Computer Sciences Commons](#)

Recommended Citation

Qerkini, Nebi, "LIDHJA SERIKE E ARDUINOVE" (2021). *Theses and Dissertations*. 2694.
<https://knowledgecenter.ubt-uni.net/etd/2694>

This Thesis is brought to you for free and open access by the Student Work at UBT Knowledge Center. It has been accepted for inclusion in Theses and Dissertations by an authorized administrator of UBT Knowledge Center. For more information, please contact knowledge.center@ubt-uni.net.



Programi për Shkenca Kompjuterike dhe Inxhinierise

LIDHJA SERIKE E ARDUINOVE
Shkalla Bachelor

Nebi Qerkini

Qershor / 2021
Prishtinë



Programi për Shkenca Kompjuterike dhe Inxhinierise

Punim Diplome
Viti akademik 2020 – 2021

Nebi Qerkini

LIDHJA SERIKE E ARDUINOVE

Mentor: Dr. Bertan Karahoda

Qershor / 2021

Ky punim është përpiluar dhe dorëzuar në përmbushjen e kërkesave të
pjeshme për Shkallën Bachelor

ABSTRAKT

Në këtë punim shtjellohet lidhja e Arduinove në seri. Ato lidhen nëpërmjet portit serik, ku edhe mund të bëhet transferimi i të dhënave prej Arduinos amë tek Arduinot tjetra. Qëllimi kryesor i lidhjes së tyre është rritja e kapacitetit memorues dhe numrit të pinave.

Për këtë hulumtim janë përdorur tri Arduino të cilat na janë ofruar në dispozicion nga ana e Kolegjit UBT. Ato janë programuar në gjuhën programuese C++ dhe janë përdorur për ta realizuar punimin në fjalë.

Hulumtimi dhe shtjellimi i literaturës së huaj dhe të gjuhës shqipe ishin bazamenti i këtij punimi. Gjithashtu janë përdorur libra të ndryshme dhe punime shkencore, për ta bërë sa më të qartë problemin dhe shpjegimin e pjesës harduerike dhe softuerike, ku janë përdorur këto metodologji studimi: hulumtimi i literaturës, shfletimi i internetit dhe metodologjia eksperimentale. Rol të veçantë ka luajtur shfletimi i internetit, përmes të cilit është bërë i mundur sqarimi i pjesës softuerike dhe është mundur shkarkimi i softuerit.

Po ashtu është e mundur të bëhet lidhja e më shumë se tri Arduinove, por duhet patur kujdes me lidhjen e saktë dhe komunikimin e tyre.

FALENDERIME

Falënderimi i takon Allahut, që më dha shëndet dhe ma mundësoi ta përfundoj me sukses këtë pjesë të studimeve.

Falënderimet e mia të përzemërta i takojnë familjes sime, që më përkrahen, këshilluan dhe financuan gjatë studimeve. I falënderoj për durimin dhe mirëkuptimin e tyre që kanë dëshmuar gjatë gjithë kohës. U jam shumë mirënjohës për të gjitha sakrificat që kanë bërë për mua. Dy fjalë janë të pamjaftueshme për të shprehur mirënjohjen time, por të jeni të sigurt se as me gjithë fjalët që njoh, nuk arrij ta përshkruaj këtë.

Kisha fatin dhe kënaqësinë të udhëhiqesha nga Prof: Bertan Karahoda, i cili më ka këshilluar dhe sygjëruar gjatë të gjitha fazave të punimit të temës së diplomës. Ndikimi i tij pozitiv ka pas efekt në ngritjen e vlerës dhe cilësisë së këtij punimi. Po ashtu falënderoj Ali Qerkinin, i cili më ka ndihmuar dhe siguruar materialin e nevojshëm për punimin e temës së diplomës. Në shenjë mirënjohje edhe falënderimi, këtë punim ua dedikoj familjës sime.

PËRMBAJTJA

LISTA E FIGURAVE	V
LISTA E TABELAVE	V
FJALORI I TERMAVE	VI
1.HYRJE	1
2. SHQYRTIMI I LITERATURËS	2
2.1 Komponentët e Arduinos	2
2.2 Hardueri i Arduinos.....	2
2.2.1 Butoni i resetimit.....	3
2.2.2 Programuesi serik In-Circuit	4
2.2.4 Mikrokontrolleri Atmega	9
2.2.5 Pinat hyrës analog	10
2.2.6 Voltazhi hyrës VIN	11
2.2.7 Tokëzimi (GND)	11
2.2.8 Pini i energjisë 5 volt.....	11
2.2.9 Pini i energjisë 3,3 volt.....	11
2.2.10 Pini i resetimit	12
2.2.11 Furnizuesi i jashtëm i energjisë	12
2.2.12 USB	13
2.2.13 Drita sinjalizuese	14
2.2.14 Referenca analoge (AREF)	14
2.2.15 Pinat digjital I/O (2-13)	14
2.2.16 Dalja dhe hyrja serike (TX/RX)	14
2.3 Softueri i Arduinos	15
2.3.1 Shkarkimi dhe instalimi i Arduinos për sistemin operativ <i>Windows</i>	15
2.3.2 Instalimi për <i>Mac OS X</i>	16
2.3.3 Instalimi për <i>Linux</i>	17

2.4 Softueri në detaj	19
2.4.1 Ndërrimi i gjuhës.....	19
2.5 Faqja startuese në detaj	19
2.5.1 Shembuj brendapërbrenda softuerit.....	20
3. DEKLARIMI I PROBLEMIT	22
4. METODOLOGJIA.....	23
5. PUNIMI KRYESOR	24
6. DISKUTIME DHE PËRFUNDIME	35
7. REFERENCAT	36

LISTA E FIGURAVE

Figura 1. Paraqitja frontale dhe komponentët përbërëse të pllakës ARDUINO	3
Figura 2. Struktura e mikrokontrolleri ATmega328	10
Figura 3. Koka e adapterit	12
Figura 4. Kabllo lidhëse në mes kompjuterit dhe pllakës ARDUINO	13
Figura 5. Mënyra e downloadimit të Arduinos	15
Figura 6. Paraqitja e procedurës së instalimit të softuerit ARDUINO	16
Figura 7. Instalimi në Mac OS X	17
Figura 8. Instalimi në Linux	18
Figura 9. Elementet e faqës startuese	20
Figura 10. Mënyra e paraqitjes së shembujve në softuerin e Arduinos	21
Figura 11. Vendosija e paisjeve në pllakat e Arduinove	24
Figura 12. Kodi i Arduinos së parë	25
Figura 13. Algoritmi i Arduinos së parë	26
Figura 14. Kodi i Arduinos së dytë	28
Figura 15. Algoritmi i Arduinos së dytë	29
Figura 16. Kodi i Arduinos së tretë	31
Figura 17. Algoritmi i Arduinos së tretë	32
Figura 18. Ndezja e dy Ledave	33
Figura 19. Fikja e dy Ledave	33
Figura 20. Fikja e Ledit të parë dhe ndezja e Ledit të dytë	34

LISTA E TABELAVE

Tabela 1. AVR-mikrokontrollerët e kompanis ATmel	7
Tabela 2. Të dhënat e pllakave të Arduinos	9

FJALORI I TERMEVE

RX – Hyrja serike
TX – Dalja serike
USB – Universal Serial Bus
LED – Drita Emituese e Diodës
IC – Çark Integrues
MDR – Matje, Dirigjim, Rregullim
CPU – Njësia Qendrore përpunimit
I / O – Hyrjet / Daljet
RISC – Instruksionet e Reduktuara të Vendosura në Kompjuter
CISC – Instruksionet e Komplikuara të Vendosura në Kode
SIMD – Instruksionet e Lira, Të dhënat e Shumëfishuara
DSP – Procesimi i Sinjalit Digjital
A / D – Analog / Digjital
GND – Pini i tokëzimit

1. HYRJE

Arduino është një pllakë harduerike, e cila përdoret për programimin e disa funksioneve me anë të kompjuterit. Nëpërmjet kësaj pllake mund të kryejmë funksione të ndryshme e sidomos me bashkëveprimin e lidhjeve të tjera, që shërbejnë për forcimin e sinjaleve dalëse apo për përshtatjen e sinjaleve marrëse nga sensor të ndryshëm.

Arduino filloi në vitin 2005 si një projekt për studentët në institutin e ndërveprimit të projektimit në Itali. Të parët që punuan në Arduino ishin studentët: Massimo Banzi, David Cuartilles, Tom Igoe, Gianluca Martino, David Mellis. Në atë kohë ata përdornin një "Vulë bazë" me një kosto prej \$ 100 e cila konsiderohej si e shtrenjtë. Pas instalimeve, platforma ishte e plotë. Hulumtuesit kanë punuar për ta bërë atë sa më të lehtë, më pak të shtrenjtë dhe në dispozicion për komunitetin burim të hapur.

Arduino është një platformë harduerike dhe softuerike, e menduar për programer, dizajner, artist, realizues të qarqeve elektrike dhe për secilin që ka interes në krijimin e mjediseve dhe objektivave bashkëvepruese. Për fillim pllaka Arduino ishte menduar për shkolla, dhe u zhvillua shpejt. Sikurse softueri i krijuar po ashtu edhe hardueri ishin shumë të përshtatshme për fillestar. Vetëm në tri vitet e para u shitën përafërsisht 50.000 pllaka Arduino. Ishte një projekt shumë i suksesshëm.

Programimi në Arduino është një version i thjeshtësuar i gjuhës programuese C dhe C ++. Nëse ju posedoni njohuri paraprake të këtyre gjuhëve, programimi i Arduino-s do të jetë fare i lehtë. Nëse jo, nuk ka nevojë për shqetësim, sepse në kapitujt në vijim, do ju sqaroj komandat e nevojshme, për përdorim fillestar.

2. SHQYRTIMI I LITERATURËS

2.1 Komponentët e Arduinos

Pas kësaj hyrje, në këtë kapitull do ju sqaroj më në detaj Harduerin e Arduinos. Por s'pari një udhëzim i rëndësishëm i sigurisë.

Kujdes: Ngarkimi elektrostatik shkatërron Arduinon.

Para se ti prekni pjesët elektrike ose pllakën e Arduinos, çdoherë duhet të bëhet “shkarkimi - zbrazja” për çështje sigurie.

Shkarkimi → Me siguri ju ka rastisë juve, që pas një fërkimi të një materiali të mëndafshhtë, kur keni prekur ndonjë njeri apo gjësend përçues, keni pas ndjenjën se ju ka zërë rryma. Shkaktari kryesor është ngarkimi statik, që shumë lehtë mund të arrij 10.000 Volt [1].

Meqenëse pjesët elektronike dhe pllaka e Arduinos janë të ndjeshme kundër këtyre tensioneve të larta, duhet që këtë ngarkesë elektrike ta largojmë disi. Këtë më së miri mund ta bëjmë, pak para prekjes së një pllake përmes masës (tokëzimit). Një masë tjetër preventive është që çdoherë duhet ta kapni pllakën në skaje. Pjesët ose portet nuk guxon të preken drejtpërdrejtë në pina. [1]

2.2 Hardueri i Arduinos

Pllaka, që do përdorim në pjesën softuerike, quhet Arduino *Uno*. *Uno* nga gjuha italiane do të thotë *Një*. Këtë emër e ka marr për hir të asaj, se pllaka Arduino ishte e para që u krijua atëbotë. Gjatë viteve, ekipi Arduino përmirësuan dizajnin e pllakës duke lëshuar disa versione të reja. Sot ka një numër të madh të platformave të Arduinos të cilat mund ti dallojmë në bazë të formës, shpejtësisë dhe hapësirës memoruese.

Disa prej tyre janë: Arduino Duo, Arduino-Robot, Arduino Esplora, LilyPad Arduino USB, Arduino Micro, Arduino Nano etj.

Në vijim është paraqitur pllaka Arduino (shih Figurën 1) nga pjesa frontale me përshkrime të komponentëve kryesore, të cilat veç e veç do përshkruhen në vazhdim, duke filluar nga butoni i resetimit në drejtim të akrepave të orës.

Arduino përbëhet prej disa komponentësh siç janë: mikrokontrolleri Atmega 328, butoni i resetimit, furnizimi i jashtëm i energjisë, USB-ja, disa pina të ndryshëm, programuesi serik, hyrja serike RX, dalja serike TX, drita sinjalizuese.

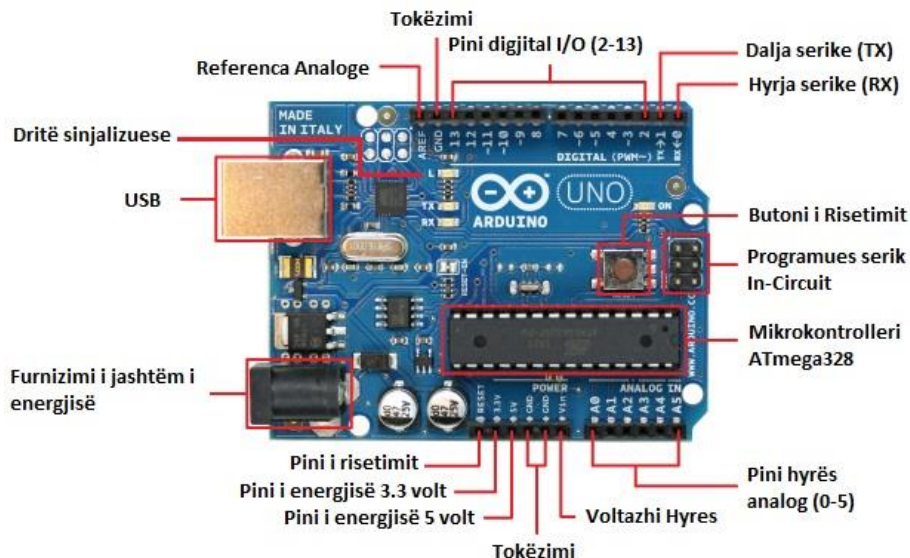


Figura 1. Paraqitja frontale dhe komponentët përbërëse të pllakës

2.2.1 Butoni i resetimit

Fjala angleze *Reset* do të thotë kthim në gjendjen fillestare. Butoni i resetimit gjendet afër programuesit serik dhe mikrokontrollerit Atmega 328. Në qoftë se ju shtypni pullën *Reset* ne pllakën e Arduinos, ristartohet pllaka si një kompjuter dhe fillon me përpunimin e softuerit të ruajtur në mikrokontroller. Me gjasë juve do ju nevojitet rrallë butoni i resetimit, se në momentin që ju e bëni furnizimin e pllakës me rrymë, ajo është gjithnjë në gjendjen fillestare. Kur pllaka e Arduinos nuk reagon më, butoni i resetimit na ndihmon shumë, që të arrijmë në pikën startuese. Arduino është dizajnuar në atë mënyrë që e lejon atë të rishkruhet nga softueri në një kompjuter të lidhur. Kur pllaka e Arduinos është e lidhur me një kompjuter nëpërmjet USB-s, ne mund të ngarkojm kod dhe të rishkruajm sa herë që është e nevojshme.

Një shembull për këtë: në qoftë se një roboti i japim komandën që të lëviz në një hapësirë të caktuar, dhe pas detektimit të një pengese, të sinjalizoj përmes dritës sinjalizuese LED çdo katër sekonda. Kjo bëhet duke i dhënë urdhërin *delay (4000)*. Por nëse pa dashje gjatë programimit shtypim më tepër zero, atëherë roboti do mbetet në atë gjendje një kohë të gjatë, në varësi nga ajo, se sa zero kemi shtypur.

Këtë problem mund ta mënjanojm me anë të butonit të resetimit, duke shtypur butonin Reset dhe programi starton prej fillimit, në vend të rrugës së mundimshme duke ja ndërprerë robotit furnizimin me rrymë elektrike.

2.2.2 Programuesi serik In-Circuit

Në pllakën e Arduinos është krahas softuerit normal, gjithashtu edhe një i ashtuquajtur **Bootloader**. Ky Bootloader mundëson aktualizimin e softuerit në mikrokontroller drejtpërdrejt përmes USB-s (Upload) dhe pa harduer special.

Në të kaluarën duhej që mikrokontrolleri për transferim të ri të largohej nga pllaka. Ata futeshin në harduer shtesë dhe pastaj riprogramoheshin. Kjo nënkuptonte që nëpërmjet largimit të vazhdueshëm dhe futjes së pjesëve nevojitej një ngarkesë e lartë mekanike dhe harduer shtesë për programim.

Me futjen e portit të programuesit serik In-Circuit ishte e mundur drejtpërdrejt të ketë qasje në pjesën programuese, pa e largu, mirëpo harduer shtesë për programim ishte i nevojshëm prapëseprapë. [2] [1]

2.2.3 Mikrokontrollerët

Mikrokontrolleri është një kompjuter, i cili përbëhet nga një chip i vetëm. Në krahasim me një kompjuter normal, i cili përbëhet prej shumë pjesëve të ndërrueshme si procesori dhe RAM memoria, këtu është i vendosur gjithçka në një IC (Integrated Circuit, engl. për qark integruar).

Mikrokontrollerët përdoren para se gjithash në teknikën e rregullimit automatik, të matjeve, dirigjimeve dhe rregullimeve teknike. Përparësia e një sistemi të mikrokontrollerit është, që në një hapësirë të vogël me energji dhe kosto efikase të bëjë matjen dhe interpretimin e madhësive fizike, si dhe të takoj vendime dhe të kryej veprime. Në thelb është çdo detyrë, që njeriu me Arduino dëshiron ta zgjidh, një punë MDR (matje, dirigjim, rregullim).

Elementet kryesore të një mikrokontrolleri janë: CPU-ja (Central Processing Unit), memoria kryesore (RAM), memoria programuese (FLASH) dhe periferia.

CPU

Njësia themelore është qendra e njësive llogaritëse (procesori). Ai mundë të kuptohet si truri i mikrokontrollerit. Aty përpunohen urdhrat dhe operacionet aritmetike.

Memoria kryesore (RAM) dhe programuese (Flash)

Memoria kryesore dhe programuese janë logjikisht të ndara. Programi përdorues, të cilin e kemi shkruar vet, vendoset në një memorie të qëndrueshme Falsh-memorie, memorie programuese.

Memoria kryesore vlen për ruajtjen e përkohshme të llogaritjeve, matjeve dhe të dhënave dirigjuese. Memoria kryesore është më e vogël se memoria programuese, por shumë më e shpejtë. Të dhënat në memorien kryesore krijohen gjatë kohës së marshit, dhe zhduken me një ristartim të kontrollërit, kurse në memorien programuese jo.

Periferia

Si periferi emërtohen komponentët e një mikrokontrolleri, të cilat nuk përfshihen në CPU, RAM dhe Flash (memorie programuese), posaçërisht komponentët, të cilat paraqesin një interface me pjesën e jashtme. Hyrjet dhe daljet digjitale (I/O) llogariten si pajisje periferike. Një numër i madh i mikrokontrollerve, gjithashtu edhe Arduino, ofrojnë një numër të madh të hyrjeve dhe daljeve analoge dhe digjitale.

Krahasimi i teknologjive: RISC dhe CISC

Sikurse edhe Arduino, po ashtu edhe AVR-mikrokontrolleri bazohen në teknologjinë RISC (Reduced Instruction Set Computer). Sipas Atmel shkurtesa AVR nuk është shkurtesë speciale, mirëpo vlen vetëm për emërtimin e kësaj familje të mikrokontrollerve.

Teknologjia-CISC

Në teknologjinë CISC (Complex Instruction Set Coding) bëhet ngarkimi i memories programuese në RAM të sistemit dhe e ndan këtë me memorien programuese. Kjo mund të kuptohet, se kodi programues dhe të dhënat ndajnë të njëjtën memorie. Kjo ishte e dobishme posaçërisht në kohërat e hershme, sepse RAM-memoria ishte e shtrenjtë.

Një karakteristikë e veçantë për mikrokontroller është struktura e komandave. Një CISC posedon komanda të rëndësishme, të cilat në hapa të veçantë ekzekutohen. Në teknikën digjitale një komandë është një renditje e bajtëve të caktuar. Një bajt mundë të merr 256 (0 deri 256) gjendje të ndryshme. Për të implementuar më shumë se 256 komanda të ndryshme, nevojiten më shumë bajt. Kështu që, një komandë speciale mundë të përbëhet prej më shumë bajtëve (p.sh. pesë bajtëve). Ngarkimi i kësaj komande zgjatë më shumë se i një komande, e cila është e gjatë vetëm një bajt.

Teknologjia-RISC

Teknologjia RISC është krijuar si ide, duke pasur parasysh mangësitë e teknologjisë CISC. Qëllimi ishte që në CPU të implementohen komanda të shkurta dhe të shpejta. Kështu që, në RISC-mikrokontrollera nuk mund të gjenden komanda, që përmbajnë më shumë se tre ose katër bajt. Teknologjia RISC nuk kërkon dekodim, dukuri e cila e bënë më të shpejtë, kursen hapësirë në chip dhe e bënë më të lirë në treg.

Për të arritur efikasitetin sikurse CISC, pjesa më e madhe e RISC teknologjive posedojnë një numër të madh të regjistrave. Një regjister është një memorie e përkohshme dhe e shpejtë në CPU. Një tjetër dallim me CISC teknologjinë është se bëhet ndarja e qartë fizikale dhe logjike në mes të memories programuese dhe asaj kryesore.

Duhet të kihet parasysh fakti se nuk ka teknologji të pastër të përbërë vetëm nga CISC apo nga RISC.

Mikrokontrolleri-Atmel

Ekzistojnë prodhues të ndryshëm të mikrokontrollerve, ata janë: Atmel, Microchip, Philips, Motorola, Texas, SGS dhe Infineon.

Në firmën Atmel bënë pjesë familja AVR, në të cilën familje bëjnë pjesë të gjithë mikrokontrollerët që përdoren për llojet e ndryshme të pllakave të Arduinos. Mikrokontrollerët-AVR përmbajnë njësi të ndryshme periferike On-Chip (integrimi i të gjitha ose një pjesë të madhe të funksioneve të një sistemi në një chip) si I/O-portet dhe A/D-konvertorët.

Familja e mikrokontrollerëve AVR

Në princip, të gjithë mikrokontrollerët e familjes AVR janë njëjtë të ndërtuar. Disa tipa dallojnë në pajisje, si p.sh. hapësira memoruese, numri dhe lloji i Timer-ve.

Në tabelën 1 është paraqitur një pamje e përgjithshme e mikrokontrollerëve AVR, të cilët kryesisht përbëhen prej 8 bitëve. Seria ATxmega emërtohet si 8/16 Bit mikrokontroller, sepse intern mund të llogaris me një gjerësi prej 16 Bit. Kjo seri është kompatible me ATtiny dhe Atmega, por jo me serinë AT32. Edhe pse i takon familjes AVR, ka një arkitekturë (DSP, SIMD) krejtësisht tjetër. Përveç tjerash, Atmel prodhon një numër tjetër të konsiderueshëm të AVR-mikrokontrollerëve special, të cilët bazohen në ATtiny ose Atmega. Për shembull, Automotive-AVR, të cilët janë të përshtatshëm për një temperaturë (-40 °C deri + 125°C), ose Battery Management AVR, të cilët

mund të operojnë me tensione të larta (9 V) dhe posedojnë pina të tensioneve të larta (High Voltage-Pins-maximal 28 V).

Siç mund të shihet edhe nga tabela 1, ATxmega-mikrokontrollerët (AVRxmega) punojnë me një tension prej 1,6 V, gjë që ata i kualifikon në Lower-Power-Controllern (mikrokontroller të fuqisë së ultë). Në Power-Down-Mode (gjendje në të cilën gjitha taktet ndalen) harxhojnë mikrokontrollerët aktual standard-AVR 100 nA, ku ju lejohet një përdorim i taktit maksimal prej 12 MHz.

AVR-Serie	Tensioni [V]	Max. frekuencë [MHz]	RAM [Bytes]	Memoria programuese [kByte]	I/Os max.	Lidhjet Shtëpiza	Features
AT90	2,5...5,5	16	500...4000	8...28	53	20-64 Pin TQFP,QFN	CAN,USB,10 Bit-A/D DAC
AT90S	2,7...6	12	64...128	1...4	20	20-32 Pin PDIP,SOIC, SSOP,TQFP	10 Bit- A/D,Analog Comparator,UART,SPI
ATtiny	0,7...5,5	20	64...1000	0,5...8	6	6-32 Pin SOT23,PDIP SOIC,MLF	10 Bit-A/D,Analog Comparator,UART
ATmega	1,8...5,5	20	1000...8000	4...265	64	28-100 Pin PDIP,QFN, TQFP,BGA	8/10 Bit-A/D,Analog Comparator,USARTs, SPI,USB,LCD
ATxmega	1,6...3,6	32	2000...16000	16...384	78	32-100 Pin QFN,TQFP, BGA	12 Bit-A/D,DACs,USB, USARTs,SPI,TWI, DMA

Tabela 1. AVR-mikrokontrollerët e kompanis ATmel

Seria ATtiny posedon mikrokontroller të tipit ATtiny43U, i cili punon me 0,7 V dhe një takt prej 8 MHz.

Sistemet standarde të Arduinos

Një Atmel-mikrokontroller i zakonshëm dallon nga një mikrokontroller i Arduinos, në atë se në memorien Flash është i programuar edhe një Boot-Loader, me qëllim që Arduino të mund të startoj dhe programimi i mëvonshëm të jetë i mundur edhe pa pajisje externe. Pothuajse secila Arduino (siç shihet nga tabela 2) posedon një USB-Interface, që hyn në përdorim për programim. Për disa tipa (p.sh. Mini, LilyPad) është i nevojshëm një adapter, me qëllim që ta pajisë Interface-sin serik të Arduinos me funksionalitetin-USB. Në tabelën 2 janë paraqitur modelet e ndryshme të pllakave të Arduinos, mikrokontrollerët dhe vetitë e tyre.

Modeli	Kontrolleri	Memoria	USB-Chip	I/O Portet	A/D Kanale	Shënim	Madhësia [mm]
Diecimilla	ATmega168	16 kB Flash 1 kB RAM 0,5kB EEPROM	FTDI-FT232RL	14	6	Power-Jumper	68,6 x 53,3
Duemilano ve	ATmega168 ATmega328	16/32 kB Flash 1/2 kB RAM 0,5/1 kB EEPROM	FTDI-FT232RL	14	6	Versione të ndryshme	68,6 x 53,3
Uno	ATmega328	32 kB Flash 2 kB RAM 1 kB EEPROM	Atmega8U2 ose Atmega16U2	14	6	Vlen si Pllakë standarde	68,6 x 53,3
Mega	ATmega1280	128 kB Flash 8 kB RAM 4 kB EEPROM	FRDI-FT232RL	54	16	4 x UART 2 x SPI	101,6 x 53,3
Mega 2560	ATmega 2560	256 kB Flash 8 kB RAM 4 kB EEPROM	Atmega8U2 ose Atmega16U2	54	16	4 x UART 2 x SPI	101,6 x 53,3
Mega ADK	ATmega 2560	256 kB Flash 8 kB RAM 4 kB EEPROM	Atmega16U2 +MAX 3421 (USB 2.0)	54	16	4 x UART 2 x SPI	101,6 x 53,3
Leonardo	ATmega32U4	32 kB Flash 2,5 kB RAM 1 kB EEPROM	Intern	20	12	UART	68,6 x 53,3
Esplora	ATmega32U4	32 kB Flash 2,5 kB RAM 1 kB EEPROM	Intern	20	12	UART Sensoren, Elemente hyrëse	68,6 x 53,3
Micro	ATmega32U4	32 kB Flash 2,5 kB RAM 1 kB EEPROM	Intern	20	12	UART	48,3 x 18
Mini	ATmega328	32 kB Flash 2,5 kB RAM 1 kB EEPROM	Per Adapter	14	8	Pllaka më e vogël me DIP-Layout	33 x 18
Nano	ATmega168 ATmega328	32 kB Flash 2,5 kB RAM 1 kB EEPROM	FTDI-FT232RL	14	8	Pllaka më e vogël me USB	43 x 18
LilyPad (2,7 V – 5,5V)	ATmega168 ATmega328	32 kB Flash 2,5 kB RAM 0,5 kB EEPROM	Per Adapter	14	6	rrumbullak, i pastrueshëm	Ø 50
LilyPad USB (3,3)	ATmega32U4	32 kB Flash 2,5 kB RAM	Intern	20	12	UART	Ø 50

		1 kB EEPROM					
Due (3,3 V)	AT91SAM (Cortex M3)	512 kB Flash 96 kB RAM	ATmega16U2	54	12	2 x D/A 4 x UART CAN	101,6 x 53,3
YUN (3,3 V)	ATmega32U4 Plus AR9331 Mit Linux	32 kB Flash 2,5 kB RAM 1 kB EEPROM	Intern	20	12	UART, USB,Ether net WiFi,SD- Card	68 x 53,3
Galileo (3,3 V)	Intel SoC X1000 mit Linux	512 kB Flash 256 MB RAM	Intern	14	6	UART,RS 232,Ethern etSD- Card,Mini PCIe	106 x 71,2

Tabela 2. Të dhënat e pllakave të Arduinos

Duke filluar nga mikrokontrollerët më të vogël për nga memoria të familjes Atmega, tregohet edhe domethënia e numrave që kanë marr prapa, si p.sh. ATmega8, do të thotë që ky mikrokontroller ka 8 kB Flash-memorie, Atmega16 ka 16 kB Flash-memorie e kështu me radhë për Atmega32, Atmega64, Atmega128. Kurse ATmega16U2 do të thotë se përveç që ka 16 kB Flash-memorie, ka edhe një USB-2,0-Interface (për atë U2).

2.2.4 Mikrokontrolleri Atmega

Mikrokontrolleri Atmega 328 gjendet afër butonit të resetimit dhe pinave hyrës analog. Ky mikrokontroller punon me një frekuencë të taktit 16 MHz, në një tension prej 5V dhe ofron 14 porte digjitale, si dhe 6 hyrje analoge me një rezolucion prej 10 Bit.

Konvertimi i sinjaleve analoge në ato digjitale në ATmega bëhet kryesisht në konvertimin e temperaturës apo edhe konvertimin e intensitetit të dritës. Pas konvertimit rezultati mund të paraqitet në ndonjë formë më të lexueshme për njeriun si për shembull në LED, indikator 7 segmentesh apo edhe në LCD Display. Saktësia e matjes së këtyre fenomeneve natyrore nuk do të thotë që është e saktë 100%, por kjo varet prej karakteristikave të konvertuesit Analog/Digjital.

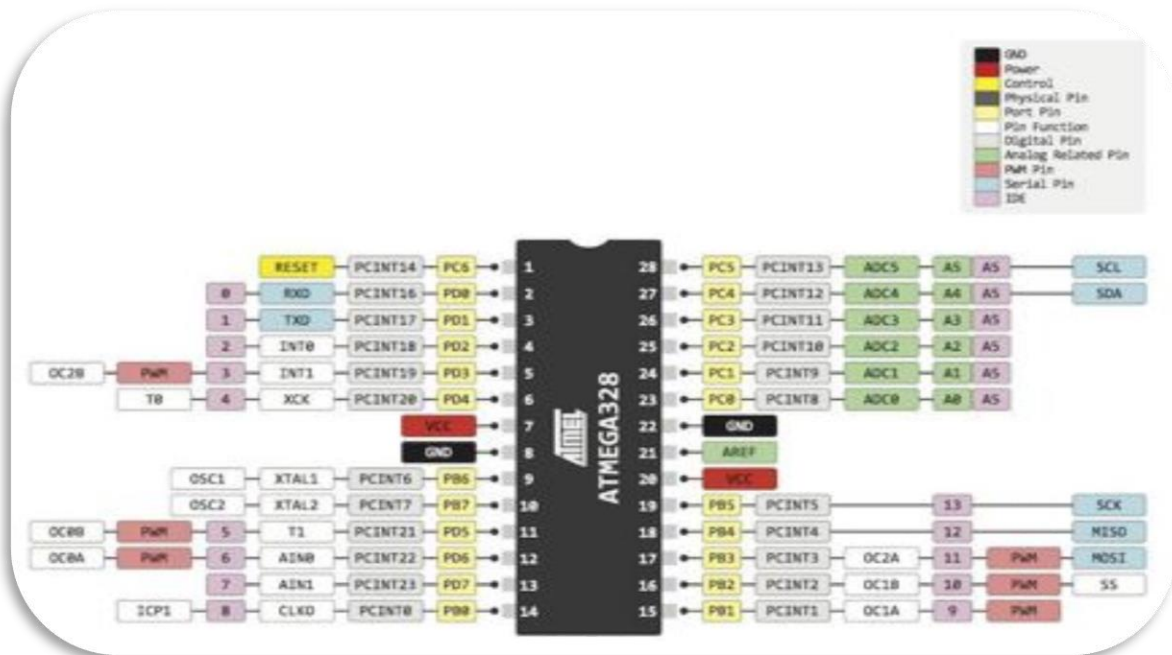


Figura 2. Struktura e mikrokontrolleri ATmega328

Në bazë të dhënave nga faqja Atmel, Atmega328 mund të operoj në një rang të temperaturës prej -40°C deri në +85°C. Varshmëria e ndryshimit të temperaturës në ndryshimin e tensionit është 10mV për K(Kelvin), ku 0mV do të thotë 0 K. Saktësia e konvertimit varet direkt prej karakteristikave të elementeve që e përbëjnë këtë proces.

ATmega ka karakteristikat e mëposhtme: 32 kB Flash (prej të cilave 0,5 kB janë për Bootloader), Memoria programore me aftësi të lexoj derisa është duke shkruar, 2Kbyte SRAM, regjistrat e punës 32bit dhe gjashtë softuer të kursimit të energjisë. Ndër të tjera ka mikrokontroller të fuqishëm që ofron një fleksibilitet dhe kosto efektive. Ky mikrokontroller është i mbështetur me një suitë të plotë të programit dhe sistemit të zhvillimit. [3] [4]

2.2.5 Pinat hyrës analog

Në krahasim me hyrjet digjitale, në hyrjet analoge (0-5) mund të vendoset një tension i çfarëdoshëm ndërmjet 0 dhe 5 volt. Analog do të thotë, që tensioni me të vërtetë mund të marr çdo vlerë brenda një intervali të definuar, si p.sh. brenda intervalit 0 dhe 5 volt.

Gjatë leximit të një tensioni analog, Arduino krahason tensionin në njërën nga pinat 0 deri në 5 me një tension standard 5 volt. Kështu ju do të merrni për shembull gjatë vendosjes së 0 V një vlerë

digjitale 0 në softuer, kurse gjatë vendosjes së 2,5 volt një vlerë 128 dhe për 5 volt 256. Kjo përlllogaritje vjen sipas sistemit decimal.

Në qoftë se dëshirojmë të lexojmë një tension më të vogël se vlera maksimale 5 volt (p.sh. 3,14 volt) dhe prapëseprapë të marrim vlera në softuer prej 0 deri 255, në këtë rast na vjen në përdorim pini AREF.

2.2.6 Voltazhi hyrës VIN

Ky pin ju takon pinave të furnizimit me rrymë dhe mund të përdoret edhe si hyrës gjithashtu edhe si dalje. Në këtë pin mund të vendosen tensione 7-12 volt, që pastaj pllakën e Arduinos e furnizojnë me energji elektrike (në qoftë se nuk është e furnizuar përmes USB-s ose përmes portit të furnizimit me rrymë). Por në qoftë se pllaka furnizohet me energji elektrik përmes USB-s ose kablllos furnizuese, atëherë në këtë pin kemi vetëm 5 volt.

2.2.7 Tokëzimi (GND)

Pini i tokëzimit (anglisht Ground, shkurtimisht GND) është pika referencë për të gjitha tensionet në pllakën Arduino. Ai është në të njëjtën kohë pol minus dhe kështu që reprezenton 0 volt. Të njëjtin funksion kryen tokëzimi i paraqitur në pjesën e hyrjeve digjitale.

2.2.8 Pini i energjisë 5 volt

Ky pin vlen vetëm si dalje. Përderisa pllaka e Arduinos furnizohet përmes USB-s ose portit të furnizimit me rrymë, vijnë në këtë pin tensione të rregulluara 5 volt. Këto tensione ju mund ti përdorni për furnizim me rrymë të pjesëve të tjera të vogla, p.sh. të sensorëve. Mos e furnizoni këtë dalje asnjëherë me një tension, se ky tension mund ta shkatërroj pllakën e Arduinos.

2.2.9 Pini i energjisë 3,3 volt

Në këtë pin dërgon Arduino një tension prej 3,3 volt, prej nga dhe ka marr emrin. Keni parasysh se këtu mund të dërgohet një rrymë maksimale prej 50 mA. E njëjta vlen edhe këtu, sikurse te pini i energjisë 5 volt, se ky pin nuk guxon asnjëherë të furnizohet me një tension në këtë dalje, sepse mund ta shkatërroj pllakën e Arduinos.

2.2.10 Pini i resetimit

Ky pin shërben për lidhje me një buton të vet të resetimit. Në momentin që ky pin lidhet me tokëzimin, bëhet resetimi i pllakës së Arduinos. Në qoftë se pllakën e Arduinos e zgjerojmë me *Shields*, mund të ndodh, që ky e bllokon butonin e resetimit në pllakë. Për këtë arsye është ky pin i dobishëm.

2.2.11 Furnizuesi i jashtëm i energjisë

Për ta furnizuar pllakën e Arduinos me rrymë elektrike, ekzistojnë dy mundësi: përmes portit të rrymës ose portit të USB-s. Në qoftë se e lidhim pllakën me një USB kabëll me kompjuterin, do të vëjm re se menjëherë drita sinjalizuese e rrymës (Power-LED) ndriçon dhe gjithçka në pllakë, furnizohet me rrymë. Pa marr parasysh, se pllaka a lidhet përmes USB-s apo portit furnizues të rrymës, ajo e njeh burimin e rrymës automatikisht.

Për të shfrytëzuar portin e rrymës elektrike na nevojitet një kokë e adapterit me diametër të brendshëm 2,1 mm (shih Figurën 3).



Figura 3. Koka e adapterit

Poli pozitiv gjendet në mesin e vrimës së kokës së adapterit, kurse poli negativ apo tokëzimi në pjesën e jashtme. Pllaka mund të veprojë në një furnizim me një tension hyrës 7 volt, përafërsisht 0,29 Wat (290mW).

Në qoftë se portin e furnizimit të rrymës elektrike e lidhim me një furnizues të jashtëm të rrymës apo bateri, duhet të kemi parasysh që vlerat kufitare të mos tejkalohen për një kohë të gjatë. Tejkalimi i vlerës 20 volt për një kohë të gjatë, mund të shkatërroj pllakën e Arduinos. Furnizuesi i jashtëm duhet të posedoj një tension të vazhdueshëm (DC=Direct Current Voltage).

Vlera të rekomanduara të tensionit hyrës janë: 7-12 volt.

Vlera të lejuara për një kohë të shkurtër janë: 6-20 volt.

2.2.12 USB

Porti i USB-s në pllakën e Arduinos ka tri funksione të ndryshme:

- Furnizimin e Arduinos me rrymë
- Interface për transfer të softuerit në pllakën e Arduinos
- Shkëmbim të dhënave ndërmjet Arduinos dhe kompjuterit

Në qoftë se lidhim një pllakë të Arduinos me kompjuter, atëherë pllaka do furnizohet me rrymë. Në këtë rast, nuk duhet bërë furnizimi extra përmes portit të furnizimit me rrymë. Funksioni i dytë është transferimi i programit të shkruar në pllakën e Arduinos, përmes të cilit komandohen të gjitha veprimet e pajisjes për të cilën përdoret pllaka. Funksioni i tretë dhe i fundit është që pllaka Arduino dërgon të dhënat në kompjuter.

Para se të bëhet lidhja e pllakës së Arduinos nëpërmjet USB-s me kompjuter, duhet për disa sisteme operative të bëni instalimin e të ashtuquajturit drajver (Driver), përndryshe mund të ndodhë që Arduino mos të njihet nga kompjuteri.

Kabllo standarde që kryejnë këto veprim janë USB 2.0 (shih Figurën 4). Gjatësia e kablllove është rreth 178 cm, ngjyrat e kablllove dhe forma mund të ndryshojnë pak nga modelet e ndryshme të Arduinove. [5]



Figura 4. Kabllo lidhëse në mes kompjuterit dhe plakës ARDUINO

Në qoftë se ju dëshironi të keni një vështrim më të afërt të portit të USB-s në pajisjen e Arduinos, atëherë ajo gjendet në anën e portit për furnizim me energji elektrike apo më saktësisht afër dritës sinjalizuese.

Pllaka e Arduinos zakonisht është e paraprogramuar me një program për të verifikuar funksionimin e pllakës. Pasi të lidhet pllaka me kompjuter apo me një furnizim tjetër të energjisë, ajo menjëherë ka një dritë që vazhdimisht ndizet dhe fiket, që sinjalizon lidhjen në mes të pajisjeve.

2.2.13 Drita sinjalizuese

Në pllakën e Arduinos me shkronjën L është paraqitur drita sinjalizuese LED. Kjo dritë na ndihmon të kontrollojmë ose ta kërkojmë gabimin. Drita LED ndezet, në qoftë se vendoset me urdhrin e duhur në gjendjen *High* dhe fiket nëse është *Low*.

2.2.14 Referenca analoge (AREF)

Ky pin është në hyrje, në të cilin mund të vendoset një tension i çfarëdoshëm ndërmjet 0 dhe 5 volt. Ai mund të përdoret vetëm në lidhje me hyrjet analoge 0 deri në 5 (Analoge IN) dhe ju shërben këtyre si referencë.

2.2.15 Pinat digjital I/O (2-13)

Pinat 2 deri në 13 shërbejnë si hyrës ose dalës. Kjo do të thotë, që një tension mund të vendoset në një pin (hyrës) ose Arduino mund të jap një tension (dalës).

Kujdes: Tensionet nën 0 dhe mbi 5 volt në njërin prej pinave dëmton pllakën e Arduinos.

Pasi që në pyetje janë hyrje/dalje digjitale, këta pina si një kompjuter njohin vetëm dy gjendje: 0 dhe 1. Në elektronikën e Arduino Uno-s reprezenton 0 volt ose 5 volt përkatësisht në formë të përgjithshme *LOW* ose *HIGH*.

2.2.16 Dalja dhe hyrja serike (TX/RX)

Nëse dëshirojmë që programin, të cilin e kemi shkruar, ta transferojmë në Arduino, kjo ndodh nëpërmjet një lidhje në mes kompjuterit dhe portit USB të pllakës së Arduinos. Kompjuteri dhe Arduino shkëmbejnë të dhëna. Në rastin kur Arduino dërgon diçka tek kompjuteri, sinjalizon TX (Transmitter = dërgues) dhe RX (Receiver = pranues) në qoftë se kompjuteri dërgon të dhëna në drejtim të Arduinos. Përderisa të dyja dritat sinjalizuese gjatë transferimit të të dhënave ndizen dhe fiken, duhet të kuptojmë se komunikimi në mes të kompjuterit dhe Arduinos është duke funksionuar pa problem. Këto dy drita sinjalizuese gjenden në pllakën e Arduinos, afër dritës sinjalizuese L (LED), me TX dhe RX të shënuara.

Nëpërmjet pinave TX dhe RX mund të bëhet një transfer serik i të dhënave me pjesët tjera. Meqenëse Arduino Uno posedon vetëm një Interface serik, që në pllakë intern me USB e lidhur

është, duhet që këta pina të mos përdoren. Për arsye se gjithnjë, në qoftë se një program transferohet në Arduino, të dhënat do të kalojnë edhe nëpër këta pina.

2.3 Softueri i Arduinos

Pasi u njohëm me harduerin e pllakës së Arduinos, na mbetet të njihemi me pjesën softuerike e cila na shërben për komunikim mes pllakës së Arduinos dhe kompjuterit si dhe për programimin e saj. Softueri Arduino mund të shkarkohet falas nga interneti dhe atë pa pasur nevojë për ndonjë licensë, vetëm se duhet të specifikoni sistemin operativ, për të cilin ju dëshironi të bëni instalimin.

2.3.1 Shkarkimi dhe instalimi i Arduinos për sistemin operativ *Windows*

Për të instaluar softuerin Arduino në *Windows*, vizitoni faqen projektuese të Arduinos (<http://arduino.cc>). Navigoni në menynë *Download* dhe pasi ta keni hapur faqen, klikoni mbi linkun *Windows Installer*, siç shihet në figurën 5. Installer është rruga më e thjeshtë për të instaluar softuerin. Nuk ka rëndësi se ju posedoni Windows 7 apo Windows 8, se është i njëjti shkarkim.

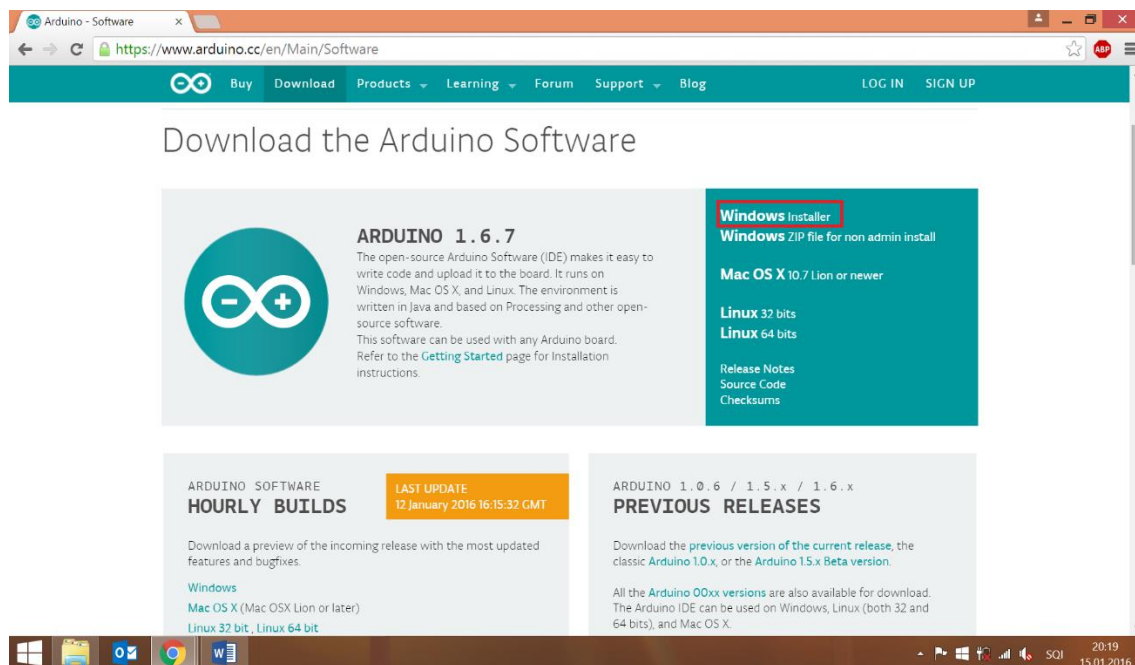


Figura 5. Mënyra e downloadimit të Arduinos

Pasi të keni klikuar, do të bëhet shkarkimi i softuerit dhe *Windowsi* zakonisht i ruan të dhënat në follderin Downloads. Për të startuar instalimin e softuerit të Arduinos, ju duhet të klikoni dy herë mbi të, pastaj ti përcjellni udhëzimet nëpërmjet figurave të mëposhtme.

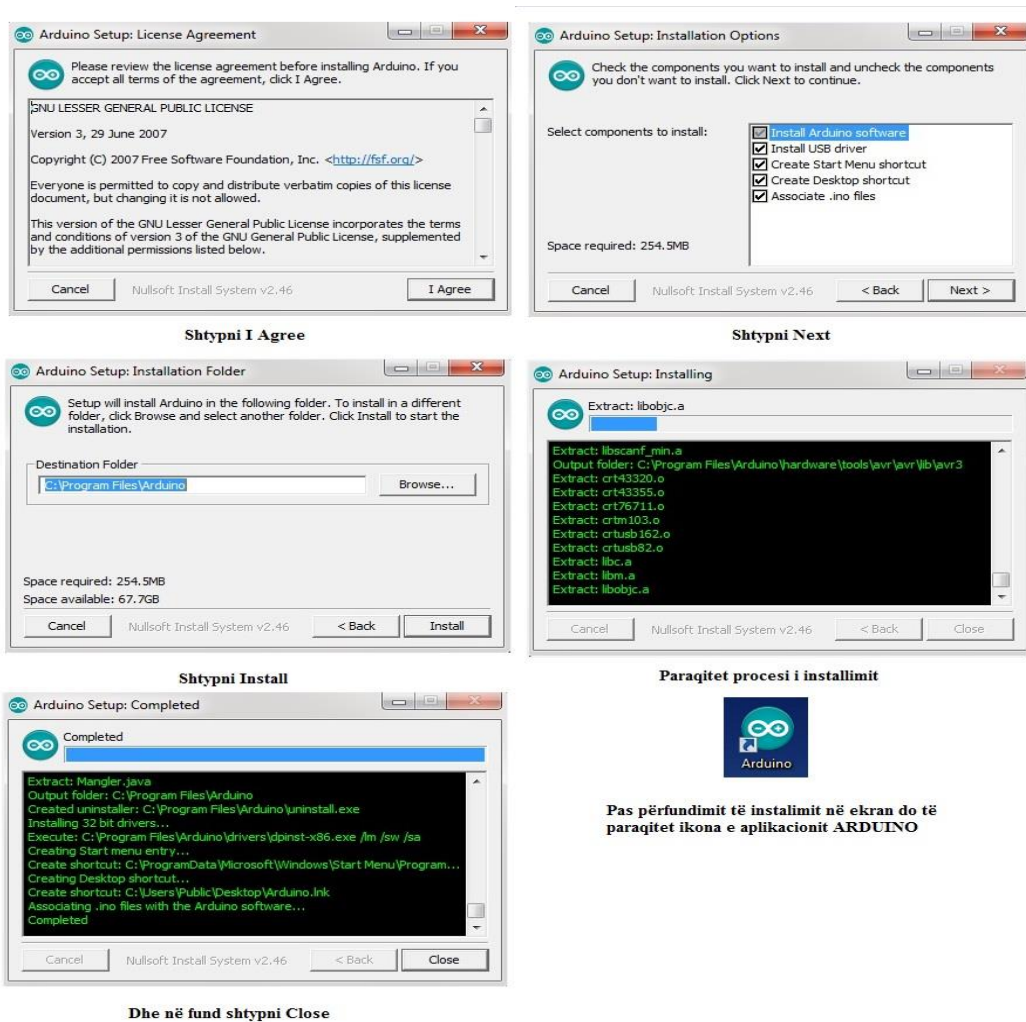


Figura 6. Paraqitja e procedurës së instalimit të softuerit ARDUINO

2.3.2 Instalimi për Mac OS X

Shkarkimi për sistemin operativ *Mac OS X* bëhet duke thirrur faqen projektuese të Arduinos, sikurse në rastin për sistemin operativ *Windows*. Pas hapjes së faqes, ju duhet të zgjidhni sistemin operativ që të bëni shkarkimin e versionit të dëshiruar. Instalimi i këtij versioni dallon nga ai për *Windows*, ngase këtu ju duhet vetëm të bëni bartjen e softuerit të shkarkuar nga direktoriumi *Downloads* në atë ku janë programet (*Applications*), siç mund të shihet edhe nga figura 7. Driver shtesë për *Mac OS X* nuk nevojitet.



Figura 7. Instalimi në Mac OS X

2.3.3 Instalimi për Linux

Ubuntu tani për tani është njëri prej sistemeve operative të *Linux-it* më i pëlqyeri nga fillestarët. Në qoftë se përdorni ndonjë version tjetër të *Linux-it*, atëherë vizitoni faqen projektuese të Arduinos, për të marr ndihmë rreth instalimit.

Nëpërmjet *Ubuntu* bëhet instalimi i softuerit të Arduinos në bazë të hapave në vazhdim. Një shkarkim nga faqja projektuese e Arduinos nuk është i nevojshëm. S`pari hapeni *Ubuntu Software-Center*.

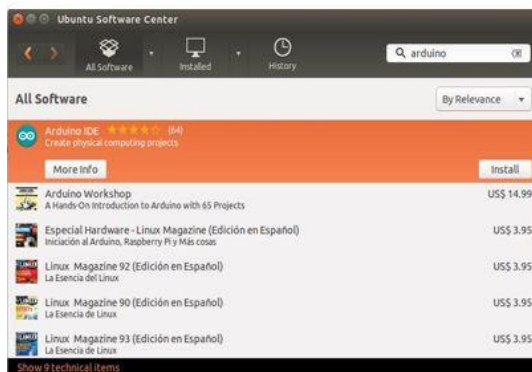
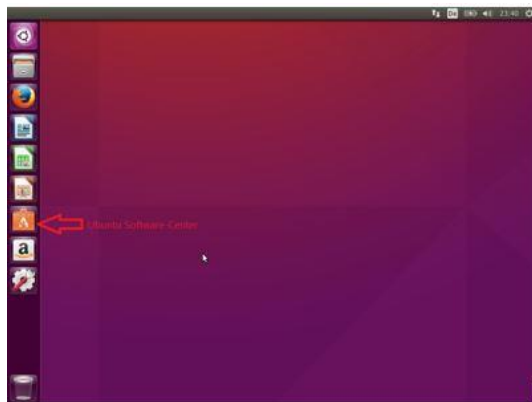


Figura 8. Instalimi në Linux

Pasi të jetë hapur *Software-Center*, klikoni në fushën e kërkimit dhe shkruani Arduino, që të kërkonte softuerin. Në momentin e gjetjes së softuerit të Arduinos në *Software-Center*, ajo do të tregohet si në figurën 8 në pjesën e djathtë lartë. Klikoni mbi *Install*, me qëllim që të startoni instalimin. Sistemi operativ *Linux* para çdo instalimi të ndonjë softueri, kërkon që përdoruesi të legjitimohet, dhe më pas vazhdon me instalimin e softuerit. Me përfundimin e instalimit, do t'ju shfaqet në ekran simboli i Arduinos. Në qoftë se jo, atëherë klikoni mbi simbolin  *Ubuntu-Dashboard* dhe kërkonte softuerin. Menjëherë pasi të keni shtypur shkronjën e parë, juve do ju paraqitet softueri dhe prej nga edhe mund të startohet për përdorim të mëvonshëm.

Me qëllim që kompjuteri juaj ta njohë pllakën e Arduinos, në prapavijë është instaluar s'bashku me softuerin edhe një Driver, çka do të thotë, ju duhet ta ristartoni kompjuterin tuaj, që të funksionoj gjithçka në rregull.

2.4 Softueri në detaj

Këtu ju merrni edhe disa udhëzime, se si softueri i Arduinos është i ndërtuar dhe çka nënkuptojnë veç e veç pikat e menysë.

2.4.1 Ndërrimi i gjuhës

Pasi programi si zakonisht shkarkohet në gjuhën angleze dhe ju dëshironi ta përdorni në gjuhën shqipe, ju mund të bëni ndërrimin e saj duke klikuar mbi *File* dhe duke zgjedhur *Preferences*. Ky softuer është përkthyer në mbi 30 gjuhë.

Për ti njohur këto ndryshime, ju duhet që softuerin e Arduinos ta mbyllni dhe ta hapni përsëri.

2.5 Faqja startuese në detaj

Në figurën 9 shihet faqja startuese e softuerit të Arduinos me të gjitha simbolet. Konkretisht simbolet dhe zonat e faqes kanë këto kuptime:

- Verifiko **1**: Kontrollon kodin programues për gabime
- Ngarko **2**: Transferon programin e kryer në harduer
- E re **3**: Krijon një program të ri, të zbrazët
- Hap **4**: Hapë programet ekzistuese
- Ruaj **5**: Ruan programin aktual
- **6**: Tregon emrin e programit aktual dhe datën së bashku me muajin
- **7**: Versioni i softuerit të instaluar të Arduinos
- **8**: Aktivizon monitorin serik
- **9**: Funkcionet për shfrytëzimin e Tabit
- **10**: Zona për editim
- **11**: Zona për tregim të mesazhit të statusit
- **12**: Tregon rreshtin, në të cilin kursori gjendet
- **13**: Tregon bordin e Arduinos dhe lidhjen aktuale

Zakonisht softueri i Arduinos gjatë startimit hap programin e fundit të përdorur. Gjatë mbylljes së të dhënës së fundit të hapur mbaron Programi, por paraprakisht ju njofton juve softueri.

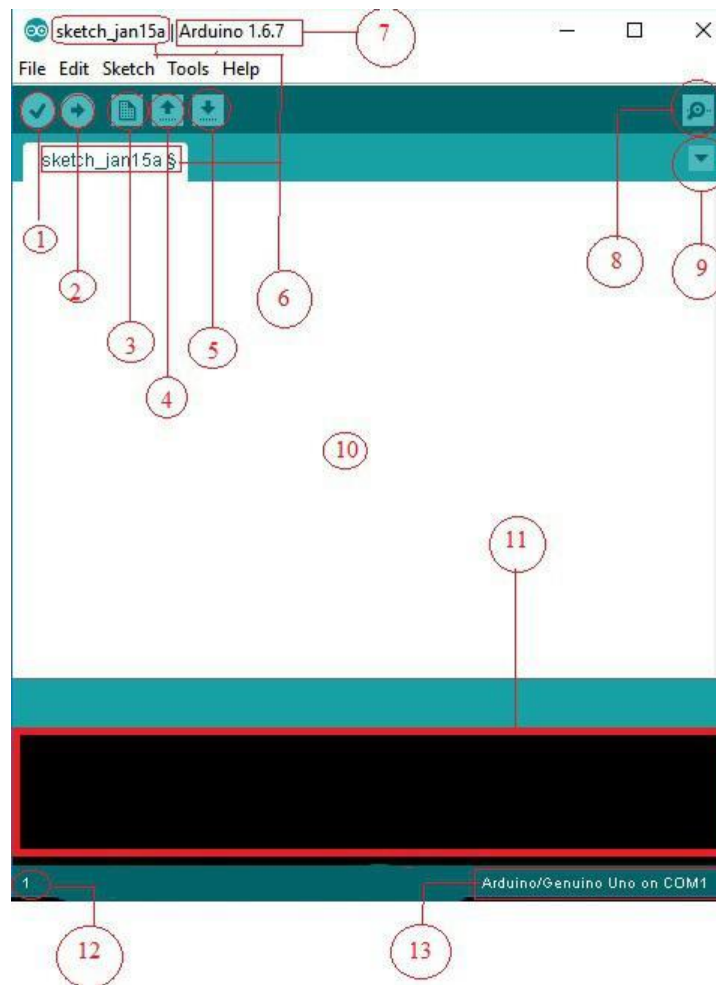


Figura 9. Elementet e faqës startuese

2.5.1 Shembuj brendapërbrenda softuerit

Siç jeni njoftuar më herët, softueri i Arduinos përmban një numër të madh të shembujve të programuar për përdorim. Këta janë të grupuar në bazë të kërkesave të ndryshme. Në figurën 7 është paraqitur selektimi i një programi komandues të servos. Gjithë shembujt janë të komentuar në gjuhën angleze. [6] [4] [7]

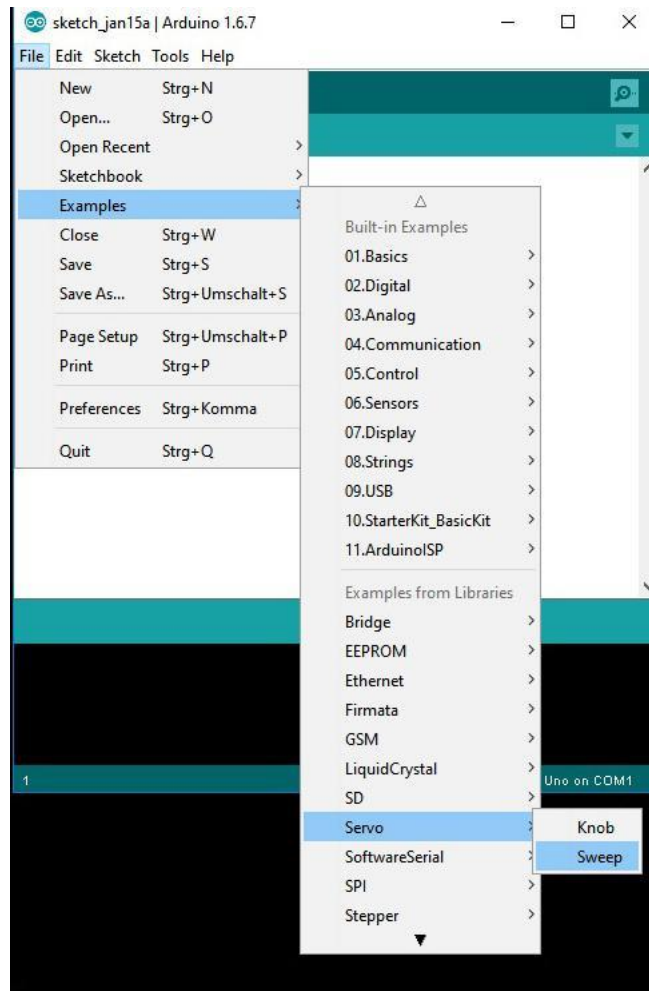


Figura 10. Mënyra e paraqitjes së shembujve në softuerin e Arduinos

3. DEKLARIMI I PROBLEMIT

Në qoftë se gjatë programimit të pllakës së Arduinos, vërehet që kodi, të cilin jemi duke e shkruar është shumë i gjatë për një Arduino, që do të thotë se kapaciteti i memories nuk është i mjaftueshëm ose pinat që kemi në dispozicion nuk mjaftojnë, atëherë mund të bëhet ndarja në dy e më shumë pllaka të Arduinos.

Problemi i parë që do të paraqitej, është lidhja e saktë e Arduinove si dhe komunikimi i tyre. Për të rritur kapacitetin memorues dhe të numrit të pinave, Arduinot lidhen nëpërmjet portit serik, ku edhe mund të bëhet transferimi i të dhënave prej njëres tek tjetra. Këtu duhet pasur parasysh se mungon sinkronizimi i duhur në mes të Arduinove. Normalisht, Arduino e parë merr rolin udhëheqës, ndërsa e dyta ekzekuton urdhrat e Arduinos së parë. Këtu, megjithatë, duhet të sigurohet se të dy Arduinot janë në dijeni se tjetra është e gatshme për komunikim. Furnizohen me rrymë që të dyja në të njëjtën kohë, atëherë fillon ekzekutimi rutinë instalimit. Megjithatë, në varësi të madhësisë, instalimi te njëra mund të përfundoj më parë se tek tjetra. Por këtu duhet ditur, se Arduino e parë, e cila ka edhe rolin udhëheqës, vendos se kur duhet Arduino tjetër të filloj me instalimin.

4. METODOLOGJIA

Për të realizuar dhe shtjelluar problemin e lidhjes serike të Arduinove janë përdorur këto metodologji studimi: hulumtimi i literaturës, shfletimi i internetit dhe metodologjia eksperimentale. Për hulumtim të literaturës janë përdorur libra të ndryshme dhe punime shkencore, për të bërë sa më të qartë problemin dhe shpjegimin e pjesës harduerike dhe softuerike. Një rol të veçantë ka luajtur shfletimi i internetit, përmes të cilit është bërë i mundur sqarimi sa më i mirë i pjesës softuerike, dhe është mundësuar shkarkimi i softuerit, prej nga edhe është bërë programimi i pllakës së Arduinos për shembujt e zgjedhur.

Metodologjia eksperimentale si koncept kryesor ka eksperimentin. Si pjesë kryesore për të zhvilluar eksperimentin është softueri i Arduinos, ku bëhet edhe ekzekutimi i tij, si dhe pllaka e Arduinos.

5. PUNIMI KRYESOR

Qëllimi kryesor i këtij punimi është lidhja serike e Arduinove. Paisjet e nevojshme për zhvillimin e këtij punimi janë: një bredboard, tre butona, dy Leda, disa fije kablllore, pesë rezistorë dhe tri Arduino. Fillimisht është bërë lidhja e nevojshme në mes të Arduinove e mandej është bërë lidhja me bredboard. Po ashtu duhet vendosur butonat dhe rezistorët në bredboard.

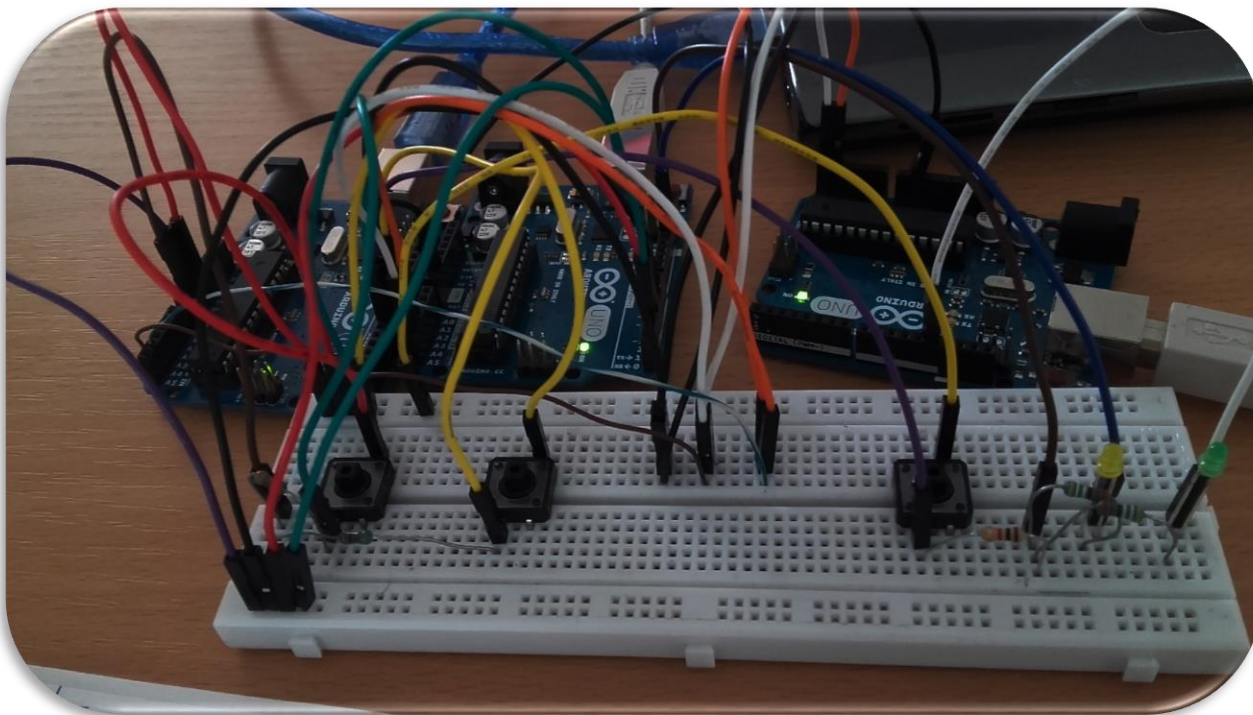


Figura 11. Vendosja e paisjeve në pllakat e Arduinove

Për realizimin e këtij punimi duhet të bëhet programimi i pllakave të Arduinove. Ato duhet të programohen veç e veç, me qëllim që të mos ngatërrohet kodi. Kodi duhet të shkruhet në Arduinon e parë si në figurën e mëposhtme, mandej duhet të ngarkohet.

```

#include <Wire.h>
const int Butoni = 6; // numri i Butonit
int GjendjaEButonit = 0; // Gjendja e Butonit
int GjendjaEButonitStop = 0;
int Butoni3 = 7;
int Butonpinstop = 9;
int a = 0;
// int Butonpin3 = 7;
int b = 0;
int butonstartadr3=0;
void setup ()
{
    Serial.begin (9600) ;
    pinMode (ButonpinStop, INPUT) ;
    Serial.begin (9600) ;
    Wire.begin ();
    pinMode (Buton1, INPUT) ;
    Serial.begin (9600) ;
    Serial.begin (9600) ;
    pinMode (Buton13, INPUT) ;
}
void loop () {
    Serial.println(GjendjaEButonit) ;
    Serial.println(GjendjaEButonitStop) ;
    GjendjaEButonit = digitalRead(Butoni) ;
    GjendjaEButonitStop = digitalRead(ButonpinStop) ;
    butonstartadr3 = digitalRead(Buton13) ;
    if (GjendjaEButonit ==1 ) {
        a=1 ;
        b=1 ;    }
    if (GjendjaEButonitStop ==1) {
        a=0 ;
        b=0 ; }
    if (butonstartadr3 ==1) {
        a=0 ;
        b=1 ;
    }
    Wire.beginTransaction (2) ;
    Wire.write (a) ;           // Dergimi i një bajt
    Wire.endTransmission ( ) ;
    Wire.beginTransaction (3) ;
    Wire.write (b) ;
    Wire.endTransmission ( ) ;
    Serial.println (b) ;
}

```

Figura 12. Kodi i Arduinos së parë

Që ta kemi kodin sa më të qartë duhet të bëhet paraqitja përmes diagramit si në figurën e mëposhtme.

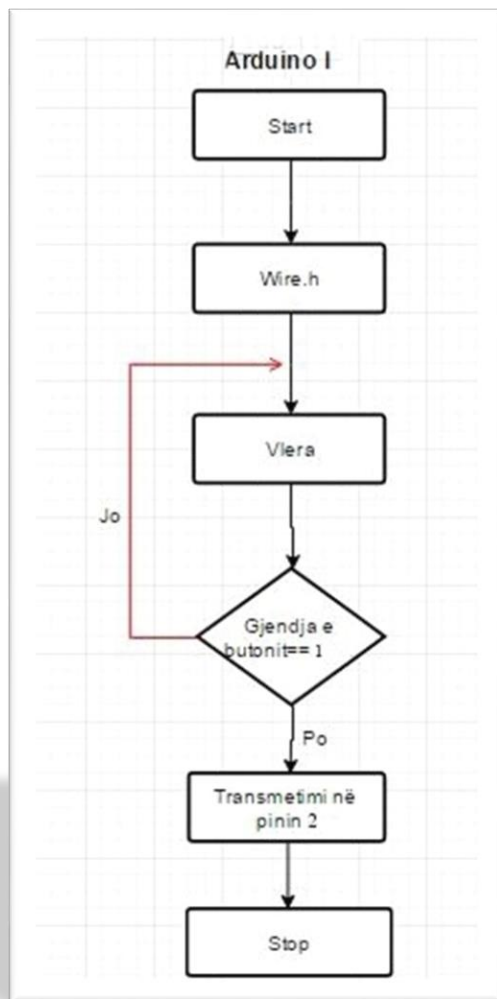


Figura 13. Algoritmi i Arduinos së parë

Mandej vazhdohet me shkrimin e kodit në Arduinon e dytë si në figurën e mëposhtme, si dhe bëhet ngarkimi i tij.

```
#include <Wire.h>
const int LedPin = 13 ; // numri i LED pinit
int dp0 = 0;
int dp1 = 0;
int dp2 = 0;
int dp3 = 0;
int adresa = 0;
int Gjendjadip0 = 0;
int Gjendjadip1 = 0;
int Gjendjadip2 = 0;
int Gjendjadip3 = 0;
int dip3 = 5;
int dip2 = 4;
int dip1 = 3;
int dip0 = 2;
void setup ()
{
  Serial.println (adresa) ;
  Serial.begin (9600) ;
  Wire.begin (2) ;
  Wire.onReceive (receiveEvent) ; //Rexhistron ngjarjen
  Serial.begin (9600) ; //Pranimi në seria
  pinMode (LedPin, OUTPUT) ;

  Serial.begin (9600) ;
  pinMode (dip0, OUTPUT) ;
  gjendjadip0 = digitalRead (dip0) ;
  gjendjadip1 = digitalRead (dip1) ;
  gjendjadip2 = digitalRead (dip2) ;
  gjendjadip3 = digitalRead (dip3) ;
  adresa = digitalRead (adresa) ;
  if (gjendjadip0 == 1);
  dp0 = 1; }
  else{
  dp0 = 0;
  }
  if (gjendjadip1 == 1);
  dp1 = 2; }
  else{
  dp1 = 0;
  }
  if (gjendjadip2 == 1);
```

```

dp2 = 4; }
else{
dp2 = 0;
}
if (gjendjadip3 == 1);
dp3 = 8; }
else{
    dp3 = 0;
}
adresa = dp0 + dp1 + dp2 + dp3;
}
void loop ( )
{
    delay(200);
    digitalWrite (LedPin, LOW);
}
void reciveEvent (int howMany);
{
    int a = Wire.read ( );
    // Serial.println(a);
    if (a == 1) {
        digitalWrite (LedPin, HIGH);
        // Serial.println(a);
    }
}

```

Figura 14. Kodi i Arduinos së dytë

Njësoj si më lartë, bëjmë paraqitjen e diagramit edhe për këtë kod, që të kemi një shpjegim më të qartë.

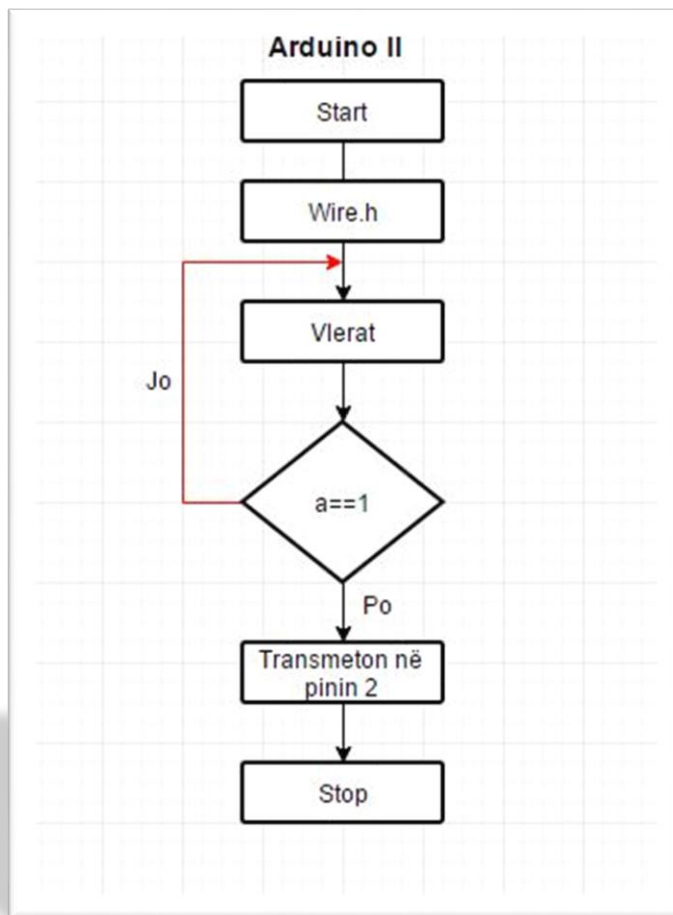


Figura 15. Algoritmi i Arduinos së dytë

Vazhdojm me procedurat e njejta edhe për pllakën e tretë, si në rastet më lart.

```

#include <Wire.h>
const int LedPin = 13 ; // numri i LED pinit
int dp0 = 0;
int dp1 = 0;
int dp2 = 0;
int dp3 = 0;
int adresa = 0;
int Gjendjadip0 = 0;
int Gjendjadip1 = 0;
int Gjendjadip2 = 0;
int Gjendjadip3 = 0;
int dip3 = 5;
int dip2 = 4;
int dip1 = 3;
int dip0 = 2;
void setup ()
{
  Serial.begin (9600) ;
  Wire.onReceive (receiveEvent) ; //Rexhistron ngjarjen
  Serial.begin (9600) ; //Pranimi në seria
  pinMode (LedPin, OUTPUT) ;
    Serial.begin (9600) ;
  pinMode (dip0, OUTPUT) ;
    gjendjadip0 = digitalRead (dip0) ;
    gjendjadip1 = digitalRead (dip1) ;
    gjendjadip2 = digitalRead (dip2) ;
    gjendjadip3 = digitalRead (dip3) ;
    adresa = digitalRead (adresa) ;
  if (gjendjadip0 == 1);
    dp0 = 1; }
  else{
    dp0 = 0;
  }
  if (gjendjadip1 == 1);
    dp1 = 2; }
  else{
    dp1 = 0;
  }
  if (gjendjadip2 == 1);
    dp2 = 4; }
  else{
    dp2 = 0;
  }
  if (gjendjadip3 == 1);
    dp3 = 8; }
  else{
    dp3 = 0;
  }

```

```

}
adresa = dp0 + dp1 + dp2 + dp3;
Wire.begin (3);
}
void loop ( )
{
    delay(200);
    digitalWrite (LedPin, LOW);
}
void reciveEvent (int howMany);
{
    int b = Wire.read ( );
    // Serial.println(b);
    if (b == 1) {
        digitalWrite (LedPin, HIGH);
        // Serial.println(b);
    }
}

```

Figura 16. Kodi i Arduinos së tretë

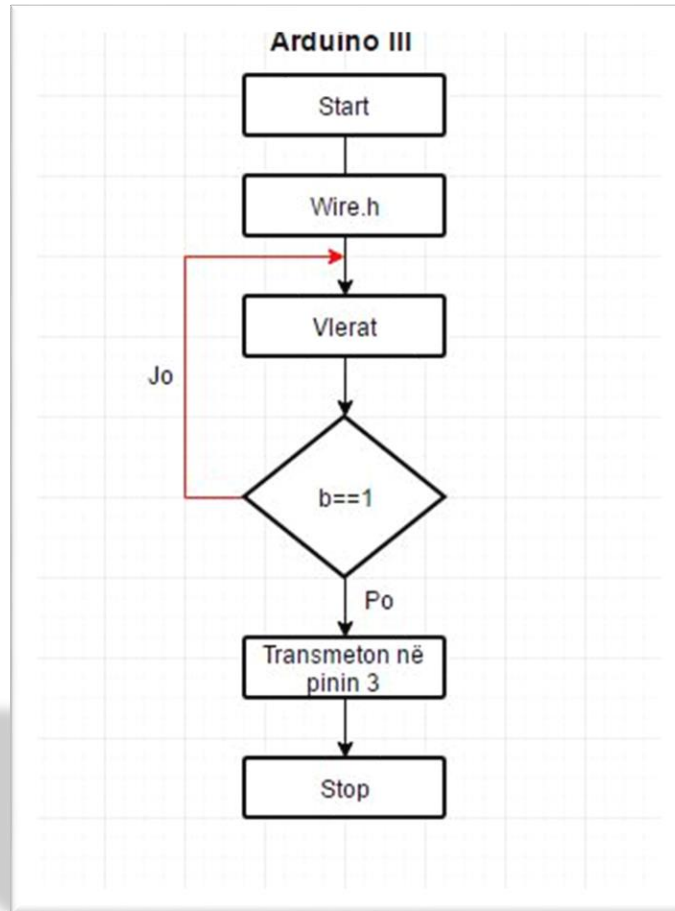


Figura 17. Algoritmi i Arduinos së tretë

Pasi të siguroheni që është ngarkuar kodin në secilën Arduino duhet të bëhet testimi i tri Arduinove dhe duhet caktuar portet e komunikimit. Me shtypjen e butonit të parë në Arduinon e parë fillon të lexohet kodi dhe Arduino e parë fillon të komunikoj me Arduinon e dytë e cila ka për detyrë që të ndez Ledat si në figurën e mëposhtme.

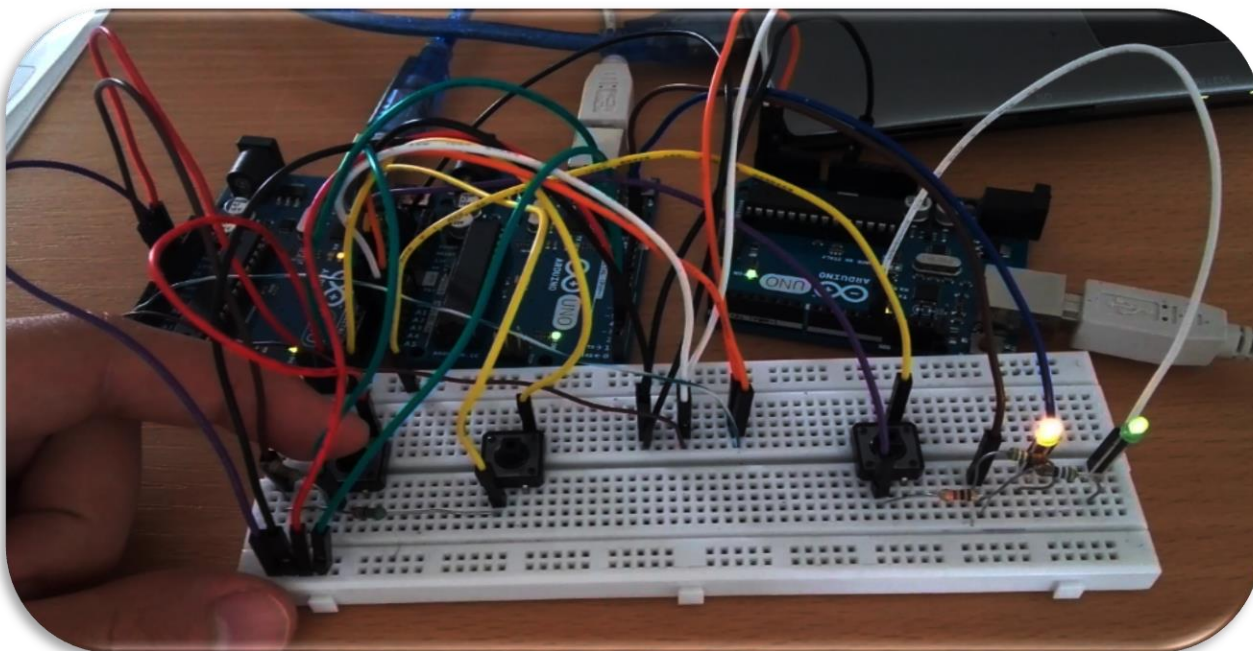


Figura 18. Ndezja e dy Ledave

Vazhdohet me butonin e dytë i cili duhet ti fik dy Ledat si në figurën e mëposhtme.

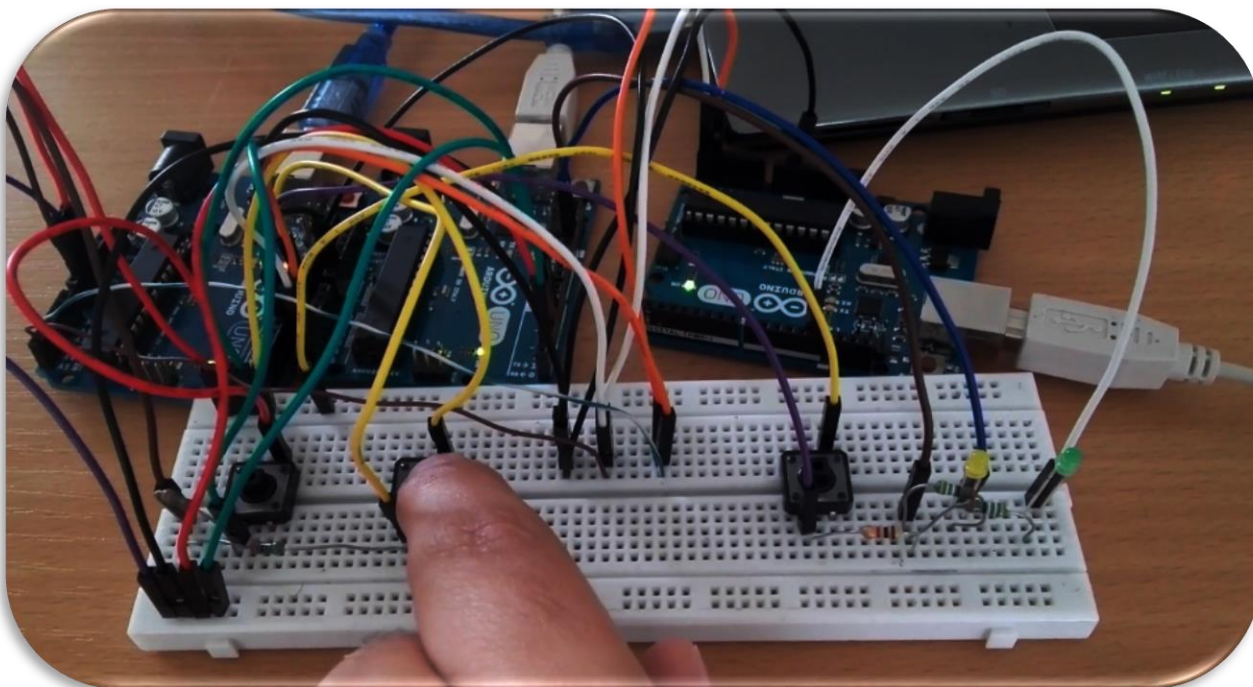


Figura 19. Fikja e dy Ledave

Butoni i tretë ka për detyrë që të mbaj të fikur Ledin e parë dhe të ndez Ledin e dytë. Këtë veprim mund ta shohim në figurën e mëposhtme.

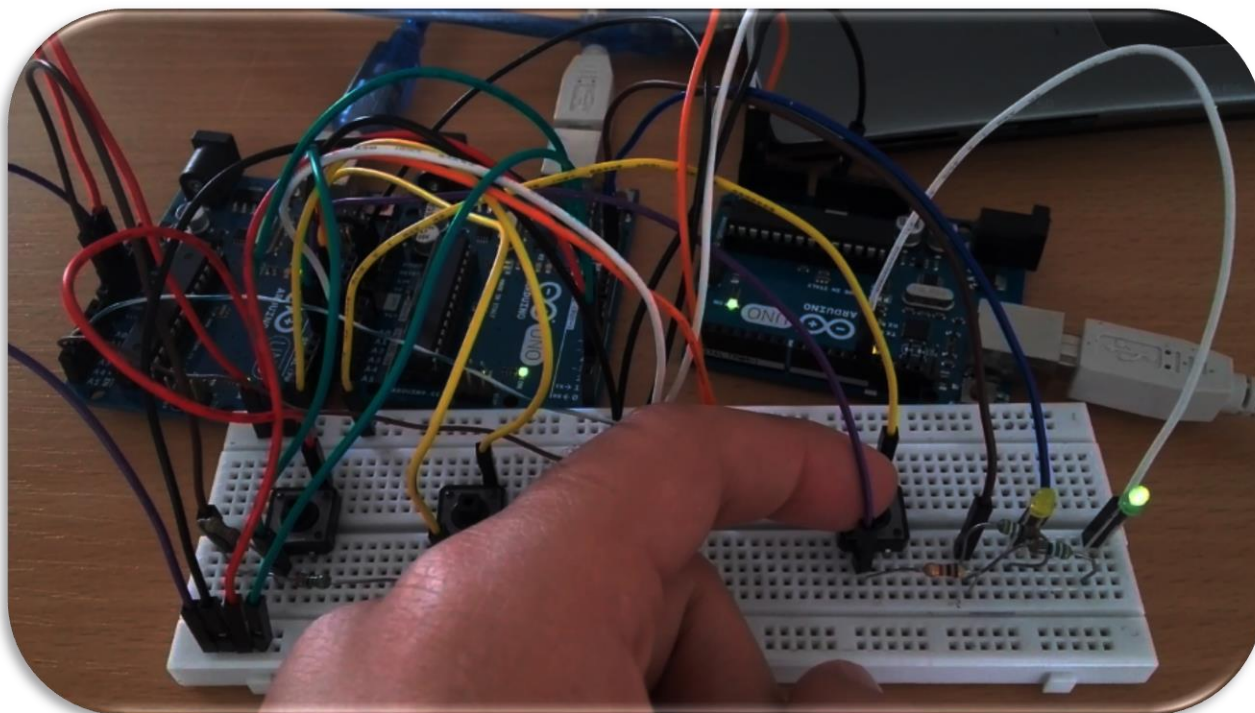


Figura 20. Fikja e Ledit të parë dhe ndezja e Ledit të dytë

Për ti fikur dy Ledat duhet të shtypet butoni i dytë dhe kështu përfundon leximi i kodit në Arduino. Nëse dëshirojm ti ndezim Ledat përsëri, duhet të bëjmë të njejtin veprim si në figurën 18.

6. DISKUTIME DHE PËRFUNDIME

Pikë kyçe e këtij punimi është lidhja serike e Arduinove. Njëra ndër veprimet më me rëndësi është lidhja e saktë si dhe komunikimi i tyre. Qëllimi kryesor i lidhjes së Arduinove në seri është rritja e kapacitetit memorues dhe numrit të pinave. Arduinot lidhen në seri nëpërmjet portit serik, ku edhe mund të bëhet transferimi i të dhënave prej njëres tek tjetra. Këtu duhet pasur parasysh që të mos mungoj sinkronizimi i duhur në mes të Arduinove. Sinkronizimi është duke funksionuar nëse ndodh ndezja dhe fikja e dritave sinjalizuese TX dhe RX, si dhe verifikohet duke u siguruar që është ngarkuar kodi në secilën Arduino. Mandej bëhet caktimi i porteve të komunikimit të Arduinos amë me Arduinot tjera. Normalisht, Arduino amë ka rolin udhëheqës, ndërsa Arduinot tjera ekzekutojnë urdhërat e Arduinos së parë. Këtu, megjithatë duhet Arduinoja dërguese të sigurohet se Arduinoja marrëse është e gatshme për komunikim. Përderisa të dyja dritat sinjalizuese TX dhe RX gjatë transferimit të të dhënave ndizen dhe fiken, duhet kuptuar se komunikimi në mes të kompjuterit dhe Arduinove në mes vete është duke funksionuar pa problem. Megjithatë, në varësi të hapësirës memoruese, ngarkimi i kodit tek njëra mund të përfundojë më parë se tek tjetra.

Për realizimin e këtij punimi është bërë programimi i pllakave të Arduinove. Emërtimi i kodeve të Arduinove në dritare të ndryshme nuk guxon të jetë i njëjtë. Kodet mund të programohen edhe në të njëjtën dritare punuese, mirëpo duhet të bëhet ndarja përkatëse.

Si përfundim mund të nxjerrim se është e mundur të bëhet lidhja e më shumë se tri Arduinove por duhet pasur kujdes me lidhjen e saktë dhe komunikimin e tyre. Po ashtu mund të shtohen edhe funksione të tjera, siç janë: shtimi i Ledave, butonave, rezistorëve, LCD display etj.

Gjatë këtij punimi janë plotësuar rezultatet e pritura ashtu siç shihet më lartë tek punimi kryesor.

7. REFERENCAT

- [1] M. Schmidt, Arduino AQuick-StartGuide, United States of America: The Pragmatic Bookshelf , 2011.
- [2] M. Knapp, Roboter bauen mit Arduino, Bonn: Galileo Press, 2015.
- [3] G. Schmitt, Mikrocomputertechnik mit Controllern der Atmel AVR-RISC-Familie, München,Wien: Oldenbourg, 2005.
- [4] U. Sommer, Mikrocontroller-Programmierung mit Arduino/Freduino, Poing/Haar bei München: Franzis, 2010/2013.
- [5] J. N. Gerhard Franken, Arduino für Dummies, Weinheim: Wiley-VCH, 2015.
- [6] K. Dembowski, Embedded-Systeme mit der Arduino-Plattform, Berlin: VDE-Verlag, 2014.
- [7] R. Walter, AVR-Mikrocontroller-Lehrbuch, Berlin: Denkholtz-Buchmanufaktur, 2009.